Reinforcement Learning Project

LunarLander 환경에서의 DQN 알고리즘 비교 분석

DQN, Double DQN, Dueling DQN의 구조적 차이점 및 학습 효율성 연구

발표자: [사용자 이름]

[소속 또는 학교명]

2025년 11월 29일

Gymnasium LunarLander-v3 Environment

프로젝트 주제 및 연구 목표

문제 정의

LunarLander-v3 환경에서 착륙선을 달 표면의 착륙 패드(깃발 사이)에 안전하고 부드럽게 착륙시키는 제어 문제를 해결합니다.

성공 기준 (Solved)

연속된 100개 에피소드에서 평균 보상 +200점 이상 획득 시 문제 해결로 간주합니다.

연구 목표

DQN 계열 알고리즘 성능 비교 분석 기본 DQN vs. Double DQN vs. Dueling DQN

학습 안정성 및 수렴 속도 평가 각 알고리즘이 목표 점수(200점)에 도달하는 에피소드 수 비교

데이터 효율성 및 최종 성공률 검증 동일한 학습 스텝 내에서 더 높은 평균 보상을 얻는지 확인

최적 하이퍼파라미터 도출 Learning Rate, Batch Size 등의 영향 분석

환경 및 데이터셋 설명

LunarLander-v3 Environment Dynamics

환경 역학 (Dynamics)

시작 상태 (Initial State)

착륙선은 달 표면 상공 중앙 부근에서 생성됩니다. 초기 위치, 속도, 각도는 무작위로 결정되어 매 에피소드마다 다양한 상황을 학습합니다.

종료 조건 (Termination)

- 성공: 착륙 패드(두 깃발 사이)에 안전하게 접지
- 실패: 달 표면에 충돌 (추락)
- 시간 초과: 1000 스텝 초과 혹은 정지

에이전트 인터페이스

Continuous 관찰 공간 (Observation Space, 8차원)

1	X 좌표 (가로 위치)
2	Y 좌표 (세로 위치)
3	X 선형 속도
4	Y 선형 속도
5	착륙선 각도
6	각속도
7	좌측 다리 접지 (Bool)
8	우측 다리 접지 (Bool)

보상 시스템 (Reward System)

Optimization Target & Feedback

Terminal State

착륙 성공

+100

착륙 패드(깃발 사이)에
안전하게 접지 시 획득

Success Criteria

목표 점수

200+

최근 100 에피소드 평균
점수가 이 값을 넘어야 함

Terminal State

추락 / 충돌

-100

달 표면에 강하게 충돌하거나
기체가 파괴될 경우

엔진 사용 비용

-0.3 / Frame

메인 엔진 점화 시마다 부여되는 연료 소비 패널티
(최소한의 연료로 효율적인 움직임 유도)

상태 패널티 (Shaping)

Variable

착륙 패드와의 거리, 이동 속도, 기체 기울기에 따라
실시간 감점 부여 (안정적 자세 유지 유도)

DQN 알고리즘 핵심 원리

Deep Q-Network Architecture & Hyperparameters

CORE CONCEPTS (3 PILLARS)

Deep Q-Network

복잡한 상태 공간 처리를 위해 Q-Table 대신 신경망을 사용
Input: State(8) Output: Q-values(4)

Replay Buffer

경험 (s, a, r, s')을 저장 후 무작위로 샘플링하여 학습 데이터 간의 상관관계(Correlation) 제거

Target Network

학습 목표(Target Q) 계산 시 별도의 고정된 네트워크 사용
Moving Target 문제 해결 및 발산 방지

KEY HYPERPARAMETERS

LEARNING RATE (LR)
네트워크 가중치 업데이트 속도 및 수렴 안정성 결정

DISCOUNT FACTOR (Γ)
미래 보상의 현재 가치 반영 비율 (근시안적 vs 원시안적)

BATCH SIZE
한 번의 그래디언트 업데이트 시 사용하는 경험 샘플 수

EPSILON DECAY (E)
초기 무작위 탐험(Exploration)에서 활용(Exploitation)으로의 전환 비율

BUFFER SIZE
학습에 재사용 가능한 최대 과거 경험 데이터 저장 용량

TARGET UPDATE
Target Network를 Main Network와 동기화하는 주기 및 방식

알고리즘 심화 비교 (Advanced Architectures)

Overestimation Control vs. Structural Innovation

Double DQN

Target Stability Optimization

목적 및 문제 해결

기본 DQN의 과대 추정(Overestimation) 편향을 해결하여, Max 연산이 노이즈를 과대평가하는 문제를 방지합니다.

핵심 원리: 분리(DECOUPLING)

행동의 선택(Selection)과 가치의 평가(Evaluation)를 분리합니다.

$$Q_{target} = R + \gamma\, Q_{target}(S', \operatorname{argmax} Q_{online}(S', a))$$

`Online Net:` 최적 행동`(a*)` 선택
`Target Net:` 선택된 행동 가치 평가

주요 효과

Q-값의 발산을 막고 학습 안정성 향상

불필요한 낙관적 행동 방지로 최적 정책 수렴

VS

Dueling DQN

Structural Efficiency

구조적 혁신

Q-함수를 상태 가치(Value)와 행동 이점(Advantage)의 두 스트림으로 분리합니다.
니다.

