심층 강화 학습에 대한 간략한 조사

(원제 :A Brief Survey of Deep Reinforcement Learning)

강한결, 최수빈

목차

1.1 인공지능(AI)과 자율 학습의 목표

AI의 목표: 경험을 통해 학습하고 최적의 결정을 내리는 것

규칙 기반 AI의 한계: 새로운 환경에서 대응 불가능

자율 학습(Self-learning)의 필요성

- 예측할 수 없는 상황에서 유연하게 대응
- 시행착오(trial & error)를 통해 최적 행동 학습

- **강화 학습(RL)의 역할**: 보상을 기반으로 스스로 학습
  - AI가 환경을 관찰하고 행동을 선택
  - 행동의 결과로 보상 또는 패널티를 받음
  - 보상을 극대화하는 방향으로 정책을 최적화
  - 이런 시행착오를 반복하면서 AI는 더 좋은 선택을 하도록 학습

- **적용 사례**: 게임 AI(알파고), 로봇, 자율 주행

1.2 강화 학습(RL) 개념

- **강화 학습(RL) :** 시행착오를 통해 최적의 행동을 학습하는 기법

- **목표:** 보상을 극대화하는 행동을 찾는 것

- **기본 개념**
  - **에이전트(Agent)**: 학습하는 AI
  - **환경(Environment)**: AI가 상호작용하는 공간
  - **행동(Action)**: 에이전트가 선택할 수 있는 행동
  - **보상(Reward)**: 행동의 결과로 얻는 점수
  - **정책(Policy)**: 최적의 행동을 선택하는 전략

- **적용 사례**: 추천 알고리즘(스마트폰 광고, 넷플릭스), 음식 배달 앱, 게임 AI, 내비게이션

1.3 기존 강화 학습의 한계

- **기존 RL의 주요 한계**
  - **고차원 환경에서 학습 어려움** : 학습해야 할 정보가 너무 많음(동작 하나하나 학습)
  - **샘플 효율성 문제** : 수많은 시행착오 과정필요. 즉 방대한 데이터가 필요함
  - **일반화(Generalization) 부족**
    - 새로운 환경에서는 처음부터 다시 학습해야 함
    - 이미 학습한 정보를 비슷한 환경에 적용시키지 못함.

- **실제 예시**
  - **자율 주행 AI**: 새로운 도시에서 다시 학습 필요
  - **게임 AI**: 한 게임을 배워도 다른 게임에서는 처음부터 다시 시작
  - **음식 추천 AI**: 새로운 사용자가 오면 처음부터 다시 데이터 수집 필요

1.4 딥러닝과 강화 학습의 결합 → DRL의 등장

- **해결책: 딥러닝(Deep Learning)과 결합**
  - **딥러닝:** 복잡한 데이터를 분석하고 패턴을 학습하는 능력
  - **강화 학습:** 시행착오를 통해 최적의 행동을 학습하는 능력

- 두 가지를 결합해 복잡한 데이터를 시행착오로 학습하는 학습

- **DRL(Deep Reinforcement Learning) 등장**
  –즉, AI가 직접 화면을 보고, 현실을 인식하면서 배우는 게 가능해졌다.–

- 예시: **알파고(AlphaGo), 게임 AI, 자율 주행, 로봇 제어**

1.5 DRL의 주요 성과와 연구 방향

- **DRL의 실제 성과 - 어디까지 발전했나?**
  - **게임 AI** → 단순한 행동에서 전략적 플레이까지
  - **자율 주행** → 가상 환경에서 학습 후 실제 도로 적용
  - **로봇 제어** → 인간 수준의 정밀한 조작 가능

- **DRL의 한계 - 아직 해결해야 할 문제들**
  - **데이터 효율성** → 학습 시간이 너무 오래 걸림
  - **일반화 문제** → 새로운 환경에서도 잘 작동해야 함
  - **탐색 전략 개선** → 시행착오를 줄이고 더 똑똑하게 학습

- **DRL의 미래 - 연구자들이 집중하는 방향**
  - **모델 기반 RL** → 시행착오를 최소화하는 학습
  - **전이 학습(Transfer Learning)** → 다른 환경에서도 활용 가능
  - **휴먼-인-더-루프(Human-in-the-loop)** → 인간과 협력하는 AI

