



Fakultät Informatik

Studiengang Künstliche Intelligenz und Computer Vision (Master)

Modulhandbuch

Stand: SS 2026

Februar 2026

Prof. Dr. Erich Müller
Studiendekan der Fakultät Informatik

Prof. Dr. Klaus Ulhaas
Studiengangkoordinator

Prof. Dr. Jürgen Brauer
Vorsitzender der Prüfungskommission

Inhaltsverzeichnis

1 Ziele und Aufbau des Studiengangs Künstliche Intelligenz und Computer Vision .	2
2 Begriffserläuterungen	6
3 Modulbeschreibungen	7
KIM101 Ausgewählte Verfahren der KI 1	7
KIM102 Mustererkennung und maschinelles Lernen	9
KIM103 Ausgewählte Verfahren der KI 2	11
KIM104 Deep Learning	13
KIM105 Computer Vision	15
KIM4101 Ethik & KI	17
KIM2106 Softwarearchitektur	19
KIM2107 Mobile Roboter	21
KIM2110 Multimodale Sensorsysteme	23
KIM2111 Algorithmen und Strategien zur Entscheidungsunterstützung	25
KIM2112 Reinforcement Learning	27
KIM2113 Deep Learning für Computer Vision	29
KIM2114 3D Computer Vision	31
KIM2115 Data Science	33
KIM2116 Big Data	35
KIM2117 Differentiable Programming	37
KIM2118 Funktionale Sicherheit	39
KIM2119 Large Language Models	41
KIM5100 Projekt	43
KIM6101 Masterarbeit	44

1 Ziele und Aufbau des Studiengangs Künstliche Intelligenz und Computer Vision

Der Masterstudiengang „Künstliche Intelligenz und Computer Vision“ soll den Studierenden Kompetenzen im Forschungs- und Entwicklungsbereich der Informatik an der Schnittstelle zwischen Künstlicher Intelligenz und Computer Vision vermitteln.

Die Studierenden erreichen mit diesem Studium die Kompetenz Problemstellungen im Bereich Künstlicher Intelligenz (KI) sowie im Bereich Computer Vision (CV) zu analysieren, ein Software-design zu erstellen und geeignete Methoden auszuwählen, zu adaptieren und umzusetzen. Das Studium vermittelt dabei Kompetenzen in den zwei Bereichen KI und CV, die miteinander sehr verwurzelt sind und gerade im Bereich der Automatisierung sehr nachgefragt werden.

Im Bereich KI werden im Studium Methoden der klassischen KI und der Mustererkennung vermittelt. Dieser Bereich erstreckt sich bis zu den aktuellen Forschungs- und Entwicklungsergebnissen aus dem Bereich des tiefen Lernens (Deep Learning).

Im Bereich CV werden klassische Methoden des maschinellen Sehens vermittelt. Neben den klassischen 2D Bildverarbeitungsmethoden können u.a. Methodenkompetenzen im Bereich Deep Learning für Computer Vision und im Bereich 3D Computer Vision erlangt werden.

Besonderes Augenmerk wird auch auf die Fähigkeit gelegt, gelernte Methoden praktisch mit modernen Programmiersprachen umzusetzen und auf neue Anwendungsfelder zu übertragen. Im Studium wird von den Studierenden auch erwartet, die heutigen Möglichkeiten des Einsatzes von Künstlicher Intelligenz und Computer Vision kritisch und ethisch zu hinterfragen.

Das Studium ist als konsekutiver Studiengang für den Bachelorstudiengang „Informatik“ der Hochschule Kempten oder eines inhaltlich nahe verwandten Studiengangs angelegt. Dabei sollen aus dem Bachelorstudium bestimmte Kompetenzen aus dem Bereich der Mathematik und Informatik vorhanden sein. Aus der Mathematik sind Lineare Algebra, Wahrscheinlichkeitsrechnung und Analysis erforderlich. Im Bereich Informatik benötigen die Studierenden fundierte Fähigkeiten in der Programmierung, Algorithmen und Datenstrukturen, Softwareengineering und der Theoretischen Informatik.

Auf Basis dieser Grundausbildung hinaus erlangen die Studierenden Kompetenzen, über die in unten stehender Tabelle ein Überblick gegeben wird:

Nr.	Studienziel	Lernergebnisse
1	Algorithmische und mathematische Kompetenz	• Verständnis aktueller mathematischer Konzepte und Algorithmen • Fertigkeit zur Auswahl und Adaption von aktuellen Vorgehensweisen für gegebene Problemstellungen • Kompetenz zur wissenschaftlich fundierten Weiterentwicklung
2	Methodische Kompetenz Künstliche Intelligenz und Computer Vision	• Tiefes Verständnis der Algorithmen und Methoden in den Bereichen Künstliche Intelligenz, Computer Vision, Deep Learning und Mustererkennung • Fertigkeit zur Auswahl und Adaption von aktuellen Vorgehensweisen für gegebene Problemstellungen • Kompetenz zur Erarbeitung einer Lösung für gegebene, komplexe Problemstellungen aus oben genannten Bereichen • Kompetenz zur wissenschaftlich fundierten Weiterentwicklung von Technologien dieser Bereiche • Fähigkeit Softwarearchitekturen für den Einsatz Künstlicher Intelligenz und Computer Vision zu entwickeln

3	Technologiekompetenz	• Kenntnis aktueller Hardware- und Software-Lösungen im Bereich Deep Learning, Computer Vision und Mustererkennung sowie deren algorithmischer Grundlagen • Fähigkeit zur Auswahl und Kombination passender Lösungen • Kompetenz zur Erarbeitung einer Gesamtlösung
4	Anwendungskompetenz	• Fähigkeit das erworbene Wissen auch in großen und fachlich komplexen Problemstellungen einzusetzen und erfolgreich anzuwenden • Befähigung zur wissenschaftlichen Arbeit • Verständnis für den Zusammenhang der verschiedenen Fachbereiche • Kompetenz zur Kombination und Verknüpfung von Problemlösungen verschiedener Fachbereiche des Studiengangs zur Lösung komplexer Problemstellungen
5	Soziale und überfachliche Kompetenzen	• Fähigkeit Systeme und Verfahren der künstlichen Intelligenz und der Computer Vision hinsichtlich ethischer und datenschutzrechtlicher Aspekte kritisch zu reflektieren und zu hinterfragen • Kompetenz größere Projekte des Fachgebiets zu planen, zu organisieren und zu leiten, sowie gegenüber Dritten zu vertreten, Probleme frühzeitig zu erkennen und zu lösen und Projekte erfolgreich abzuschließen • Fähigkeit, im Team fachlich als auch leitend verantwortliche Funktionen zu übernehmen • Kompetenz gemischte Teams zu koordinieren und zu leiten, um ein Gesamtprojekt zu realisieren

Ein wesentlicher Teil des Kompetenzerwerbs und der Lernergebnisse wird durch die Pflichtmodule Ausgewählte Verfahren der KI 1 und der KI 2, Mustererkennung und Maschinelles Lernen, Deep Learning und Computer Vision erreicht. Einen nicht unerheblichen Anteil zur Erlangung einer gewissen sozialen Kompetenz im Umgang mit ethischen und datenschutzrechtlichen Fragen nimmt das Pflichtseminar KI & Ethik ein.

Die Module Ausgewählte Verfahren der Künstlichen Intelligenz 1 und 2 vermitteln die klassischen Verfahren der künstlichen Intelligenz. Diese werden auch heute in Anwendungen der Mustererkennung und der klassischen Computer Vision eingesetzt. Die Computer Vision liefert Merkmale und Bildeigenschaften für die Mustererkennung und den klassischen Neuronale Netze, wohingegen Deep Learning komplexe Netzstrukturen bereitstellt, die besonders auf die Bereiche 2D und 3D Computer Vision ausgerichtet sind.

Mit der Anfertigung einer Masterarbeit, die im Interesse einer raschen Praxiseingliederung der Studierenden entweder im Rahmen eines Projektes mit einem Partner aus der Industrie oder in einem der Forschungsprojekte der Hochschule erstellt wird, erreichen die Studierenden eine solide Anwendungskompetenz gerade im Zusammenhang mit der Befähigung zur wissenschaftlichen Arbeit. In ihr sollen die Studierenden ihre Fähigkeit nachweisen, die im Studium erworbenen Kenntnisse, Fertigkeiten und Kompetenzen in einer selbstständig angefertigten, anwendungsorientiert-wissenschaftlichen Arbeit auf komplexe Aufgabenstellungen anzuwenden.

