Akira

渡辺 順哉

2018 年 11 月 26 日

1 概要

学生のときに研究で使っていた 9 路盤のプログラムをスタートにして、最初の AlphaGo 論文を参考にしつつ趣味で開発しているプログラムです。現状では学習局面の自動生成に手を出していないこともありまだまだ弱い（2018 年 12 月の cgos でレート 2500 くらい）です。とりあえず自分よりは強くなった気はするので最初の目標は達成しましたが、最終的には AlphaGo Lee ぐらいの強さにすることを目指しています。また、「楽しさ」の観点から、学習済みモデルを特定のプレイヤー（武宮先生とか秀策など）の棋譜で追加学習する等してプレイヤーの個性を実現することにも興味を持っています。

2 探索部

探索アルゴリズムは PUCT を使っています。

3 学習

3.1 PolicyNet

PolicyNet はフィルター数 128 で 15 ブロックの ResNet を TYGEM9d の棋譜約 140 万局で学習し（一致率は約 57%）、そこから cgos の強いプログラムの棋譜を使って少し Finetuning したものを使っています。

3.2 ValueNet

ValueNet はフィルター数 192 で 13 層のものを KGS,TYGEM,AYA の棋譜合計約 400 局面で学習したものを使っています。ただ、あまり精度は高くなく、ValueNet を入れても自己対戦で勝率 8 割程度にしか向上していません。

3.3 Playout Policy

3x3 パターンやその他のよくある特徴を MM 法で学習したものを用いています。また、プレイアウトの初手は PolicyNet の着手確立で選択しています。

4 並列化

ルートノードから深さ 1 でスレッド数分のノードを展開して、展開した各ノードをそれぞれ親ノードとした PUCT 探索を各スレッドに行わせます。そして、探索終了時の各親ノードのプレイアウトと ValueNet の勝率に PolicyNet の着手確立を加味した評価値を基準に手を選択しています。なお、PolicyNet の着手確立を加味しているのは、探索部があまり正確な勝率を返してくれず、勝率だけで手を選択しているとたまにとんでもない手を打ったりするのでその対策としてです。

あまりいい並列化の手法とは思いませんが、とりあえず実装してみたら少し強くなったのでそのまま使っています。

5 今後

一番の課題は棋譜の自動生成などで ValueNet の精度を上げることです。Zero 方式にも興味はありますが、計算資源の問題やプログラムの楽しさの観点から、しばらくは現状のスタイルで開発を続けたいと思っています。