Duale Hochschule Baden-Württemberg Mannheim

Reinforcement Learning

Portfolio

Projekt

Studiengang Wirtschaftsinformatik

Studienrichtung Data Science

Verfasser:	Rico Joel Siegelin
Matrikelnummer:	6577951
Kurs:	WWI20SB
Studiengangsleiter:	Bernhard Drabant
Dozent:	Janina Patzer
Bearbeitungszeitraum:	22.05.2023 – 30.07.2023

Ehrenwörtliche Erklärung

Ich versichern hiermit, dass wir die vorliegende Arbeit mit dem Titel selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Oberzent, 26.07.2023

Ort, Daum

Rico Joel Siegelin

Inhaltsverzeichnis

Abbildungsverzeichnis

Formelverzeichnis

Abkürzungsverzeichnis

RF	Reinforcement Learning

1. Wahl und Beschreibung der Umgebung

Ich habe mich für die Umgebung des Gaming entschieden, da ich selbst gerne Mobile Games oder auch auf der Konsole spiele. Auch mit Dota habe ich bereits Erfahrungen gemacht, was auch der Grund dafür ist, dass ich mich für die Gaming Richtung entschieden habe. Denn hier gab es die vergangenen Jahre die Einbindung von Reinforcement Learning in die Dota Welt. Das trainierte Modell spielte hierbei gegen die anderen echten Spiele und gewann haushoch (vgl. Berner et al. 2019:1). In diesem Reinforcement Learning Projekt wird ein bekanntes Spiel, nämlich „Flappy Bird" mithilfe eines Agenten trainiert und somit selbstständig gespielt. Flappy Bird ist ein einfaches 2D-Spiel, bei dem der Spieler einen fliegenden Vogel steuert. Das Ziel ist es, den Vogel durch eine Reihe von Röhren zu bekommen, die sich auf dem Bildschirm befinden. Der Vogel fliegt automatisch, und der Spieler muss ihn durch einfaches Tippen hochfliegen lassen. Das Spiel endet, wenn der Vogel eine Röhre berührt oder den Bildschirmrand trifft. Dabei fällt der Vogel direkt zu Boden, wenn keine Taste betätigt wird. Für jede passierte Röhre bekommt man einen Punkt. Ziel ist es den Agenten durch Punkte zu belohnen und ihn so besser zu machen. Nichtstun würde in dem Fall auch nichts bringen oder den Vogel auf derselben Höhe zu halten, da sich die Höhe und Größe der Röhren mit aufsteigenden Punkten ändert und schwieriger wird.

2. Entwicklung des RL-Agenten

Die Entwicklung eines geeigneten Agenten für das Spiel Flappy Bird erfolge auf Grundlage und auf Basis des Q-Learning Algorithmus. Das Q-Learning funktioniert dabei so, dass dieser Algorithmus die beste Aktion für einen gewissen Zustand wählt. Darüber hinaus lautet die Formel des Q-Learning-Algorithmus wie folgt:

$$Q^{new}(s_t, a_t) \leftarrow (1-\alpha) \cdot \underbrace{Q(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha}_{\text{learning rate}} \cdot \overbrace{\left(\underbrace{r_t}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \cdot \underbrace{\max_a Q(s_{t+1}, a)}_{\text{estimate of optimal future value}} \right)}^{\text{learned value}}$$

Formel 1: Q-Learning Algorithmus

Bei jedem Zeitschritt „*t*" befindet sich der Agent in einem bestimmten Zustand „s_t" *und* wählt eine Aktion „a_t" aus. Die Umgebung führt danach die ausgewählte Aktion aus und gibt die jeweilige Belohnung an den Agenten zurück. Das Ziel des Agenten ist dabei, die

Gesamtbelohnung bzw. den Q-Wert oben zu maximieren. Um das zu erreichen, muss der Agent eine optimale Richtlinie π finden, die eine Wahrscheinlichkeitsverteilung eines bestimmten Zustands über Aktionen darstellen. Hierbei stellt das „Q" die Action-Value-Funktion. Die Lernrate ist ein wichtiger Hyperparameter der angepasst werden muss, da dieser die Konvergent steuert (vgl. Amine „Q-Learning Algorithm: From Explanation to Implementation" 2020).

In dieser Funktion soll der Q-Wert maximiert werden, wobei Q für die Qualität der Aktion steht. Dabei wird der Agent immer dann belohnt, wenn der Vogel weiterkommt als in der Epoche zuvor. Um das Ganze in Zahlen auszudrücken, bekommt der Agent für +15, wenn er im nächsten Frame noch lebt. Darüber hinaus wird der Vogel sehr viel härter bestraft, wenn er stirbt. Dabei bekommt er nämlich -1000 Punkte. Das Training erfolgt dabei über 3000 Episoden bzw. Epochen und wurde aus Zeitgründen bei dort abgebrochen.

