

蒼天幻想ナイツ・オブ・タヌキ アピール文書

蒼天幻想ナイツ・オブ・タヌキ
製作委員会

野田久順 岡部淳 鈴木崇啓
日高雅俊 那須悠 河野明男

目次

- 蒼天幻想ナイツ・オブ・タヌキ
- 主な工夫点
 - 自動化・効率化
 - 自己対戦の棋譜を用いた大規模機械学習
 - 進行度推定
 - 探索パラメーターの自動調整
 - クラスタリング

蒼天幻想ナイツ・オブ・タヌキ

- 磯崎元洋氏(やねうらお)による『やねうら王』から派生したコンピューター将棋ソフトです
- 開発者がかけた手間に対するレーティング向上比を高めることを目標としています
 - 上記を実現するための技術的な工夫として、開発プロセスの自動化・効率化に力を入れています

蒼天幻想ナイツ・オブ・タヌキ(続)

- ソフト名の元ネタは『SQUARE ENIX』が運営するMMORPG『ファイナルファンタジーXIV』に登場するボス戦イベント名です
- 第4回電王トーナメント直前、やねうらお師匠とのSkypeミーティングにて

師匠「まだソフトの名前考えてないねん…」

自分「『蒼天幻想やねうら王』はどうですか？」

師匠「…」

- 結局不採用になり、自分たちで採用することにしました

主な工夫点

- 開発プロセスの自動化・効率化
- 自己対戦の棋譜を用いた大規模機械学習
 - 他のソフトより深い探索
- 進行度の推定
- 探索パラメーターの自動調整
 - 探索パラメーターの進行度による切り替え
- 楽観合議制クラスタ

開発プロセスの自動化・効率化

- 全ての実験タスクを Jenkins上で管理することにより、効率的な開発を実現しています
 - タスクのキューイング
 - タスク終了後のメール通知
 - タスク進捗状況のブラウザ上での確認



The screenshot displays the Jenkins web interface. On the left, a sidebar contains navigation links: '新規ジョブ作成' (New Job), '開発者' (Developer), 'ビルド履歴' (Build History), 'Jenkinsの管理' (Manage Jenkins), '設定の変更履歴' (Configuration History), 'Job Priorities', and '認証情報' (Credentials). Below these links are two expandable sections: 'ビルドキュー' (Build Queue) showing 'ビルド待ち' (Build Pending) and 'ビルド実行状態' (Build Execution Status) showing a single job 'tanuki-optimizer.2017-01-14' with a progress bar and build number '#56'.

The main content area shows a table of builds. The table has columns for 'S' (Status), 'W' (Weather icon), and '名前 ↓' (Name). The builds listed are:

S	W	名前 ↓
●	☀	apery_learn.2016-08-29
●	☀	generate_kifu.2016-11-29
●	☀	learn.2016-11-02
●	☁	selfplay.2016-10-20
●	☁	selfplay.2017-01-25
●	☁	tanuki-optimizer.2017-01-14
●	☁	zoukin.2016-12-05

Below the table, there is a link for 'アイコン: S M L'.

自己対戦の棋譜を用いた 大規模機械学習

- 2016年後半に流行した自己対戦の棋譜を用いた大規模機械学習を用いています
- 「浅い探索の評価値で深い探索の評価値を近似する手法」を繰り返します
- 通称『雑巾絞り』

