5-5 将棋概要

1. はじめに

今回、AlphaZero 深層学習 強化学習 探索 人工知能プログラミング 実践入門(布留川英一)内で紹介されているどうぶつ将棋のプログラムを参考に、AI と 5 五将棋で対戦することができるプログラムを作成した。

2. 5 五将棋のルール

5五将棋とは、5×五の盤で、歩兵、角行、飛車、銀将、金将、王将(玉将)を各陣1枚ずつ図1のように配置した状態から始まる将棋のことである。基本的なルールは通常の将棋と同じであるが、陣地は手前1行のみであり、一番奥の行に達したときに成ることができる。またAIの成りについて、既存の動きに新たな動きが追加され、成ることにデメリットのない歩兵、角行、飛車については必ず成ることとした。一方成ることにデメリットがある銀将については、50%の確率でランダムに判断されることとした。

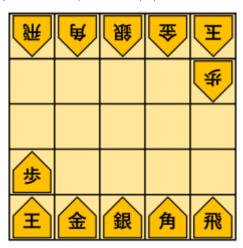


図 1-1 5 五将棋の初期譜面

- 5 五将棋の禁じ手としては以下のようなものがある.
- ・打ち歩詰め:持ち駒の歩を打つことで相手の王将を詰んではいけない。
- ・二歩:自陣の歩がいる列に持ち駒の歩を打ってはいけない.
- ・行きどころのない駒: 持ち駒の歩を敵陣に直接打つことはできない. また, 歩が敵 陣に入ったら必ず成らなければならない.
- ・王手放置の禁:王手をかけられたら必ず回避しなければならない。また、自らの王 将が王手になるような手を打ってはいけない。

・千日手の回避: 持ち駒も含めて全く同じ局面が4回以上現れる「千日手」の状態となった場合は、先手側が千日手を回避する手を打たなければならない。 ただし連続王手による千日手の場合、王手をかけている側が手を変えなければならない。

今回の5五将棋プログラムでは、打ち歩攻め、二歩、行きどころのない駒の3つの禁じ手ついては実装したが、王手放置の禁、千日手の回避の2つの禁じ手については省略した。また、千日手によりゲームが終了しないことを防ぐため、300手を越えたら引き分けとすることとした。

3. プログラムとその構造について

5 五将棋プログラムフォルダーのディレクトリ構造は図 3-1 のようになっている.

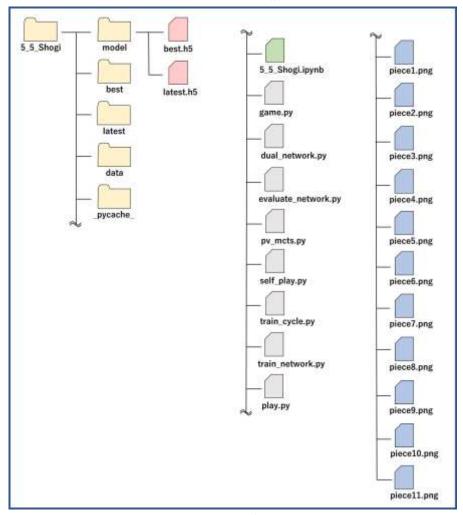


図 3-1 5 5 Shogi のディレクトリ構造

また、各フォルダ、ファイルの役割・働きは以下のようになっている.

- ・model: 1番優秀なモデル best.h5 と最新の学習によって得られたモデル latest.h5 が保存されている.
- ・best:過去の優秀だったモデル best.h5 のバックアップが保存されている.
- ・latest:過去の latest.h5 モデルのバックアップが保存されている.
- ・data: AI 同士の対戦の譜面が.history ファイル形式で保存されている.
- ・5 5 Shogi.ipynb: Google Colaboratory で学習させる際に実行するファイル.
- ・game.py:5 五将棋のルールを定める python コード
- ・dual_network.py:学習のためのデュアルネットワークの定義と作成を行う python コード
- ・evaluate_network.py: 最も優秀なモデル best.h5 と最新学習で得られたモデル latest.h5 を対戦させてモデルの評価を行い、勝った方を新たな best.h5 として更新する python コード (1 評価当たりのゲーム数:9)
- ・pv_mcts.py:手を推測する際に使用するモンテカルロ木の定義とその探索を行う python コード(1 推論当たりのシミュレーション回数:50 回)
- ・self_play.py: AI 同士の対戦と対戦譜面データを作成する python コード (自己対 戦回数:100回)
- ・train_cycle.py:AI 同士の対戦による学習の一連の流れが記された python コード
- ・train_network.py: 譜面データからネットワークを学習させ, 最新のモデル latest.h5 を作成する python コード (学習回数: 20 回)
- ・play.py: AI 対人間の対戦を行う python コード
- ・piece1~11.png:駒の画像

4. プログラムの動作方法について

4-1:学習

学習は Google Colabratory を用いて行うことを前提としている。そのほかの環境ではバージョンによって実行できない可能性がある。特に tensorflow や keras のバージョンによっては、ModuleNotFoundError や ImportError の発生、h5 形式のモデルの保存や読み込みができない可能性が高いため注意が必要である。

学習の際は 5_5_Shogi フォルダを google Drive の直下にアップロードし、5_5_Shogi.ipynb を上のセルから順に実行する. ただし、play.py と piece1~11.png の 11 個の画像ファイルについては学習に使用しないため、必ずしもアップロードする必要はない. また、best フォルダと latest フォルダは Google Colabratory のセッション切れ対策のためのバックアップを保存するフォルダであり、学習に必ずしも必要では ない、学習回数によっては Drive 容量を圧迫するため、プログラム内の best フォルダ

と latest フォルダにアクセスするコードを削除・コメントアウトしてしまってもよい.

