Ausarbeitung – Interaktives Lern-Tool für neuronale Netze

# Arbeitsaufteilung

|  |  |  |
| --- | --- | --- |
| Phase | Patrick Kaserer | Lukas Willmann |
| Analysieren | Ausarbeitung Idee | Ausarbeitung Idee |
|  | Erstellung Personas |
|  | Erstellung Szenario |
| Interpretieren | Recherche nach Tools | Erstellung Figma Kontext |
|  | Testen NN (neuronales Netz) from Scratch |  |
| Kreieren | Tests mit ThreeJS (Kreise und Linien | Erstellung Wheel und Slider Interaktion mit ThreeJS |
|  | Tests Tensorflow |  |
|  | Ergänzen von Infos bei Zwischenpräsentation | Umsetzung Zwischenpräsentation |
|  | Dynamisches Erzeugen von Layers + Tensorflow | Erzeugung TikTakToe und Studentenwohnheim Datensatz |
|  | Tests-NN TikTakToe Problem | Tests-NN TikTakToe Problem + Tests Studentenwohnheim |
|  | Gestaltung von Prototypen für Experteninterview | Umsetzung eines Prototyps für das Experteninterview (Interaktion + Studenten NN) |
|  | Debuggen Prototyp bei Generierung von Ergebnissen in Output-Layer | Erzeugung von Ergebnissen, nachdem neuer Input ausgewählt wurde |
| Bewerten | Vorbereitung + Durchführung von Experteninterview | Erstellung Fragebogen für Interview + Anfragen |
| Analysieren | Auswertung Interview |  |
| Interpretieren | Planung aus dem Ergebnis des Interviews |  |
| Kreieren | Vorbereiten des gemeinsamen Standes für die Einarbeitung der Ergebnisse des Interaktionsprototypen | Einarbeitung der Funktionen des Interatkionsprototypen |
| Bewerten | Durführung des AB-Tests | Durchführung des AB-Tests |

# Idee: Patrick und Lukas

Wir haben gemeinsam eine Idee evaluiert, die darauf abzielt, ein Tool zu entwickeln, mit dem Lernende durch einen konstruktivistischen Lernansatz ein neuronales Netzwerk durch Interaktion kennenlernen können. Es gibt bereits ähnliche Tools ([Interaktiv: Neuronales Netz – Hart und Trocken](https://www.hartundtrocken.de/my-product/interaktiv-neuronales-netz/)), jedoch keines, bei dem die internen Parameter des Netzwerks angepasst werden können. Die Ausarbeitung der Idee und die Entwicklung der Anwendung wurden von uns gemeinsam erarbeitet und vorangetrieben. Eine strikte Trennung der Arbeitsteilung war hier nicht möglich. Trotzdem gab es in bestimmten Bereichen Schwerpunkte, die im Folgenden aufgezeigt werden.

## Fragestellung

Welchen Einfluss haben unterschiedliche Benutzerschnittstellen auf die Effizienz und Benutzerfreundlichkeit bei der Anpassung von Gewichtungen in XAI-Anwendungen? Ein Vergleich zwischen der Verwendung eines einfachen Sliders und der direkten Manipulation von Neuronen und deren Verbindungen mittels Mausklick und Mausrad.

# Arbeitsbereich Lukas Willmann

## Analysieren

Für den ersten Abschnitt habe ich vier Personas erstellt. ([ID\_XAI\_TS/Documents/Personas at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/tree/main/Documents/Personas)) Die Personas bilden lernwillige Personen ab, welche sich fortbilden wollen. Anhand dieser Personas erstellte ich ein Szenario für eine der Personas in einem möglichen Studenten-Alltag. ([ID\_XAI\_TS/Documents/Szenario at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/tree/main/Documents/Szenario))

## Interpretieren

Mithilfe der Personas konnte ein „klickbarer“-Figma Prototyp erstellt werden, welcher für uns den Use-Case verdeutlichte. ([NN\_Explainer – Figma](https://www.figma.com/file/IPAPgOgXqxD9lLfVlNVUVv/NN_Explainer?type=design&node-id=0-1&mode=design)) Lernende würden an einem Desktop-PC auf einer Website Informationen über Neuronale Netze in verschiedenen Kapitel abrufen können. Auf dieser Website wäre das interaktive Tool als eigenes Kapitel oder am Ende eines passenden Kapitels eingebettet.

In unserem Projekt konzentrierten wir uns auf die Umsetzung des Tools. Nicht auf die komplette Umsetzung der Website mit dem Tool, da dies den Rahmen für uns gesprengt hätte. Dieser Figma-Kontext half uns aber das Tool auf bestimmte Weise zu entwerfen, wie z.B. die Beachtung von verschiedenen Desktop-Größen (Kamera-Handling in der Szene), jedoch nicht die Auslegung für Mobile, da wir davon ausgehen, dass die Nutzer für eine konzentrierte Lernatmosphäre eher an Desktop-PCs sitzen mit einer Scroll-Möglichkeit (Maus oder Touchpad eines Laptops).