Modul	Studienziel				
	1	2	3	4	5
Ausgewählte Verfahren der KI 1	++	++	+	+	+
Ausgewählte Verfahren der KI 2	++	++	+	+	+
Mustererkennung und Maschinelles Lernen	+	++	++	++	
Deep Learning	+	++	++	++	
Computer Vision	++	++	+	+	+
Ethik & KI			+	++	++
Master Projekt	+	+	+	++	++
Deep Learning für CV	+	++	++	+	+
3D Computer Vision	++	++	+	++	+
Mobile Roboter	++		+	++	
Multimodale Sensorsysteme	+	++	++	++	
Data Science	+	+	+	++	
Softwarearchitektur		++	+	+	
Big Data	++		+	++	
Algorithmen und Strategien zur Entscheidungsunterstützung	++	+		++	+
Masterarbeit	++	++	+	++	+

Aufbau des Studiengangs Künstliche Intelligenz und Computer Vision

Der Masterstudiengang ist modularisiert. In Anlehnung an das European Credit Transfer System (ECTS) werden für die drei Semester des Masterstudiums insgesamt 90 Leistungspunkte (CP) vergeben, und zwar pro Semester 30 Leistungspunkte. Insgesamt sind 25 CP für Pflichtmodule vorgesehen, 20 Leistungspunkte für Wahlpflichtmodule, 10 Leistungspunkte für das Masterprojekt, 5 ECTS für ein Seminar und 30 Leistungspunkte für die Masterarbeit. Das Studium kann sowohl im Winter- als auch im Sommersemester begonnen werden. Wird das Studium im Wintersemester begonnen, ändert sich nur die Abfolge der Semester. Durch die Konzeption der Module als abgeschlossene Einheiten entsteht daraus für die Studierenden kein Nachteil.

Künstliche Intelligenz und Computer Vision, Master (M.Sc.)

Hochschule für angewandte Wissenschaften Kempten

Semester

3	Masterarbeit ⁽⁴⁾																																																										
2	Ausgewählte Verfahren der KI 2 ⁽¹⁾					Deep Learning ⁽¹⁾					Computer Vision ⁽¹⁾					WP-Modul ⁽²⁾					WP-Modul ⁽²⁾					WP-Modul ⁽²⁾																																	
1	Ausgewählte Verfahren der KI 1 ⁽¹⁾					Mustererkennung und Maschinelles Lernen ⁽¹⁾					WP-Modul ⁽²⁾					Ethik & KI (Seminar) ⁽³⁾					Master-Projekt ⁽³⁾																																						
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30																																																											
ECTS-Punkte																																																											
Legende:																																																											
1			Pflichtmodule																											3			Projekt / Seminar																										
2			Wahlpflichtmodule																											4			Masterarbeit																										

Das Masterstudium kann auch als Teilzeitstudium in sechs Semestern absolviert werden. Im

Teilzeitstudium sind pro Semester 15 Leistungspunkte zu erbringen. In den ersten vier Semestern erfolgt die theoretische Ausbildung und die beiden letzten Semester sind für die Masterarbeit reserviert.

Künstliche Intelligenz und Computer Vision, Master (M.Sc.) - Teilzeit

Hochschule für angewandte Wissenschaften Kempten

Semester

6	Masterarbeit ⁽⁴⁾														
5	Masterarbeit ⁽⁴⁾														
4	WP-Modul ⁽²⁾					WP-Modul ⁽²⁾					WP-Modul ⁽²⁾				
3	Ethik & KI (Semin- nar) ⁽³⁾					Master-Projekt ⁽³⁾									
2	Ausgewählte Verfah- ren der KI 2 ⁽¹⁾					Deep Learning ⁽¹⁾					Computer Vision ⁽¹⁾				
1	Ausgewählte Verfah- ren der KI 1 ⁽¹⁾					Mustererkennung und Maschinelles Ler- nen ⁽¹⁾					WP-Modul ⁽²⁾				

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

ECTS-
Punkte

Legende:



Pflichtmodule



Wahlpflichtmodule



Projekt / Seminar



Masterarbeit

Für den Studiengang Künstliche Intelligenz und Computer Vision stehen folgende Ansprechpartner zur Verfügung:

Studiengangkoordinator:

Prof. Dr. Klaus Ulhaas

Fachstudienberater:

Prof. Dr. Klaus Ulhaas

Vorsitzender der Prüfungskommission:

Prof. Dr. Jürgen Brauer

2 Begriffserläuterungen

ECTS - European Credit Transfer System

Diese Vereinbarungen zur Anrechnung, Übertragung und Akkumulierung von Studienleistungen basieren auf dem Arbeitspensum, das Studierende durchzuführen haben, um die Ziele des Lernprogramms zu erreichen. Für jede studienbezogene Leistung wird der voraussichtliche durchschnittliche Arbeitsaufwand angesetzt und auf das Studienvolumen angerechnet. Der Arbeitsaufwand umfasst Präsenzzeit und Selbststudium ebenso wie die Zeit für die Prüfungsleistungen, die notwendig sind, um die Ziele des vorher definierten Lernprogramms zu erreichen. Mit dem ECTS können Studienleistungen international angerechnet und übertragen werden.

Arbeitsaufwand (Workload) und Leistungspunkte (ECTS-LP)

Der Arbeitsaufwand der Studierenden wird im ECTS in credit points angegeben. Deutsche Übersetzungen für credit point sind die Begriffe Leistungspunkt oder ECTS-Punkt. Ein Arbeitsaufwand von 30 Zeitstunden bedeutet einen Leistungspunkt. Der Arbeitsaufwand von Vollzeitstudierenden entspricht 60 Leistungspunkten pro Studienjahr, also 30 Leistungspunkten pro Semester. Das sind 1.800 Stunden pro Jahr oder 45 Wochen/Jahr mit 40 Stunden/Woche.

Der Arbeitsaufwand setzt sich zusammen aus:

- Präsenzzeit
- Zeit für die Vor- und Nachbereitung des Vorlesungsstoffs,
- Zeit für die Vorbereitung von Vorträgen und Präsentationen,

Die Bachelorstudiengänge mit sieben Semestern bescheinigen erfolgreichen Studierenden 210 ECTS-LP, die dreisemestrigen Masterstudiengänge weitere 90 ECTS-LP. Damit ist die Forderung nach 300 ECTS-LP für ein erfolgreich abgeschlossenes Masterstudium erfüllt.

Semesterwochenstunden und Präsenzzeit

Der Studiengang setzt sich aus Modulen zusammen. Ein Modul repräsentiert eine inhaltlich und zeitlich zusammengehörige Lehr- und Lerneinheit. Module werden in der Regel in einem Semester abgeschlossen. Modulgruppen sind Zusammenfassungen von Modulen mit einem weiteren inhaltlichen Zusammenhang. In allen Fällen stellt ein Modul oder ein Teilmodul eine Einheit dar, für die innerhalb und am Ende eines Semesters eine Prüfungsleistung erbracht werden kann, für die Leistungspunkte vergeben werden. Die Lehrveranstaltungen werden derzeit in deutscher Sprache gehalten.

Studienbegleitende Prüfungen und Studienfortschritt

Sämtliche Prüfungen erfolgen über das gesamte Studium verteilt studienbegleitend und stehen in direktem Bezug zur Lehrveranstaltung. Prüfungsbestandteile können je nach Lehrveranstaltung begleitend oder nach Abschluss des Moduls stattfinden, beispielsweise als Referat, Klausurarbeit, mündliche Prüfung, Hausarbeit mit Kolloquium, Entwurf mit Kolloquium, Laborbericht, Exkursionsbericht oder einer Kombination. In den Beschreibungen der einzelnen Module wird im Modulhandbuch die jeweilige Prüfungsform festgelegt. Eine Wiederholung der Prüfung eines Moduls erfolgt bei Nichtbestehen im folgenden Semester.

3 Modulbeschreibungen

KIM101 Ausgewählte Verfahren der KI 1

Allgemeines

Im Modul werden grundlegende Konzepte und Hintergründe der Künstlichen Intelligenz und des Maschinellen Lernens und der dort verwendeten Verfahren betrachtet. Die Studenten und Studentinnen sollen nicht nur einen Einblick in verschiedene Strategien und Konzepte erhalten um sie anwenden zu können, sondern auch in der Lage sein, die Verfahren zu implementieren und weiterzuentwickeln.

