아무도 하지 않는 게임에서 딥러닝까지 아득바득 써가며 스코어보드를 점령하기까지

By 보더콜리

BBConf 24 Winter

🤔 흥미로운 사실

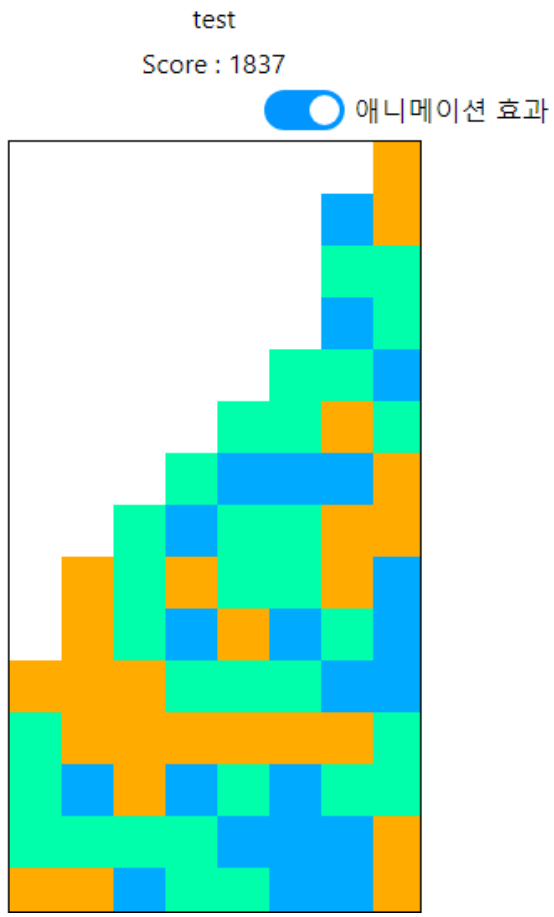
- 이거 자소서에도 썼음
  - 대체 무슨 정신이었을까 싶지만...
    - 지금 회사가 IT 서비스 회사가 아니라서 잘 봐준듯 하다

"선행 프로젝트의 부재에도 문제를 창의적으로 해결하다"

퍼즐게임을 플레이하는 AI 봇을 개발하는 토이 프로젝트를 진행한 적이 있습니다. 이 프로젝트는 Poptile이라는 웹 기반 퍼즐게임에서 리더보드 1등을 달성하는 AI 봇을 개발하는 것이 목표였습니다. 문제는 이 게임에 대한 분석 자료가 사실상 전무했기 때문에 따라 할 수 있는 레퍼런스가 없다는 점이었습니다. 저는 이 문제를 해결하기 위해, 다음과 같은 접근을 취했습니다.

Table of Contents

🧩 들어가며

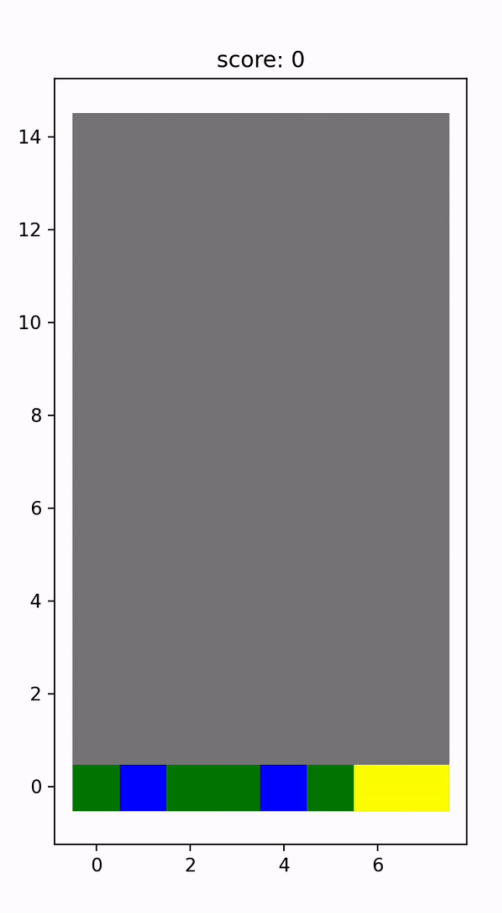


SINGLE PLAY LEADERBOARD

Name	Score	Touches	Score per touch
Magister2	559,657	5,369	104.2
뽕뽕이	545,676	3,205	170.3
포키리	395,687	2,154	183.7
Magister	336,864	3,342	100.8
다크모드 요망	279,350	1,757	159
judy5	242,427	1,517	159.8

