

# 名人コブラアピール文書

松山洋章

## 概要

名人コブラは第32回世界コンピュータ将棋選手権で第4位となりました。参加プログラムの要件である「主要な開発者が思考部に技術的に何らかの明示的な工夫を施した」部分は、結果的に定跡部分のみでした。この文書ではその定跡部分の作成方法を中心に、思考エンジン、評価関数、ハードウェアについてもそれぞれ記述いたします。

## 定跡

### 使用ライブラリ

- Python-dlshogi2
- cshogi

### 使用データ

- Floodgateの棋譜データ (2019年-2022年4月分。レーティング3500以上。50手以上。)

### 開発動機

昨年大会優勝ソフトelmoの主な工夫点が定跡データであったように、優秀なライブラリの登場により強豪ソフトが多数出場するようになった結果、昨今は序盤定跡の重要性が高まったように思われます。開発開始が大会直前であったことから自身の計算資源を使用することは諦めて、Floodgateでレーティングが高いソフト同士の対局データを使用することにしました。

### 開発過程

Python言語のシンプルさや豊富なライブラリ群と、山岡氏の制作した将棋特化ライブラリ群のおかげで、極めて迅速に開発ができました。

大会前にテストをして勝率などを測ったりはしていません。2次予選の前半では、定跡フォーマットを勘違いして生じたバグにより、先手番時に定跡が利用されていませんでした。名人コブラは2次予選の後半以降から調子を上げてきたことから、定跡が無いよりはある程度強くなっているかと思われま

## 作成方法

Floodgate棋譜データにある評価値を勝率に変換したものと、対局結果による勝率をそれぞれ加工して、二つの勝率の平均値をとって局面の評価値を作成しています。

評価値を勝率に変換したものは、前方への指数移動平均を取って利用しています。指数移動平均を取ることで評価値が平滑化されます。またその方向を前方(終局方向)とすることにより、より対局結果に近い評価値となります。

勝敗情報による勝率は同一局面が1回しか出現していない場合は100%か0%になってしまう等、対局数が少ない場合には信頼性が低いものとなってしまいます。それを避けるための工夫として勝率45%のダミーデータを10個加えて、出現回数が少ない場合には勝率45%に近づき、出現回数が多くなるにつれて実際の勝率に近づくようにしています。

指数移動平均の係数やダミーデータの数などはできた定跡データを確認しながら目見当で決定いたしました。

## 思考エンジン

### 使用ライブラリ

- やねうら王

大会時最新版のやねうら王をそのまま使用しました。2次予選でやねうら王と当たったときにレアなバグが出てしまったため、Mizarさんのアドバイスに従い、オプションを変更してビルドし直しました。

### ライブラリ選定理由

昨年のWCSCでやねうら王ベースのelmoが優勝し、WCSCの比較的長い持ち時間ではやねうら王の方がdlshogiより強いと思われたため。

## 評価関数

### 使用ライブラリ

- 水匠3

### ライブラリ選定理由

確認不足のまま、第1回電竜戦用に作成したキメラ評価関数を使用したつもりになっていました。電竜戦時には多数のキメラ評価関数を作成いたしましたが、どの評価関数をWSCS32で使用したかは未確認だったため、大会後、この文書を書くために数十ある手元の評価関数と比較して確定作業を行いました。その結果、電竜戦の準備でベンチマークで使用していた水匠3を今回のWSCS32で使用していたことが分かりました。(互角局面を使用したテストで、第1回電竜戦に使用したキメラ評価関数は水匠3より僅かながら強かったです。)

### 開発過程

当初提出したアピール文書のとおり、最初はやねうら王 (NNUE) と dlshogi の評価値、局面情報を入力特徴としたスタッキング評価関数を作成し、手元の環境ではある程度の効果があることが確認できました。しかしながら本番環境でのテストをするには時間と予算が不足するために、標準 NNUE のやねうら王を単体で使用することになりました。

## ハードウェア

- Amazon EC2 c6a.metal

### ハードウェア選定理由

前回大会までは Google Computing Engine を使用していましたが、現在のところ GCE には Amazon EC2 の c6a.metal に匹敵するマシンタイプが存在しないため、今回は初めて EC2 を使用しました。GCE にも第3世代 AMD EPYC のインスタンスはありますが、スレッド数で EC2 の c6a.metal に及びません。

EC2 と GCE では用語が異なるものの機能的にはほぼ同じなため、初めてでもそれほど違和感なく使用できました。

## 終わりに

今大会ではdlshogiをベースとしたソフトが1位と2位を占めたことと、約2年前のNNUE評価関数で決勝上位に食い込めたことから、標準NNUE評価関数は学習が飽和してしまっている印象を更に強く持ちました。そのため、次回以降のWCSCではdlshogiをベースとしたソフトで出場したいと思っています。

最後になってしまいましたが、今大会ではスタッフの皆様、参加者の皆様、フリーソフトウェアの作者の皆様のおかげで、とても楽しい時間を過ごすことができました。深く感謝申し上げます。