Modulverantwortliche(r):	Prof. Dr. Stefan Rieck
Dozent(en):	Prof. Dr. Stefan Rieck
Modultyp:	Pflichtmodul
Voraussetzungen:	
Verwendbarkeit:	
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage,

- die in der Vorlesung vorgestellten Verfahren auf konkrete Probleme anzuwenden
- die Verfahren in einer geeigneten Programmiersprache zu implementieren
- die Verfahren weiterzuentwickeln

Lehrinhalte:

Die folgende Liste gibt eine Übersicht der behandelten Standard-Themen:

- Grundlegende Konzepte: Definition von KI, KI-Paradigmen, Intelligente Agenten
- Suchverfahren
- Ausgewählte Verfahren zum Maschinellen Lernen
- Wissensrepräsentation

Darüber hinaus werden aktuell in der Forschung befindliche Verfahren vorgestellt.

Literatur:

-
- Stuart Russell, Peter Norvig : Artificial Intelligence: A Modern Approach, Global Edition, Pearson 2021
 - Aktuelle Papers zu KI

KIM102 Mustererkennung und maschinelles Lernen

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Stefan Rieck
Dozent(en):	Prof. Dr. Stefan Rieck
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Künstliche Intelligenz und Computer Vision (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung/Praktikum
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden beherrschen die grundlegenden Begriffe des Maschinellen Lernens (ML). Sie können Verfahren der Mustererkennung und des Maschinellen Lernens adäquat auswählen und Mustererkennungsaufgaben in der Praxis lösen.

Lehrinhalte:

- Skripting in Python
- Grundlagen des Maschinellen Lernens
- Data Preprocessing (Vorverarbeitung, Merkmale)
- Grundlegende Verfahren zur Erkennung einfacher Muster (Bayes Classifier, Support Vector Machines, Neuronale Netze)
- Erkennung komplexer Musterfolgen und strukturierter Muster
- Kombination von Klassifikatoren

Literatur:

- Christopher M. Bishop: Pattern Recognition and Machine Learning. Springer, Berlin 2006, ISBN 387310738.
- Anselm Lingnau: Programmieren in Python: Eine praktische Einführung, online: <https://www.tuxcademy.org/download/de/pyth/pyth-de-manual.pdf>
- Pedregosa et al.: Scikit-learn: Machine Learning in Python, JMLR 12, pp. 2825-2830, 2011.
- Aurélien Géron. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Con-

cepts, Tools, and Techniques for Building Intelligent Systems. O'Reilly. 2017, ISBN: 1491962291

- Sebastian Raschka und Vahid Mirjalili: Machine Learning mit Python, mitp, 3. Auflage 2021, ISBN: 9783747502136

KIM103 Ausgewählte Verfahren der KI 2

Allgemeines

Im Modul werden grundlegende Konzepte und Hintergründe der Künstlichen Intelligenz und des Maschinellen Lernens und der dort verwendeten Verfahren betrachtet. Die Studenten und Studentinnen sollen nicht nur einen Einblick in verschiedene Strategien und Konzepte erhalten um sie anwenden zu können, sondern auch ein möglichst umfassendes Verständnis der zugrundeliegenden Informatik gewinnen. Das Modul schließt an das Modul "Ausgewählte Verfahren der KI 1" an, baut aber nicht darauf auf.

Modulverantwortliche(r):	Prof. Dr. Elmar Böhler
Dozent(en):	Prof. Dr. Elmar Böhler
Modultyp:	Pflichtmodul
Voraussetzungen:	
Verwendbarkeit:	Studiengang Künstliche Intelligenz und Computer Vision (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Lehrinhalte:

Die Lehrinhalte können abhängig vom aktuellen Wissensstand variieren; mögliche behandelte Themen sind:

- Aussagenlogik
- Prädikatenlogik
- Automatisches Beweisen, Resolution, Hornalgorithmus
- Logische Inferenz
- Probabilistische Verfahren
- Probabilistische Automaten, Hidden Markov Models
- Bayessche Netze
- Probably Approximately Correct Learning (PAC)
- Algorithmische Wahrscheinlichkeit und Kolmogorov-Komplexität
- Minimum Description Length (MDL)

- Theoretische Fundierung der KI

Literatur:

Stuart Russell, Peter Norvig : Artificial Intelligence: A Modern Approach, Global Edition, Pearson 2021

KIM104 Deep Learning

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Jürgen Brauer
Dozent(en):	Prof. Dr. Jürgen Brauer
Modultyp:	Pflichtmodul
Voraussetzungen:	Analysis, Lineare Algebra, Programmierkenntnisse
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	4 SWS Seminaristischer Unterricht
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht mit Praxisbeispielen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden kennen mindestens drei der wichtigsten DL Architekturen im Detail
- Für jeden dieser drei Ansätze haben Sie fundiertes Wissen über den Aufbau und Funktionsweise gesammelt und können dieses Wissen bei Problemstellungen im Bereich Klassifikation, Regression und Spracherkennung durch Anwendung eines geeigneten DL Verfahrens umsetzen.
- Die Studierenden verstehen, welche Eigenschaften die DL Verfahren besonders auszeichnet und von klassischen Machine Learning Verfahren unterscheidet
- Sie kennen die historische Entwicklung des DL Gebietes und können abschätzen welche weiteren Entwicklungen zu erwarten sind

Lehrinhalte:

- Motivation Deep Learning: kürzliche Erfolge
- Geschichte des Deep Learnings
- Wieso boomt Deep Learning erst jetzt?
- Das biologische Vorbild
- Technische Neuronenmodelle
- Ein Neuronales Netz ohne Deep Learning Bibliothek selber implementieren
- Perzeptron / Perzeptron-Lernregel
- Multi-Layer-Perzeptron
- Backpropagation-Algorithmus
- Deep Learning Bibliotheken

- Rekurrente Neuronale Netze
- Deep Learning zur Textverarbeitung
- Die Transformer-Architektur
- Optimierer und Deep Learning Tricks
- Ausblick: wohin entwickelt sich der Ansatz?

Literatur:

- Yann LeCun, Yoshua Bengio, Geoffrey Hinton. Deep Learning. Nature, 2015.
- Jürgen Schmidhuber. Deep Learning in Neural Networks: An Overview. Neural Networks Journal 61 (2015): 85-117
- Dive into Deep Learning: <https://d2l.ai/>

KIM105 Computer Vision

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Klaus Ulhaas
Dozent(en):	Prof. Dr. Klaus Ulhaas
Modultyp:	Pflichtmodul
Voraussetzungen:	
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage

- den theoretischen Hintergrund klassischer Verfahren und moderner Verfahren zu erklären.
- die gelernten Verfahren zu kombinieren und mittels Pytorch und OpenCV auf Beispiele aus der Praxis zur Objekterkennung und -segmentierung zu übertragen.
- mathematische Zusammenhänge zu verstehen und anzuwenden.

Lehrinhalte:

Folgende Inhalte werden in der Veranstaltung bearbeitet:

- Einführung in Python, OpenCV und Pytorch
- Klassische 2D Computer Vision Verfahren wie z.B. HOG- und SIFT-Detektor
- Basisoperationen für Tiefe Neuronale Netze (Faltung, Max-Pooling, Max-Unpooling, Up-Convolution)
- Convolutional Neural Networks
- Segmentierung (klassisch: Selective Search & modern: U-Net)
- Objektlokalisierung und Objektdetektion
- Moderne Objektdetektoren: R-CNN, Fast-RCNN, Faster R-CNN, YOLO
- Implementierung von klassischen Computer Vision Verfahren mittels OpenCV und modernen Computer Vision Verfahren mittels Pytorch
- Metriken zur Objektdetektion wie z.B. Precision, Recall, F β -Maß, Konfusionsmatrix,

mAP, Multiclass-Confusion-Matrix

Literatur:

Literatur und Empfehlungen zu Einzelthemen werden fortlaufend in der Vorlesung bekanntgegeben

KIM4101 Ethik & KI

Allgemeines

Künstliche Intelligenz verspricht Innovationen, die den Menschen in seiner Arbeit und im Alltag hilfreich unterstützen könnten. Dabei geht man meist von teilautonomen und autonomen Systemen aus. Diese werden von uns alleine gelassen, müssen unabhängig von uns zwischen Optionen auswählen und zu Aktionen fähig sein. Sie müssen selbstständig entscheiden und handeln.

Die Akzeptanz von diesen Systemen, hängt maßgeblich davon ab, ob die Menschen Vertrauen in die Technologie setzen. Dieses Vertrauen hängt wiederum von der Antwort auf die Frage ab, wie diese Systeme zu ihren Entscheidungen und Bewertungen kommen. Dafür benötigen sie zum einen Formen der Intelligenz, zum anderen Formen der Moral.

