

チーム Barrel house のアピール文書@第 29 回世界コンピュータ将棋選手権

Barrel house とは岡山の駅前にあるビアバーです。昨年チームメイトを求めさすらって行きついたところですが、マスターに許可頂いたので名前をお借りしました。その後、メンバーが変わって当初の方向性とは全く違って来ましたが、まあ一度出した名前を変更するのもアレなので今年もそのままです。

プログラム名：Kristallweizen

昨年の Hefeweizen が濁った白ビールでしたが、Kristallweizen はフィルターでろ過した透き通った白ビールです。命名経緯は上記のビアバーに起因します。

命名時は昨年より洗練されたソフトにしたいとの気持ちでしたが、準備ペースは昨年同様で洗練とは程遠い感じです。ちなみに、マスターによると Kristallweizen は樽で輸入するのが難しいので瓶になりますが大丈夫ですかとのことでした。祝勝会では数が揃わないので Hefeweizen で乾杯する予定です。

チームの特徴

昨年は段取りが分からず全関係者を加えておりましたが開発者に絞りました。今年からのメンバーもお願いして参加頂きました。若いメンバーも大口スポンサーありませんので、多方面に精通した開発者の独創的なアイデアだけが勝負どころと考えています。昨年は独創的な評価関数生成と先行先読み Multi Ponder が冴え渡り優勝致しましたが、今年は別のブルーオーシャンへ漕ぎ出す予定でしたが結局レッドオーシャンだったようです。

昨年同様別々に開発を進めてネット上で連絡を取り合う程度の疎結合チームです。

CSA 使用可能ライブラリ使用表明

最終的には以下のものだけ利用しました。

やねうら王, tanuki-, 技巧

ライブラリ選定理由

やねうら王：探索部が高速なため主に探索部の利用。定跡部作成時にも利用。

tanuki-：評価関数に利用。

技巧：Multi Ponder のオーダーリングに利用。

使用マシン

普通のノートパソコンに加えて、クラウドの力をお借りしました。昨年同様 AWS の m5.24xlarge を 5 台使用しました。

クラスタリングについて

昨年 Multi Ponder のクラスタリングを行いました。今年は Go 言語で再実装して追加機能を加える予定でしたが通信遅延の隠ぺい程度の成果に終わっています。リアルタイムで各ノードの情報が得られるようになったので今後の追加機能の下準備ができたと考えます。

評価関数の設計思想

中終盤の評価値が拮抗する難解な局面において、あえて最善手を選ばずに、次善手を選択するようにすることで、相手の読み筋を意図的に外して Ponder hit させない。これは昨年の Hefeweizen で実現していた特徴で、将棋倶楽部 24 で有段者の方々とスパーリングを行って細かく調整した賜物である。将棋倶楽部 24 のユーザーさんから「まるで大山将棋のようだ」という、絶賛の声をいただいた。調整にご協力いただいた将棋倶楽部 24 のユーザーの皆様、この場を借りて御礼申し上げます。

Hefeweizen 同士の対局で生成した棋譜を教師局面としているため、千日手、手数制限による引き分けの棋譜が多数あったためか、異様に粘り強く、負け辛い指し直しを行う。また、すぐに入玉したがるようになっている。いいんだか悪いんだか……。

学習部自体はやねうら王 Ver4.83 を使用し、人造棋士 18 号以来の改良として、低スペック PC でも学習を行えるようにメモリ下限を下げる修正を行い、評価関数はやねうら王にマージされている NNUE (tanuki-チーム提供) を使用している。これは、WCSC28 で Hefeweizen を開発していた時から感じていた「KPPT の学習の頭打ち感」を打破したいという思いがあり、打破するにはどうしたらいいかと考えていたところで NNUE が公開され、やねうら王にマージされたことから、流行に乗ってみようという安易な発想から採用した。

また、WCSC28 で優勝した KPPT 型の評価関数の Hefeweizen の持つ、中盤のねじり合いの局面では最善手をあえて外して次善手を選択するようにし、相手の Ponder hit を外すことによって探索部の特徴を最大限に活かす指し手選択を行うという特徴を、そのまま NNUE 型にも引き継がせることを目標にして作成した。

評価関数ができあがったのが、5/2 の 7:56 (タイムスタンプより) という、ギリギリの状態であったが、親である Hefeweizen の棋風を受け継ぎつつ、その棋力を向上させた NNUE 型の評価関数とすることができた。

完成までの行程

1. NN 構成の選択

やねうら王にマージされている NNUE のデフォルトの NN 構成は、HalfKP-256x2-32-32 となっており、KPPT の探索速度に匹敵するスピードを得ることを目的に最適化されており、学習精度、探索速度のバランスでこれを超える NN 構成を模索したが、選手権までに発見することはできなかったため、最終的にデフォルトの NN 構成を採用した。

ただ、学習方法の工夫によって、学習パラメータの選択方法がおぼろげながら見えてきたため、再び NN 構成の検証を再開する予定である。

2. 教師局面生成用評価関数

初期学習のための教師局面生成に WCSC28 で優勝した Hefeweizen を使用して居飛車、振り飛車を指させ、depth10 で約 3.5 億局面を生成した。本当はもっと生成したかったのだが、手元にあるマシンリソースの関係上、これが限界だった。そのため、この少ない教師を効率よく学習させる工夫を行って使い回すこととした。

