

将棋における個人に適応した着手推定モデルの構築

山内 智晴^{1,a)} 鶴岡 慶雅^{2,b)}

概要: 本研究では、特定のプレイヤーの数百件程度の棋譜データから、その人物の着手推定モデルを作成することを目指す。着手推定モデルの実現によって、将棋プログラムの利用法をより広げることが可能である。これを実現するための提案手法として、特定のプレイヤーの小規模な棋譜データと多数の差し手による大規模な棋譜データとを組み合わせることで評価関数の学習を行う。また、指し手の探索過程においても、プレイヤーごとの特徴である読みの深さを探索深さに反映させる。将棋ソフト「技巧」を用いて実験を行ったところ、提案手法によって着手推定の正解率の向上が確認できた。

Developing Move Prediction Models for Individual Players in Shogi

TOMOHARU YAMAUCHI^{1,a)} YOSHIMASA TSURUOKA^{2,b)}

Abstract: The purpose of this work is to develop move prediction models for individual players in Shogi by using their own several hundred records of games as training data. Such move prediction models will expand the application area of Shogi programs. In this paper, we propose a method for learning an evaluation function that uses an individual player's game records in combination with a large number of game records of professional Shogi players. We also attempt to reflect the search depth of a human player's thinking process, which is a feature of an individual player, in the search depth of a Shogi program. We conducted experiments on the proposed methods using a state-of-the-art Shogi program, Gikou. The experimental results show that the accuracy of move prediction can be improved by the proposed methods.

1. はじめに

近年、コンピュータ将棋プログラムはトッププロとの対局にも勝利するなど、強さという点においては大きな進歩を遂げつつある。また、世界コンピュータ将棋選手権の出場ソフトが市販されるといった形で実際に一般の人々によって利用される機会も増えつつある。このように一般の人々にも利用されるようになってくるにつれて、将棋ソフトウェアには単なる強さ以外のものが求められるようになって考えられる。本研究では、そのような要求の一つとして考えられる、特定のプレイヤーの差し手を推定するモデル

の作成を目標とする。本研究では、この特定のプレイヤーの指し手を推定するモデルを着手推定モデルと呼ぶことにする。

特定の個人の着手推定モデルの作成が可能であれば、将棋プログラムに新たな利用法を生み出すことが可能である。例えば指導対局やユーザが自身の希望する特定のプロ棋士や知人の指し手を模倣した将棋プログラムとの対局、コンピュータとのペアプレイ、といったことが実現できる。幅広い活用を実現するためにも、本研究ではプロ等でないような一般の将棋愛好家程度のプレイヤーの着手推定モデル作成を視野に入れる。従って、数百局程度の棋譜しか持たないようなプレイヤーであっても実現可能な着手推定モデルの作成手法について検討する。先行研究では、少数とはいえ個人としてはかなり多くの棋譜が入手可能なトッププロ棋士を対象とした研究がほとんどであり、数百件程度の棋譜で学習を行うという取り組みは本研究の特色である。

強さではなく着手推定モデルの構築を目的とする本研究

¹ 東京大学工学部電子情報工学科
Department of Information and Communication Engineering, The University of Tokyo

² 東京大学大学院工学系研究科電気系工学専攻
Department of Electrical Engineering and Information Systems, Graduate School of Engineering, The University of Tokyo

a) yamauchi@logos.t.u-tokyo.ac.jp

b) tsuruoka@logos.t.u-tokyo.ac.jp

でも、既存の将棋プログラムで広く使われている評価関数の学習手法(文献[1][2]等を参照)を基本として評価関数の学習を行う。しかしながら、着手推定精度の向上を目的とした学習データの設定やパラメータの調整が必要となる。

着手の探索についても、着手推定モデルの作成においては闇雲に深く探索することよりも着手推定の対象となる個人の読みの深さを反映することが大事である。また、読みの深さに限らず人間のプレイヤーは読み抜けをはじめとするミスも冒すこともあれば、戦術に対する得意不得意、あるいは好みもプレイヤーごとに違うものであると考えられる。これらの人間の特性は、評価関数の調整だけではカバーしきれない可能性がある。そこで、探索の過程において既存の探索手法になんらかの工夫を加えることで、これらの特性を反映した探索を行えるようにすることが課題になる。

特定の個人の棋風を再現する研究として文献[3]がある。この研究では、序盤の棋風の再現については定跡データベースを用いて上手く実現しているが、中終盤の局面評価の模倣はあまり上手くいかなかったとしている。それに対して本研究では、中終盤においても着手を推定するために定跡は用いず、局面評価と探索の工夫によってこれを実現する。

2. 関連研究

近年の将棋プログラムは、多くの場合プロ棋士の棋譜を用いた教師あり機械学習の手法で性能を高めている。例えば、2006年の世界コンピュータ将棋選手権優勝プログラムである Bonanza [4] は、トッププロの棋譜を多数集め教師データとしている。それらの棋譜中に登場する各局面において「棋譜で指された手が最善手」と仮定して教師あり機械学習の手法を用い、プログラムの出力が棋譜で実際に指された着手に近づくように評価関数を調整している [1][2]。この手法は、多くの将棋プログラムの評価関数学習の基本となっている。

文献[1]では、プログラムの出力を実際に指された着手に近づける、という操作を次の式(1)によって示される損失関数 J を最小化することによって実現している。

$$J(P_0, P_1, \dots, P_{n-1}, v) = \sum_{i=0}^{n-1} l(P_i, v) \quad (1)$$

ここで、 P_i は教師データにおける各学習局面、 v は評価関数の各特長量に対する重みのベクトルである。また、 $l(P, v)$ はこの棋譜中の手と他の指し手の評価値の差の度合いを表現する関数であり、式(2)で与えられる。

$$l(P, v) = \sum_{m=1}^M T[\xi(p_m, v) - \xi(p_{m=0}, v)] \quad (2)$$

p_m は局面 P を合法手 m により 1 手進めた局面で、合法手の数を M 、教師データでの実際の着手は $m=0$ とする。 $\xi(p_m, v)$ は局面 p の minimax 探索の結果としての評価値で