Aber was bedeutet Moral? Wie kommt die Moral in die Maschine? Wie können ethische Prinzipien formalisiert werden damit diese in der KI genutzt werden können? Welche Herausforderungen ergeben sich für die verantwortungsbewusste Entwicklung und Anwendung von KI-Systemen, die Werte wie Selbstbestimmung, Gerechtigkeit, Schutz der Privatheit und der Persönlichkeit bewahren? Was können Individuen, Unternehmen und die Gesellschaft tun?

Und falls wir irgendwann intelligente Maschinen bauen sollten, die moralische Fähigkeiten haben und Empathie zeigen, wie würden wir mit ihnen umgehen?

Modulverantwortliche(r):	Prof. Dr. Rafael Mayoral Malmström
Dozent(en):	Prof. Dr. Rafael Mayoral Malmström
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Künstliche Intelligenz und Computer Vision (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	Seminar
Leistungspunkte:	5
Arbeitsaufwand:	15 Stunden Präsenzzeit Vorträge und anschließender Diskussion 135 Stunden selbständiges Arbeiten, Vorbereitung der Präsentation, Ausarbeitung der Studienarbeit
Leistungsnachweis und Prüfung:	Studienarbeit und Kolloquium
Zur Prüfung zugelassene Hilfsmittel:	ohne / keine Einschränkung, alle Hilfsmittel zugelassen

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage

- sich in ein anspruchsvolles, wissenschaftliches Thema im Bereich der Ethik und KI einzuarbeiten
- entsprechende Literatur zu recherchieren und aufzuarbeiten
- einen Vortrag zu einem wissenschaftlichen Thema zu erarbeiten, zu präsentieren und bei der nachfolgenden Diskussion zu vertreten

- eine schriftliche Ausarbeitung zu einem Thema im Bereich Ethik und KI nach den Grundsätzen guter wissenschaftlicher Praxis zu erstellen

Lehrinhalte:

Anhand von praktischen Beispielen aus der Künstlichen Intelligenz und Robotik sowie mit Hilfe von Texten von Maschinenethik-Expertinnen und -Experten widmen sich die Studierenden u.a. folgenden Themen:

- Ethik und Moral, moralische Vorstellungen und Wertesysteme
- Maschinenethik und Roboterethik
- Formalisierung ethischer Prinzipien, moralische Agenten, Verantwortung von Maschinen
- Beispielanwendungen aus Sicht der Maschinenethik: autonomes Fahren, Pflegeroboter, KI in Medizin, KI und die Demokratie, KI und Kunst, Sexroboter, autonome Waffensysteme, KI und Kunst, usw.
- Artificielle Superintelligenz, Singularität, Transhumanismus und Posthumanismus

Literatur:

- Christoph Bartneck et al.: Ethik in KI und Robotik, Hanser Fachbuch, 1. Aufl. (2019)
- Oliver Bendel (Hrsg.): Handbuch Maschinenethik, Springer VS, 1. Aufl. (2019)
- Klaus Mainzer (Hrsg.): Philosophisches Handbuch Künstliche Intelligenz, Springer Fachmedien, 1. Aufl. (2022)
- Catrin Misselhorn: Grundfragen der Maschinenethik, Reclam, 4. Aufl. (2018)
- Luís Moniz Pereira, António Barata Lopes: Machine Ethics, Springer, 1. Aufl. (2020)
- Luís Moniz Pereira, Ari Saptawijaya: Programming Machine Ethics, Springer, 1. Aufl. (2016)
- Weitere spezifische wissenschaftliche Literatur

KIM2106 Softwarearchitektur

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Georg Hagel
Dozent(en):	Prof. Dr. Georg Hagel
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Leistungsnachweis in der Übung, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzungen.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage,

- Zu erklären, welche Faktoren Einfluss auf die Architektur einer Software haben.
- Zu beschreiben, welche Architekturprinzipien beim Entwurf einer Softwarearchitektur beachtet werden müssen.
- Bekannte Architekturmuster bei der Erstellung der Softwarearchitektur anzuwenden.
- Aspekte von Softwaresystemen zu beschreiben.
- Lösungsansätze für Aspekte von Softwaresystemen anzugeben.
- Softwarearchitekturen mit aktuellen Beschreibungsmethoden zu dokumentieren.
- Continuous Integration, Continuous Deployment und DevOps zu beschreiben und anzuwenden.

Lehrinhalte:

- Aufgaben von Softwarearchitekten
- Entwurf von Softwarearchitekturen
- Architekturmuster
- Architektur Aspekte
- Dokumentation von Softwarearchitekturen
- Continuous Integration, Continuous Deployment und DevOps

Literatur:

- Richards, Mark, Ford, Neil: Handbuch moderner Softwarearchitektur, O'Reilly, 2021
- Starke, Gernot: Effektive Software-Architekturen, Carl Hanser Verlag, 9. Auflage (2020)
- Bass, Len et al.: Software Architecture in Practice, Fourth Edition, SEI Series in Software Engineering, Addison-Wesley (2021)
- Martin, Robert, C.: Clean Architecture, Prentice Hall, 2018
- Richardson, Chris: Microservices Patterns, Manning, 2019
- Oestereich, Bernd: Die UML 2.5 Kurzreferenz für die Praxis, Oldenbourg Verlag, 6. Auflage (2014)
- Buschmann, Frank et al.: POSA Band 1 - Pattern-orientierte Softwarearchitektur - Ein Pattern-System

KIM2107 Mobile Roboter

Allgemeines

Mobile Roboter sind immer noch nicht in unserem Alltag allgegenwärtig. Zwar gibt es einige wenige Beispiele wie Staubsaugroboter, Drohnen, Marserkundungsroboter, autonome Fahrzeuge oder Fußballspielende Roboter beim RoboCup, doch meist handelt es sich bisher um Nischenlösungen. Wieso ist es so schwer mobile Roboter zu bauen, die Arbeiten übernehmen können, die heute nur von Menschen ausgeführt werden können? In der Vorlesung soll ein breiter Überblick über die verschiedenen Aspekte des Themas mobile Roboter erarbeitet werden. Nachdem ein Verständnis für die vielseitigen Herausforderungen der mobilen Robotik geschaffen wird, werden wir auf geeignete Sensorik und Aktorik für Roboter eingehen. Die Organisation der Informationsverarbeitung ist in einem mobilen Roboter ein wichtiges zentrales Thema. Entsprechend werden hier verschiedene Ansätze für sog. Roboterkontrollarchitekturen vorgestellt. Ein ebenso zentrales Thema ist das Thema der Navigation: wie kann ein mobiler Roboter robust durch eine bekannte oder sogar unbekannte Umgebung navigieren? Schließlich behandeln wir das Thema des Lernens von geeigneten Aktionen für Roboter. Hier haben sich verschiedene Ansätze wie z.B. das Reinforcement-Learning und das Imitationslernen etabliert. Am Ende der Vorlesung werden wir auf Besonderheiten der kooperativen Robotik eingehen und es werden Ansätze vorgestellt, die eine Zusammenarbeit von mehreren Robotern ermöglichen sollen. In den Übungen sollen zentrale Algorithmen anhand von Robotersimulatoren vertieft werden.

