

Playtesting in Match 3 Game Using Strategic Plays via Reinforcement Learning

YounKyung Jang, 2022.03.28

논문 선택 계기

- 수많은 Reinforcement Learning Algorithm이 쏟아지고 있음.
- 알고리즘의 성능의 좋을수록 문제를 잘 풀게 될까?
- OpenAI Gym / Mario 등 이외의 custom한 환경에서의 RL
- 알고리즘 선택 그 이후?
 - Observation Space 정의
 - Action Space 정의
 - 수많은 실험 진행 및 최적해 찾기
- 생각했던 Flow
 - Custom 환경 만들기
 - Observation Space / Action Space / Algorithm 모듈화
 - 실험 결과 보기 (parameter tuning)
 - Specific 환경에서의 최적해 찾기

Introduction

- In Match 3 games, the game screen essentially comprises blocks and obstacles, and the players are required to solve missions assigned to each level before exhausting a given number of moves, by moving objects such as blocks and items
- In Match 3 games, the game screen essentially comprises blocks and obstacles, and the players are required to solve missions assigned to each level before exhausting a given number of moves, by moving objects such as blocks and items
- The biggest concern for Match 3 game designer is to design 'levels' to keep users continually interested in the game
 - Measuring the difficulty of each level



Related Works

- In Match 3 games, the game screen essentially comprises blocks and obstacles, and the players are required to solve missions assigned to each level before exhausting a given number of moves, by moving objects such as blocks and items
- The biggest concern for Match 3 game designer is to design 'levels' to keep users continually interested in the game
 - Measuring the difficulty of each level, error checking via playtesting, and establishing overall level balance are often extremely time-consuming
 - Creating new game levels, the job of a level designer also involves estimating the intended difficulty of a level by personally playing that level and considering the placement of building blocks or tools.
 - The development of a level is time-consuming because of these processes, and the subsequent modifications of level-related parameters based on heuristic experiences often produce unexpected results.

Methods

- Proposed Methods
 - playtesting using strategic plays, and the difficulty of a level is measured by the number of moves allowed for the level.
 - Randomly generated boards are learned using advantage actor-critic (A2C) to determine the current state of the board
 - Selecting blocks offering the biggest rewards from among the blocks that enable strategic action.
 - The numbers of moves used in completing the missions by a player and the proposed Strategy Agent and are compared and the performance of the method is evaluated in Jewels Star Story,² a Match 3 game.
- Game Environment
 - 9×9 2D grid environment, blocks, obstacles, and special blocks are placed
 - Available Missions
 - Collecting blocks • Removing obstacles • Sending item to specific position • Creating special block
 - Information for Strategy
 - Position of blocks by colors • Positions related to mission • Positions of movable blocks.

Methods

- Definition of Strategic Play in Match 3 Game
 - Defines five strategic plays that are commonly used in missions
 - Level Difficulty : the number of obstacles / moves is increased / special blocks / map design ..
- 5 Strategies



FIGURE 1. Various strategic plays used in Match 3 games: (a) removing mission-related blocks, (b) eliminating blocks of a specific color, (c) using associated special blocks, (d) creating a special block, and (e) removing matched blocks.

Methods

- Association between Moves and Mission and Difficulty Analysis
 - The level designer assigns an appropriate number of moves to each level based on the corresponding perceived difficulty during the development process.
 - The overall level balancing and enjoyment gained by players are optimal only when difficulty levels are set according to planned intentions.
 - Level balancing fails -> Low User Retention !!!

