Flip 7 카드 게임에서의 전략적 의사결정을 위한 심층 강화학습 에이전트 개발 및 행동 양식 분석

전남대학교 인공지능학부 213802 최대한

2025년 강화학습 기말 프로젝트

Abstract

본 연구는 확률적 불확실성이 존재하는 카드 게임인 'Flip 7' 환경에서 최적의 의사결정을 수행하는 심층 강화학습(Deep Q-Network, DQN) 에이전트를 제안한다. 제안된 에이전트는 명시적인 규칙 입력 없이도 카드 카운팅과 위험 관리 전략을 스스로 학습하였다. 실험 결과, 에이전트는 단독 플레이 시 200점 도달에 평균 11.38라운드를 기록하여 수학적 기댓값 기반의 기준 모델('Daehan Player')과 대등한 효율성을 보였으며, 1:1 대결에서는 52.0%의 승률로 우위를 점하였다. 또한 행동 양식 분석을 통해 에이전트가 상황에 따라 선별적으로 위험을 감수하는 전략적 유연성을 가짐을 확인하였다. 본 연구는 강화학습이 불완전 정보 게임에서 인간 수준의 합리적 추론을 넘어설 수 있는 가능성을 시사한다.

주제어: 강화학습, Deep Q-Network (DQN), Flip 7, 카드 게임 AI, 불완전 정보 게임

코드 공개: 본 연구의 소스 코드는 다음 리포지토리에서 확인할 수 있다.
https://github.com/dae-hany/flip_seven_reinforcement_learning

1 서론

1.1 연구 배경 및 필요성

Flip 7은 덱에 남아있는 카드의 구성을 추론하고, 현재의 위험과 보상을 저울질하여 'Hit' 또는 'Stay'를 결정해야 하는 대표적인 'Press-your-luck' 메커니즘 기반의 카드 게임이다. 이 게임은 단순한 규칙을 가지고 있지만, 매 턴마다 확률적 불확실성이 존재하며, 사용된 카드가 버려짐에 따라 확률 분포가 동적으로 변하는 특징을 가진다.

기존의 게임 AI 연구에서 이러한 환경은 주로 확률 기반의 휴리스틱이나 규칙 기반 알고리즘으로 접근해왔다. 본 프로젝트에서 비교 대상으로 삼는 'Daehan Player' 역시 남은 카드를 카운팅하여 기댓값을 계산하고, 이를 바탕으로 합리적인 의사결정을 내리도록 설계되었다.

하지만 이러한 규칙 기반 모델은 사전 정의된 수식에 의존하기 때문에, 게임의 복잡한 상황 변수를 유연하게 반영하여 장기적인 보상을 극대화하는 전략을 수립하는 데에는 한계가 있다. 이에 본 연구에서는 심층 강화학습, 그중에서도 DQN(Deep Q-Network) 알고리즘을 적용하여 Flip 7 게임 환경에서의 최적 의사결정 에이전트를 개발하고자 한다.

에이전트는 수많은 에피소드를 통해 환경과 상호작용하며, 명시적인 규칙 입력 없이도 승리를 위한 효율적인 카드 카운팅 능력과 위험 관리 능력을 스스로 학습하게 된다.

1.2 연구 목표

본 연구의 최종 목표는 인간 수준 이상의 성능을 발휘하는 Flip 7 AI 에이전트를 개발하고, 그 행동 양식을 분석하는 것이다. 구체적인 세부 목표는 다음과 같다.

- **효율성 극대화:** 단독 플레이 시, 목표 점수인 200점에 도달하기 위해 소요되는 평균 라운드 수를 최소화한다. 본 연구에서는 평균 11라운드 이내 도달을 목표로 한다.
- **합리적 에이전트와의 경쟁 우위 확보:** 수학적 기댓값에 기반한 강력한 규칙 기반 에이전트인 'Daehan Player'와의 1:1 대결 시뮬레이션에서 승률 우위를 점하는 정책을 학습한다. 이때 대결 방식은 각 플레이어가 **라운드 단위로 번갈아 가며(Alternating Rounds)** 게임을 진행하는 것을 기본으로 한다.
- **확장성 및 행동 분석:** 학습된 에이전트가 다인원(6인) 경쟁 환경에서도 유효한 성능을 내는지 검증하고, 위험 상황에서의 의사결정 패턴을 분석하여 강화학습 에이전트가 기존 알고리즘 대비 어떤 전략적 우위를 가지는지 규명한다.