Modulverantwortliche(r):	Prof. Dr. Jürgen Brauer
Dozent(en):	Prof. Dr. Jürgen Brauer
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Analysis, Lineare Algebra, Programmierkenntnisse
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	4 SWS Seminaristischer Unterricht
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht mit Praxisbeispielen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden können einschätzen, welche Fähigkeiten heute von mobilen Robotern erwartet werden können und welche nicht
- Sie wissen welche Sensorik für eine robuste Umfelderkennung notwendig ist und kennen Ansätze zur Umfelderkennung auf Basis geeigneter Sensordaten
- Sie haben fundiertes Wissen über verschiedene Ansätze von Roboterkontrollarchitekturen und kennen deren Vor- und Nachteile

- Die Studierenden kennen unterschiedliche Navigationsverfahren und können diese selber implementieren und anwenden
- Sie haben einen breiten Überblick über Verfahren die das Erlernen von Aktionen erlauben, kennen den theoretischen Hintergrund der Verfahren und können die Verfahren selber umsetzen
- Den Studierenden sind mögliche Auswirkungen der Verfügbarkeit intelligenter mobiler Roboter auf die Gesellschaft bewusst und können kritisch beurteilt werden

Lehrinhalte:

- Einführung: Geschichte der mobilen Roboter
- Herausforderungen für mobile Roboter
- Roboterkontrollarchitekturen
- Navigation, Lokalisation und Erstellung von Karten (SLAM), Pfadplanung
- Lernen von Aktionen, Reinforcement-Learning, Imitationslernen
- Kognitive Architekturen für Roboter
- Ausblick: wohin entwickelt sich der Bereich mobile Roboter?
- Mögliche Auswirkungen der Verfügbarkeit intelligenter mobiler Roboter auf die Gesellschaft

Literatur:

- Joachim Hertzberg et al. Mobile Roboter: Eine Einführung aus Sicht der Informatik Springer Vieweg Verlag. 2012.
- Roland Siegwart et al. Introduction to Autonomous Mobile Robots MIT. 2011

KIM2110 Multimodale Sensorsysteme

Allgemeines

Die Lehrveranstaltung befasst sich mit der Verarbeitung von Sensordaten - insbesondere der Fusion von Sensordaten, die aus mehreren (auch unterschiedlichen) Sensordatenquellen stammen. Dies können zum Beispiel mehrere Kamerabilder sein, die zu einer Objektdetektion fusioniert werden. Als grundlegende Techniken werden vor allem der rekursive Bayessche Schätzer als etablierter Zustandsschätzer sowie lernende Fusionsarchitekturen, wie das Convolutional Neural Network, behandelt.

Modulverantwortliche(r):	Prof. Dr. Jürgen Brauer
Dozent(en):	Prof. Dr. Jürgen Brauer
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Lineare Algebra, Programmierkenntnisse
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	4 SWS Seminaristischer Unterricht
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht mit Praxisbeispielen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden wissen wieso die Sensordatenfusion in realen multi-modalen Sensorsystemen notwendig ist und können Fusionsansätze klassifizieren
- Sie kennen und verstehen den probabilistischen Hintergrund des Zustandsschätzproblems
- Sie kennen eine konkrete Realisierung des rekursiven Bayesschen Schätzer durch den Partikel-Filter
- Sie kennen und verstehen neuere Architekturen zur Sensordatenfusion wie Convolutional Neural Networks (CNNs) und wie diese für konkrete Aufgaben wie die Objektdetektion eingesetzt werden können

Lehrinhalte:

- Warum ist Sensordatenfusion wichtig?
- Klassifikation von Fusionsansätzen
- Probabilistischer Rahmen des Zustandsschätzproblems: der rekursive Bayessche Filter
- Realisierung des rekursiven Bayesschen Schätzers über Partikel-Filter
- Repräsentation von Sensordaten in hierarchischen Fusionsarchitekturen wie CNNs zur Objektdetektion

- Representation Learning

Literatur:

- Francois Chollet. Deep Learning with Python. Manning, 2018.

KIM2111 Algorithmen und Strategien zur Entscheidungsunterstützung

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Jochen Staudacher
Dozent(en):	Prof. Dr. Jochen Staudacher
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	45 Stunden Präsenzzeit Vorlesung 15 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Die erfolgreiche Teilnahme an den Übungen gilt als Zulassungsvoraussetzung für eine 20-minütige mündliche Prüfung am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden können entscheidungs- und spieltheoretische Konzepte auf praktische Fragestellungen aus Informatik und Wirtschaftswissenschaften anwenden.
- Sie können beschreiben und analysieren, welchen Einfluss die Struktur eines sozialen oder ökonomischen Netzwerks auf strategisches Handeln besitzt.
- Sie können für ausgewählte Anwendungsfälle aus großen Datensätzen sinnvolle Information extrahieren, die zur Entscheidungsunterstützung eingesetzt werden kann.
- Sie können die vorgestellten Methoden und Algorithmen in R umsetzen.
- Sie können zu den vorgestellten Ansätzen und Algorithmen zur Entscheidungsunterstützung kritisch Stellung nehmen und deren Potenziale und Grenzen charakterisieren.

Lehrinhalte:

- Entscheidungskriterien, Entscheidungsmodelle, Entscheidungsalgorithmen
- Nutzenfunktionen, rationales Entscheiden bei Risiko, Bernoulli-Prinzip
- Interaktives Entscheiden, Algorithmische Spieltheorie und Anwendungen in Informatik und Ökonomie
- Anpassung und Erweiterung spieltheoretischer Ansätze zur Analyse ökonomischer und sozialer Netzwerke

- Ökonomien mit Netzwerkeffekten
- Statistische Entscheidungstheorie
- Datenbasiertes Entscheiden: Algorithmen und Anforderungen an die Datenqualität

Literatur:

Es gibt nicht das EINE Lehrbuch zu dieser Vorlesung; verschiedene Teile der Vorlesung werden durch die folgenden Lehrbücher abgedeckt:

- James N. Webb: Game Theory: Decisions, Interaction and Evolution, Springer, 2007
- Michael Maschler, Eilon Solan, Shmuel Zamir: Game Theory, Cambridge University Press, 2013
- Hans-Jürgen Zimmermann: Operations Research, Vieweg, 2. Auflage, 2008
- Matthew O. Jackson: Social and Economic Networks, Princeton University Press, 2008
- David Easley, Jon Kleinberg: Networks, Crowds, and Markets: Reasoning About a Highly Connected World, Cambridge University Press, 2010
- Pierre Lafaye de Micheaux, Remy Drouilhet, Benoit Liqueur: The R Software, Springer, 2014
- Larry Wasserman: All of statistics, Springer, 2004

KIM2112 Reinforcement Learning

Allgemeines

Das Modul vermittelt die theoretischen Grundlagen und praktischen Methoden des Reinforcement Learning (RL), einer zentralen Teildisziplin des maschinellen Lernens, in der lernende Agenten durch Interaktion mit ihrer Umgebung optimale Handlungsstrategien entwickeln. Zu Beginn werden die konzeptionellen und mathematischen Grundlagen eingeführt, insbesondere Markov-Entscheidungsprozesse, Belohnungsfunktionen, Zustands- und Aktionsräume sowie das zentrale Spannungsfeld zwischen Exploration und Exploitation. Darauf aufbauend werden klassische Ansätze des Reinforcement Learning behandelt, darunter Dynamic Programming, Monte-Carlo-Methoden sowie Temporal-Difference-Verfahren wie SARSA und Q-Learning. Im weiteren Verlauf des Moduls wird der Übergang zum Deep Reinforcement Learning (DRL) vollzogen, bei dem neuronale Netze als Funktionsapproximatoren eingesetzt werden, um komplexe Zustandsräume zu bewältigen. Behandelt werden zentrale DRL-Algorithmen wie Deep Q-Networks (DQN) und deren Erweiterungen (Double DQN, Dueling DQN), Policy-Gradient-Methoden (REINFORCE, A2C/A3C) sowie moderne Verfahren wie Proximal Policy Optimization (PPO). Ein besonderer Fokus liegt auf den Herausforderungen bei der Stabilisierung des Lernprozesses, der effizienten Nutzung von Erfahrungen durch Replay Buffers sowie auf der Balance zwischen Lernstabilität und Exploration. Abschließend werden ausgewählte Anwendungen und aktuelle Forschungsthemen vorgestellt, unter anderem der Einsatz von RL in Spielen (z. B. Atari, Go, AlphaZero), in der Robotik, in autonomen Systemen sowie im Multi-Agent Reinforcement Learning. Darüber hinaus werden aktuelle Herausforderungen wie Sample Efficiency, Sicherheit, Interpretierbarkeit und ethische Fragestellungen diskutiert. Begleitend zur Vorlesung vertiefen die Studierenden die Inhalte in praktischen Übungen, in denen sie beispielsweise RL-Agenten in Simulationsumgebungen (z. B. Gymnasium) implementieren, trainieren und evaluieren.

Modulverantwortliche(r):	Prof. Dr. Jürgen Brauer
Dozent(en):	Prof. Dr. Jürgen Brauer
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	
Verwendbarkeit:	
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	4 SWS Seminaristischer Unterricht
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht mit Praxisbeispielen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden erklären die grundlegenden Konzepte des Reinforcement Learning und ordnen es als Teildisziplin des maschinellen Lernens ein.

