

Decision Making in Starcraft

Fachgebiet: Informatik

Betreuer: Prof. Dr. Jürgen Eckerle

Experte: Dr. Federico Flückiger (Eidgenössisches Finanzdepartement)

Jedes Jahr wird an der AIIDE ein Starcraft Broodwar-Turnier veranstaltet, an dem künstliche Intelligenzen des Spiels gegeneinander antreten. Beim Recherchieren über die Umsetzung eines solchen Bots stiessen wir auf ein Dokument, in dem ein theoretischer Ansatz beschrieben ist, wie die gegnerische Strategie während des Spiels mittels Data Mining erkannt werden kann. Im Rahmen unserer Bachelor Thesis wurde dieser theoretische Ansatz erweitert und umgesetzt.

Ausgangslage

Zu Beginn der Arbeit hatten wir Daten aus über 5'000 Spielen zwischen professionellen Starcraft-Spielern gesammelt, welche alle Gebäude und deren Bauzeiten sowie die vom Spieler verfolgte Strategie beinhalten. Ziel war es, diese Daten für das Trainieren eines Klassifikationsalgorithmus zu verwenden, mit dem die gegnerische Strategie zur Laufzeit klassifiziert werden kann.

Als Schnittstelle zum Spiel diente die BWAPI (Brood War API), die es ermöglicht mittels C++ – Code Spielzustände abzufragen und Einheiten Befehle zu erteilen.

Lösung

Für das Erkennen der gegnerischen Strategie waren zwei Komponenten relevant: das Scouting-Verhalten des Bots und der Klassifikationsalgorithmus. Das Scouting beinhaltet das Ausspionieren und Aufzeichnen der gegnerischen Gebäude während des Spiels. Unter einem Klassifikationsalgorithmus versteht man eine Logik, welche anhand von aktuellen Daten einen Output, auch Klasse genannt, liefert.

In der Konzeptionsphase verglichen wir drei verschiedene Algorithmen mit dem Data Mining Programm WEKA und evaluierten den, der die genauesten Klassifikationsresultate lieferte. Daraus resultierte ein Decision Tree, welcher anhand der gegnerischen Gebäude und deren Gebäudebauzeiten die Strategie des Gegners erkennen kann.

Damit der Decision Tree zur Laufzeit angewendet werden kann, müssen Informationen über den Gegner bekannt sein. Weil die Karte durch den sogenannten «Fog Of War» verdunkelt ist, der die gegnerischen Aktivitäten verdeckt hält, muss ein eigener Spion in die gegnerische Basis gesendet werden, um die Gebäude des Gegners aufzudecken. Zeichnet der Scouter zu späte Gebäudebauzeiten auf, verfälscht dies die Klassifikation. Deshalb wurde die Scouting-Logik so umgesetzt, dass die Gebäude direkt entdeckt werden, wenn

sie die der Gegner baut. Das heisst, der Spion hält sich bis zum Klassifikationszeitpunkt in der gegnerischen Basis auf und darf nicht sterben. Wird der Spion angegriffen, flüchtet er aus der gegnerischen Basis und kehrt zu einem späteren Zeitpunkt wieder zurück.

Wurde während des Spiels die gegnerische Strategie ermittelt, leitet der Bot regelbasiert eine Gegenstrategie ein. Die 30 möglichen Gegenstrategien sind als XML-Dateien abgelegt, welche die Befehle für den Rest des Spiels beinhalten. Die Wahl der XML-Datei ist von dem Klassifikations-Output abhängig.

Testing

Für das Testen der Klassifikation entwickelten wir einen zweiten, schlanken Bot, der Strategien als XML-Datei einliest und ausführt. Damit hatten wir die Grundlage für das gezielte Testen der Strategie-Klassifikation. Weiter entwickelten wir ein eigenständiges Programm in Java, das die Resultate der Klassifikation aus einer Log-Datei ausliest und diese lesbar als Baum in einer HTML-Seite darstellt. So konnten wir die Klassifikation zurückverfolgen und Fehler besser erkennen.

Resultat

Die resultierende künstliche Intelligenz weist eine Klassifikationsgenauigkeit von 72 Prozent auf. Das heisst, der Bot klassifiziert mit hoher Genauigkeit die gegnerische Strategie, was eine gezielte Gegenstrategie zur Folge hat. Der Ansatz, die gegnerische Strategie mittels «Data Mining» zu erkennen, hat sich also bewährt.

Fazit

Der Ansatz, die gegnerische Strategie zur Laufzeit zu ermitteln und eine Gegenstrategie einzuleiten, ist einer von vielen, welche für das effiziente Spielen einer Starcraft-Partie benötigt werden. Für eine Teilnahme an der AIIDE reicht allein diese Umsetzung nicht aus. Die künstliche Intelligenz könnte noch im Angriffs- und Verteidigungsverhalten enorm verbessert werden.



Moritz Hulliger



Benjamin Lauener