## Kreieren

Damit die Kamera zentral auf die Layers gerichtet ist in Abhängigkeit von der Anzahl der Neuronen wie viele in einem Layer sind und wie viele Layer es gibt habe ich eine Kamera-Steuerungsklasse erstellt, mit welcher das neuronale Netz zentriert im Bild bleibt. ([ID\_XAI\_TS/src/classes/CameraHandler.ts at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/blob/main/src/classes/CameraHandler.ts))

Um die Kommunikation zwischen unseren Programmierteilen zu verbessern habe ich ein Klassendiagramm erzeugt. ([ID\_XAI\_TS/Documents/interactUML.drawio at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/blob/main/Documents/interactUML.drawio))

Die Interaktionen des Nutzers werden über die Klassen in der Interaktions-Lib gesteuert. ([ID\_XAI\_TS/src/classes/InteractionLib at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/tree/main/src/classes/InteractionLib)) Durch einen Parameter in der „InteractionFactory“ ([ID\_XAI\_TS/src/classes/InteractionLib/InteractionFactory.ts at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/blob/main/src/classes/InteractionLib/InteractionFactory.ts)) kann gesteuert werden, ob deine „Wheel“ oder eine „Slider“ Interaktion erzeugt werden soll. Diese Klassen wurden zuerst in einem JavaScript Projekt erstellt. Bei der Umstellung auf TypeScript wurden die Klassen aber angepasst.

Um Datensätze zu haben mit welchen wir NNs trainieren konnten erzeugte ich einmal einen TikTakToe-Datensatz und einen Wohnungen-Datensatz. Diese wurden durch Brute-Force-Python Scripte erzeugt. Bei dem TikTakToe Datensatz wurde viel mit Zufalls-Zahlen gearbeitet und in dem Datensatz der Wohnungen wurden logische Regeln und ebenfalls Zufallszahlen verwendet um die Daten zu generieren und zu labeln. ([ID\_XAI\_TS/DataSets at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/tree/main/DataSets))

Um die NNs zu testen welche aus den Datensätzen entstehen können habe ich mit Jupyter-Notebooks gearbeitet. ([ID\_XAI\_TS/DataSets/JupyterTests at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/tree/main/DataSets/JupyterTests))

Da die Kombination zwischen unserem Hauptprojekt und dem Projekt an welchem ich mit den Interaktion gearbeitet habe zu komplex war um in der Zeit bis zur Endpräsentation zu kombinieren, habe ich einen ersten testbaren Prototyp in meinem Projekt erzeugt. Das Modell ist in seinen Layern, Neuronen und Output Möglichkeiten statisch. Das Modell wird beim Start 200 mal mit dem Wohnungs-Datensatz trainiert. Danach kann der Tester die beiden Interaktionen unter Anleitung testen. [LksWllmnn/NeuroInteract: Interaction with a small Neuronal Network (github.com)](https://github.com/LksWllmnn/NeuroInteract)

## Bewerten

Um unsere wissenschaftliche Fragen für das Projekt zu bearbeiten habe ich einen einfachen Fragebogen erstellt. ([ID\_XAI\_TS/Documents/Fragebogen Interaktion der Anwendung KI.pdf at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/blob/main/Documents/Fragebogen%20%20Interaktion%20der%20Anwendung%20KI.pdf)) Im ersten Teil des Fragebogens gibt es qualitative Fragen zur Nutzung des Prototypen. Hierbei war die Intuition, dass wir ein genaueres Verständnis bekommen welche Interaktion für NNs angenehmer ist und nicht nur ob eine Testperson das Mausrad oder einen Slider lieber mag. Im zweiten Teil haben wir den wissenschaftlichen „SUS“-Fragebogen verwendet um einen wissenschaftlich Bewährten Fragebogen für unsere Evaluation heran ziehen zu können. ([(PDF) SUS: A quick and dirty usability scale (researchgate.net)](https://www.researchgate.net/publication/228593520_SUS_A_quick_and_dirty_usability_scale))

# Arbeitsbereich Patrick Kaserer

## Interpretieren

Für die Umsetzung der Anwendung wurde eine Recherche durchgeführt, um passende Werkzeuge zu finden. Obwohl anfangs die Idee bestand, die Inhalte von Grund auf selbst zu entwickeln, wurde von diesem Vorhaben recht schnell Abstand genommen. Für die Umsetzung wurden mehrere Repositories herangezogen. Das erste war eine reine JavaScript-Umsetzung (<https://github.com/Z3r0cks/Interaktionsdesign_XAI>), in deren Code auch Ansätze zu finden sind, wie die Entwicklung ohne Libraries hätte erfolgen können (z.B. die Datei [src/simple\_NN.js](https://github.com/Z3r0cks/Interaktionsdesign_XAI/blob/main/src/simple_NN.js#L61)). Hierbei war wichtig, dass die Anwendung möglichst einfach gestaltet wird und kein Wert auf hohe Komplexität gelegt wird. Ziel war es, ein einfaches neuronales Netz zu schaffen, das die XAI-Anwendung mit einem einfachen Modell repräsentiert.

