

Maturarbeit - Konzept

Titel (Änderbar): Chess Master – Erstellen von KI-Gegnern mit Deep Learning

Leitfrage: Wie realistisch kann man die mit Deep-Learning trainierten Bots auf verschiedene Schwierigkeitsgradstufen anpassen?

In meiner Maturarbeit will ich als Produkt ein funktionierendes Schachspiel haben, welches KI-Gegner beinhaltet, gegen die man schlussendlich spielen kann. In meiner Maturarbeit will ich untersuchen, ob es mir gelingt, eine KI zu kreieren, die sich in bestimmten Niveaus anpassen kann.

Dazu teile ich die Maturarbeit in folgende Phasen ein;

PHASE 0 - Verständnis

In dieser Phase will ich sicherstellen, dass ich die Grundlagen von Deep Learning verstehe und beherrsche. Dies beinhaltet:

- Visuelles verstehen von neuronalen Netzwerken
- Mathematisches Verständnis von neuronalen Netzwerken
- Detailliertes Verständnis in CNN's
- Pytorch lernen / numpy
- Evtl. Erste einfache Anwendungen von deep Learning

**Phase 0 könnte schon in den Sommerferien absolviert werden*

PHASE 1 – Sammeln der Daten

Damit die KI lernen kann braucht sie zuerst die entsprechenden Datasets. Dazu habe ich folgende csv-File gefunden:

<https://www.kaggle.com/datasets/arevel/chess-games>

Diese csv-File enthält über 6mio Schach Partien auf verschiedenen Niveaus. Falls mehr spiele benötigt werden (von tieferen Niveaus hat es eher wenig daten), können diese noch aufgesucht bzw. selbst erstellt werden (selber spielen oder von Freunden sammeln)

PHASE 2 – Schachspiel

Da ich meinen Fokus eher auf das Trainieren der KI's lege, würde ich gerne das bereits existierende «python-chess» in meine Arbeit implementieren, und so auf das Erstellen der Spielregeln verzichten. Jedoch müsste Folgendes selber gemacht werden:

- Schachbrett
- Spielfiguren
- Einzelnen Züge
- Anzeige Schachmatt

(Alles, was python-chess nicht enthält, was jedoch essenziell ist)

PHASE 3: trainieren der 3 KI's (erstellen der CNN's)

Diese Phase beinhaltet das Trainieren der 3 einzelnen KI's. Dies würde gerne auf pytorch machen. Ich habe mir tensorflow angeschaut, jedoch bin ich mehr überzeugt von pytorch.

Inspiriert habe ich mich von folgendem Video;

<https://www.youtube.com/watch?v=aOwvRvTPQrs&t=850s>

Ich würde auch versuchen die entsprechenden Neuralen Netzwerke zu bilden, jedoch angepasst auf die entsprechenden Niveaus; (Bsp. Chess data = chess_data_raw ['WHITEELO' > 1000]). Ich würde gerne 3 verschiedene KIs trainieren, eine auf Anfängerstufe (0-700), Fortgeschritten (700-1500), und Profi-Niveau (1500-2000)

In der Theorie können für alle Bots dieselben CNN's benutzt werden, der Unterschied liegt lediglich darin, dass sie mit anderen Daten «gefüttert» werden. Die Vorgehensweise ist jedoch anpassbar.

Die entsprechenden KI's würde ich versuchen, für einen bestimmten Zeitraum (z.B 24h – Zeit anpassbar) zu trainieren. Um die KI's zu testen, könnte ich die verschiedenen KI's gegen bereits vorhandene Chess bots oder Freunde spielen lassen, und somit noch anpassen

PHASE 4: Auswertung

In dieser Phase geht es darum, die trainierten KI's zum Spielen zu bringen. Dazu werde ich auf chess.com (je nach Nutzung von Datasets andere Website) die Züge meiner KI kopieren und schauen, wann das maximum erreicht wird. Danach wird sich zeigen, ob sich die KI's auf das entsprechende Niveau angepasst haben oder nicht.

Ein anderer weg wäre, Spieler auf entsprechendes Niveau zu suchen und 10 Spiele gegen die entsprechende KI spielen zu lassen.

PHASE 5: erstellen eines Programms/Website

Falls genügend Zeit vorhanden ist, würde ich mich noch darum sorgen, ein Programm (exe) /Website zu erstellen, um es noch attraktiv zu gestalten. Je nach Datenverbrauch der KI's müssen andere Methoden gewählt werden

Vorkenntnisse:

Python, JavaScript, HTML, CSS,

Ich selbst würde mich nicht als Profi-Programmierer ausgeben, jedoch finde ich zeige ich Einiges an Erfahrung mit Programmieren.