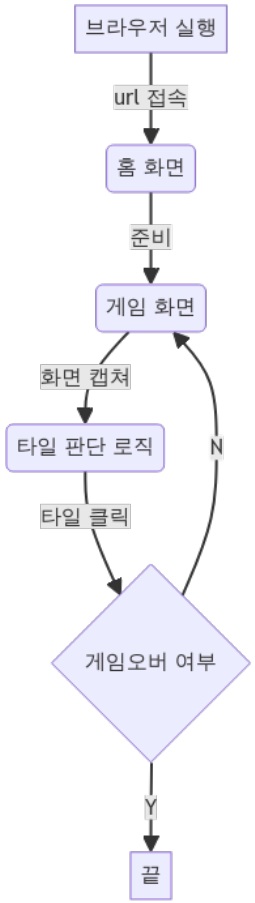
- 아직 안해본 분은 해보십쇼. 재밌습니다.
- https://poptile.panty.run/

🔨 학습에 사용할 poptile 환경 구현



- 붙어있는 같은 색 타일끼리 다 터뜨리기
- 그 다음에 빈 공간 아래로 밀어넣기
- 구현 난이도는 대충 G4~G5정도 되는 것 같음

🔨 웹에서 실제 플레이할 bot 개발



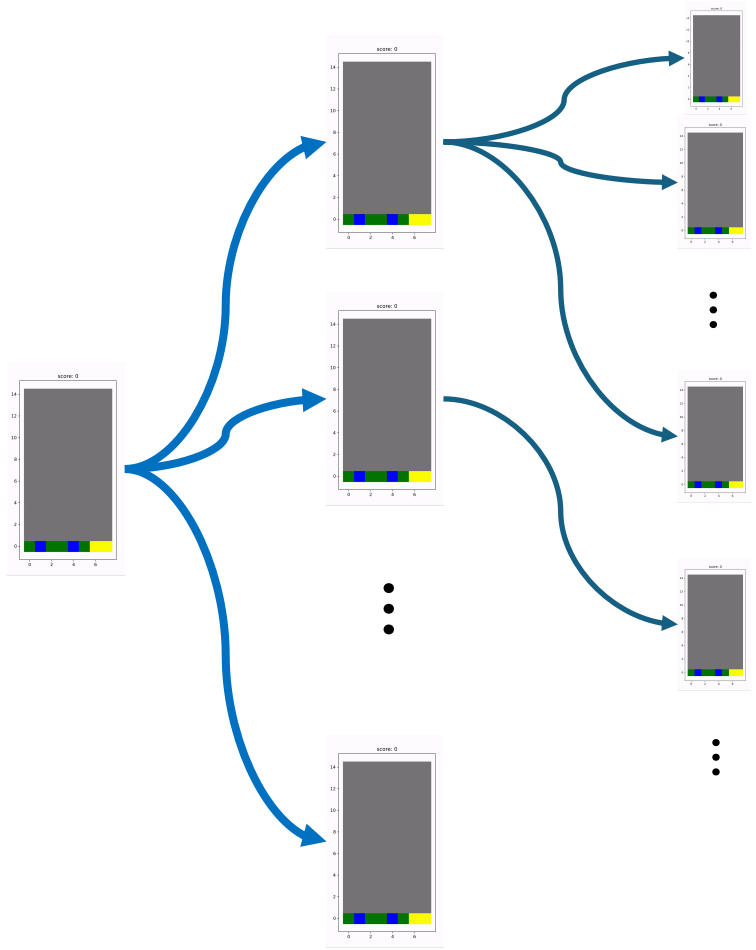
- selenium으로 구현했음
- html 버튼이 아니라 canvas라서 별 수 없었습니다.
- 대충 캡처하고 np.array로 저장하고, 판단 로직 거쳐서 click한다는 뜻

🔨 휴리스틱으로 baseline 설정

```python
def state_value(board: Board) -> int:
    temp_values = {
        'top_height': get_top_height(board),
        'components': count_components(board),
        'var': avoid_valley(board)
    }
    return sum([
        temp_values['top_height'] * 100,
        (temp_values['components'] ** 2) / 10,
        temp_values['var'] * 10
    ])
```

