Exploring the synergies between transfer in reinforcement learning and procedural content generation

Müller-Brockhausen, M.F.T.

Samenvatting

In dit proefschrift (getiteld: *Exploring the Synergies between Transfer in Reinforcement Learning and Procedural Content Generation*) is onderzocht hoe de twee in de titel genoemde onderzoeksvelden *Transfer in Reinforcement Learning (TRL)* en *Procedural Content Generation (PCG)* synergetisch zouden kunnen samenwerken.

Allereerst vergeleken wij verschillende AI-algoritmes in het bordspel *Tetris Link*. We vergeleken (1) een heuristische benadering, (2) Monte Carlo Tree Search (MCTS) en (3) Reinforcement Learning (RL).

Voor beide laatste algoritmes (MCTS & RL) is het aantal mogelijke zetten één van de indicaties hoe moeilijk een spel is. In Tetris Link ligt het aantal mogelijke zetten tussen die van schaken en go. Verder heeft Tetris Link de eigenschap dat het moeiljik is terug te komen van een achterstand. Samen maken deze eigenschappen dit spel een interessant onderzoeksonderwerp. Onze verwachting was dat MCTS hier het beste zou presteren, vanwege de snelheid van onze implementatie. Toch was MCTS zwakker dan RL en de heuristische benadering. Ook RL is door onze heuristiek verslagen. Hetgeen verbazend is, gezien bekende voorbeelden in de literatuur waarbij RL uitstekend presteert. Dit heeft ons ertoe aangezet de reden verder te onderzoeken, waarbij we ons op transfer in RL hebben gericht.

Om een idee van het onderzoeksveld te krijgen hebben we een literatuurstudie geschreven. De belangrijkste conclusie van onze survey is dat TRL zich voornamelijk richt op een statische hoeveelheid van taken die niet meer veranderen. Daarom zijn we ons gaan richten op PCG, waarin gemakkelijk dynamische veranderingen kunnen worden onderzocht.

Zodra we begonnen met experimenteren met TRL stuitten we snel op het probleem van reproduceerbaarheid. Het trainen van RL-policies gebaseerd op neurale netwerken is lastig reproduceerbaar. Reproduceerbaarheid is een belangrijke eigenschap in machine learning, vooral in de context van TRL waar elke taakwisseling een keten van niet-reproduceerbaarheid creëert. Hoewel we geen algemene oplossing voor het probleem gevonden hebben, hebben we wel een verbetering voorgesteld: Replay traces. De meeste RL omgevingen zijn deterministisch en dus reproduceerbaar als dezelfde actie-sequentie met dezelfde initiële random seed worden uitgevoerd. In deze benadering wordt een set van verschillende random seeds gebruikt om diverse maar reproduceerbare

playouts te garanderen. Figuren en tabellen in wetenschappelijke publicaties zijn met replay traces vergelijkbaar en verifieerbaar. Voortbouwend op de basis van replay traces, onderzochten wij hoe TRL en PCG elkaar kunnen versterken. Daarvoor hebben we het spel *Linerider* veranderd van 2D naar 3D. Het spel zelf is puur creatief en heeft geen vast doel. Dit heeft als gevolg dat er allerlei creaties kunnen worden gemaakt zoals tracks die synchroon op muziek worden afgespeeld. Na het specificeren van de track zorgen krachten zoals zwaartekracht voor de beweging van de rijder. Om dit spel toegankelijk voor RL te maken hebben we er een omgeving voor geprogrammeerd. Daarvoor moesten we een doel voor de algoritme definiëren. Het interessante aan deze omgeving is dat de op te lossen taak zelf een toepassing van PCG is. Initiële experimenten toonden aan dat de afstand tussen de startpositie en het doel klein moet zijn om RL de taak te laten leren. Door kleine incrementele verhogingen in afstand waren we in staat om een RL policy te trainen die ook geschikt was voor tracks van grotere afstanden. Tenslotte hebben we in de lijn van PCG ook Large Language Models (LLMs) onderzocht die inhoud voor games genereren. Gerelateerd onderzoek focust vooral op dynamische dialogen waarin de gebruiker alles kan invoeren wat hij wil, wat redelijk goed werkt. Gebruikersinvoer heeft echter het nadeel dat LLM's misbruikt kunnen worden om veiligheidssystemen te kraken en ze te laten ontsporen om toxische inhoud uit te spuwen. Derhalve hebben wij het concept van Chatter als alternatief bedacht, waarbij NPC's alleen korte zinnen genereren op basis van vooraf gedefinieerde aanwijzingen van ontwikkelaars. We laten empirisch zien dat dit goed werkt. Bovendien laten we in dit werk zien dat de meeste gaming hardware die consumenten in huis hebben krachtig genoeg is om een lokale LLM te laten draaien naast een AAA-game zoals Cyberpunk. Dit proefschrift laat zien dat PCG en TRL elkaar kunnen aanvullen. PCG kan worden toegepast om veel verschillende taken te genereren die helpen bij het trainen van generaliserende RL policies. En TRL kan helpen om betere resultaten te bereiken bij het toepassen van RL, bijvoorbeeld op een PCG-probleem zoals Linerider.