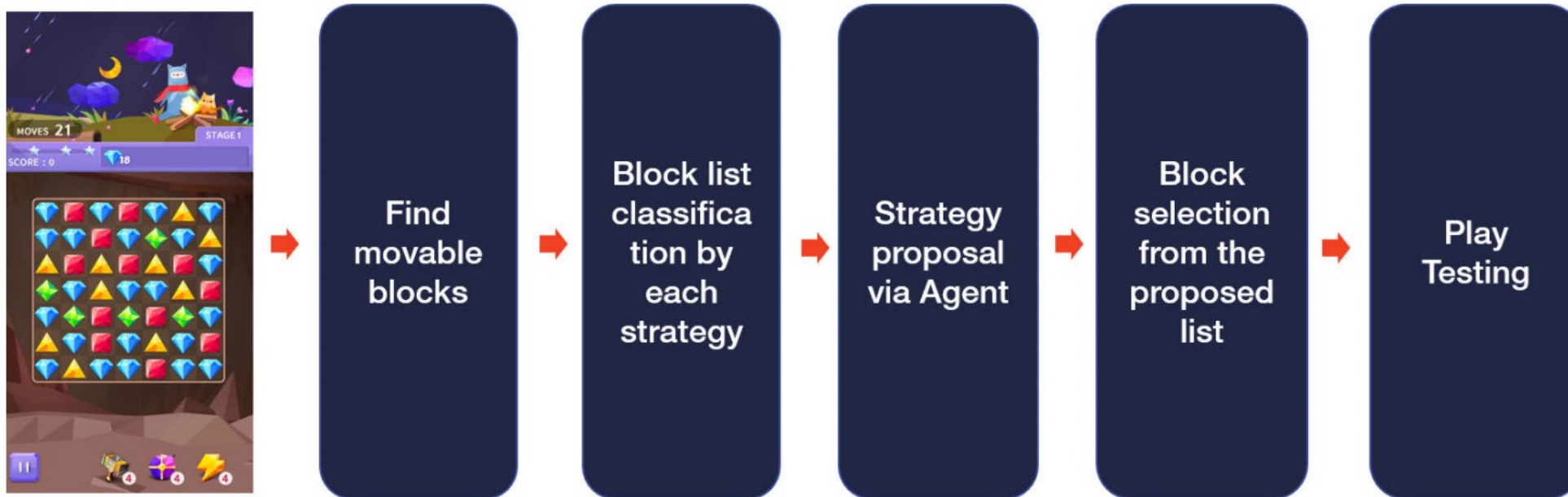


FIGURE 2. Overview of the playtesting process using the Strategy Agent.

Methods

- A2C AGENT
 - A2C agent, a policy-based reinforcement learning method
 - Current state of the board is updated whenever the agent selects a block via a network using OpenAI Gym interface and TCP/IP
 - The probabilistic values of the five strategic plays are determined at this step
 - The action with the highest probabilistic value is selected. If it does not correspond to the state of the board, the action with the second highest probability is selected.
 - The agent is trained to learn the best strategies by checking the current state of board and mission using a policy-based method
 - Find and remove blocks that can complete the mission >> to simply match blocks.

Methods

- Agent Input and Output
 - A2C agent, a policy-based reinforcement learning method (Actor and Critic)
 - 1) Positions of movable blocks — “1” and “0” represent movable and non-movable blocks
 - 2) Positions of grids related to mission — “1” and “0” represent grids related and unrelated to the mission
 - 3) Block types defined on board — “0” represents blocks undefined on the board, and numbers greater than “0” and less than “1”
 - 4) Colors defined on board — “0” and “1” represent the colors of the blocks and special blocks

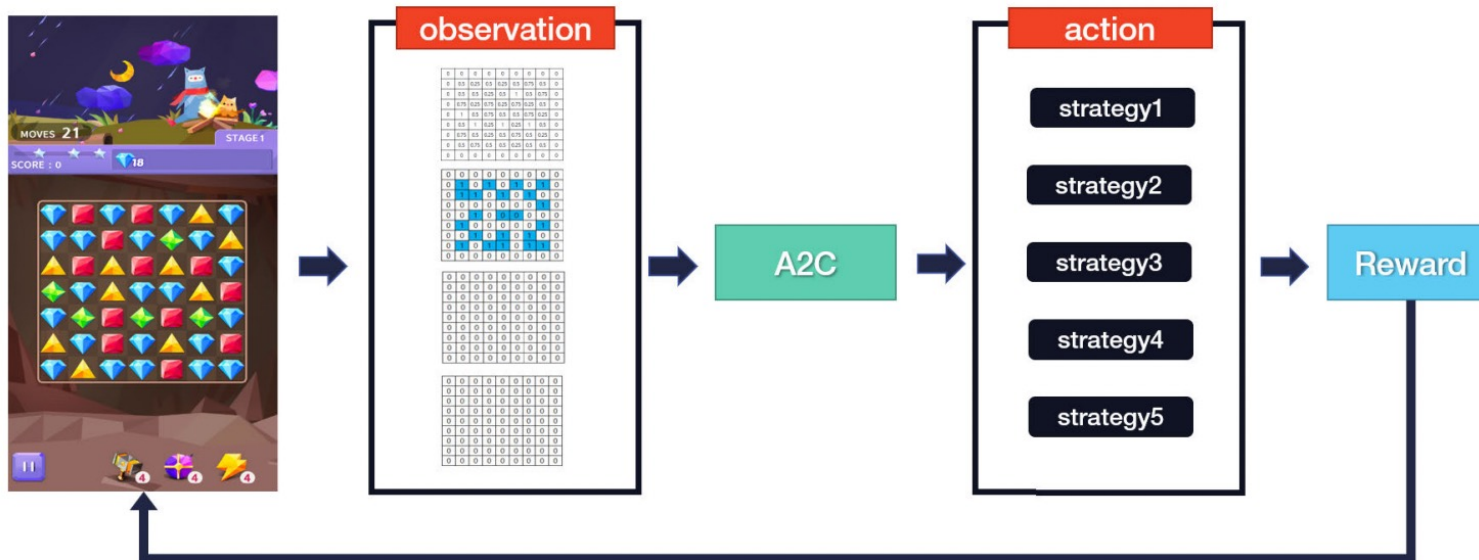
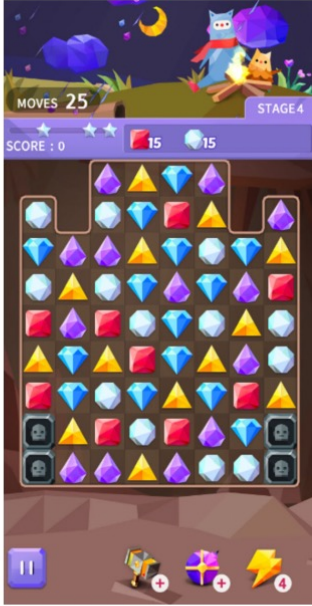