본 보고서는 위 목표를 달성하기 위해 구축한 Gymnasium 기반의 환경, 적용된 DQN 알고리즘의 구조, 그리고 다양한 시나리오에서의 실험 결과를 상세히 기술한다.

2 문제 정의 및 환경 구축

본 장에서는 Flip 7 카드 게임의 규칙을 정의하고, 이를 강화학습 에이전트가 학습할 수 있는 마르코프 결정 과정(Markov Decision Process, MDP) 형태로 모델링한다. 또한, Python의 Gymnasium 라이브러리를 활용하여 구축한 시뮬레이션 환경의 구체적인 구현 내용을 기술한다.

2.1 Flip 7 게임 규칙

Flip 7은 총 85장으로 구성된 고정된 덱을 사용하는 카드 게임이다. 플레이어는 200점에 먼저 도달하는 것을 목표로 하며, 매 턴마다 카드를 한 장 더 뽑을지('Hit') 혹은 현재 점수를 보전할지('Stay') 결정해야 한다. 구체적인 규칙은 다음과 같다.

- **덱 구성:** 숫자 카드(0–12) 79장과 수정자 카드(Modifier Cards) 6장으로 구성된다. 숫자 카드는 해당 숫자의 값만큼 덱에 포함된다(예: '12' 카드는 12장, '5' 카드는 5장). 수정자 카드는 점수를 추가하거나($+2, \ldots, +10$) 배로 만드는($\times 2$) 특수 효과를 가진다.
- **행동(Action):**
  - **Hit:** 덱에서 카드를 한 장 뽑는다. 만약 뽑은 숫자 카드가 현재 손패에 이미 있는 숫자라면 파산(Bust)이 되어 라운드가 즉시 종료되고 0점을 획득한다. 중복되지 않은 경우 손패에 추가된다.
  - **Stay:** 라운드를 종료하고 현재 손패의 점수를 획득하여 총 점수에 누적한다.
- **특수 규칙 (Flip 7):** 손패에 중복 없는 숫자 카드가 7장이 모이면 'Flip 7'을 달성하게 되며, 라운드가 즉시 종료되고 계산된 점수에 15점의 보너스를 추가로 획득한다.
- **점수 계산:** 숫자 카드의 합에 $\times 2$ 수정자가 있을 경우 2배를 한 뒤, 나머지 + 수정자 카드의 값을 더한다.
- **덱 관리:** 라운드가 종료되면 사용된 카드는 버림 더미로 이동하며, 덱이 완전히 소진되기 전까지는 섞지 않는다. 이는 카드 카운팅 전략이 유효함을 의미한다.

2.2 Flip 7 게임 규칙의 MDP 모델링

본 연구에서는 Flip 7 게임을 에이전트가 환경 상태 S_t에서 행동 A_t를 선택하고, 그에 따른 보상 R_{t+1}과 다음 상태 S_{t+1}을 관측하는 MDP 문제로 정의하였다.

1. **상태 공간 (State Space, $\mathcal{S}$):** 에이전트가 카드 카운팅을 수행하고 현재의 위험도를 판단하기 위해 다음과 같은 정보를 상태로 정의하였다.
   - `current_hand_numbers`: 현재 손패에 보유한 숫자 카드들의 존재 여부를 나타내는 이진 벡터 (크기 13).
   - `current_hand_modifiers`: 현재 손패에 보유한 수정자 카드들의 존재 여부를 나타내는 이진 벡터 (크기 6).
   - `deck_composition`: 덱에 남아있는 각 카드 종류별 잔여 매수를 나타내는 정수 벡터 (크기 19). 이는 카드 카운팅을 위한 핵심 정보이다.
   - `total_game_score`: 현재까지 누적된 게임 총 점수.
2. **행동 공간 (Action Space, $\mathcal{A}$):** 이산적인 두 가지 행동으로 정의된다.
   - $a = 0$ (Stay): 라운드 종료 및 점수 획득.
   - $a = 1$ (Hit): 카드 뽑기.
3. **보상 함수 (Reward Function, $\mathcal{R}$):** 학습의 효율성을 높이기 위해 다음과 같이 보상을 설계하였다.

- **파산(Bust):** 0점 (라운드 실패).
- **Stay / Flip 7:** 게임 규칙에 따라 계산된 최종 라운드 점수.
- **Reward Shaping:** 최종 라운드 점수를 보상으로 설정하였다.

2.3 Gymnasium 환경 구현

OpenAI Gymnasium 인터페이스를 상속받아 `FlipSevenCoreEnv` 클래스를 구현하였다. 이 환경의 핵심적인 구현 특징은 다음과 같다.