3. Bewertung & Visualisierung

Abschließend kann gesagt werden, dass das Training des Agenten sehr erfolgreich war und er einen sehr hohen Score erreicht hat. Der Agent hat eine hohe Performance erreicht. Da auch das Flappy Bird Spiel implementiert wurde ist der Ansatz sehr nahe an der Realität. Alternative Ansätze wäre zum einen das Deep-Q-Learning, was tiefe neuronale Netze verwendet, um die Q-Funktion zu approximieren. Auch kann man die Policy Gradient Methode angewandt werden. Hierbei wird darauf konzentriert die Richtlinie (Policy) zu optimieren, die die Wahrscheinlichkeit einer Aktion in einen gegebenen Zustand bestimmt. State-of-the-Art Ansatz ist dabei die Q-Learning Funktion, wobei es mit dem Deep-Q-Learning eine erhöhte Rechenleistung benötigt oder ein vorzeitiges Abbrechen bei einem gewissen Score. Anhand der Grafik kann man die Performance des Agenten darstellen.

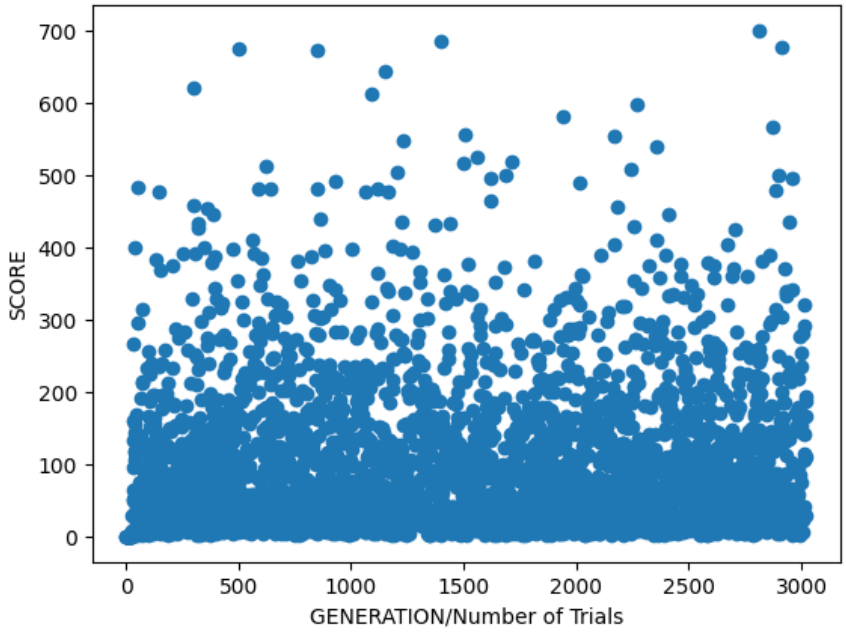

Abbildung 1: Auswertung der Ergebnisse im Dot-Plot

Um das Ergebnis noch weiter zu verbessern könnte man den Agenten über noch mehr Epochen trainieren oder zu einen Deep-Q-Learning Algorithmus wechseln.

Literaturverzeichnis

Amrani, Amine (2023): Q-Learning-Algorithmus: Von der Erklärung zur Umsetzung | von Amrani Amine | Auf dem Weg zur Datenwissenschaft. Online verfügbar unter https://towardsdatascience.com/q-learning-algorithm-from-explanation-to-implementation-cdbeda2ea187, zuletzt aktualisiert am 27.07.2023, zuletzt geprüft am 27.07.2023.

Berner, Christopher; Brockman, Greg; Chan, Brooke; Cheung, Vicki; Dębiak, Przemysław; Dennison, Christy et al. (2019): Dota 2 with Large Scale Deep Reinforcement Learning. OpenAI. Online verfügbar unter https://arxiv.org/pdf/1912.06680.

GitHub (2023): kyokin78/rl-flappybird: Use reinforcement learning to train a flappy bird which NEVER dies in Python. Online verfügbar unter https://github.com/kyokin78/rl-flappybird, zuletzt aktualisiert am 26.07.2023, zuletzt geprüft am 26.07.2023.

Q-Learning-Algorithmus: Von der Erklärung zur Umsetzung | von Amrani Amine | Auf dem Weg zur Datenwissenschaft (2023). Online verfügbar unter https://towardsdatascience.com/q-learning-algorithm-from-explanation-to-implementation-cdbeda2ea187, zuletzt aktualisiert am 27.07.2023, zuletzt geprüft am 27.07.2023.

Li, Anthony (2021): Reinforcement Learning in Python with Flappy Bird - Towards Data Science. In: Towards Data Science, 20.05.2021. Online verfügbar unter https://towardsdatascience.com/reinforcement-learning-in-python-with-flappy-bird-37eb01a4e786, zuletzt geprüft am 26.07.2023.

YouTube (2023): Understanding Reinforcement Learning with Flappy Bird. Online verfügbar unter https://www.youtube.com/watch?v=yyNOa9aB77E, zuletzt aktualisiert am 27.07.2023, zuletzt geprüft am 27.07.2023.

Anhang

Github-Link: https://github.com/Ricooooo31/Reinforcment_Learning_Flappy_Bird