自己対戦の棋譜を用いた 大規模機械学習（続）

※第4回電王トーナメントPR文書より再掲

- 問題を定式化すると以下ようになります

$$\operatorname{argmin}_{\omega} L(\omega)$$

ここで

ω : 調整対象となる重みベクトル
= 評価関数ファイルの中身

L : 目的関数

自己対戦の棋譜を用いた 大規模機械学習（続）

$$L(\omega) = \sum_i l(\omega, \varphi_i, \omega', \varphi'_i)$$

ここで

l : 損失関数

i : 自己対戦で生成した棋譜の局面の通し番号

φ_i : i 番目の局面に対する浅い探索のPVの末端ノードの特徴量ベクトル

ω' : 自己対戦に用いた重みベクトル=評価関数ファイル

φ'_i : i 番目の局面に対する自己対戦中の深い探索のPVの末端ノードの特徴量ベクトル

自己対戦の棋譜を用いた 大規模機械学習（続）

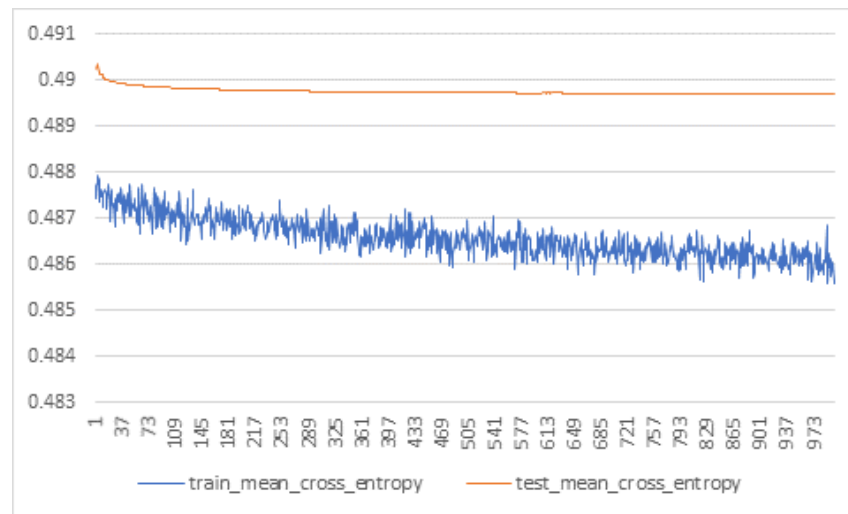
- 損失関数として考えられる関数は…
 1. 評価値の差の2乗
 2. 評価値から推定した勝率[1]の差の2乗
 3. 評価値から推定した勝率の確率分布の交差エントロピー
 4. …
- 蒼天幻想ナイツ・オブ・タヌキでは3.の交差エントロピーを用いています
 - ー 交差エントロピーの詳細については第4回電王トーナメントPR文書を参照してください

自己対戦の棋譜を用いた 大規模機械学習（続）

- 機械学習は以下の条件で行っています
 - 深い探索の深さ: $8 \cdot 10 \cdot 12$
 - 浅い探索: 0手読み+静止探索 置換表無効化
 - データ量: $5 \cdot 2.5 \cdot 1$ 億局面×複数回
 - 学習手法: ミニバッチ勾配降下法
 - 最適化手法: Adam[2] FOBOS L2正則化[3]
 - ミニバッチサイズ: 100万局面
 - 学習率: 評価値に対し $1/64$ 学習率減衰無し
 - 損失関数: 交差エントロピー
 - 次元下げ: 左右対称のみ
 - 棋譜データの評価値の絶対値の上限: 3000
- 他のソフトに比べ、より深い探索による評価値を学習データとしています

自己対戦の棋譜を用いた 大規模機械学習（続）

- 実験結果例
 - 雑巾絞り6周目
 - 探索深さ: 10
 - 生成局面数: 2.5億局面
 - 学習局面数: 10億局面
(2.5億×4周分)
 - 検証用局面数: 100万
局面



自己対戦の棋譜を用いた 大規模機械学習（続）

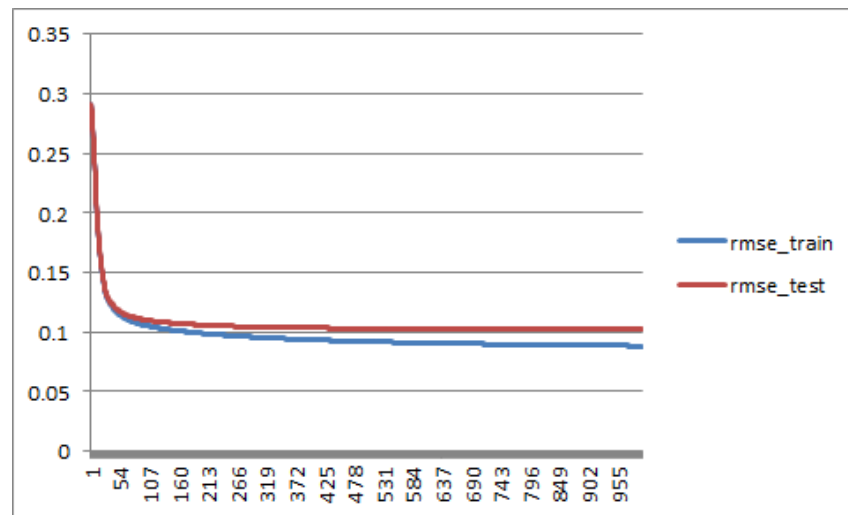
- 実際の作業手順は…
 1. 学習用の棋譜を生成する
 2. 誤差の測定に使う検証用の棋譜を生成する
 3. 機械学習により評価関数を生成する
 4. 過去の評価関数と自己対戦を行う
 5. 以下繰り返し…
- 以上をJenkins+Pythonスクリプト等で自動的に行えるようにしています
- 深さ12の探索を行う場合、1周あたりにかかる時間は10日前後です