- **에피소드와 게임의 분리:** 강화학습의 에피소드 단위는 하나의 '라운드'로 설정하였다. `step()` 함수는 라운드가 종료될 때 `terminated=True`를 반환한다. 반면, 실제 게임의 승리 조건인 200점 도달 여부는 외부 루프에서 `total_score`를 통해 관리된다.
- **상태 유지:** `reset()` 메서드가 호출될 때, 손패는 초기화되지만 `total_score`와 덱의 상태(남은 카드 및 버림 더미)는 유지되도록 구현하였다. 이를 통해 에이전트는 라운드가 바뀌어도 이전 라운드에서 빠진 카드 정보를 기억하여 다음 라운드의 의사결정에 활용할 수 있다.
- **셔플 로직:** 실제 게임 규칙에 따라, `step()`에서 'Hit'을 수행할 때 덱이 비어있는 경우에만 버림 더미를 가져와 셔플하도록 구현하여 카드 카운팅 환경을 엄격히 준수하였다.

3 방법론

본 장에서는 Flip 7 게임 환경에서 최적의 정책을 학습하기 위해 적용한 심층 강화학습 알고리즘의 구조와, 성능 비교를 위해 설계된 기준 모델인 'Daehan Player'의 알고리즘, 그리고 실험에 적용된 구체적인 학습 설정을 기술한다.

3.1 심층 강화학습 알고리즘 (DQN)

본 연구에서는 Q-Learning을 심층 신경망으로 확장한 DQN(Deep Q-Network)을 사용하였다. Flip 7 게임의 상태 공간은 서로 다른 속성의 정보(이진 벡터인 손패, 정수형인 덱 구성, 스칼라 값인 점수)로 구성되어 있다. 이를 효과적으로 처리하기 위해, 본 연구에서는 단순한 다층 퍼셉트론(MLP) 대신 입력 특징별로 특화된 서브 네트워크를 거친 후 통합하는 구조를 제안하였다.

3.1.1 신경망 아키텍처 (Q-Network)

관측 상태 딕셔너리의 4가지 요소를 각각 별도의 완전 연결 층으로 처리한 후 결합한다.

- **Hand Numbers Net:** 숫자 카드 상태(13차원) $\rightarrow$ 32차원 특징 벡터
- **Hand Modifiers Net:** 수정자 카드 상태(6차원) $\rightarrow$ 16차원 특징 벡터
- **Deck Composition Net:** 덱 구성 정보(19차원) $\rightarrow$ 64차원 특징 벡터
- **Score Net:** 현재 점수(1차원) $\rightarrow$ 8차원 특징 벡터

위 과정을 통해 추출된 총 120차원의 특징 벡터는 공유된 MLP 레이어를 거쳐 최종적으로 행동 공간의 크기인 2개의 Q-Value($Q(s, \text{Stay})$, $Q(s, \text{Hit})$)를 출력한다.

3.1.2 학습 알고리즘

- **Experience Replay:** 에이전트의 경험 $(s_t, a_t, r_t, s_{t+1}, \text{done})$을 리플레이 버퍼에 저장하고, 매 학습 단계마다 무작위로 미니 배치를 샘플링하여 학습함으로써 데이터 간의 상관관계를 제거하고 학습의 안정성을 높였다.
- **Target Network:** Q-Value의 타겟값 계산에 사용되는 네트워크를 별도로 두어, 일정 주기마다 메인 네트워크의 가중치로 업데이트함으로써 학습 발산을 방지하였다.

3.2 기준 모델 (Baseline Model: Daehan Player)

강화학습 에이전트의 성능을 객관적으로 평가하기 위해, 수학적 확률 계산에 기반한 규칙 기반 에이전트인 'Daehan Player'를 기준 모델로 설정하였다. 이 에이전트는 현재 덱에 남아있는 카드 정보를 완벽하게 인지하고 있으며, 다음과 같은 기댓값(Expected Value, EV) 계산을 통해 행동을 결정한다.

1. 현재 덱 구성을 바탕으로 다음 카드가 파산을 유발할 확률(P_{bust})과 안전할 확률(P_{safe})을 계산한다.
2. 안전한 카드들의 평균 획득 가능 점수($E[\text{Gain}]$)를 계산한다. 이때 $\times 2$ 카드의 효과 등을 반영한다.
3. 각 행동에 대한 기댓값을 비교한다.

$$EV_{\text{stay}} = \text{Current Score} \tag{1}$$

$$EV_{\text{hit}} = P_{\text{safe}} \times (\text{Current Score} + E[\text{Gain}]) + P_{\text{bust}} \times 0 \tag{2}$$

(단, $P_{\text{bust}} \times 0$ 항은 파산 시 점수가 0이 됨을 의미하며 실제 계산에서는 소거된다.)

