|  |  |
| --- | --- |
| Seminarthema: | [Implementierung des neuronalen Netzes](https://schule.informatik.uni-rostock.de/course/view.php?id=2#section-4) |
| Zeitaufwand: | 3 – 5 Woche (270 min) |
| Jahrgangsstufe: | 12 |
| Kursangebot: | Grundkurs & optionale Hinweise für den Leistungskurs |
| Voraussetzungen: | * Ziele der vorherigen Woche * Structorizer - Werkzeug zur Erstellung von Struktogrammen * Informatikraum (max. 2-3 SuS/Lernplatz), Programmierumgebung (Spyder, PyCharm, THONNY o.ä.), * TensorFlow, numpy, MNIST-Datensätze jeweils als Import (Import-Skript) |
| Ziele: | * Modelle in einer objektorientierten Programmiersprache (hier Python) zu analysieren und zu modifizieren * Klassendefinitionen sowie die Funktionalität von Methoden zu interpretieren und zu implementieren * eine grafische Oberfläche mit Beschriftungselementen, Schaltflächen, Eingabefeldern einschließlich der Fachklassenanbindung zu analysieren * Wichtigkeit der Trainingsdaten/Testdaten für die KI zu erläutern |
| Methodische Überlegungen: | * Implementieren einer Klasse anhand bereits implementierter Klassen * Anlegen von Dateien und Ordnern * Einlesen in den Code, die Klassen * Unklarheiten beseitigen * Coden/Testen * Abhängig von der Anzahl der Computer Einzel- oder Paararbeit |
| Motivation, Hilfen, Impulse | * Hilfestellungen/Tipps sind in den Aufgaben und können von der Lehrkraft nach belieben hinzugefügt oder weggelassen werden |

| Zeitbudget | Inhalte | Methodik | Material |
| --- | --- | --- | --- |
| 15 min | * **Einführung** | * Übergang vom UML-Diagramm zur Entwicklung * Den Schüler\*innen die benötigten Dateien zur Verfügung stellen. * Den Schüler\*innen das 1. Aufgabenblatt geben und Unklarheiten besprechen. | * Dateien aus dem Git: <https://github.com/Schupax/KI-OCR-Schulprojekt> * 1. Aufgabenblatt * Beamer |
| 45 - 60 min | * **Selbstständiges Arbeiten der Schüler\*innen am 1.Aufgabenblatt** * Layer-Klasse implementieren * Methoden und Validierung implementieren * Vor- & Nachteile der Vererbung und Aggregation erklären | * Einzelarbeit der Schüler\*innen sollte Paararbeit vorgezogen werden * Individuelle Hilfestellung/Betreuung bei den Aufgaben | * 1. Aufgabenblatt * Beamer * <https://github.com/Schupax/KI-OCR-Schulprojekt/blob/main/Vorlagen%26Aufgaben/Layer.py> |
| 15-30 min | * **Auswertung des 1. Aufgabenblattes** | * Einzelne Ergebnisse/Lösungen zu den Aufgaben von den Schüler\*innen vorstellen lassen und vergleichen * Vor- & Nachteile der Vererbung und Aggregation auswerten | * Beamer * Lösungen der Schüler\*innen |
| 120 min | * **Selbstständiges Arbeiten der Schüler\*innen am 2.Aufgabenblatt** * Fehleranalyse/ Algorithmus korrigieren * Algorithmus erstellen & implementieren | * Den Schüler\*innen das 2. Aufgabenblatt geben und Unklarheiten besprechen. * Einzelarbeit * Aufgaben mit Struktogramm können auch auf dem Papier gelöst werden * Auswerten der ersten Aufgabe nach den ersten 30-45 Minuten | * 2.Aufgabenblatt * Structorizer (optional) * <https://github.com/Schupax/KI-OCR-Schulprojekt/tree/main/Vorlagen%26Aufgaben> * Die Vorlagen-Struktogramme für die Schüler\*innen (siehe Link) |
| 180 min | * Implementierung der kompletten Konsolenanwendung als Tochterklasse von View | * Freihand lassen bei der Entwicklung * Individuelle Hilfestellung |  |