2.1 DRL의 기본 원리

- DRL(Deep Reinforcement Learning)
  - **강화 학습**(RL) + **딥러닝**(DL) **결합** → 복잡한 환경에서도 학습 가능
  - RL: 시행착오로 학습 & 보상을 극대화
  - DL: 신경망을 이용해 고차원 데이터를 분석

- DRL의 핵심 원리
  - **신경망**(Deep Neural Network) → 복잡한 데이터를 처리하는 역할
  - **Q-러닝**(Q-Learning) → 행동의 가치를 평가하여 최적의 선택
  - **경험 재사용**(Experience Replay) → 학습 속도를 높이는 기법

- DRL의 특징
  - 복잡한 환경에서도 학습 가능 (예: 이미지 기반 학습)
  - 기존 RL보다 빠르고 정교한 학습 가능

2.2 마르코프 결정 과정(MDP)

- MDP란? : 강화 학습의 기본 수학적 모델
  - **현재 상태만 고려하여 최적의 행동을 선택하는 과정**

- MDP의 구성 요소
  - **상태**(State, S) : 환경의 현재 상태
  - **행동**(Action, A) : 에이전트가 선택할 수 있는 행동
  - **보상**(Reward, R) : 행동의 결과로 얻는 점수
  - **전이 확률** (Transition Probability, P) : 행동에 따라 상태가 어떻게 변하는지
  - **정책**(Policy, π) : 최적의 행동을 선택하는 전략

- **마르코프 성질** (Markov Property)
  - **"현재 상태만 알면 미래를 예측할 수 있다"**

2.3 가치 함수와 정책

- **정책(Policy, π)**
  - 에이전트가 특정 상태에서 어떤 행동을 선택할지 결정하는 규칙.
  - 어떤 행동을 하도록 유도하고 설정하는 규칙. 즉 행동 방침을 정해줌.
  - **종류**: 결정론적 정책(Deterministic) / 확률적 정책(Stochastic)

- **행동 가치 함수(Q-Value, Q)**
  - 특정 상태에서 **특정 행동을 했을 때 받을 기대 보상**
  - **Q-러닝(Q-Learning)에서 사용**

- **가치 함수(Value Function, V)**
  - 상태(State)의 **좋음 정도(가치)를 평가하는 함수**
  - 가치가 높은 상태일수록 보상을 받을 가능성이 큼

2.4 신경망을 활용한 가치 근사

- **가치 근사(Value Approximation)**
  - 모든 상태의 가치를 **정확히 계산하기 어려울 때, 신경망을 이용해 근사**

- **가치 근사를 활용하는 이유**
  - 상태와 행동이 너무 많을 경우, **표 형태(Tabular)로 저장하는 것이 불가능**
  - **신경망을 사용하면 복잡한 관계를 학습 가능**

- **딥 Q-네트워크(DQN)의 활용**
  - Q-값을 신경망으로 근사하여 학습
  - Atari 게임 등에서 인간 수준의 AI 성능을 달성

3.1 가치 기반 학습 -DQN

- **가치 기반 학습**
  - **Q-값(Q-Value)을 학습하여 최적의 행동을 선택하는 방식**
  - 상태의 가치(Value)나 Q-값을 계산하여 정책을 결정

- **대표 알고리즘: DQN (Deep Q-Network)**
  - **Q-러닝을 신경망으로 확장**하여 고차원 데이터 학습 가능
  - Atari 게임 등에서 **인간 수준의 AI 성능을 달성**

- **DQN의 주요 기법**
  - **경험 재사용(Experience Replay)** → 학습 안정성 향상
  - **타깃 네트워크(Target Network)** → 학습 안정화

3.2 정책 기반 학습– 정책 그래디언트, PPO

- **정책 기반 학습**
  - **Q-값을 계산하지 않고, 최적의 정책(Policy)을 직접 학습하는 방식**
  - 확률적으로 행동을 선택하여 더 나은 전략을 탐색 가능

- **대표 알고리즘: 정책 그래디언트(Policy Gradient)**
  - 행동을 선택하는 **정책(π)** 자체를 신경망으로 표현하고 최적화
  - DQN과 달리, **연속적인 행동 공간에서도 학습 가능**