Hefeweizen は KPPT 型であるが、人造棋士 18 号の評価値調整ツールによって汎化が比較的うまくいっており、中終盤の指し手の多様さが魅力である。この評価関数が生成する教師局面によって、実戦で出てきやすい局面に対して様々な応手で対局した局面が生成できたため、深層学習の学習時の問題点である過学習に比較的陥りにくかったと考えられる。

また、生成時には千日手や手数制限によって引き分け扱いになった対局データも多数生成されるのだが、それもそのまま教師データとして学習に使用している。選手権時に他の開発者と学習について話す機会があったのだが、引き分けデータは勝率に結びつかないとして教師データから取り除いているとのことであったが、千日手も 0.5 勝という立派な勝ちであるので、これを取り込まない手はないと思う。そのせいか、異様に粘り強く、辛い指し直しを行うようになった。

3. 学習方法の工夫

NNUE の学習、というかやねうら王に実装されている学習部の仕様は、1epoch 目のベクトルの傾きによって学習度合い、按点のハマリ具合が大きく左右されることがわかっている。この問題に対処するため、以下の手順で学習することで、安定した学習を行えるようにした。以下、これを彫刻作業になぞらえて説明する。

3.1. シード (種) 学習 …… 木の切り出し

学習ゼロの状態でききなり大量の教師データで学習した場合、1epoch 目の学習ベクトルの傾きが芳しくないと、いくら大量に学習したとしても棋力が向上しないことが検証により判明した。こうなってしまうと学習をやり直すしかないのだが、こういった無駄な時間を少しでも減らすため、居飛車、振り飛車両方の局面をほどよく混ぜた教師データを 200 万局面用意し、eta 1、lambda 1 で学習させ、ロスと accuracy を確認し、問題があれば再度やり直すという「傾きガチャのリセマラ」を短時間で繰り返すことによって、不要な過学習、鞍点へのハマリを抑えるようにした。

これはなかなか骨の折れる作業であるため、自動化も考えたが、自動化のための基準をコーディングする手間のほうがかかってしまうのは本末転倒であるので、目視によって確認していった。

3.2. ミニミニバッチ学習 …… 粗削り

生成した約 3.5 億の教師データを、1 ファイルにつき 200 万局面 (つまりシード用の教師データと同じ単位) に分割し、それを適当なフォルダに放り込んでおき、1 ファイルずつランダムに取得しながら (targetdir でフォルダ指定するのではなく、Linux の ls コマンドの結果をランダムソートして for ループで取り出している) nn_batch_size 10、batchsize 10000、eta 1、lambda 1 で学習するループを適度に行った。ファイルを読み込む度に初期ベクトルは変わり、バッチサイズも小さいため精度も低くなるが、多様な傾きでの学習が行われることで、精度は低いものの、それなりに均等な値を持たせることができたと思っている。

3.3. 通常のミニバッチ学習 …… 彫刻刀による彫刻

約 3.5 億の教師データを今度はシャッフルしながら 1 ファイルにまとめ、nn_batch_size 1000、batchsize 10000000、eta 0.1、lambda 0.5 で loop パラータで回数指定して学習を行った。こうすることで、起動時に傾きガチャを行って下る方向を決めたら、後はその傾きで一気に学習を行うことで、粗削りな評価値を大きめのバッチサイズで精度を上げつつ、accuracy を上げていった。その際、適宜 lambda を調整しながら勝敗項稼ぎのループ、勝率項稼ぎのループを交互に回しながら上げていった。この段階では、move accuracy は 37% が限界だった。

3.4. 最終調整 …… やすりかけ

3.3 の状態では、教師の素である Hefeweizen にも勝ち越せない程度でしかなく、途方に暮れていたのだが (この時点で 5/1 だった)、最後にやけくそで eta 0.01、lambda 0.1 で 3.3 の学習を 1 ループだけ行ったところ、奇跡的に move accuracy がぐんぐん伸び、最終的に 42% まで上昇した。この状態で仮想敵としていた orqha1018+やねうら王 4.83 と 5000 万ノード指定で 10 対局させたところ、8 勝 1 敗 1 分という結果が出た。対局数が少ないと思われるかもしれないが、5/1 時点でそれほど対局を回せるはずもなく、また、全対局の指し直しを目視していて、異様に辛くて粘り強い、ビールに辛子と納豆を入れたような棋風に仕上がっていたため、これを採用とした。

このように、学術的な根拠などまるで皆無な思いつきで作成された評価関数であるが、それでも昨年の Hefeweizen の棋風を受け継ぎつつ、その棋力を向上させることに成功したのは、僥倖としかいいようがない。

実験結果や追試について記述を求められていますが、上記のような選手権開催前日に完成した評価関数を持ち込んだため事前の実験結果というべきものは皆無です。事後実験に可能なものは公開させて頂きましたので、御興味ある方は是非実験して結果を公開して頂ければ関係者皆が喜ぶと思います。