2023 年 1 月 10 日 B3 西村昭賢

進捗報告

1 やったこと

- 「何もしない」を行動空間に含めた際に「何もしない」が損であるかどうか学習できているかの確認
- ランダムに対戦するプレイヤー同士の対戦環境におけるバランス調整
- 構築環境への DQN の適用実験

2 「何もしない」を行動空間に含めた場合の学習

コストや, プレイヤーの HP を追加した環境において, 行動空間に「何もしない」という行動を追加した場合には結果としてルールベースの相手に対して高い勝率を残すことができなかった.

2.1 実験

そこで表 1 に示す行動空間に従い, 学習序盤と学習後期として 30000 ステップ, 3000000 ステップとステップ数 DQN で実験し 10000 回対戦を実行し勝率とエージェントが選択する行動を観測し「何もしない」という行動が損であるかどうか確認した.

表 1: 実験で定義した行動空間 (太字は「何もしない」に対応する行動)

行動説明	次元数
手札 1 ～ 9 を自盤面に出す	9
手札 1 ～ 9 を自盤面に出さない	**9**
自盤面 1 が敵盤面 1 ～ 5 に攻撃 or 敵プレイヤーに攻撃	6
自盤面 2 が敵盤面 1 ～ 5 に攻撃 or 敵プレイヤーに攻撃	6
自盤面 3 が敵盤面 1 ～ 5 に攻撃 or 敵プレイヤーに攻撃	6
自盤面 4 が敵盤面 1 ～ 5 に攻撃 or 敵プレイヤーに攻撃	6
自盤面 5 が敵盤面 1 ～ 5 に攻撃 or 敵プレイヤーに攻撃	6

2.2 結果

表 2 に勝率を示す. 3000000 ステップ学習したエージェントのほうが高い勝率を残している. また表 3 , 4 にエージェントがとった行動の総数を示す.

表 2: 勝率

ステップ数	勝率
30000	0.1169
3000000	0.3656

表 3: 30000 ステップ学習したエージェントの 10000 回の対戦実行における行動総数上位 5 つ (盤面のカードの攻撃については実験に関係ないので除外)

行動説明	総数
手札 1 を盤面に出さない	92698
手札 2 を盤面に出さない	87254
手札 3 を盤面に出さない	81907
手札 4 を盤面に出さない	61406
手札 5 を盤面に出さない	37569

表 4: 3000000 ステップ学習したエージェントの 10000 回の対戦実行における行動総数上位 5 つ (盤面のカードの攻撃については実験に関係ないので除外)

行動説明	総数
手札 1 を盤面に出さない	109802
手札 2 を盤面に出さない	76544
手札 2 を盤面に出す	72931
手札 3 を盤面に出さない	62698
手札 1 を盤面に出す	58919

2.3 考察

学習が進むにつれ, 手札を盤面に出す行動の回数が増えていることが分かる. しかし, 依然として「盤面に出さない」という選択肢も選択されている. これは行動空間の与え方によるエージェントのターンエンドに原因があると考えられる. エージェントは表 1 に与える行動空間に沿って, 盤面と手札に存在する全てのカードについて何かしら行動をしていたらターンエンドとしている. この方法で学習が上手く行ったカードのコストが無い以前の環境は, 手札について考えると「何もしない」すなわち盤面に出さないという行動を選択することはあえてカードを出さないという戦略的な意味を持った行動となる.

しかし, 今回のカードのコストが存在するカードでは,「何もしない」という行動はあえてカードを出さないといった戦略的な意味を持った行動だけでなく, プレイヤーのコスト的にどうしても盤面に出せないからターンエンドを迎えるために選択するといった戦略的には意味のない行動にもなりうる. そのため学習の際に「何もしない」という行動が過大評価されて上手く学習が進まないと考えられる.

3 ランダムに行動するプレイヤー同士の対戦におけるバランス調整

ランダムに対戦する動詞の勝率は先攻, 後攻の勝率は約 50 % となった.