4-2:ゲームプレイ

5 五将棋をプレイする際は、play.py を実行する. このプログラムは自身の環境で実行することを前提としている. 実行すると図 4-1 のようなウィンドウが表示され、プレイすることが可能となる. また、先手はプレーヤー、後手は AI となっている. 基本的な操作方法は以下で示す.

~駒を動かす~

- ① 動かしたい駒をクリックして選択する.
- ② 選択した駒に赤いカーソルが表示される. 選択を解除したい場合はもう一度クリックする. 選択状態が解除された場合はカーソルが消える.
- ③ 動かしたい盤目をクリックする.

~持ち駒を配置する~

- ① 配置したい駒をクリックして選択する.
- ② 選択した駒に赤いカーソルが表示される. 選択を解除したい場合はもう一度クリックする. 選択状態が解除された場合はカーソルが消える.
- ③ 配置したい盤目をクリックする.

~成る~

- ① 敵陣に入ると図 4-2 のようなウィンドウが表示される. ただし歩兵の場合, ウィンドウが表示されることはなく自動的にと金に成る.
- ② 「はい」か「いいえ」を選択する.選択するとウィンドウは自動的に消える. 「はい」を選択した場合, 駒は成る.



図 4-1 ゲームウィンドウ



図4-2 成りの確認ウィンドウ

5. 考察

参考にした AlphaZero 深層学習 強化学習 探索 人工知能プログラミング 実践入門 (布留川英一)で紹介されていたどうぶつ将棋では、evaluate_network.py の 1 評価当 たりのゲーム数は 10, self_play.py の自己対戦回数 500 回, train_network.py の学習回 数は 100 回であった.しかしながら,盤目の数の増加や「成り」というルールの実装, 禁じ手の設定などにより、どうぶつ将棋より学習にかかる時間が大幅に増加してしま った. 実際に, Google Colaboratory の GPU を利用しても, train_cycle.py の 1 サイク ルに約5時間かかってしまった. そのため, self_play.py の自己対戦回数を100回に減 らすとともに、データ数の減少による過学習を防止するため train_network.py の学習 回数も 20 回に減らした. また, 少しでも良いモデルを残すため, evaluate_network.py の1評価当たりのゲーム数を奇数である9とした.その結果,train_cycle.pyの1サイ クルに要する時間は 1.0~1.5 時間となった. それでも時間の関係上 train_cycle.py は 34 サイクルしか実行することができず, 完成した AI はとても弱いものになってしまった. 特に、明らかな王手をかけられた際にも回避しようとしないことが多いのが問題であ る. また、学習させている途中で game.py のコードに問題があり、持ち駒の歩兵を配 置できない状態となっていることが判明した. その後訂正を行い, 続きから学習を再開 させたため、26 サイクルまでの best.h5 モデルでは持ち駒の歩兵を配置することはな かった. データ数や学習回数の少なさに加え, このようなミスも完成した AI の弱さの 原因となっていると考えられる.

6. 所感

今までの課題でも CPU の学習に限界を感じることが多かったが、今回の5五将棋の学習は CPU ではほぼ不可能なものであった。そのため Google Colabratory を使用したが、90 分でセッションが切れる問題により学習データを破棄してやり直さなければならないことが多々発生した。また、Google Colabratory の GPU 使用可能時間が1日当たり12時間であると思っていたが、実際には5~6時間で使えなくなり、24時間待たなければならなかった。さらには、参考にした AlphaZero 深層学習 強化学習 探索 人工知能プログラミング 実践入門(布留川英一)で使われていた tensorflow と keras のバージョンと、Google Colaboratory でのバージョン、自身の環境でのバージョンがそれぞれ異なっていたため、keras のインポートや h5形式のファイルをロードできない問題に直面した。いままでバージョンの違いについてあまり意識してこなかったが、場合によってはソースコードを大きく書き換えなければならないこともあることを知り、これからは注意しなければならないと感じた。

~参考文献~

- ・布留川英一. AlphaZero 深層学習 強化学習 探索 人工知能プログラミング 実践入門. 初版, 東京, ボーンデジタル. 2019, 333p.
- ・強化学習 AlphaZero20(どうぶつ将棋). PYTHON と RPA で遊ぶ. 2019-10-14. https://ailog.site/2019/10/14/zero12/, (参照 2022-08-05).
- ・新堀穂高. 機械学習を用いた AI 開発. 2020-2-1. https://www.info.kindai.ac.jp/~takasi-i/thesis/2020_17-1-037-0120_H_Shinpori_thesis.pdf, (参照 2022-08-10).