- Die Studierenden beschreiben Markov-Entscheidungsprozesse einschließlich Zustandsräumen, Aktionsräumen, Belohnungsfunktionen und Übergangsdynamiken.
- Die Studierenden erläutern das Spannungsfeld zwischen Exploration und Exploitation und dessen Bedeutung für den Lernprozess.
- Die Studierenden wenden klassische Reinforcement-Learning-Verfahren wie Dynamic Programming, Monte-Carlo-Methoden und Temporal-Difference-Lernen an.
- Die Studierenden erklären die Funktionsweise zentraler TD-Algorithmen wie SARSA und Q-Learning.
- Die Studierenden beschreiben die Motivation und Grundlagen des Deep Reinforcement Learning unter Verwendung neuronaler Netze als Funktionsapproximatoren.
- Die Studierenden erläutern Aufbau und Funktionsweise von Deep-Q-Networks sowie deren Erweiterungen wie Double DQN und Dueling DQN.
- Die Studierenden erklären Policy-Gradient-Methoden einschließlich REINFORCE, A2C/A3C und Proximal Policy Optimization.
- Die Studierenden analysieren Herausforderungen der Lernstabilisierung, der Experience-Replay-Nutzung und der Balance zwischen Stabilität und Exploration.
- Die Studierenden beurteilen ausgewählte Anwendungen von Reinforcement Learning in Spielen, Robotik, autonomen Systemen und Multi-Agenten-Szenarien.
- Die Studierenden diskutieren aktuelle Forschungsthemen und Herausforderungen wie Sample Efficiency, Sicherheit, Interpretierbarkeit und ethische Fragestellungen.
- Die Studierenden implementieren, trainieren und evaluieren Reinforcement-Learning-Agenten in Simulationsumgebungen anhand praktischer Übungen.

Lehrinhalte:

Literatur:

- Sutton, R. S. & Barto, A. G. (2018): Reinforcement Learning: An Introduction (2nd Edition). MIT Press.
- Graesser, L. & Keng, Wah Loon (2020): Foundations of Deep Reinforcement Learning. Pearson / Addison Wesley Data & Analytics Series

KIM2113 Deep Learning für Computer Vision

Allgemeines

In diesem Modul werden Deep Learning Modelle und Techniken für die Verarbeitung von Bildern vermittelt. Im Fokus hierbei steht das Convolutional Neural Network (CNN), aber auch Varianten und neuere Ansätze werden vorgestellt. CNNs stellen heute den "Kern" vieler State-of-the-Art (SOTA) Computer Vision Algorithmen dar.

Wir starten das Modul mit einer kurzen Geschichte der Bildverarbeitung. Die Herausforderungen der Bildverarbeitung und wie diese über viele Jahrzehnte hinweg systematisch unterschätzt worden sind, werden aufgeführt. Danach wird das CNN eingeführt, das heute das "Arbeitspferd" der Deep Learning basierten Bildverarbeitung darstellt und sehr oft als sog. "Feature Extractor" oder "Backbone" für viele neuere Bildverarbeitungsmodelle verwendet wird. Um das CNN Basismodell richtig zu durchdringen werden wir die einzelnen Komponenten eines CNNs (Filterschichten, Poolingschichten, Klassifikatorschicht) genauer untersuchen und die Studierenden sollen anhand von Experimenten in den Übungen mit Varianten dieser Schichten arbeiten, um eine Intuition für die Bedeutung im Rahmen der gesamten Verarbeitung eines Bildes zu entwickeln.

Es wird dann im Modul auch auf weiterführende Modelle eingegangen, die das CNN für Bildaufgaben im Kern benutzen, beispielsweise Objektdetektionsverfahren.

Modulverantwortliche(r):	Prof. Dr. Jürgen Brauer
Dozent(en):	Prof. Dr. Jürgen Brauer
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Das Modul Deep Learning ist explizit keine (!) Voraussetzung
Verwendbarkeit:	
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	4 SWS Seminaristischer Unterricht
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht mit Praxisbeispielen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden kennen die Herausforderungen bei der Bildverarbeitung, können diese durch konkrete Beispiele benennen.
- Die Teilnehmer des Moduls haben einen Überblick über die Geschichte der Bildverarbeitung bis hin zu Deep Learning basierten Modellen der Bildverarbeitung.
- Die Studierenden verstehen die Bedeutung der einzelnen Schichten eines Convolutional Neural Networks und wie die verschiedenen Schichtenarten bei der Bildklassi-

fikation zusammenspielen.

- Die Teilnehmer des Moduls kennen aktuelle Modelle zur Objektdetektion und wissen wie diese sich unterscheiden. Sie verstehen die interne Arbeitsweise und können Vor- und Nachteile der einzelnen Ansätze benennen.
- Die Studierenden kennen Modelle zur Erzeugung realistisch aussehender Bilder und können deren Funktionsweise nachvollziehen.
- Die Studierenden können einzelne ausgewählte Deep Learning Verfahren zur Bildverarbeitung mit aktuellen Deep Learning Bibliotheken umsetzen.

Lehrinhalte:

- Computer Vision - Eine kurzer geschichtlicher Rückblick
- CNN - Gesamtüberblick über das Convolutional Neural Network
- CNN - Bedeutung und Funktionsweise der einzelnen Komponenten
- Deep Learning Modelle zur Objektdetektion
- Neuere Ansätze für die Feature Extraktion wie Vision Transformer
- Generative Deep Learning Modelle um realistisch aussehende Bilder zu erzeugen

Literatur:

KIM2114 3D Computer Vision

Allgemeines

Das Modul 3D Computer Vision behandelt ausgewählte Methoden der Projektiven Geometrie, der Posenbestimmung von Objekten und der 3D-Punktwolken-Analyse.

Es werden mathematischen Grundlagen zur Lösung linearer und nichtlinearer Gleichungssysteme behandelt und im Bereich der 3D Computer Vision angewendet. Dabei spielen homogene Koordinaten und homogene Gleichungssysteme eine entscheidende Rolle. Es werden Verfahren zur Kalibration von Kamerasystemen (Single View und Multiple View) behandelt und Methoden zur Triangulation auf der Basis der "Structure From Motion" vorgestellt.

Modulverantwortliche(r):	Prof. Dr. Klaus Ulhaas
Dozent(en):	Prof. Dr. Klaus Ulhaas
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	
Verwendbarkeit:	Studiengang Master Künstliche Intelligenz und Computer Vision, Master Informatik, Master Game-Engineering und Visual Computing
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen/Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage

- 3D Punkte auf mehrere Kamerabildflächen zu projizieren
- aus Bildpunkten mehrerer Kameras entsprechende 3D Punkte zu rekonstruieren
- ein Mehrkamerasystem zu Kalibrieren
- die Epipolargeometrie zum Matching zu nutzen
- Lineare homogene Gleichungssysteme zu nutzen um unbekannte Posen-, Structure- oder Kameraparameter zu lösen
- metrische Rekonstruktionen durchzuführen

Lehrinhalte:

Das Modul 3D Computer Vision behandelt folgende Themen:

- Projektive Geometrie

- Homographie
- Kamerakalibration
- Epipolargeometrie: Essential und Fundamentalmatrix
- Triangulation und metrische Rekonstruktion
- Lineare und Nichtlineare Optimierung
- Bündelblock-Methoden (Bundle Adjustment)
- Pose Estimation
- Iterative Closest Point (ICP)
- Structure From Motion

Literatur:

Die Literatur wird zu den einzelnen Themen auf den Folien bekanntgegeben. Folgende Bücher sind zu empfehlen:

- 3D Point Cloud Analysis, Liu, Shan, Zhang, Min, Kadam, Pranav and Kuo, C. -C. Jay, Springer International Publishing, 2021
- Multiple View Geometry in Computer Vision, Hartley, Richard and Zisserman, Andrew, Cambridge University Press, Second Edition, 2003

KIM2115 Data Science

Allgemeines

Data Science ist ein interdisziplinäres Wissenschaftsfeld mit Schnittmengen zur Informatik und zur Statistik und beschäftigt sich mit der Extraktion von Wissen aus Daten. Das Ziel ist es, Erkenntnisse und Handlungsempfehlungen in konkreten Problemstellungen zu generieren und zugleich, wo möglich, das Maß der Unsicherheit in den Erkenntnissen zu quantifizieren. Angesichts der sprunghaft wachsenden Datengenerierung in allen Bereichen der modernen Gesellschaft findet Data Science vielzählige kommerzielle Einsatzmöglichkeiten, beispielsweise im Kontext Industrie 4.0, insbesondere, aber nicht ausschließlich, in Big Data Szenarien. Die Veranstaltung geht auf Grundlagen der Statistik ein, erklärt grundlegende Methoden des Maschinellen Lernens, betont die Rolle der Datenvisualisierung und des Storytellings und versucht darüberhinaus das Verständnis für typische Fallstricke und allgemeiner für die Grenzen der Dateninterpretation zu schärfen. Mathematik-Kenntnisse im Umfang der Grundvorlesungen (Summationsformeln, Integration, einfache Beweise) werden vorausgesetzt. Der Umgang mit Python und gängigen Data Science Paketen in Verbindung mit Jupyter Notebooks wird in den Übungen erklärt und einstudiert.