- state_value : 상태가 얼마나 좋은지에 대한 휴리스틱 함수
- state_value를 가장 좋게 만드는 action을 선택하는 전략 = BFS 하겠다는 뜻입니다
- depth=1은 너무 띨빵하고, depth=3은 너무 오래 걸린다

🔨 휴리스틱으로 baseline 설정



- state_value : 상태가 얼마나 좋은지에 대한 휴리스틱 함수
- state_value를 가장 좋게 만드는 action을 선택하는 전략 = BFS 하겠다는 뜻입니다
- depth=1은 너무 띌빵하고, depth=3은 너무 오래 걸린다

😴 중간평가

Magister	336,864	3,342	100.8

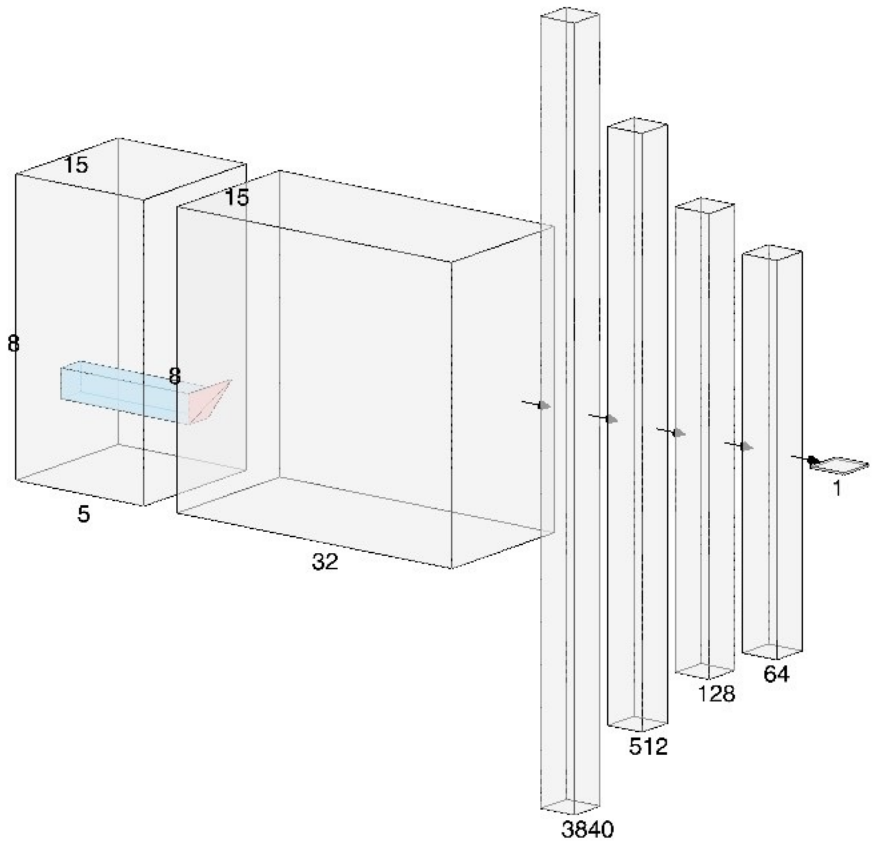
- 게임오버 되면 계속 다시 시도하도록 틀어놓고 잤음
- 자고 일어나니까 33만점(당시 1등) 떴음.
- 귀찮아서 방치했음

😴 중간평가

포키리	395,687	2,154	183.7
Magister	336,864	3,342	100.8

- 포키리님 누구세요? 덕분에 2년만에 다시 잡았습니다.

🔨 (유사)DQN 구현 – 모델 구조



- 이것저것 시도해 보았지만 3x3 conv 1층, Dense layer 4층 정도가 가장 좋았음.
  - dense만 쓰면 ‘붙은 타일’이라는 당연한 패턴을 너무 힘겹게 학습함
- policy network도 써봤는데 전혀 학습이 안됨. 이유는 못찾음
  - 아마도 중력에 의해 타일이 재조정 되는 패턴이 학습하기 어려운 것 같다고는 생각중
- Residual block도 두텁게 쌓고 이것저것 다 해봤는데 별 효과 없었음

🤔 첨언) 사실 엄밀한 의미의 DQN은 아님

Q(s,a) = r + a * maxQ(s′ + a′)