4. $EV_{\text{hit}} > EV_{\text{stay}}$인 경우에만 'Hit'을 수행하고, 그렇지 않으면 'Stay'를 선택한다.

이 모델은 인간이 수행할 수 있는 가장 이상적이고 합리적인 플레이를 모사하므로, 강화학습 에이전트가 이를 능가하는지 검증하는 것은 매우 중요한 의미를 가진다.

3.3 학습 설정

DQN 에이전트의 학습을 위해 설정한 주요 하이퍼파라미터는 표 1와 같다.

4 실험 및 결과

본 장에서는 제안한 DQN 에이전트의 학습 과정과 다양한 시나리오에서의 성능 평가 결과를 제시한다. 모든 실험은 1,000회 이상의 반복 실험을 통해 결과의 분산을 줄이고 통계적으로 유의미한 성능 차이를 확인하고자 하였다. 또한, 기준 모델인 'Daehan Player'와의 비교를 통해 에이전트의 효용성을 검증하였다.

Table 1: DQN Training Hyperparameters

Hyperparameter	**Value**
Total Training Games	1,000
Optimizer	Adam
Learning Rate	1×10^{-4}
Discount Factor (γ)	0.99
Replay Buffer Size	50,000
Batch Size	64
Target Update Frequency	10 Games
Epsilon (ϵ) Schedule	$1.0 \rightarrow 0.01$ (Decay: 0.995 per Game)

4.1 학습 곡선 및 수렴성

에이전트는 총 1,000번의 게임을 수행하며 학습되었다. 그림 1는 학습 진행에 따른 손실 값과 200점 도달에 소요되는 라운드 수의 변화를 보여준다.

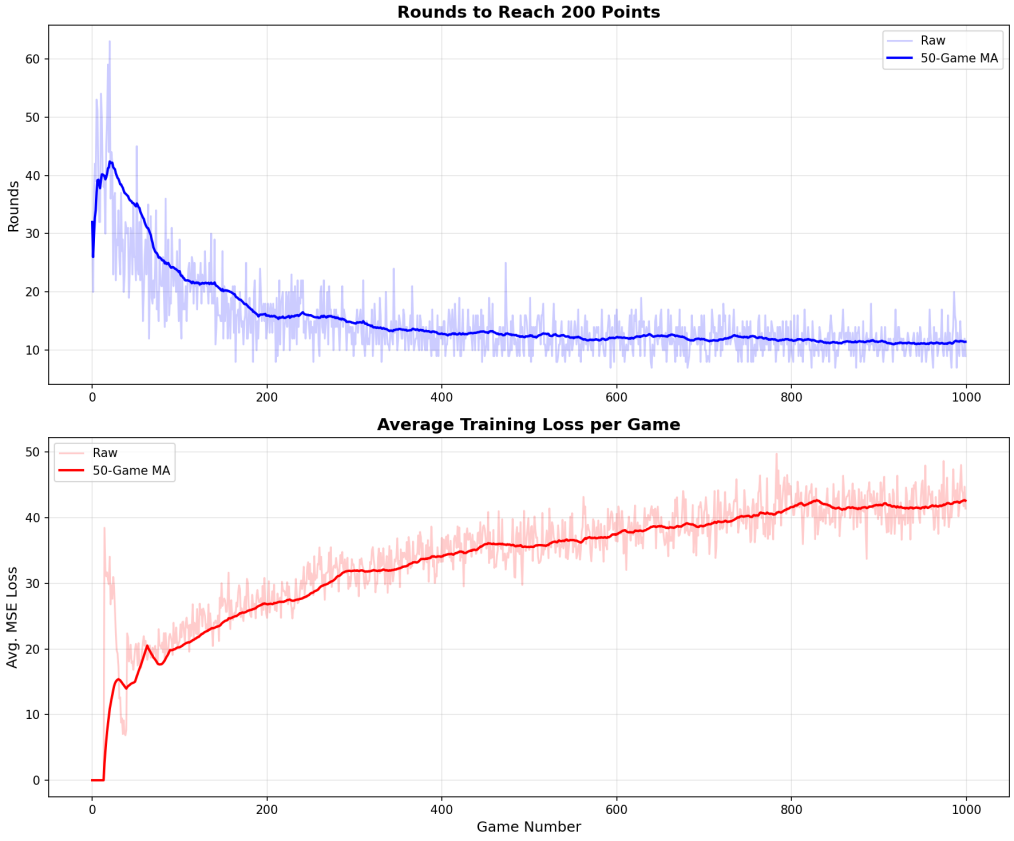

Figure 1: Training History: Average Rounds to Reach 200 Points (Top) and Training Loss (Bottom)