進行度推定

- 『激指』『技巧』等が導入している進行度推定を行っています
- 特徴量にKP絶対を用いたロジスティック回帰を用いています

進行度推定（続）

- 実験結果例
 - － 学習手法: バッチ法
 - － 最適化手法: Adam
 - － 教師局面: Floodgate上
の上位ソフト同士の対
局の棋譜34,000局面
 - － 検証用局面数: 1000局
面



進行度推定（続）

- 実装は1日程で終わりました
 - 『技巧』のソースコードを参考にしつつ実装しました

探索パラメーターの自動調整

- 第26回世界コンピューター将棋選手権で採用したTree-structured Parzen Estimator (TPE)とGaussian Processを用いた探索パラメーターの自動調整を行っています

探索パラメーターの自動調整（続）

※第26回世界コンピュータ将棋選手権PR文書より再掲

- 探索パラメーターを入力、レーティングを出力とする関数を考える
- 関数の値を最大化する探索パラメーターを探す
 - 大域的最適化問題
- 得られるサンプルには大きなノイズが乗っている

探索パラメーターの自動調整（続）

- Tree-structured Parzen Estimator (TPE)
 - 機械学習のハイパーパラメーターの最適化に用いられるアルゴリズムの1つ
 - PythonライブラリHyperoptに実装されている
 - 1探索パラメーターセットあたり24対局、数百パラメーターセット分の自己対戦の勝率をサンプリングする
 - Hyperoptの出力をそのまま使ったらダメ！
 - Hyperoptは最高の値を出力した最初の値を出力する
 - サンプルの誤差が大きい場合、本来弱いはずの探索パラメーターが最終結果として出力されてしまう場合がある

探索パラメーターの自動調整（続）

- Gaussian Process (GP)
 - ノンパラメトリックな回帰分析などに用いられる
 - sklearn等の実装されている
 - Hyperoptで得られたサンプルをGaussian Processでノンパラメトリック回帰分析する
 - 探索パラメーターに対する勝率の関数の確率分布が得られる
 - サンプルの中で関数の確率分布の平均値が最も高いものを最終的な解として出力する

探索パラメーターの自動調整（続）

前回からの変更点1

- 前はサンプリングの際の思考時間が0.1秒と短かったため、短い思考時間に最適化されていた可能性が高い
- 思考時間を10秒とし、ある程度長い思考時間に適応させる

探索パラメーターの自動調整（続）

前回からの変更点2

- 将棋は序盤と終盤でゲームの特性が変わるため、序盤から終盤まで同一の探索パラメーターを用いるのは適切ではない可能性がある
- 進行度0.5を閾値に序盤と終盤で別の探索パラメーターを用いる

探索パラメーターの自動調整（続）

- 実際の作業手順は…
 - 探索パラメーターセットと勝率のサンプリングに数日～数週間
 - GPによる探索パラメーターセットの決定に数分
 - 勝率の測定に数時間
- 数週間のうち数回手動で操作するだけで作業が終わるようになっていきます

楽観合議制クラスタ

- 第26回世界コンピュータ将棋選手権予選2日目第8回戦の最中に実装した楽観合議制クラスタを使用予定
 - ノード数未定

参考文献

- [1] https://twitter.com/issei_y/status/589644174275674112
- [2] Diederik Kingma and Jimmy Ba (2015): "Adam: a method for stochastic optimization," the 3rd International Conference for Learning Representations (ICLR 2015).
- [3] Efficient Online and Batch Learning Using Forward Backward Splitting John Duchi, Yoram Singer; Journal of Machine Learning Research 10(Dec):2899 – 2934, 2009
- [4] 海野裕也, 岡野原大輔, 得居誠也, 徳永拓之 (2015): "オンライン機械学習 (機械学習プロフェッショナルシリーズ)," 講談社.

ご支援ご愛顧のほどよろしくお願い申し上げます