Q(s) = r + a * maxQ(s′)

헤@으응

- 원래 DQN은 Q(state, value) 로 표현하거든요
- 그래서 model도 *policy = Network(state)*를 사용하는게 일반적임
- 근데 말했다시피 안돼서, *value = Network(state)* 로 1-depth BFS할란다 하고 타협했음

🔨 (유사)DQN 구현 – ‘reward’ 정의하기

```
future_reward = future_reward * discount + game[idx].score - game[idx - 1].score
```

- Q – learning에 대해 배워볼 시간입니다

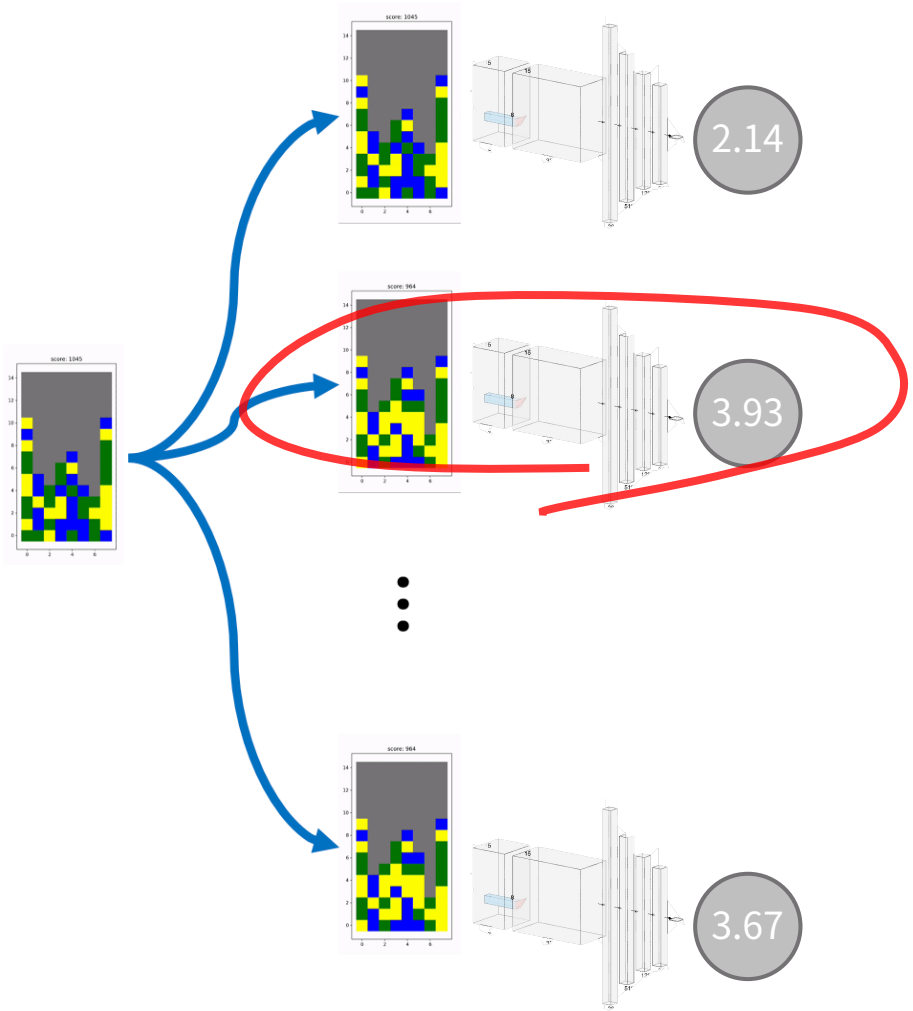
- reward는 dqn에게 줄 ‘얼마나 좋은지에 대한 척도’
- 잠깐 구리더라도 나중에 좋아진다면 반영해야 하니까 discount * 미래 reward를 줘야함

🔨 (유사)DQN 구현 – ‘reward’ 정의하기

```
future_reward = future_reward * discount
future_reward += np.sqrt(game[idx].score - game[idx - 1].score) * (1 - STABILITY_ALPHA)
future_reward += (15 - self.get_top_height(game[idx - 1].state)) * STABILITY_ALPHA
```