FIGURE 3. The Strategy Agent trains the Agent using the A2C algorithm, and the input returns the probability values corresponding to the five strategies that can be used with the four Feature Maps, and the training is performed via compensation.

Experiments

- Comparison of the number of Moves (random / a2c agent / human)



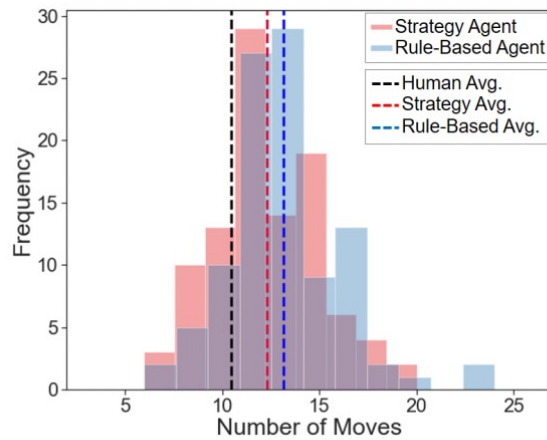
(a)



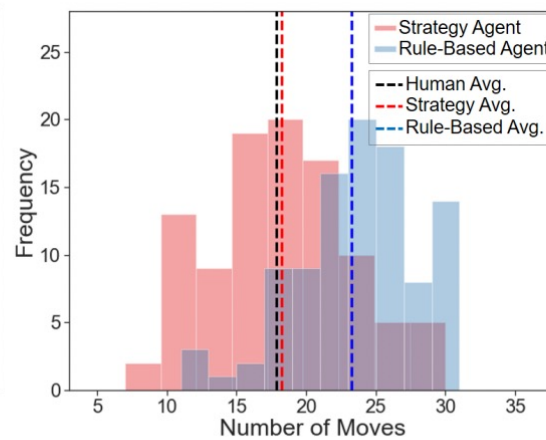
(b)



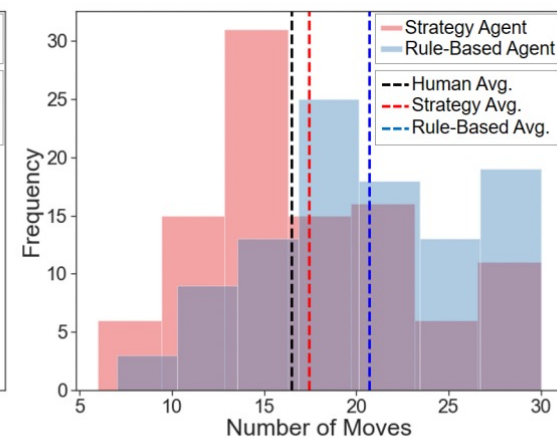
(c)



(d)



(e)



(f)

Questions

- Observation Space
 - 블록별 레이어
- Action Space
 - Movement 위치 & 방향
- Reward Functions
- 알고리즘
- 구현한다면?
 - 게임 연결 - Unity ml agents / Open ai
- 게임 쪽 Custom RL 논문 / open sources

References

- <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9034187>
- <https://github.com/andxet/unity-match-three>
- <https://www.youtube.com/c/NicholasRenotte/videos>
- <https://github.com/karan1149/candy-crush-rl>