- **정책 기반 학습의 특징**
  - **탐색(Exploration)과 활용(Exploitation) 균형 유지**
  - **고차원, 연속적인 행동 공간에서도 효과적으로 작동**

3.3 액터-크리틱 방법 – A3C, DDPG

- **액터-크리틱(Actor-Critic)**
  - **가치 기반 학습(Value-based) + 정책 기반 학습(Policy-based)을 결합한 방법**
  - **액터(Actor)**: 정책을 학습하여 행동을 결정
  - **크리틱(Critic)**: 가치 함수를 학습하여 액터를 평가

- **대표 알고리즘**
  - A3C (Asynchronous Advantage Actor-Critic) → 병렬 학습으로 속도 향상
  - DDPG (Deep Deterministic Policy Gradient) → 연속적인 행동 공간에서 활용 가능

- **액터-크리틱의 장점**
  - 정책 기반 학습의 **탐색 능력**과 가치 기반 학습의 **안정성**을 모두 활용
  - 연속적인 행동 공간에서도 효율적으로 학습 가능

4.1 DQN(가치기반)의 Atari게임 학습

- **DQN이 Atari 게임을 학습한 방법**
  - 픽셀 데이터를 입력받아 **CNN(합성곱 신경망)** 을 활용해 상태를 인식
  - Q-값을 학습하여 최적의 행동을 결정

- **DQN의 성과**
  - 사람보다 더 높은 점수를 기록한 게임 다수 (예: 브레이크아웃, 스페이스 인베이더)
  - 강화 학습이 **고차원 환경에서도 적용 가능하다는 것**을 증명

- **DQN의 한계**
  - 학습 속도가 느리고, 일반화가 어려움
  - 탐색 전략이 부족하여 새로운 환경 적응이 어려움

4.2 AlphaGo와 DRL

- AlphaGo
  - 딥마인드(DeepMind)에서 개발한 **세계 최초의 바둑 AI**
  - 심층 강화 학습(DRL)과 **몬테카를로 트리 탐색(MCTS)** 을 결합하여 학습

- AlphaGo의 학습 방법
  - **정책 신경망(Policy Network)**: 다음 수를 예측하여 최적의 수 선택
  - **가치 신경망(Value Network)**: 현재 상태에서 승리 확률 평가
  - **강화 학습을 통해 스스로 수많은 게임을 플레이하며 학습**

- AlphaGo의 성과
  - **2016년 이세돌 9단과의 대결에서 4:1 승리**
  - 기존의 규칙 기반 AI를 넘어 **스스로 전략을 학습하는 AI의 시대를 열음**

4.3 로봇 공학 및 자율주행에서의 DRL 적용 사례

- **로봇 공학에서의 DRL 활용**
  - 로봇이 직접 **시행착오를 통해 최적의 동작을 학습**
  - **예제:** 로봇 팔이 물체를 집는 법을 스스로 학습

- **자율 주행에서의 DRL 적용**
  - 시뮬레이션 환경에서 **도로 주행, 보행자 회피, 신호 인식 등을 학습**
  - 실제 도로에서 안전성을 검증하기 전, 수백만 번의 학습 진행

- **DRL이 로봇과 자율 주행에 적합한 이유**
  - **고정된 규칙 없이 환경 변화에 적응 가능**
  - **데이터가 부족한 경우에도 스스로 학습하여 최적의 행동을 찾아감**

5.1 샘플 효율성 문제와 해결책

- **샘플 효율성**
  - AI가 학습할 때 필요한 **데이터(경험) 대비 성능 향상 정도**
  - DRL은 시행착오를 통해 학습하기 때문에 **엄청난 양의 데이터가 필요**

- **샘플 효율성 문제의 원인**
  - 무작위 탐색이 많아 **비효율적인 학습 과정 발생**
  - 기존 RL은 **하나의 경험을 한 번만 사용 → 학습 속도 저하**

- **해결책(ex.자율주행)**
  - **경험 재사용(Experience Replay)**: 이전 데이터를 반복 학습하여 데이터 낭비 방지
  - **모델 기반 RL(Model-based RL)**: 환경을 예측하는 모델을 활용해 실제 시행착오 감소
  - **전이 학습(Transfer Learning)**: 다른 환경에서 배운 경험을 활용하여 학습 시간 단축