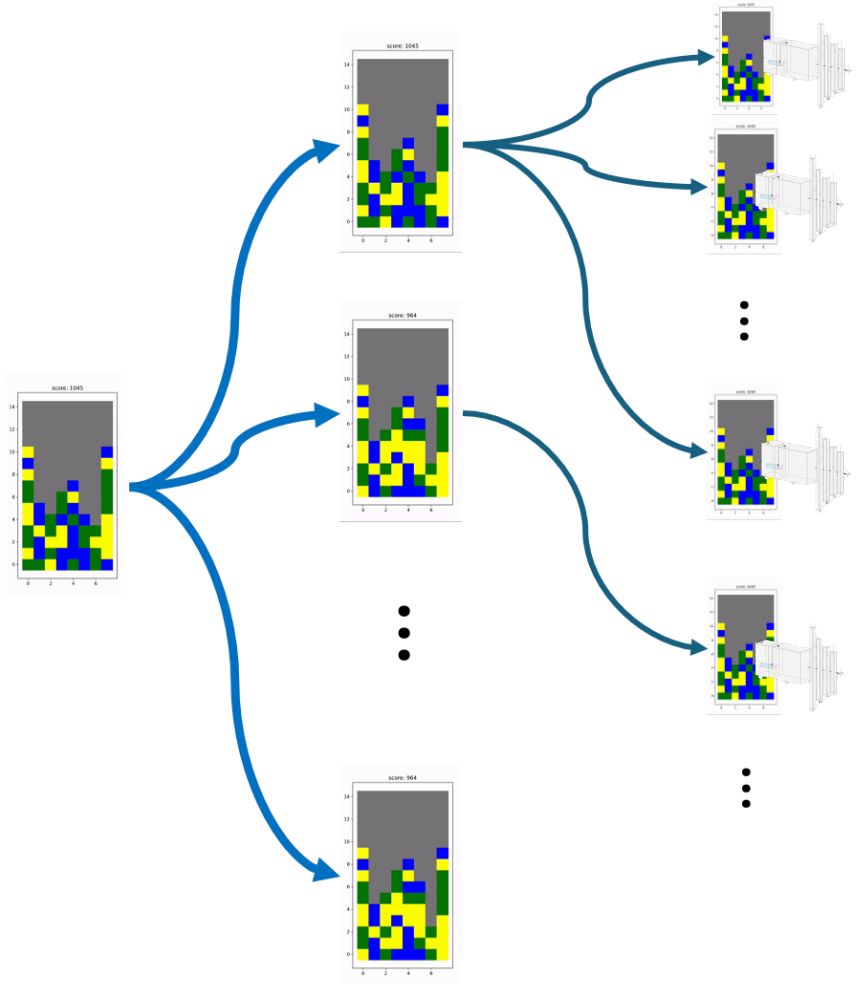
- 실제로 쓴건 이렇다
- 일단 score가 터트린 타일 갯수 제곱으로 늘어나서 훈련에 방해됨
- 안죽는게 우선이라서 덜 위험할수록 어드벤티지를 줬음
- (heuristic이랑 결국 비슷해졌죠?)

🔨 (유사)DQN 구현 – Agent의 행동 전략



- 학습용으로는 딱 한 단계 lookup 했음 (BFS 했다는 뜻)
  - 모든 타일에 대해 depth=1로 시도해보고, 점수가 높았던 걸로 선택했다는 뜻

🔨 (유사)DQN 구현 – Agent의 행동 전략



- 학습용으로는 딱 한 단계 lookup 했음 (BFS 했다는 뜻)
  - 모든 타일에 대해 depth=1로 시도해보고, 점수가 높았던 걸로 선택했다는 뜻
- 스코어링 할때는 depth=2
  - 최적화 하려면 pruning, MCTS같은거 할 수 있었을 것 같기는 하다

😴 스코어링

Name	Score	Touches	Score per touch
Magister2	559,657	5,369	104.2
포키리	395,687	2,154	183.7
Magister	336,864	3,342	100.8

- 자고 일어나니까 55만점 떠서 1위 탈환.
- 100만점 띄우려다가 걍 종료했습니다.
- 또 뺏기면 다시 할 생각 있음

💬 잡설

- 여러분도 도전해보세요.
  - 또 기록 깨지면 저도 다시 해볼라니까
  - 코드 필요하면 공유할 수는 있는데 하나도 정돈 안되어있음.

End of Document

감사합니다