초기 탐험 단계에서는 높은 엡실론(ϵ) 값으로 인해 무작위 행동이 많아 라운드 소모가 많았으나, 학습이 진행됨에 따라 평균 라운드 수가 급격히 감소하여 약 400 게임 이후부터는 평균 11–12 라운드 수준으로 수렴하였다. 이는 에이전트가 게임의 목표인 '최소 라운드 내 200점 도달'을 위한 효율적인 정책을 성공적으로 학습했음을 시사한다.

4.2 단독 플레이 성능

에이전트의 순수한 득점 효율성을 평가하기 위해, 방해 요소가 없는 단독 플레이 환경에서 200점 도달 속도를 측정하였다. 1,000회 시뮬레이션 결과는 그림 2와 같다.

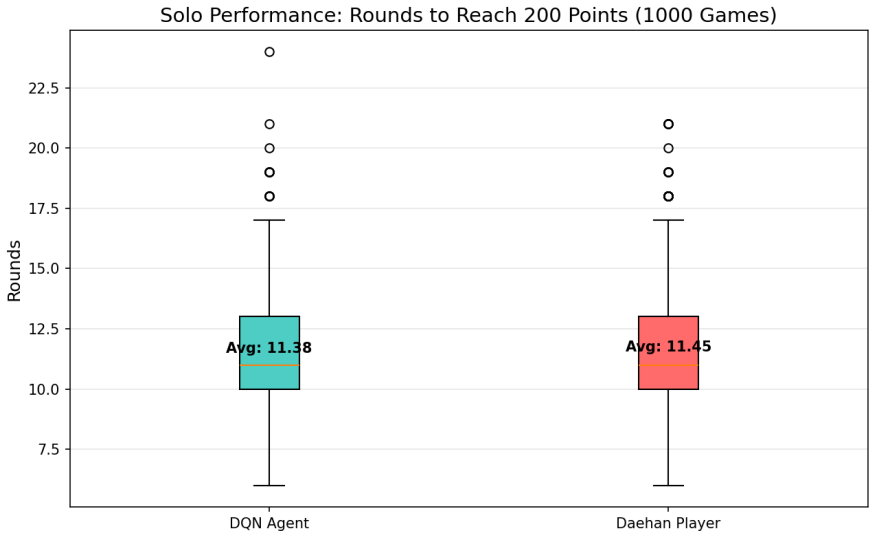

Figure 2: Distribution of Rounds to Reach 200 Points (Solo Play)

실험 결과, DQN 에이전트는 평균 **11.38 라운드**, Daehan Player는 평균 **11.45 라운드**를 기록하였다. 이는 강화학습 에이전트가 별도의 규칙 입력 없이도, 수학적으로 계산된 최적의 휴리스틱 모델과 대등한 수준의 득점 효율성을 스스로 찾아냈음을 의미한다.

4.3 1:1 대결 성능

본 연구의 핵심 목표인 경쟁력을 검증하기 위해, 동일한 덱을 공유하며 **한 플레이어의 라운드가 완전히 종료된 후 다음 플레이어가 라운드를 진행하는 방식(Alternating Rounds)**으로 1:1 대결 시뮬레이션을 10,000회 수행하였다.