3.1 実験 1

後攻プレイヤーの「初期手札枚数」,「初期コスト」,「最大 HP 」の変更がどれほど後攻プレイヤーの勝率に影響を及ぼすかを後攻の勝率が 55 % を超えるまで調べた. 各パラメータの値の初期値はそれぞれ 3, 1, 20 である. なお, シミュレーションの際は 50000 回対戦を実行し, 10000 回おきに勝率を平均し 5 個の値の平均値を勝率とした. 表 5 に結果を示す.

初期手札枚数, 初期コストは値を 1 変えただけで後攻の勝率が 55 % を超えて約 6 割まで増加した. HP の最大値を変更させた場合は 24 から勝率が 55 % を超え始め, 26 , 27 付近で初期手札枚数, 初期コストを 1 変更した場合と同じ程度の勝率を記録した.

表 5: 後攻プレイヤーのパラメータを変更することによる勝率の変化 (55 % 超えは太字で示す)

変更説明	勝率
初期値	0.49424
初期手札 3 → 4	**0.59276**
初期コスト 1 → 2	**0.59096**
HP 最大値 20 → 21	0.51692
HP 最大値 20 → 22	0.53216
HP 最大値 20 → 23	0.54676
HP 最大値 20 → 24	**0.56362**
HP 最大値 20 → 25	**0.58058**
HP 最大値 20 → 26	**0.58834**
HP 最大値 20 → 27	**0.60598**

この変化の度合いをカードのパラメータ調整に反映したかったが, 各パラメータがどんな影響を及ぼしてどう調整に反映するのか検討がつかず, 現在行き詰まってます. (例えば初期手札枚数を変えて勝率が大きく変わったということは, 初期ドロー運に大きく左右されるということでデッキ全体のカードバランスが悪いと考えられるが, これをどうバランス調整に持ち込むのか)

3.2 実験 2

実験 1 と趣向を変え, バランスブレイカーの検出およびデッキ全体のナーフバフによるバランス調整を検討した.

ハースストーンの異なるデッキ間のバランス調整に取り組んでいた研究 [1] で用いられていた WRP (Win Rate when Played) を取り入れ, デッキ内のカードのバランスを確認した.

表 6 に示す実験で用いるデッキは以前から変わらない.

表 6: デッキの内容

ID	攻撃力	HP	コスト	特殊効果	枚数
0	1	1	0	無し	2
1	2	1	1	無し	2
2	3	2	2	無し	2
3	4	3	3	無し	2
4	5	4	4	無し	2
5	2	2	2	召喚	2
6	2	3	3	召喚	2
7	1	1	1	循環	2
8	1	3	2	循環	2
9	2	1	2	速攻	2
10	3	1	3	速攻	2
11	1	2	2	攻撃	2
12	2	3	3	攻撃	2
13	1	1	1	治癒	2
14	2	1	3	治癒	2

カードの特殊効果は,

- 盤面に出したら (攻撃力 , HP) = (1 , 1) のユニット追加で出す. (召喚)
- 盤面に出したら自プレイヤーの HP を 2 回復　(治癒)
- 盤面に出したら敵プレイヤーの HP を 2 削る　(攻撃)
- 盤面に出したら自プレイヤーは 1 枚カードをドロー　(循環)
- 盤面に出たターンに攻撃できる (速攻)

となっている. なお, 以下に結果として示しやすいようにカードに ID を割り振っている. デッキがミラーでランダムに対戦した場合の WRP を表 に示す.

表 7: カードごとの WRP (降順)

カード ID	WRP
4	0.56502
3	0.54911
6	0.54446
5	0.54069
12	0.53849
8	0.53489
2	0.53134
7	0.52547
11	0.51873
14	0.51707
10	0.51669
1	0.51444
13	0.51408
9	0.51043
0	0.50788

この値の最大値と最小値が何らかの基準以下になるようにナーフとバフを繰り返すことで, バランス調整できそうと考えているが, カードにおける 攻撃力 , HP , コストのどの数値を調節すればよいのかが定まっていないためこの実験も行き詰まっている.