Modulverantwortliche(r):	Prof. Dr. Frank Schirmeier
Dozent(en):	Prof. Dr. Frank Schirmeier
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	
Verwendbarkeit:	
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Leistungsnachweise in den Übungen, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzung.
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage, in einem Unternehmen im Gespräch mit einem Mitarbeiter der Führungs- oder Fachebene eine datenbezogene Problemstellung zu analysieren, die relevanten Daten zu identifizieren, diese in einer geeigneten Infrastruktur zu erfassen, weiterzuverarbeiten und in nachfolgenden Analyseschritten Handlungsempfehlungen zu erarbeiten. Hierzu wird Python in Verbindung mit Jupyter Notebook eingesetzt. Es werden grundlegende Verfahren der Statistik und des Maschinellen Lernens beherrscht. Bei ungenügender Datenqualität oder -quantität sind die Studierenden befähigt, zu argumentieren, dass keine valide Aussage getroffen werden kann.

Lehrinhalte:

- Deskriptive Statistik (Mittelwert, Median, Pearson- und Spearman-Korrelation, Quantile, ...)
- Statistische Tests (u-Test, t-Test) und Bootstrapping
- Underfit und Overfit (Bias-Variance-Tradeoff)
- Grundlegende Methoden des Maschinellen Lernens (Dimensionsreduktion, Regression, Kernel Density-Estimation) und Anwendungen zur Signalanalyse und Anomalie-detektion
- Datenreinigung und Feature Engineering
- Projektmanagement im Data Science-Umfeld (insb. CRISP-DM)
- Python und Jupyter Notebook mit gängigen Paketen wie numpy, pandas, scikit-learn oder matplotlib

Literatur:

- Jake VanderPlas: Data Science Handbook (online verfügbar)
- Wes McKinney: Python for Data Analysis
- J. Grus: Einführung in Data Science
- S. Chan: Introduction to Probability for Data Science
- T. Hastie, R. Tibshirani, J. Friedman: The Elements of Statistical Learning
- G. James, D. Witten, T. Hastie, R. Tibshirani: An Introduction To Statistical Learning
- Christopher M. Bishop: Pattern Recognition and Machine Learning

KIM2116 Big Data

Allgemeines

Die Datenmengen, die in unserer Kommunikations- und Wissensgesellschaft anfallen wachsen unaufhörlich und die Datenvielfalt nimmt zu. Sie bieten ein riesiges Potential um mehr über uns, unsere Gesellschaft, unsere Umwelt und unsere Wirtschaft zu erfahren. Daten müssen jedoch zunächst einmal gespeichert, verarbeitet und analysiert werden. Dies stellt ganz neue Anforderungen an die zugrundeliegenden Systeme, da die klassischen Ansätze mit relationalen Datenbanken für strukturierte Daten hierfür nicht mehr geeignet sind. In Big Data Systemen werden die Daten verteilt auf viele Knoten gespeichert und verarbeitet. Die Daten selbst sind oft semistrukturiert oder unstrukturiert und müssen ggf. erst aufbereitet und gefiltert werden. Analysemethoden werden immer ausgefeilter und zunehmend ist eine Analyse in Echtzeit gefordert, die wiederum neue Technologien und Systeme erfordert.

In der Vorlesung zu Big Data werden verschiedene Problemstellungen und Lösungsansätze für Big Data Systeme diskutiert. Im Praktikum werden diese Ansätze mit einigen der derzeit leistungsfähigsten Systeme auf reale Daten angewendet.

Modulverantwortliche(r):	Prof. Nikolaus Steger
Dozent(en):	Prof. Nikolaus Steger
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Praktikum 90 Selbststudium
Leistungsnachweis und Prüfung:	Aktive Teilnahme am Praktikum. Schriftliche Prüfung 90 Minuten am Ende des Semesters. Erfolgreiche Teilnahme am Praktikum ist Zulassungsvoraussetzung für die Prüfung.
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage

- die Funktionsweise, die theoretischen Grundlagen und wichtige Algorithmen und Datenstrukturen zur Speicherung, Analyse und verteilten Verarbeitung großer Datenmengen zu verstehen
- konkrete Komponenten für Big Data Systeme zu kennen und in der Praxis einzusetzen
- komplexe Big Data Systeme zu konzipieren und aufzubauen

Lehrinhalte:

- Einsatzbereiche von Big Data Systemen
- Map-Reduce-Paradigma, Apache Hadoop, HDFS
- Batchverarbeitung mit Apache Spark, Resilient Distributed Datasets, Data Frames und Spark SQL
- Data Warehousing
- SQL-Erweiterungen für Analytics und Data Warehousing
- Column Store, Datenkomprimierung und Datenstrukturen zur Speicherung analytischer Daten
- Online Analytics und Stream Processing mit Apache Flink und Flink SQL
- Anforderungen, allgemeine Konzepte und Komponenten für Big Data Systeme

Literatur:

- Learning Spark, Lightning-Fast Data Analysis- Damji, Wenig, Das, O'Reilly and Associates

KIM2117 Differentiable Programming

Allgemeines

Einführung in Differentiable Programming als Programmierparadigma.

Modulverantwortliche(r):	Prof. Dr. Patrick Scharpfenecker
Dozent(en):	Prof. Dr. Patrick Scharpfenecker
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Analysis, Lineare Algebra, Programmierkenntnisse
Verwendbarkeit:	Studiengang Künstliche Intelligenz und Computer Vision (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht an den Übungen; schriftliche Prüfung (90 Minuten) am Ende des Semesters; erfolgreiche Teilnahme an den Übungen ist Zulassungsvoraussetzung
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden verstehen die Prinzipien von Automatic Differentiation, einschließlich Forward- und Reverse-Mode, sowie deren Implementierung in Computation Graphs (z. B. mit PyTorch). Sie beschreiben die mathematischen Grundlagen und unterscheiden zwischen ableitbaren und nicht ableitbaren Programmierkonstrukten (Schleifen, Bedingungen) in Julia und PyTorch.
- Die Studierenden können ableitbare Programme für komplexe Aufgaben implementieren. Sie wenden Differentiable Rendering-Techniken (z. B. Neural Radiance Fields/NeRFs) zur 3D-Rekonstruktion an und integrieren ableitbare Methoden in Computer Vision (mit Kornia) oder der Robotik (z. B. Trajektorienoptimierung unter physikalischen Constraints).
- Die Studierenden bewerten die Eignung von Differentiable Programming für spezifische Anwendungen.
- Die Studierenden realisieren (ggfs. in Teams) eine End-to-End Anwendung mit Differentiable Programming und nutzen geeignete Hardware-Ressourcen effizient.

Lehrinhalte:

- Backpropagation und Automatic Differentiation
- Differenzierbare Programme/Primitive (Mathematische Operationen, Schleifen, ...)
- Differentiable Programming in Pytorch und/oder Zygote (Julia)

- Ableitungen höherer Ordnung
- Anwendungen wie z.B.:
 - Ableitbare ODEs (ODENets)
 - Differentiable Rendering (NeRFs)
 - Differentiable Computer Vision (Kornia)
 - Differentiable Robotics (Dojo)
 - Differentiable Physics (DiffTaichi)

Literatur:

- Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep Learning. MIT Press, 2016. <http://www.deeplearningbook.org>.
- Brunton, Steven L., and J. Nathan Kutz. Data-driven science and engineering: Machine learning, dynamical systems, and control. Cambridge University Press, 2022.
- Zhang, Aston and Lipton, Zachary C. and Li, Mu and Smola, Alexander. Dive into Deep Learning. Cambridge University Press, 2023. <https://D2L.ai>.
- Griewank, Andreas, and Andrea Walther. Evaluating derivatives: principles and techniques of algorithmic differentiation. Society for industrial and applied mathematics, 2008.
- Naumann, Uwe. The art of differentiating computer programs: an introduction to algorithmic differentiation. Society for Industrial and Applied Mathematics, 2011.