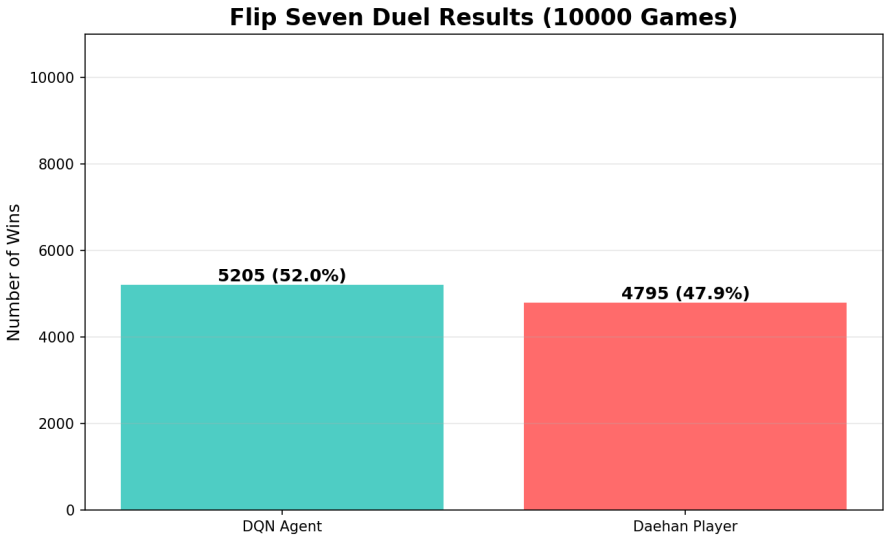

Figure 3: Win Rates in 1:1 Duel (10,000 Games)

그림 3에서 볼 수 있듯이, DQN 에이전트는 **5,205승 (52.0%)**, Daehan Player는 **4,795승 (47.9%)**을 기록하여 에이전트가 근소한 우위를 점하였다. 이는 에이전트가 단순히 기댓값만을 쫓는 것이 아니라, 상황에 따라 상대보다 먼저 200점에 도달하기 위한 전략적 위험 감수(Risk Taking)를 수행했기 때문으로 분석된다.

4.4 다인원 게임 확장성

마지막으로, 학습된 에이전트가 다수의 플레이어가 존재하는 환경에서도 성능을 유지하는지 확인하기 위해 6인 게임 시뮬레이션을 수행하였다. 참가자는 DQN 에이전트 1명, Daehan Player 2명, 그리고 서로 다른 성향을 가진 3명의 더미 플레이어(Dummy Player)로 구성되었다. 각 더미 플레이어의 행동 규칙은 다음과 같이 정의하였다.

- **보수적 플레이어 (Conservative Player):** 위험 회피적인 성향을 모사한다. 현재 라운드 점수가 **15점 이상**이면 무조건 'Stay'를 선택하고, 그 미만일 때만 'Hit'을 수행하여 안정적인 점수 획득을 목표로 한다.
- **공격적 플레이어 (Aggressive Player):** 고득점을 노리는 위험 감수 성향을 모사한다. 현재 라운드 점수가 **30점 미만**인 경우 무조건 'Hit'을 선택하며, 30점 이상이 되었을 때만 'Stay'를 선택한다. 이는 파산(Bust) 위험이 높더라도 'Flip 7' 보너스나 고득점을 적극적으로 노리는 전략이다.
- **무작위 플레이어 (Random Player):** 전략적 판단 없이 매 턴마다 50%의 확률로 'Hit' 또는 'Stay'를 무작위로 선택한다.

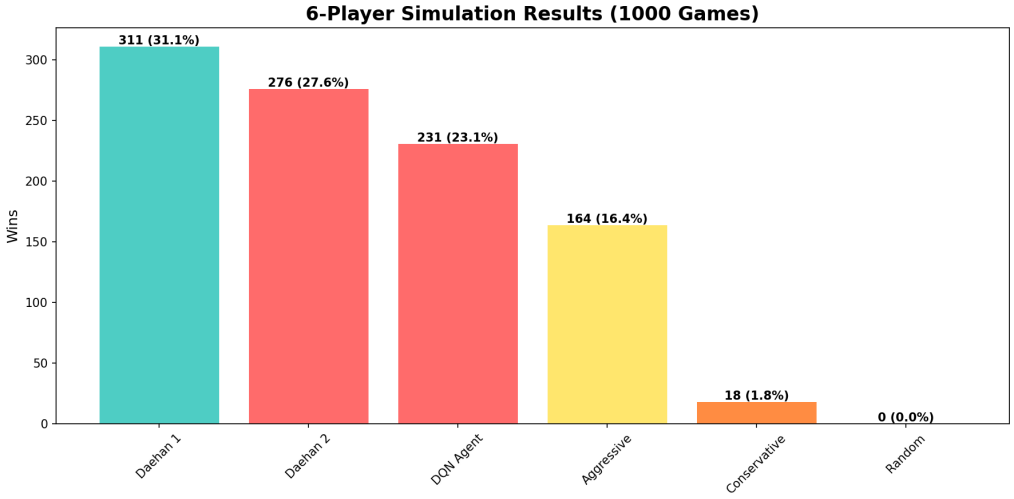

Figure 4: Win Distribution in 6-Player Game

1,000회 시뮬레이션 결과(그림 4), Daehan Player들이 각각 약 27–31%의 승률로 1, 2위를 차지하였으며, DQN 에이전트는 **23.1%**의 승률로 3위를 기록하였다. 이는 에이전트가 자신의 차례(Round)가 다시 돌아오기 전까지 다른 5명의 플레이어들에 의해 덱이 급격하게 소모되는 환경 변화를 학습 과정에서 경험하지 못했기 때문으로 판단된다.

