



第27回世界コンピュータ将棋選手権 GAN将棋アピール文章

Katsuki Ohto 2017/5/4

GAN将棋コンセプト

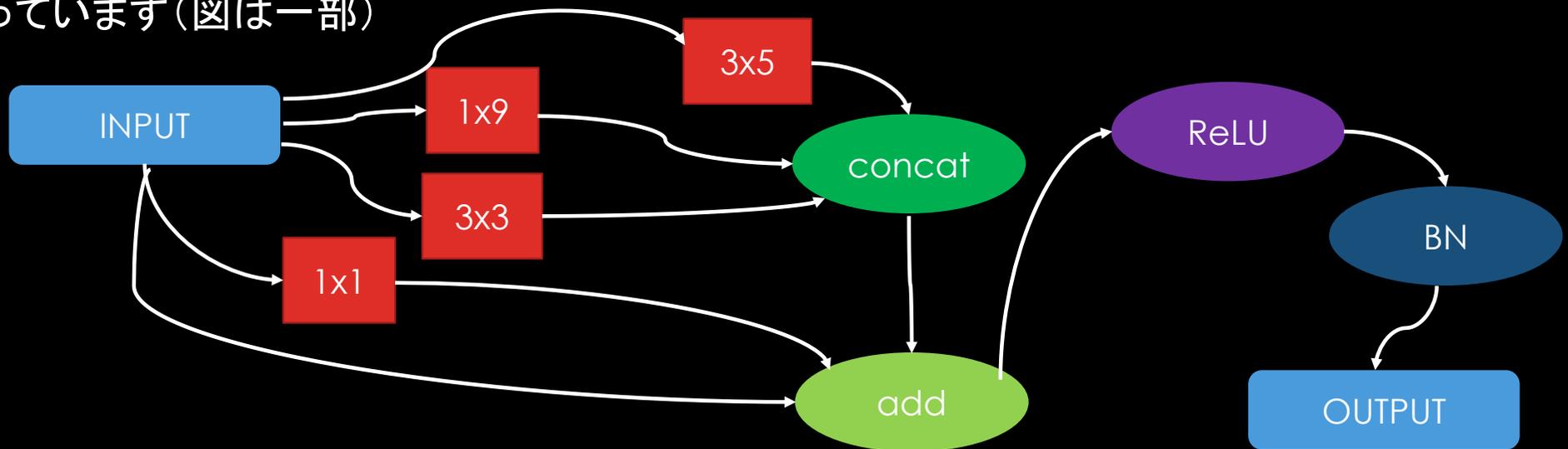
- GAN (Generative Adversarial Network) を使って評価関数を学習したいと思ってこの名前で登録
- Ponanza Chainer 発表後、既存ライブラリ + ディープラーニングで強くする方向に方針転換
- Tensorflowライブラリを使ったディープラーニング方策単体のプログラムShogiNetが一応動くところまでいったが、それ以上は何もできませんでした
- そういうわけで次頁からはShogiNetの紹介に移ります

SHOGINET概要

- ShogiNet ... ディープラーニングで学習した1手も読まない将棋方策関数プログラム。
2017/5/2時点でfloodgateでのレート(最近2週間)は1370程度
- 学習元データ...プロ棋譜6万局、floodgateの棋譜5万局、uuunuuunさんの公開棋譜2万局程度(計 1700 万局面) uuunuuunさんありがとうございます
- 全ての(局面, 手)からのシンプルな教師あり学習。王手ラッシュまで再現...
教師の手との一致率 51.8% (非合法手も候補に入れると 48.4 %)
- ベースにはAperyプログラムを利用、<http://github.com/YuriCat/apery> の nn ブランチで管理しています

SHOGINETのネットワーク

- 一致率が最も高い最新モデル(の各層)は以下のような構成で、全13層、内部は192チャンネルです
- 将棋の駒は様々な動き方をするので、それに合わせて色々な形の畳み込みフィルターを使っています(図は一部)



SHOGINETの入力

- ニューラルネットへの入力画像は 11×11 のサイズ(盤端にパディングあり) で以下の計107チャンネル
後手番の場合は盤面を反転させて入力

盤上の駒 14×2

持ち駒 (歩は4枚、香桂銀金は2枚を限度とし、超過は無視) 16×2

盤上の駒の利き 14×2

香車、角系、飛車系の隠れた利き 3×2

各マスに利いている駒の数 (自分と相手に別々、最大3つまで考慮) 3×2

AperyのCheckInfo::dcBB 1

AperyのCheckInfo::pinned 1

歩のある筋 1×2

成りゾーン 1×2

盤内 1

SHOGINETの出力

- 駒の移動元と移動先を最終層で別々に出力
- 移動元 盤面 11×11 + 持ち駒 7 の計 128 クラス
- 移動先 盤面 $11 \times 11 \times$ 駒種 14 (成駒含む) の計 1694 クラス
- 移動先を駒種別に分けることで、「ここにこの駒があればいいな」という感覚の獲得を目指しています
- 移動元、移動先のそれぞれを分類問題とし softmax cross entropy を誤差関数として学習
- 合法手集合からの分類ではないこと、2つの分類問題に分割していることで少々いびつな感じですが
- 実行時は移動元と移動先でそれぞれ確率を出力し掛け合わせて手のスコアとします

感想

- ネットワークの構成や入出力を色々といじってみました。それ以前のデータの量と質の問題がとても大きいようです
- 現在は教師の手に一様に合わせるだけの学習ですが、より発展させてネットワークの弱点(探索するプログラムが選んだ手の価値の差が大きい局面)を重点的に学習するとさらに強くなると思います
- 序盤が綺麗で、中盤以降は頑張れば勝てそう、劣勢でも逆転の目があり絶望感がない(どうせ読めてないし)ということで人間の対戦相手としては面白い存在だと感じました