## Kreieren

Durch den zu hohen Aufwand entschieden wir uns auf das Tool TensorFlow.js (TsJs) zu wechseln, was die Implementation einfacher gestalten sollte. Da das erste Repository als ein Testdurchlauf angelegt wurde, entschieden wir uns daraufhin für eine strukturiertere Variante mit TsJs und Typescript. Der erste richtige Prototyp war [dieser Prototyp](https://github.com/Z3r0cks/ID_XAI_TS). Hierbei wurde stark auf die Lesbarkeit des Codes geachtet, und auf die Modulierung der Inhalte mit Klassen. Dieser Prototyp mit der Bezeichnung „Dynamischer Prototyp“ ist dafür da, die Eingabe im Front-End zu repräsentieren, bei dem dynamisch die Anzahl der Layer und Neuronen bestimmt werden kann. Im Hintergrund ist TsJs integriert, bei dem ein TensorFlow-Model erstellt wird, sobald der entsprechende Button gedrückt wurde. Dieser Prototyp sollte die Hauptanwendung darstellen, bei dem alle Inhalte im Nachhinein implementiert werden.

Wir stellten fest, dass wir für unsere Anwendung auch ein einfaches mathematisches Problem benötigen, auf das das Modell approximiert werden kann, um als Beispiel für die XAI-Anwendung zu dienen. Die Idee, die dabei entstand, war ein TicTacToe-Spiel mit allen möglichen Ausgängen zu entwerfen und das Modell darauf zu trainieren. Trotz mehrerer Versuche und intensiver Arbeit beim Experimentieren und Debuggen konnte das Modell nicht zufriedenstellend abgeschlossen werden. Auch beim nachfolgenden Modell, dem "Studentenwohnheim", war weiteres Debugging erforderlich.

## Bewerten

Für das Experteninterview musste entsprechende Formulare erstellt und vorbereitet werden. Anschließend wurde das Experteninterview mit Prof. Dr. Lasowski durchgeführt. Hier wurden die beiden Interaktionsmöglichkeiten (Slider, Mausrad) in der Anwendung von Lukas Willmann prototypisch implementiert (<https://github.com/LksWllmnn/NeuroInteract>). Hierfür wurde zur Vorbereitung noch Veränderungen an der Anwendung durchgeführt, um den Test auf die Nutzerinteraktion anzupassen und das Hauptaugenmerk auf die Fragenstellung zu lenken.

Beim Expertentest wurde die Anwendung im Vornherein erläutert und zum Testen der beiden Eingabebereiche übergeben. Anschließend wurden zwei Fragebogen ausgefüllt und ein Interview über die Anwendung und die Eingaben durchgeführt, um ein ausführliches Ergebnis zu erhalten.

## Analysieren

Frau Lasowski, eine renommierte Expertin auf dem Gebiet der Künstlichen Intelligenz und Pädagogik, besonderes Augenmerk auf die Notwendigkeit multifunktionaler Eingabemethoden, betonte die Überlegenheit von Slider-basierten Interfaces für mobile Plattformen in ihrer Anwendbarkeit und Benutzerfreundlichkeit. Frau Lasowski, die das Tool selbst im Bildungskontext einsetzen würde, hob wiederholt die Bedeutung des "Playground"-Features hervor, welches dazu dient, komplexe Themen intuitiver zu vermitteln. Sie präferierte die Mausradeingabe mehr und empfand sie als intuitiver und einfacher zu benutzen.

Trotz der Anerkennung der Benutzerfreundlichkeit, kritisierte sie die Gestaltung des Sliders, insbesondere im Hinblick auf dessen Klarheit und Verständlichkeit. Sie merkte an, dass es schwierig sei, die korrespondierenden Werte den spezifischen Neuronen zuzuordnen. Eine visuelle Darstellung dieser Werte direkt an den Verbindungen könnte ihrer Meinung nach zur Verbesserung beitragen.

Des Weiteren empfahl Frau Lasowski, die Implementation von extremen Fallbeispielen, insbesondere zur Veranschaulichung von Overfitting, einem häufigen Problem in der Modellbildung. Sie schlug vor, dass die Dicke der Verbindungen im Tool proportional zu den Gewichtswerten der neuronalen Netzwerkmodelle sein könnte, um eine intuitivere und effektivere Visualisierung zu ermöglichen.

# Ausblick

Weitere Schritte der Anwendung wäre, die Analyse von Frau Dr. Laskowski in der Anwendung umzusetzen und anschließend die Eingabemöglichkeiten in das Hauptprojekt zu implementieren. Anschließend wäre ein AB-Test für eine weitere Iteration notwendig, um die Anwendung zu Evaluieren und in die Anwendung entsprechend zu integrieren.