5 분석 및 고찰

본 장에서는 앞선 실험 결과에서 나타난 DQN 에이전트의 성능 우위 원인을 정량적 지표를 통해 분석하고, 에이전트가 학습한 정책이 실제 게임의 전략적 요소들을 어떻게 해석하고 있는지 시각화하여 규명한다. 또한, 다인원 환경에서의 성능 저하 원인을 강화학습의 관점에서 고찰한다.

5.1 우위 요인 분석

DQN 에이전트가 수학적으로 최적화된 규칙 기반 모델인 Daehan Player를 상대로 승률 우위를 점할 수 있었던 원인을 파악하기 위해, 10,000회의 대결 데이터를 바탕으로 상세 지표를 분석하였다(그림 5).

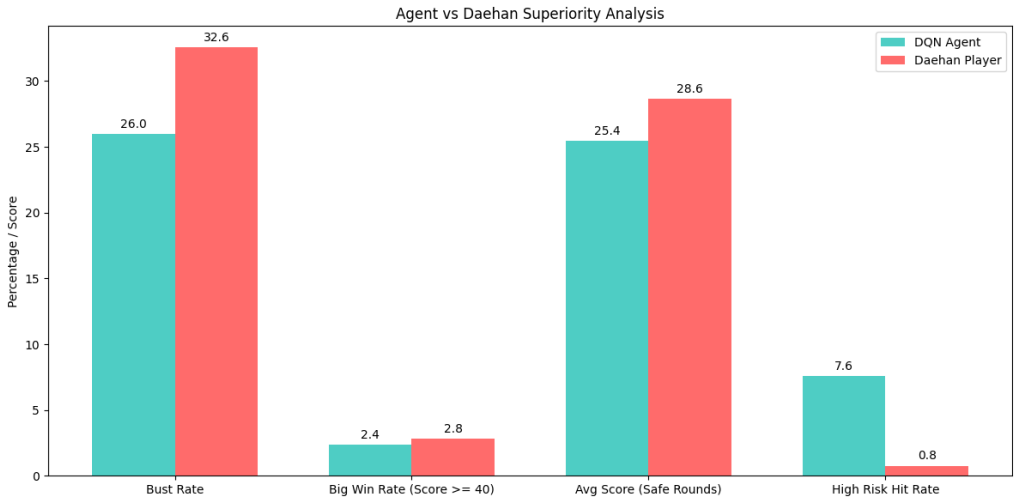

Figure 5: Superiority Analysis: Comparison of Behavioral Metrics

- **안정적인 운영 (Low Bust Rate):** 에이전트의 파산 확률(Bust Rate)은 **26.0%**로, Daehan Player의 32.6%보다 낮게 나타났다. 이는 에이전트가 불필요한 위험을 감수하지 않고 안정적으로 점수를 획득하는 경향이 있음을 보여준다.
- **결정적 순간의 과감함 (Calculated Aggression):** 반면, 파산 위험이 30% 이상인 고위험 상황에서의 'Hit' 비율은 에이전트가 **7.6%**로, Daehan Player(0.8%)에 비해 약 10배 더 높았다. 이는 에이전트가 평소에는 보수적으로 플레이하지만, 승부처에서는 과감하게 위험을 감수하여 고득점을 노리는 '선별적 공격성'을 학습했음을 시사한다.

5.2 정책 해석

에이전트가 실제로 카드 카운팅과 수정자 카드의 가치를 이해하고 있는지 검증하기 위해, 특정 시나리오에서의 Q-Value 변화를 분석하였다.

5.2.1 카드 카운팅 능력 검증

특정 숫자 카드(예: '7')가 손패에 있을 때, 덱에 해당 숫자가 남아있는 경우(Case A)와 없는 경우(Case B)의 Q-Value를 비교하였다. 실험 결과, 모든 숫자 카드(0–12)에 대해 덱에 해당 카드가 없을 때(안전할 때)의 $Q(s, \text{Hit})$ 값이 덱에 카드가 남아있을 때(위험할 때)보다 유의미하게 높게 나타났다. 이는 에이전트가 `deck_composition` 상태를 통해 자신의 손패와 충돌하는 카드의 잔여 여부를 정확히 인지하고 있음을 증명한다.

5.2.2 수정자 카드의 전략적 활용

×2와 같은 특수 카드가 상황에 따라 다르게 평가되는지를 분석하였다.

- **저위험/저득점 상황 (점수 12점):** ×2 카드가 포함될 경우, 잠재적인 점수 폭발력을 기대하여 $Q(s, \text{Hit})$ 값이 급격히 상승하였다.
- **고위험/고득점 상황 (점수 42점):** 동일하게 ×2 카드가 있더라도, 이미 높은 점수를 확보한 상태에서는 $Q(s, \text{Stay})$ 값이 압도적으로 높게 나타났다. 이는 "42점을 84점으로 확정 짓는 것"이 추가적인 위험 감수보다 더 큰 가치를 지님을 에이전트가 학습했음을 보여준다.

5.3 6인 게임에서의 성능 저하 원인

앞선 실험에서 에이전트는 1:1 대결에서는 우위를 보였으나, 6인 게임에서는 승률이 하락하였다. 이는 강화학습에서의 전형적인 **환경 변화(Distribution Shift)** 문제로 해석할 수 있다. 본 에이전트는 자신이 매 턴마다 연속적으로 의사결정을 내리는 환경(1인 또는 1:1 라운드 교대)에서 학습되었다. 그러나 6인 게임에서는 에이전트가 한 라운드를 마친 후 다시 자신의 차례가 돌아오기까지 5명의 다른 플레이어가 덱에서 카드를 소모하게 된다. 즉, S_t 시점에서 에이전트가 예측한 덱의 상태와, 실제 다음 라운드가 시작되는 S_{t+1} 시점의 덱 상태 사이에 큰 괴리가 발생한다. 현재 모델은 타인의 행동으로 인한 덱의 급격한 소모를 예측하는 기능이 부재하므로, 다인원 환경에서는 이러한 불확실성에 취약할 수밖에 없다.

6 결론

본 연구에서는 불확실성이 존재하는 카드 게임인 Flip 7에서 최적의 의사결정을 내릴 수 있는 심층 강화학습 에이전트를 개발하고, 그 성능과 행동 양식을 심층적으로 분석하였다. 제안된 DQN 에이전트는 1,000회의 학습을 통해 200점 도달에 평균 11.38라운드를 소요하며, 수학적으로 최적화된 규칙 기반 모델인 Daehan Player(11.45라운드)와 대등한 효율성을 달성하였다. 특히 1:1 대결 시뮬레이션에서는 52.0%의 승률을 기록하여 근소한 우위를 점하였는데, 이는 에이전트가 평소에는 안정적인 플레이(Low Bust Rate)를 지향하면서도 승부처에서는 과감한 위험 감수를 하는 전략적 유연성을 학습했기 때문으로 확인되었다. 또한, 정책 분석을 통해 에이전트가 카드 카운팅 능력과 수정자 카드의 가치를 정확히 인지하고 있음을 검증하였다.

그러나 다인원(6인) 게임 환경에서는 승률이 23.1%로 하락하는 한계를 보였다. 이는 에이전트가 자신의 라운드가 아닌 동안 발생하는 급격한 환경 변화(덱 소모)를 예측하지 못하는 '단일 에이전트 학습의 한계'에서 기인한다.

6.1 향후 연구 방향

향후 연구에서는 이러한 한계를 극복하기 위해 다음과 같은 방향을 제안한다.

- **상대방 모델링(Opponent Modeling):** 상대 플레이어의 행동 패턴을 추론하는 모듈을 추가하여, 자신의 차례가 돌아올 때의 덱 상태를 예측하는 능력을 강화한다.
- **다중 에이전트 강화학습(MARL):** 여러 에이전트가 동시에 경쟁하는 환경에서 학습을 진행하여, 다인원 상황에서의 동적 변화에 적응하도록 한다.
- **RNN 기반 메모리 구조 도입:** LSTM이나 GRU와 같은 순환 신경망을 적용하여, 과거의 카드 흐름(History)을 기억하고 이를 바탕으로 현재 덱 상태를 더 정교하게 추론하도록 개선한다.

본 연구는 단순한 규칙 기반 알고리즘을 넘어, 강화학습이 불완전 정보 게임에서 복잡한 전략을 스스로 학습하고 인간 수준의 합리성을 뛰어넘을 수 있는 가능성을 보여주었다는 점에서 의의가 있다.