5.2 탐색 전략 개선

- **탐색 vs. 활용 문제**
  - **탐색(Exploration)**: 새로운 행동을 시도하여 더 나은 전략을 찾음
  - **활용(Exploitation)**: 현재까지 배운 최적의 행동을 반복하여 보상을 극대화
  - **문제**: 너무 탐색하면 비효율적이고, 너무 활용하면 더 나은 전략을 놓칠 수 있음

- **해결책**
  - **입실론-탐욕적(ε-Greedy) 전략**: 일정 확률로 무작위 행동을 선택하여 탐색 유지
  - **어퍼 신뢰 한계(UCB) 알고리즘**: 보상이 좋았던 행동 + 덜 선택된 행동을 모두 고려해서 다음에 학습할 행동을선택
  - **정책 기반 학습(Policy-based RL)**: 확률적으로 행동을 선택하여 탐색과 활용의 균형 조절

5.3 다중 에이전트 학습

- **다중 에이전트 학습이란?**
  - 여러 개의 AI(에이전트)가 **협력 또는 경쟁하면서 학습하는 강화 학습 방법**
  - **예시:** 자율 주행 차량 간 통신, 로봇 협력 작업, 전략 게임 AI

- **다중 에이전트 학습의 도전 과제**
  - **환경이 동적으로 변화** → 상대방의 전략에 따라 최적의 행동이 달라짐
  - **협력과 경쟁의 균형** → 공동 목표 vs. 개별 목표 조정 필요

- **해결책**
  - **중앙 집중식 학습 - 분산 실행 (CTDE)** → 훈련은 공동으로, 실행은 개별적으로
  - **상대의 전략을 예측하는 모델 학습** → 게임 이론 기반 접근
  - **보상 공유 메커니즘 적용** → 협력을 유도하는 인센티브 구조 설계

5.4 전이 학습 과 일반화 문제

- **전이 학습(Transfer Learning)**
  - AI가 한 환경에서 학습한 지식을 **다른 환경에서도 활용할 수 있도록 하는 기법**
  - **예시**: 로봇이 공장에서 학습한 동작을 새로운 공정에서도 그대로 사용

- **일반화 문제(Generalization Problem)**
  - AI가 학습한 환경이 바뀌면 **성능이 급격히 저하될 수 있음**
  - **예시**: 자율 주행 AI가 서울에서 학습했지만, 뉴욕에서는 다시 학습해야 하는 문제

- **일반화 문제의 해결책**
  - **도메인 랜덤화**: 환경의 여러 변수를 랜덤으로 변화시켜 학습하여 일반화 성능 향상
  - **메타 학습(MAML)**: 작은 Task 단위로 학습하여 새로운 작업의 학습 속도 향상
  - **사전 학습 모델**: 대량의 데이터로 미리 학습한 후 세부 조정(Fine-tuning)

6.1 DRL의 현재 성과 요약

- **DRL이 가져온 혁신적인 변화**
  - **게임 AI**: 인간보다 뛰어난 성능 (DQN, AlphaGo)
  - **자율 주행**: 시뮬레이션을 활용한 학습 가능
  - **로봇 공학**: 강화 학습을 통한 정밀한 동작 학습

- **DRL의 한계**
  - 학습 속도 문제 → 많은 데이터와 계산 자원 필요
  - 일반화 문제 → 새로운 환경에서 다시 학습 필요
  - 탐색과 활용 문제 → 최적의 전략을 찾는 과정이 비효율적

6.2 향후 연구 방향 및 해결해야 할 과제

- DRL의 미래 연구 방향
  - **모델 기반 강화 학습(Model-based RL)**: 시행착오를 줄이고 학습 속도 향상
  - **휴먼-인더-루프 학습(Human-in-the-loop RL)**: 인간 피드백을 활용하여 더 효율적인 학습 가능(ex.ChatGPT)
  - **강화 학습과 생성 AI의 결합**: 창의적인 문제 해결과 자율적인 의사결정

- DRL이 해결해야 할 주요 과제
  - **샘플 효율성 개선**: 데이터를 덜 사용하면서 더 빠르게 학습할 방법 필요
  - **일반화 문제 해결**: 다양한 환경에서도 잘 작동하는 AI 개발
  - **안정성과 윤리적 문제**: 강화 학습 AI가 안전하고 공정하게 동작하도록 설계 필요