KIM2118 Funktionale Sicherheit

Allgemeines

Einführung in Funktionale Sicherheit mit besonderer Beachtung der Anforderungen von Informatik, Fahrerassistenzsystemen und Automatisierung & Robotik.

Modulverantwortliche(r):	Prof. Dr. Rolf Jung
Dozent(en):	Prof. Dr. Rolf Jung
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	Übung/Praktikum findet während des Semesters in Blöcken statt.
Leistungspunkte:	5
Arbeitsaufwand:	45 Stunden Präsenzzeit Unterricht 15 Stunden Präsenzzeit Praktikum/Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	Benutzung von Teilen der Normen ISO 26262 und ISO 13849 nach Vorgabe, Skript (Ausdruck mit eigenen Notizen), nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

- Studenten kennen den Begriff 'Funktionale Sicherheit' und damit verbundene Definitionen aus den Normen
- Teilnehmer kennen die Elemente des Sicherheitsprozesses und können sie dem Entwicklungsprozess zuordnen
- Sie kennen Managementmethoden und können diese dem Sicherheitslebenszyklus zuordnen
- Studenten können die Methode der HARA anwenden
- Teilnehmer können zwischen reparierbaren und nicht reparierbaren Elementen unterscheiden und die Funktionen Unzuverlässigkeit und Nichtverfügbarkeit technischer Elemente erklären
- Studenten kennen die unterschiedlichen logischen Operatoren der Wahrscheinlichkeitskombinatorik und kennen die Modellbildung in der FTA bzw. anderen Sicherheitsanalysen
- Studenten können die Methode FMEDA im Sicherheitsprozess einordnen sowie Sicherheitskennzahlen bestimmen und anwenden
- Teilnehmer kennen Testmethoden bezüglich System, Hardware, Software und können aus System- und Produktanforderungen Testanforderungen erstellen
- Studenten können die Anforderungen der Norm ISO26262 zum Produktionsprozess

im Sicherheitslebenszyklus anwenden

- Sie kennen den Begriff der Zugangssicherheit im Kontext von Funktionaler Sicherheit
- Studenten können Softwarewerkzeuge klassifizieren
- Die Studenten kennen einen Zulassungsprozess und können Dokumente für eine Sicherheitszulassung zusammenstellen

Lehrinhalte:

- Einführung in die Gesetzeslage und Normenüberblick mit Begriffserklärungen
- Erläuterung der Elemente eines Sicherheitsprozesses und Erstellung eines Sicherheitsplans und Sicherheitsnachweises
- Analyse der Elemente und Schritte in einem Sicherheitslebenszyklus
- Durchführung einer Risiko- und Gefährdungsanalyse
- Ermittlung von Sicherheitskennzahlen an einem Fallbeispiel
- Einführung in mathematischen Methoden der Zuverlässigkeitstheorie und Durchführung einer Sicherheitsanalyse für Fehlerkombinationen
- Erarbeitung einer FMEDA mit Fallbeispiel
- Management der Funktionalen Sicherheit
- Kennenlernen von Testmethoden und Erstellen eines V&V-Plans
- Einführung in Zugangssicherheit, Cybersecurity in Verbindung mit Funktionaler Sicherheit
- Klassifizierung von Softwarewerkzeugen entsprechend Sicherheitsnormen

Literatur:

- Josef Börcsök, Funktionale Sicherheit: Grundzüge sicherheitstechnischer Systeme, VDE Verlag 2014
- Löw, Pabst, Petry: Funktionale Sicherheit in der Praxis, dpunkt.Verlag
- Ross, Hans-Leo: Funktionale Sicherheit im Automobil, Hanser Verlag

KIM2119 Large Language Models

Allgemeines

Seit Einführung von ChatGPT erfreuen sich große Sprachmodelle großer Beliebtheit mit großem Anwendungspotential im privaten wie im industriellen Umfeld. Zugrunde liegt ein autoregressives Transformermodell, das statistische Gesetzmäßigkeiten auf großen Textbeständen lernt und reproduziert. Die Vorlesung führt ein in die mathematischen und technischen Grundlagen und zeigt mögliche Anwendungsfelder auf. Es werden Architekturelemente mit Python-Code nachvollzogen, daher werden Grundkenntnisse in Mathematik und Machine Learning vorausgesetzt und eine Vertrautheit mit Python und Jupyter Notebook empfohlen. Das Modul wird angeboten von Dr. Fabio Lischka und Christoph Wald, wissenschaftliche Mitarbeiter am Forschungszentrum Allgäu.

Modulverantwortliche(r):	Prof. Dr. Frank Schirmeier
Dozent(en):	Prof. Dr. Frank Schirmeier / N.N. Lehrbeauftragter / N.N. Lehrbeauftragter
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	
Verwendbarkeit:	
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Leistungsnachweise in den Übungen, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzung.
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage, unterschiedliche Architekturen von Large Language Models auf mathematische Formeln zu reduzieren und zu vergleichen. Sie sind in der Lage, ein einfaches Large Language Model in Python zu implementieren, Textdaten für das Training vorzuverarbeiten, und ein solches Training durchzuführen. Die Studenten erkennen Probleme in der Qualität der Trainingsdaten und können Gegenmaßnahmen zu ergreifen. Sie sind in der Lage, die grundlegenden mathematischen Modellannahmen zu erklären, auf denen Large Language Models aufbauen, und daraus Vorhersagen über deren Verhalten abzuleiten. Sie können LLMs in größere Anwendungen integrieren und wenden grundlegende Prinzipien bei der Einbettung von LLMs (LLMOps) an.

Lehrinhalte:

Inhalte zu nachfolgenden Themenfeldern werden bearbeitet:

- Natural Language Processing
- Architekturelemente von LLMs, insb. Transformer-Architektur
- Bereinigung von Textdaten
- Trainingsstrategien und Hyperparameter-Tuning für LLMs
- Instruction Tuning
- Prinzipien zur Anwendung von LLMs ("LLMOps")

Literatur:

Das Modul stützt sich neben aktuellen Veröffentlichungen u.a. auf nachfolgende Literatur:

- Chaumont et al.: LLM Engineer's Handbook
- Gazit et al.: Mastering NLP from Foundations to LLMs
- Raschka: Build a Large Language Model (from Scratch)
- Alto: Building LLM Powered Applications Create Intelligent Apps and Agents with Large Language Models

KIM5100 Projekt

Allgemeines

Modulverantwortliche(r):	Professoren der Fakultät
Dozent(en):	Professoren der Fakultät
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Künstliche Intelligenz und Computer Vision (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	Projekt
Leistungspunkte:	10
Arbeitsaufwand:	15 Stunden Präsenzzeit Unterricht 285 Stunden selbständiges Arbeiten
Leistungsnachweis und Prüfung:	Projektbericht Präsentation
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage,

- ein Projekt zu planen, zu organisieren, zu leiten und gegenüber Dritten zu vertreten
- Probleme im Projekt frühzeitig zu erkennen und zu lösen
- ein Projekt erfolgreich abzuschließen
- das erworbene Wissen auch in großen, fachlich komplexen, forschungsnahen Anwendungsszenarien einzusetzen und erfolgreich anzuwenden
- Methoden der künstlichen Intelligenz und Computer Vision zu realisieren und ihren Einsatz zu reflektieren und zu vertreten
- sich schnell in neue Anwendungsgebiete, Technologien und Grundlagen einzuarbeiten

Lehrinhalte:

projektabhängig

Literatur:

projektabhängig

KIM6101 Masterarbeit

Allgemeines

Modulverantwortliche(r):	Betreuender Professor
Dozent(en):	Betreuender Professor
Modultyp:	Pflichtmodul
Voraussetzungen:	Siehe Studien- und Prüfungsordnung
Verwendbarkeit:	Studiengang Künstliche Intelligenz und Computer Vision (M.Sc.)
Angebot und Dauer:	jedes Semester
Lehrformen:	
Leistungspunkte:	30
Arbeitsaufwand:	900 Stunden
Leistungsnachweis und Prüfung:	Abschlussarbeit
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Mit der Masterarbeit beweisen die Studierenden, dass sie komplexe, forschungsnahe Problemstellungen unter Einsatz der im Studium erworbenen Kenntnisse, Fertigkeiten und Kompetenzen mit wissenschaftlichen Methoden zu einer praktisch verwendbaren, konsistenten und überzeugenden Lösung führen können. Die Abschlussarbeit darf mit Zustimmung der Prüfungskommission in einer Einrichtung außerhalb der Hochschule ausgeführt werden.

Lehrinhalte:

Literatur: