Match-3 Game 스테이지 구성을 위한 PPO 기반 강화학습 에이전트 설계

홍자민, 정재화
한국방송통신대학교 정보과학과
lbypra, jaehwachung@knou.ac.kr

Design of PPO-based Reinforcement Learning Agents for Match-3 Game Stage Configuration

Jamin Hong, Jaehwa Chung
Dept. of Computer Science, Korea National Open University

요 약

Match-3 Game 은 스테이지 구성 및 난이도 설정이 중요한 게임이나 다양한 밸런스 요소로 인해 스테이지 구성에 중요한 요소인 난이도 설정에 많은 시간이 소요된다. 특히 게임을 플레이하는 유저가 재미를 느끼는 수준으로 난이도를 설정하는 것이 중요하며, 이를 자동화하기 위해 실제 유저의 플레이 데이터를 활용하여 사람과 유사한 수준의 자동 플레이 에이전트 개발이 진행되었다. 하지만 플레이 데이터의 확보는 쉽지 않기에 연구 방향은 플레이 데이터가 없는 강화학습으로 확장되고 있다. 스테이지 구성에 중요한 요소인 난이도를 설정하기 위함이라면 각 스테이지 간의 상대적인 난이도 차이를 파악하는 것으로 가능하다. 이를 위해 게임의 규칙을 학습한 강화학습 에이전트로 밸런스 요소의 변화에 따른 다양한 난이도의 스테이지를 50 회씩 플레이하여, 평균 획득 점수를 기준으로 스테이지 구성에 필요한 각 스테이지들의 난이도를 파악할 수 있었다.

1. 서론

여러 게임 장르 중에서 Match-3 Game 은 단순한 게임 규칙을 가지고 있으면서도 난이도 설정에 많은 시간이 소요되는 게임이다. 기본 규칙은 선택한 블록을 상하좌우에 인접한 다른 블록과 위치를 바꾸는 과정을 통해 같은 종류의 블록을 3 개 이상 맞추어 점수를 획득하는 것이다. Match-3 Game 은 다양하게 발전해왔는데, 시간제한이 있는 기록 경쟁 방식과 스테이지 클리어 방식으로 크게 나눌 수 있다. 특히 스테이지 방식은 유저(user)가 클리어해야 하는 스테이지가 나열되어 있어, 자신과 다른 사람들의 달성율을 비교할 수 있고 이러한 과정에서 경쟁과 커뮤니케이션이 발생한다. 다만, 스테이지 방식은 스테이지를 지속적으로 추가해야 유저들의 관심을 유지할 수 있고, 해당 장르가 발전하면서 특수블록, 아이템, 캐릭터 효과 등 다양한 밸런스 요소가 추가되었다.

이로 인해 난이도 설정은 어려워지고 있으며, 개발 과정에서 난이도를 평가하고 조정하는 과정이 반복된다. 새로운 밸런스 요소가 추가될 때마다 기존의 난이도가 어떻게 변화할 것인지를 확인해야 할 필요가 있다. 이러한 스테이지의 복잡성은 게임의 공식 출시 전에 알아야 하므로 복잡성을 테스트하기 위해서 별도의 부서가 존재하며, 테스트에 많은 시간이 소요된다[1].

이를 해결하기 위해 게임 플레이 에이전트는 규칙 기반에서 지도학습 또는 강화학습 등 다양한 기술을 사용하여 개발되었으며, 게임을 최적으로 플레이하는 것에서 시작되어 사람과 유사한 방식으로 플레이하는 에이전트를 만들기 위해 많은 노력을 기울였다[2].

유저의 플레이 데이터를 사용하는 지도학습은 사람과 유사한 에이전트를 개발하는데 용이하나 플레이 데이터를 얻지 못하는 경우 적용이 불가능하다[3].

따라서 이러한 문제를 해결하기 위해 본 논문에서는 플레이 데이터가 없는 상황에서 게임의 난이도를 비교/분석할 수 있는 방법을 제시하고자 한다. PPO 알고리즘[4]으로 게임의 규칙을 학습한 강화학습 에이전트가 다양한 난이도의 스테이지에서 플레이를 통해 획득한 점수는 난이도 평가 기준으로 산정할 수 있으며 이를 기준으로 각 스테이지의 난이도를 비교하여 스테이지를 구성할 수 있다. 이를 통해 유저의 플레이 데이터 없이 각 스테이지의 상대적인 난이도를 비교할 수 있다.

본 논문의 구성은 다음과 같다. 2 장에서는 Match-3 Game 의 자동 플레이 관련 기존 연구와 제안하는 방식의 동기에 대해 살펴본다. 3 장에서는 Match-3 Game 의 환경과 에이전트 설계 방식을 정의하고 에이전트 학습 및 학습 결과를 분석한다. 4 장에서는 설계한 에이전트로 실험한 결과를 통해 난이도 비교가 가능함을 증명하며, 5 장에서 결론을 맺는다.

2. 관련 연구

Match-3 Game 장르에서 유저들의 인기를 꾸준히 유지하기 위해서는 정교하게 레벨 디자인된 스테이지 구성을 연속적으로 업데이트하는 것이 중요하며, 신속한 업데이트를 위해 레벨 디자인을 검토할 자동 플레이 AI 개발이 필요하다[5]. 그런 이유로 Match-3 Game 을 자동으로 테스트하기 위한 여러 방법이 제시되었으며, 크게 유저의 플레이 데이터를 활용해 사람의 행동 재현이 목적인 방식과 강화학습을 통한 최고 성능 구현이 목적인 방식으로 구분할 수 있다[6]. 실제 게임 서비스에 적용하기 위해서는 에이전트가 잘 플레이하는 것도 중요하지만, 사람과 비슷한 수준의 플레이를 하는 것이 중요하다. 사람이 재미를 느끼는 수준의 난이도를 설정하기 위해서는 사람과 비슷한 수준의 플레이를 하는 것이 중요하기 때문이다.

사람과 비슷한 수준의 에이전트를 개발하는데 있어서 에이전트의 훈련에 유저의 플레이 데이터를 활용하는 방법은 대규모 학습 데이터를 확보하지 않은 이상 적용이 어렵다[3].

다른 방법으로는 훈련된 에이전트가 스테이지를 완료하는데 필요한 이동 횟수와 실제 플레이어의 대규모 샘플의 스테이지 완료율과 비교함으로써 결과적으로 에이전트가 유저와 비슷한 수준을 가지고 있는지를 비교하였다[7]. 전략적인 플레이를 통한 방법[6]에서는 유저들이 일반적으로 사용하는 전략적 플레이를 미리 정의하여 사람과 같은 플레이 방법을 모방하는 방식을 제시하고 실제 유저의 평균 이동 횟수와 비교하였다. 앞선 두 방식은 에이전트의 학습에 유저 데이터를 사용하진 않았으나 에이전트가 유저와 비슷한 수준인지를 확인하기 위해 유저 데이터가 필요하며, 데이터가 없는 상태에서는 에이전트가 유저와 비슷한 수준인지 파악하기 어렵다.

딥러닝을 이용한 연구는 유저의 플레이 데이터가 없는 강화학습을 통한 연구로 확장되고 있는데, 이유는 플레이 데이터를 이용한 학습 데이터 구축에 대한 한계점을 극복하지 못하고 있기 때문이다[8].

앞선 연구[7]에서 흥미로운 점은 수준 간의 행동 측면에서의 차이는 인간 행동의 차이와 높은 상관관계가 있다는 것으로, 수준 이하의 수행에도 불구하고 에이전트의 성능을 사용하여 플레이어 메트릭을 추정하는 것이 가능하다고 하였다.

사람과 유사한 에이전트를 개발하려는 이유는 게임을 플레이하는 사람이 재미를 느낄 수 있는 수준의 난이도로 설정하기 위함이다. 다양한 난이도를 가진 스테이지의 순서를 조정하기 위함이라면 각 스테이지의 상대적인 난이도를 파악하는 것으로 가능하다.

3. 에이전트 설계

3.1 게임 환경

Match-3 Game 의 기본 규칙에 특수블록과 이동불가 위치 요소가 포함된 환경에서 학습을 진행한다. 학습 및 실험에서 보드 사이즈는 8*8 로 고정하고, 최대 9 개의 블록을 사용한다. 특수블록은 4 종으로 가로/세로 한 줄 파괴 블록, 범위 파괴 블록에 같은 종류의 블록을 전부 파괴하는 폭탄 블록으로 구성하며 특수블록끼리 교환 시 효과로 더 많은 블록을 파괴할 수 있도록 한다. Match-3 Game 의 목표는 많은 블록을 파괴하여 높은 점수를 획득하는 것으로 한정한다. 또한 블록을 움직일 수 없는 상태가 되는 경우 보드의 블록을 갱신하여 다시 이동 가능한 블록이 존재하도록 조정한다.

3.2 에이전트 설계 방식

유저의 플레이 데이터가 없는 상황에서 각 스테이지의 난이도를 비교할 수 있는 방법을 제안한다.

단순히 스테이지의 난이도를 비교하기 위함이라면 게임의 규칙을 학습한 에이전트를 통해 밸런스 요소의 변화에 따른 다양한 난이도의 스테이지를 플레이하여 획득한 점수에 따라 난이도를 비교할 수 있다.

다른 스테이지보다 획득한 점수가 높은 스테이지가 쉬운 스테이지라고 할 수 있으며, 이러한 기준으로 획득 점수에 따라 스테이지의 순서를 난이도 기준으로 설정할 수 있다. 학습에 유저의 플레이 데이터를 사용하지 않고, 에이전트가 스테이지에서 획득한 점수를 기준으로 난이도를 상대적으로 설정하는 방식이다. 이를 통해 유저의 플레이 데이터가 없더라도 상대적인 비교로써 각 스테이지의 난이도 비교 분석이 가능해진다.

3.3 에이전트 학습

게임의 규칙을 학습할 수 있도록, 본 논문에서는 난이도 설정 대상이 되는 블록 개수와 이동불가위치 개수를 지정 범위 내에서 랜덤하게 설정하여 학습을 진행하였다. 학습이 진행될수록 이동불가위치 개수의 랜덤 범위를 조금씩 늘림으로써 좀 더 어려워질 수 있는 환경에서 에이전트를 학습하였다.

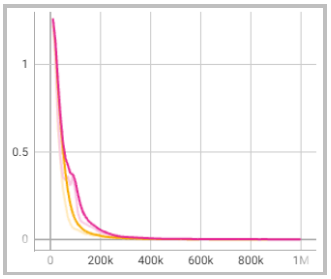
Match-3 Game 은 블록이 파괴되면 기존 블록이 아래로 내려오고 위의 빈 공간에는 새로운 블록이 랜덤으로 생성된다. 이때 랜덤으로 생성되는 블록에 따라 연속으로 블록이 파괴되는 우연성이 존재하며, 이러한 운에 따라 같은 스테이지에서도 점수 차이가 크게 나는 경우가 발생하기도 한다. 한 번에 여러 개의 블록을 확실히 파괴하기 위해서는 특수블록 위주의 학습이 필요하며, 특수블록의 생성/파괴가 어려운 스테이지의 경우는 획득 점수 위주의 학습이 유리하다. 또한 사람에 따라 다양한 플레이 타입이 있으므로, 에이전트는 점수 기반과 특수블록 기반 두 종류로 구분하여 학습을 진행하였다.

<표 1> 에이전트 구분

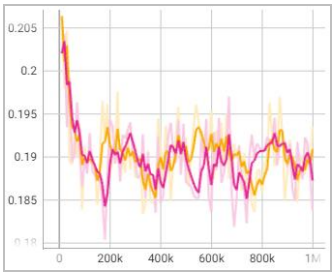
	점수 기반	특수블록 기반
공통	・보드 크기 : 가로 8 칸, 세로 8 칸으로 고정 ・블록 개수 : 6~9 개 범위 내에서 랜덤 ・이동불가위치 개수 : 3~8 개 범위 내에서 랜덤	
보상 기준	1 회 이동으로 획득한 점수	・블록 파괴 시 : +3 ・특수블록 생성/파괴 시 : - 생성 개수 * 4 - 파괴 개수 * 5 ・1 회 이동으로 발생한 위의 과정을 합산
목표	높은 점수를 획득하도록 유도	특수블록 생성/파괴 위주의 플레이를 유도

3.4 학습 결과

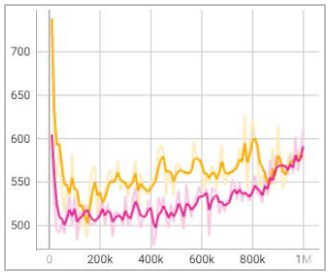
에이전트의 학습이 후반으로 갈수록 난이도가 어려워질 수 있도록 설정했기에 학습된 결과 그래프는 다음과 같은 양상을 보인다.



(a) Entropy



(b) Policy Loss



(c) Value Loss

● 점수 기반 ● 특수블록 기반

(그림 1) Tensorboard Graph

Entropy 가 훈련 중에 지속적으로 감소하는 것을 확인할 수 있다. 다만 Policy Loss, Value Loss 의 경우 초반에 감소하는 모습을 보여주다가 이후에는 일정 범위 내에서 진동하거나 조금 증가하는 모습을 보여주고 있다. 이는 새로운 블록이 랜덤하게 결정되어 게임의 획득 점수가 운에 따라 영향을 받을 수 있는 부분과 학습이 후반으로 갈수록 어려운 난이도의 스테이지가 학습될 확률을 높여서 진행한 것이 원인으로 분석된다.

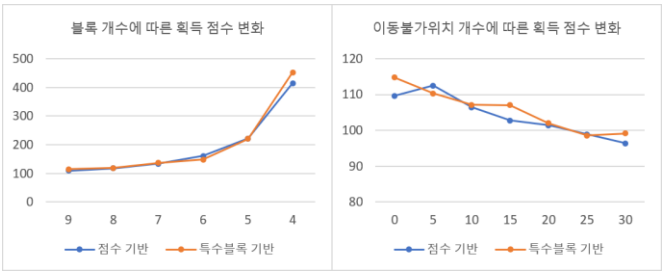
4. 실험 결과

4.1 실험 환경

Ubuntu 20.04, Unity 2020.3.33f1 (LTS), Unity ML-Agents Release 19(Unity Package 2.2.1), python 3.7.4 환경에서 PPO 알고리즘으로 에이전트를 학습하였다.

4.2 난이도 설정 대상 분석

먼저 난이도 설정 대상인 블록 개수와 이동불가위치 개수의 변화에 따른 점수의 변화를 살펴보면 다음과 같다. 게임의 규칙을 학습한 점수 기반, 특수블록 기반 에이전트로 50 회씩 반복 플레이하여 획득한 점수의 평균을 계산하였다.

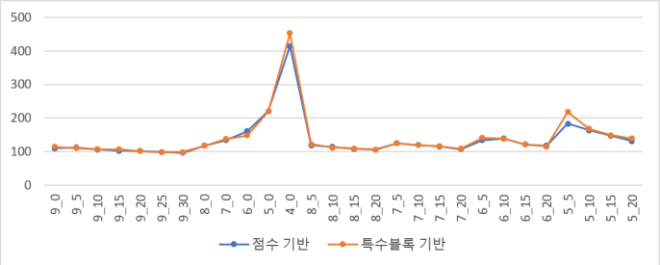


(그림 2) 난이도 요소에 따른 획득 점수 변화

하나의 난이도 요소에 변화를 줄 때 다른 요소는 고정한 상태로 획득 점수의 변화를 살펴보았다. 블록의 개수가 적으면 매치가 쉬워지므로 획득 점수가 증가하며, 이동불가위치 개수가 증가하면 매치가 어려워지므로 획득 점수가 감소하는 것을 확인할 수 있다. 이를 통해 두 요소는 난이도 설정 대상으로 적합한 것을 확인할 수 있다.

4.3 스테이지 비교 분석

앞서 학습된 에이전트를 다양한 난이도의 스테이지에서 50 회씩 반복 플레이하여 획득한 점수의 평균을 비교하였다.



(그림 3) 스테이지별 획득 점수 변화

(그림 3)에서 x 축은 스테이지의 명칭이며 y 축은 획득 점수를 나타낸다. 스테이지의 명칭은 '블록개수_이동불가위치개수'의 형식으로 표현된다. 전체적으로 비슷한 수준의 획득 점수를 보이며, 총 28 개의 스테이지 중에 19 개의 스테이지에서 특수블록 기반 에이전트가 좀 더 나은 성능임을 확인할 수 있다.

<표 2> 에이전트에 따른 스테이지 난이도 순서 차이

구분	획득 점수별 내림차순 정렬
점수 기반	4_0, 5_0, 5_5, 5_10, 6_0, 5_15, 6_10, 6_5, 7_0, 5_20, 7_5, 6_15, 7_10, 8_5, 6_20, 8_0, 7_15, 8_10, 9_5, 9_0, 8_15, 7_20, 9_10, 8_20, 9_15, 9_20, 9_25, 9_30
특수블록 기반	4_0, 5_0, 5_5, 5_10, 5_15, 6_0, 6_5, 5_20, 6_10, 7_0, 7_5, 8_5, 6_15, 7_10, 8_0, 7_15, 6_20, 9_0, 8_10, 9_5, 8_15, 7_20, 9_10, **9_15, 8_20(a)**, 9_20, **9_30, 9_25(b)**

앞서 (그림 3)의 그래프를 획득 점수를 기준으로 내림차순 정렬하여 스테이지 순서를 <표 2>로 정리하였다. 에이전트에 따른 난이도 순서에 차이가 있는 경우 회색 음영으로 표시하였다. 이를 통해 현재 스테이지 중에서 4_0 일 때 가장 쉬운 난이도이고, 9_25/9_30 일 때 가장 어려운 난이도라는 것을 확인할 수 있다. <표 2> (b)에서 9_25 가 가장 뒤에 위치하지만, 9_30 과 9_25 의 점수 차이는 0.6 에 불과하므로, 난이도의 차이가 거의 없음을 알 수 있다. 마찬가지로 <표 2> (a)에서 9_15, 8_20 의 획득 점수 차이는 1.42 정도로 거의 차이가 없다. 이를 통해 서로 다른 난이도 요소로 구성된 스테이지에서도 획득 점수를 기반으로 난이도를 비교할 수 있었다.

이러한 과정을 통해 게임의 규칙을 학습한 에이전트가 여러 난이도의 스테이지를 플레이하면서 획득한 점수를 기준으로 상대적인 난이도를 비교할 수 있음을 확인하였다. 이는 유저의 플레이 데이터가 없더라도 스테이지를 난이도 순서로 정렬할 수 있음을 의미하며, 사람과 유사한 에이전트를 개발하지 않더라도 게임의 레벨 디자인에 참고할 수 있는 난이도 기준을 획득할 수 있음을 의미한다.

5. 결론

Match-3 Game 의 난이도를 테스트하는데 있어서 사람과 유사한 수준의 에이전트를 개발하는 것이 가장 효과적이라고 할 수 있다. 그러한 에이전트의 개발을 위해서는 비교 대상인 유저의 플레이 데이터가 필요한데 데이터를 얻기 어려운 경우가 많다.

플레이 데이터가 없더라도 게임의 규칙을 학습한 에이전트가 다양한 난이도의 스테이지를 플레이하고 획득한 점수를 통해 상대적인 난이도를 파악할 수 있다는 것을 확인하였다.

이는 난이도 설정에 기준으로 삼을 수 있을 것이며 이를 통해 게임 레벨 디자인에 효과적으로 적용할 수 있을 거라 생각된다. 또한 Match-3 Game 뿐만 아니라 난이도 설정이 중요한 다른 게임에도 적용할 수 있을 것이다. 본 연구에서는 난이도 설정 대상을 한정하였으나, 다양한 밸런스 요소를 추가한다면 폭넓게 활용할 수 있으리라 기대된다.

참고문헌

[1] Kamaldinov, Ildar, and Ilya Makarov. "Deep reinforcement learning in match-3 game." 2019 IEEE conference on games (CoG). IEEE, 2019.

[2] Hingston, Philip. "A turing test for computer game bots." IEEE Transactions on Computational Intelligence and AI in Games 1.3 (2009): 169-186.

[3] Gudmundsson, Stefan Freyr, et al. "Human-like playtesting with deep learning." 2018 IEEE Conference on Computational Intelligence and Games (CIG). IEEE, 2018.

[4] Schulman, John, et al. "Proximal policy optimization algorithms." arXiv preprint arXiv:1707.06347 (2017).

[5] 박대근, and 이완복. "매치 3 게임 플레이를 위한 PPO 알고리즘을 이용한 강화학습 에이전트의 설계 및 구현." 융합정보논문지 11.3 (2021): 1-6.

[6] Shin, Yuchul, et al. "Playtesting in match 3 game using strategic plays via reinforcement learning." IEEE Access 8 (2020): 51593-51600.

[7] Kristensen, Jeppe Theiss, Arturo Valdivia, and Paolo Burelli. "Estimating player completion rate in mobile puzzle games using reinforcement learning." 2020 IEEE Conference on Games (CoG). IEEE, 2020.

[8] 신유철. "강화학습 기반 매치 3 플레이테스팅 연구." 한국컴퓨터정보학회 학술발표논문집 29.2 (2021): 611-612.