2 つの実験に共通して, 調整するカードのパラメータ (攻撃力 , HP , コスト) を何にするか決定ができていない.

4　構築環境への DQN の適用

研究発表会で頂いた指摘, 気づいた点など踏まえて修正して実験を行った. まず, アルゴリズム 1 に示すように, 対戦相手の行動について相手のカードが全てプレイヤーに飛んできた時に負けの場合においては敵盤面の中で最も攻撃力が高いカードを攻撃するように改良した. また, 実験結果においては勝率と学習時の報酬の推移だけでなく, エージェントのとった行動の総数, エージェントがプレイしたカードの種類とその総数を記録して行動の傾向を示すことができるようにした. アルゴリズム 1 の対戦相手に対して, 表 8 に示す行動空間で 1000000 ステップ先攻で学習させ, 10000 回対戦を実行した.

結果は, 勝率が 0.9739 となった. 表 10 にエージェントがとった行動の総数の上位 5 個を示す. また, 表 10 に各カードにおけるエージェントがプレイした総数を示す. これらを追加することで考察に幅が出ると考えられる. 今回の実験では, 表 10 からわかるように相手プレイヤーをひたすら攻撃して先攻有利を押し付けている.

Algorithm 1 敵の行動

```
for 手札のカード do
  if 盤面にプレイできる then
    カードをプレイ
  else
    pass
  end if
end for
for 自盤面のカード do
  if 敵の盤面に 1 回の攻撃で倒せるカードがある then
    そのカードを選んで攻撃
  else
    if 敵の盤面のカードの攻撃力の総和が自分の残り体力以上 then
      敵盤面の中で最も攻撃力が高いカードを攻撃
    else
      敵プレイヤーを攻撃
    end if
  end if
end for
```

表 8: 実験で定義した行動空間

行動説明	次元数
手札 1 ～ 9 を自盤面に出す	9
自盤面 1 が敵盤面 1 ～ 5 に攻撃 or 敵プレイヤーに攻撃	6
自盤面 2 が敵盤面 1 ～ 5 に攻撃 or 敵プレイヤーに攻撃	6
自盤面 3 が敵盤面 1 ～ 5 に攻撃 or 敵プレイヤーに攻撃	6
自盤面 4 が敵盤面 1 ～ 5 に攻撃 or 敵プレイヤーに攻撃	6
自盤面 5 が敵盤面 1 ～ 5 に攻撃 or 敵プレイヤーに攻撃	6
ターンエンド	1

5 今後の課題

- ゲームバランス調整

  カードのどのパラメータを調整すればよいのか決定する方法を見つけることができていない.

- DQN 追加実験

  アグロが最適解すぎるのでブロッキングと全破壊の特殊効果を追加した実験もしてみる.

参考文献

[1] Fernando de Mesentier Silva, Rodrigo Canaan, Scott Lee, Matthew C. Fontaine, Julian Togelius, and Amy K. Hoover. Evolving the Hearthstone Meta. *arXiv e-prints*, p. arXiv:1907.01623, July 2019.

表 9: 先攻で 1000000 ステップ学習したエージェントの 10000 回の対戦実行における行動総数上位 5 つ (ターンエンドは除外)

行動説明	総数
手札 1 を盤面に出す	76955
自盤面 1 で相手プレイヤーに攻撃	52531
自盤面 2 で相手プレイヤーに攻撃	27875
手札 2 を盤面に出す	15191
自盤面 3 で相手プレイヤーに攻撃	10130

表 10: 先攻で 1000000 ステップ学習したエージェントの 10000 回の対戦実行における各カードのプレイされた総数 (降順)

カード ID	総数
0	8463
13	8069
1	7971
7	7845
11	7693
2	7672
5	7663
9	7483
8	7372
3	6910
12	6876
4	6862
6	6720
10	6630
14	6567