

▼ AI 竜星戦 2018 参加プログラム「nlp」アピール文

開発責任者 若井 建志

「nlp」の思考部のソースコードは開発責任者が作成しているが、基本的な構成としては

1. policy network、2. rollout policy、3. value network (間に合えば！) の3部分によるモンテカルロ木探索という、Silver, Huangらが2016年に発表したAlpha Go（文献1）と同様である。

またnetwork構成、機械学習の方法についても

1. policy network — residual network 20ブロック×2層一人間の棋譜の着手を教師としてSGD (Stochastic Gradient Descent) により学習

2. rollout policy — Bradley-Terry モデル一人間の棋譜の着手を教師としてMM法 (Minorization-Maximization) により学習（文献2, 3）

3. value network — residual network 10ブロック×2層一人間の棋譜の着手と勝敗を教師とし、マルチタスクのSGDにより学習（文献4, 5）。ただし勝敗を出力する部分（value head）のみを使用

という既存の手法である。

唯一「nlp」に特徴的と言える点は囲碁の局面の表現に、佐藤らが提案し（文献6）、たかはしのんき氏が囲碁プログラムに適用（文献7）したBWモデルという数理モデルを適用していることかもしれない。このモデルでは局面を、連接点を表す行列F（19路盤であれば19*19行19*19列の0-1行列）、黒石または空点を表すベクトルb（19路盤であれば19*19次元の0-1ベクトル）、白石または空点を表すベクトルw（同）で表現する（あとコウの位置を表すベクトルが必要）。連接点は、碁盤上のある点からみて同じ連に含まれる点、あるいは同じ連に含まれる点に隣接する点と定義されている。

このBWモデルにより、着手による局面の変化が論理演算を中心として容易に求められる。隣接関係を基礎としたモデルであり、碁盤の形状を隣接点を表す行列F0（19路盤であれば19*19行19*19列の0-1行列、自分自身も隣接点とみなす）で定義するため、盤端を特別扱いする必要がなくなりアルゴリズムが単純化される。連のダメの数や位置なども論理演算で簡単に得られるため、連の詳細情報を保持する方法よりは低速と思われるが、実用的な速度でかつ実装の見通しが立てやすい利点が大きい。「nlp」（19路盤の場合）ではプログラム言語 C++のbitset<19*19>によりベクトルb, wを、vector<bitset<19*19>>（要素数19*19）により行列F, F0を表現している。

参考文献

1. Silver D, Huang A, Maddison CJ, et al. Mastering the game of Go with deep neural networks and tree search. Nature 2016; 529: 484-489.

2. 美添一樹, 山下 宏 (松原 仁, 編) . コンピュータ囲碁—モンテカルロ法の理論と実践. 共立出版, 東京 2012, 157-164.
3. Coulom R. MM法のサンプルソース, <https://www.remi-coulom.fr/> Amsterdam2007 (最終閲覧 2018.11.25) .
4. Silver D, Schrittwieser J, Simonyan K, et al. Mastering the game of Go without human knowledge. *Nature*. 2017; 550: 354-359.
5. 山口 祐. 実践！GPUサーバでディープラーニング. 第3回 GPUサーバで作るミニ囲碁プログラム（前編）. *Software Design* 2018; 5月号 112-117.
6. 佐藤真史, 穴田浩一, 堤 正義. 囲碁の数理モデル化とその応用, 第16回 ゲーム・プログラミング ワークショップ 2011 論文集 2011; 100-103.
7. たかはしのんき. グラフ理論を用いた囲碁 数式一覧, <http://www.nonkit.com/javascript/equations3.html> (最終閲覧 2018.11.25) .