VeduNet

Erarbeitungsort: 91710 Gunzenhausen

Projektbetreuer/in: Dr. –Ing Kiesmüller

Thema des Projekts: Deep Learning mit Fokus auf Bildung und Vermittlung

Fachgebiet: Informatik/Mathematik

Wettbewerbssparte: Jugend forscht

Bundesland: Bayer

Wettbewerbsjahr: 2021

Wie erkenne ich, ob ich mit der Wand rede oder mit einem Menschen? Woher weiß ich, wo das Gesicht meines Gesprächspartners ist und wie kann ich verstehen, was er meint bzw. was er sagt?

Mit dieser Frage befassen sich Informatiker seit Jahrezenten (McCorduck 1979) mit dem Ziel, die Lebensqualität der Menschheit zu verbessern. Lange Zeit waren nur Informatiker in diesem Thema eingeweiht, doch seit der Entdeckung des Gradientverfahrens gab es nicht nur eine unvorstellbare Explosion in dem Entwicklungsstand (und Entwicklungsgeschwindigkeit) von Künstlichen Netzen, sondern es wurden Künstliche Intelligenzen ein extrem mathematisches Teilgebiet der Informatik. Mathematik ist heutzutage in KIs so relevant, dass es von vielen Experten und Hochschulprofessoren als rein mathematisch-logisches Erzeugnis gesehen wird.

Genau das machte mich so heiß darauf, das Ganze zu lernen und somit zu begreifen. Einerseits trieb mich die Neugier an, zu verstehen, wie Bilderkennungen (sog. Image classification) etc. funktionieren und programmiert werden. Andererseits war die Herausforderung der Mathematik, die mich, als Schülerstudent der FernUni Hagen, schon immer begeistert hat, ein enormen Ansporn für das gesamte Projekt. Also setzte ich mich sofort am nächsten Tag dran und fing an nach „Künstlichen Intelligenzen“ zu googlen, worauf mich eine schier unbegrenzte Menge an Unterteilbereichen überrollte. Markov-Ketten, Neuronale Netze, Convolutional Neornal Network, Deep reinforcement learning etc. sind nur eine Hand voll der ganzen Unterthemen in in Wikipedia/Google zu finden sind. Eine richtige Wahl hatte ich nicht, so fing ich erstmal an die ganzen Themen zu überfliegen und zu schauen, welche Themen mich am meisten ansprechen würden und welche auch die zukunftrellevantesten Themen wären. Recht schnell fand ich große Begeisterung für Neornale Netze (egal ob faltend oder normal) und Deep reinforcement learning, die mich aber auch sehr schnell, von meinen Kapazitäten her, auslasteten. Mit dem YoutubeKanal 3Blue1Brown fing ich also an mit alles anzuschauen und schnell begriff ich die Arbeitsweisen der Systeme, was mich dazu veranlasste die Themen mit Literatur zu vertiefen.

Das ganze führte mich dann zu einer großen Problematik: Da das Thema sehr frisch und neu ist, gibt es kaum OnlineSeminare, Simulations-Programme und Apis, zum selberbauen und austesten der Systeme, womit man die Arbeitsweisen und Algorithmen ausprobieren und begriefen kann. So dachte ich mich: „Wie kann man denn einem Uneingeweihten diese Teilbereiche der künstlichen Intelligenz am besten Erklären, ohne zu mathematisch oder vorkenntnisslastig zu werdne?“ - „Wie wäre es mit einem Drag n‘ Drop Sandbox API/Programm welches dir alle Zwischenschritte visualisiert. Somit könnte man die Algorithmen, Arbeitsweisen und Prinzipien schneller,besser und effektiver begreifen“

So wurde das Projekt geboren. Das Ziel ist es, ein (wie oben bereits erwähnt) Drag n‘ Drop Sandbox API/Programm zu machen, womit jeder auf einem Blick begreifen kann, was in einer Künstlichen Intelligenz vor sich geht. Um künstliche Intellignzen der Art zu vereinfachen, aber immer noch realistisch zu halten, muss man wahrscheinlich aber auch komplett anders an die Vorstellung eines Netzes ran gehen, aber das wird sich noch im Laufe der Projektarbeit ergeben…

Zu aller erst ist es empfehlenswert der Art komplexe Projekte in kleinen Unterprojekten runter zu brechen und sich denen separiert zu widmen. Neuronale Netze sind auf dem ersten Blick simple aufgebaut, jedoch gibt es ein paar Punkt die zu den schwierigsten Stellen im ganzen Projekt gehören. Diese Stellen sind:

* Backpropagation
* Convolutional Layers - Multidimenasionale Strukturenmultiplikation

Zuerst widmete ich mich der Backprogation. Neuronale Netze lernen dadurch, dass sie anhand von Faktoren/Konstanten (sog. Wichtungen) und den Abweichungen von ihrem Ergebnis zu dem wahren Ergebnis, neue Faktoren/Konstanten bilden, welche sie dann für die nächste Praxisanwendung anwenden. Es gibt viele Algorithmen die dafür in Frage kämen, doch nach lesen zahlreicher Bücher kann ich sagen, dass es 2 gängige Algorithmen gibt die diese Aufgabe übernehmen könnten.