

# Maschinelles Lernen am Spiel Vier Gewinnt

## Aufgabenstellung

- Angelehnt an die von Google implementierte KI für AlphaGo eine KI implementieren
- Anwendung von maschinellem Lernen

## Wissenschaftlicher Charakter

- Vergleich von zwei Methoden von Maschinellern Lernen

## Methoden

- 1) Reinforcement Learning: Q-Learning
  - ➔ KI erhält für Spielzüge positive oder negative Rewards und sammelt dadurch künstlicher Erfahrungen
  - ➔ Rewards werden anhand des Spielausgangs vergeben
- 2) Neuronale Netze
  - ➔ Die nichtlineare Diskriminanzfunktion wird durch Wert der Knoten (Neuronen) und das Gewicht der Neuronen verbindenden Kanten (Synapsen) berechnet. Durch Training wird dieses Gewicht aktualisiert, um beste Strategie herauszufinden
  - ➔ Neuronale Netze gilt als eine klassische Methode des Maschinellen Lernens, und kann zum Vergleich mit RL geeignet sein.

## Umsetzung

- 1) Suche nach fertigen Implementationen und Anpassung
  - Viele der Gefunden Implementationen waren unbrauchbar, da sie nicht in der Größe anpassbar waren.
  - Viel Zeit in eine Implementation investiert, die sich zwar in der Größe anpassen ließ, jedoch war es uns nicht möglich eine zweite KI korrekt an das Spiel anzubinden.
- 2) Einlesen zum Thema Reinforcement-Learning, Q-Learning und Neuronale Netze
- 3) Eigene Implementierung des Spiels
  - Um nicht noch mehr Zeit zu verlieren haben wir letztlich unser eigenes 4-Gewinn-Spiel implementiert.
- 4) Nicht-lernende KI für das Training implementiert
  - Um eine Basis zu haben mit der die KI(s) lernen könne haben wir eine KI geschrieben, die allein mit if-Abfragen und Schleifen arbeitet.
  - Den Algorithmus haben wir aus einer anderen Implementation übernommen.
- 5) Datenbank
  - Datenbank die intern mit einer zwei ineinander verschachtelten HashMaps implementiert ist. Dabei enthält die erste HashMap, den aktuellen State (Spielfeld) und die darin enthaltene HashMap, die Actions (Züge, also Spalten in die geworfen wird) enthält und das Value (Bewertung) für diese Action.
  - Methoden zum Einfügen, Updaten, sowie Speicher und Laden der Datenbank.
- 6) Binär-Codierung
  - Zum Speichern der Spielzustände sind zwei Methoden implementiert, die zwei-dimensionale Arrays in eine Dimension umwandeln und umgekehrt, ohne dass Informationen verloren gehen.

#### 7) Turnier-Modus

- Zur Auswertung der KI(s) gedacht
- KI lernt in diesem Modus nicht weiter sondern zeigt nur wie gut die mit dem bisherigem Training spielen kann

#### 8) Q-Learning

- KI geniert mit einem Zuggenerator den besten möglichen Zug basierend auf einer Datenbank, die Belohnungen oder Bestrafungen für die KI speichert, je nachdem, ob sie gewonnen oder verloren hat.
- KI betrachtet bei jedem Zug, die darauffolgenden nächsten Züge und den maximal möglichen Wert, den sie damit erreichen kann und wertet dementsprechend den aktuellen Zug mit einem gewissen Lernparameter, der bestimmt wie wichtig langfristig mögliche Züge sind.

#### Zwischenstand

- 4-Gewinnt-Spiel inkl. Turnier-Modus implementiert
- Nicht-lernende KI für das Training implementiert
- Datenbank für RL implementiert
- Q-Learning weitgehend implementiert

#### Ausblick

- Q-KI trainieren
- Neuronale Netze
- Vergleich von Q-Learning und Neuronalen Netzen