결합 방식 (AGGREGATION)

두 스트림을 다시 합쳐 최종 Q-값을 산출합니다.

$$Q(s, a) = V(s) + (A(s, a) - mean(A(s, a')))$$

`V(s):` 상태 자체의 본질적 가치
`A(s, a):` 각 행동의 상대적 우위

주요 효과

행동 차이가 미미한 상태에서 V(s) 학습 가속

데이터 효율성(Data Efficiency) 및 수렴 속도 증가

실험 셋업 (Experimental Setup)

Environment & Evaluation Metrics

HARDWARE
MacBook Pro M1

ENVIRONMENT
Gymnasium LunarLander-v3

REPRODUCIBILITY
Multiple RNG Seeds (5 runs)

Evaluation Metrics

평균 리턴 (Average Return)

에이전트 성능의 가장 직접적인 지표. 누적 보상의 평균값을 통해 학습 수준을 평가합니다.

성공률 (Success Rate)

최근 100 에피소드 중 성공적으로 착륙한 비율을 측정하여 신뢰성을 검증합니다.

에피소드 길이 (Length)

종료 시점까지 걸린 스텝 수. 짧고 효율적인 경로로 착륙했는지 판단하는 기준입니다.

수렴 속도 (Speed)

목표 점수(+200)에 도달하기까지 필요한 에피소드 수를 통해 학습 속도를 비교합니다.

학습 안정성 (Stability)

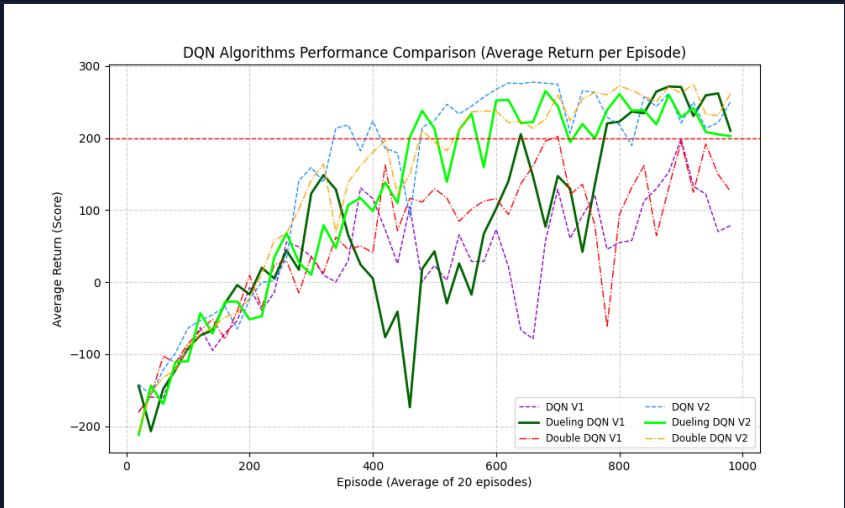
학습 곡선의 분산(Variance)과 진동 폭을 분석하여 알고리즘의 견고함을 평가합니다.

데이터 효율성 (Efficiency)

동일한 학습 스텝 내에서 얼마나 빠르게 높은 보상을 획득하는지 측정합니다.

실험 결과 및 분석 (Experimental Results)

Performance Comparison & Code Analysis



Algorithm	평균 리턴	성공률
DQN v1	9.17	25%
Double DQN v1	162.23	50%
Dueling DQN v1	-36.51	25%
DQN v2	**252.55**	**95%**
Double DQN v2	**158.16**	**60^**
Dueling DQN v2	**253.62**	**85%**

코드 구현 차이점 (v1 vs v2)

구분	Code v1	Code v2
Hyperparameters	LR (0.005) Discount Factor (0.98)	LR (0.001) Discount Factor (0.99)
	Batch Size (32)	Batch Size (64)
Target Update	Soft Update	Hard Update

차이점에 따른 성능 영향 분석

Dueling DQN

$V(S)$와 $A(S, a)$의 구조적 분리가 학습 효율성을 극적으로 높였습니다.

장기 계획 (γ=0.99)

γ=0.99는 V1의 0.98보다 장기적인 관점을 에이전트에 부여했습니다. 이 덕분에 V2의 모든 알고리즘(DQN V2 포함)이 V1의 동일 알고리즘보다 높은 최종 평균 리턴을 기록했습니다. γ의 차이가 LunarLander의 성공적인 착륙에 결정적인 영향을 미쳤습니다.

최적 파라미터 조합 (V2 Dueling)

V2의 Dueling DQN은 γ=0.99, Hard Update, B=64라는 안정적인 조합 덕분에 6가지 테스트 중 가장 빠르게 200점 목표를 달성하고 가장 높은 성능을 보였습니다.

토의 및 결론 (Conclusion & Discussion)

핵심 요약 (Key Findings)

Double DQN의 안정성 (Stability)

과대 추정(Overestimation) 문제를 효과적으로 완화하여, 학습 후반부의 Q-값 발산을 \
방지하고 안정적인 정책 수렴을 유도했습니다.

Dueling DQN의 효율성 (Efficiency)

상태 가치(V)와 행동 이점(A)을 분리 학습하여, 행동 간 차이가 미미한 상황에서도 유의미한 특징을 빠르게 학습했습니다.

FINAL RECOMMENDATION

"Dueling Double DQN (D3QN)이 LunarLander 태스크에 최적"

한계 및 보완점

단순 Replay Buffer 사용으로 중요 경험이 희석될 가능성 존재 (PER 도입 필요)

Soft Update 파라미터(τ) 및 Epsilon Decay 스케줄의 정교한 튜닝 부족

통계적 유의성 확보를 위한 더 많은 Random Seed 실험 필요

향후 연구 방향

Optuna 등을 활용한 하이퍼파라미터 자동 탐색(AutoML) 적용

Prioritized Experience Replay (PER)와 D3QN의 결합 모델 검증

연속 행동 공간(Continuous Action Space)을 위한 PPO, SAC 등과의 비교 연구