ある。 $T(x)$ は、棋譜で実際に指された指し手と他の合法手との評価値の差を、棋譜との指し手の一致度を表す指標に変換するための関数である。例えば、 $T(x)$ としてステップ関数を用いると、 $T(x)$ は「全合法手のうち、実際の手よりも評価値の高い手の数」を表す。従って、これを最小化することで棋譜との指し手の一致度を高めることができる。ただし、最小化とはすなわち停留値を求めることになるため、実際に用いる $T(x)$ は 1 段階微分可能なものを用いると都合がよい。

将棋プログラムは、このようにして学習した評価関数を用いてゲーム木の探索を行うことによって実際の着手を決定している。ゲーム木の探索には minimax 法をベースとして枝刈りを行うことで計算量を削減した $\alpha-\beta$ 法や、さらに探索量を減らすための Null Window Search 等の手法が用いられる。これにより探索量を減らし、その分探索の深さを深くしている。また、局面の実現確率を計算し、実現確率の高い局面を優先的に探索するという手法によって局所的に深い探索を実現する手法も用いられている [5]。いずれの手法にしても、深く探索することによってより強くなる、という考え方が根底にある。

評価関数やその学習に工夫を加えることでなんらかの特性を再現しようとする試みとしては、例えばコンピュータ将棋プログラムの弱点とされてきた入玉時の評価を改善する研究を行った滝瀬らの研究 [6] が挙げられる。入玉時の評価を改善するための滝瀬らの実験の一つとして、入玉勝ちした棋譜データのみを教師データとして学習を行う、という実験がある。入玉勝ちによって決着のついた棋譜データ 6,000 局を用意し、それらの棋譜データのうち実際に入玉勝ちしたプレイヤーの側の指し手のみを教師データとして Bonanza を用いて学習を行った。文献 [6] では、これによって学習された評価関数について、テストデータとして「入玉勝ちした棋譜 1,000 局」および「非入玉棋譜 1,000 局」を用意し、これらに対する着手推定の正解率にあたるパラメータの変化を確認したところ、非入玉棋譜に対しては学習を繰り返すごとに精度が悪化していたとしている。また、入玉勝ちした棋譜についても学習 3 回程度で最もよい精度を示しており、その後は精度が悪化していた。一方学習データに対する精度は上昇していたことから、過学習に陥っている可能性があると言われている。

また、生井らは、特定の個人の棋風を再現する研究を行っている [3]。この研究では、個人の棋風を再現する手法として、「対象者の棋譜データを活用した定跡データベースの作成」と「対象者の棋譜データからの評価関数の学習」を提案している。文献 [3] では、このうち後者の評価関数学習手法については、実際に対象者の棋譜データのみを使っての学習実験を行っている。実験には Bonanza を用いて、事前に十分な量の棋譜を用いて学習した評価関数のパラメータを初期値として学習を行ったが、学習を重ね学習データ

に対する着手推定の一致率が向上するにつれ、テストデータに対する着手推定の一致率は低下するという結果になった、としている。

この生井らの実験を受けて大森らは、大量の棋譜から作成した評価関数のパラメータを初期値とし特定の棋士の棋譜で学習を行うことによって微調整をする手法に、複数の正規化パラメータを加えることを提案している [7]。具体的には、パラメータの重みを 0 に近づける項とパラメータの重みを初期値からなるべく離れないようにする項の 2 つを目的関数に追加する手法を提案している。すなわち、目的関数は式 (3) で表される $J^P(\omega)$ となる。

$$J^P(\omega) = J(P, \omega) + J_C(\omega) + J_R(\omega) + J_{R'} \quad (3)$$

ω は各特徴量に対する重みのベクトルである。また、 J_C は駒の重みに関しての制約を加える関数であり、文献 [1] に詳しく示されている。 J_R および $J_{R'}$ が大森らの着目した制約項であり、それぞれ「パラメータの重みを 0 に近づける項」と「パラメータの重みを初期値からなるべく離れないようにする項」にあたる。これらは以下の式 (4)、(5) で示される。

$$J_R(\omega) = \lambda_1 |\omega''| \quad (4)$$

$$J_{R'} = \frac{C}{2} \|\omega - \omega_0\|^2 \quad (5)$$

ω_0 がパラメータの初期値、 ω'' は重みのベクトルのうち、駒の位置関係に関する重みベクトルである。この目的関数を用いて学習を行い、損失項の重みの組み合わせと学習結果の関係について調査している。重みの組み合わせ次第では、学習した棋譜との着手の一致率を向上を抑え、過学習を抑制する効果があることが示されている。ただし、文献 [7] では学習に用いなかったデータ (いわゆるテストデータ) との一致率の測定は行われていない。

これらとは別に、人間の冒すエラーをモデル化し、コンピュータ将棋プログラムにヒューマンエラーを導入する手法について伊藤らが行った研究 [8] がある。この研究においては、ヒューマンエラーを「生物学的限界」、「技量不足」、「無意識」、「意識的」の 4 つに、また対局者の思考段階を「局面認識」、「候補手生成」、「先読み」、「評価」、「手の決定」の 5 段階に分類した上で、各種エラーを実現するためにはどの思考段階にどのような処理を加えるとよいかについてまとめている。文献 [8] では、将棋プログラムにおけるヒューマンエラーの実現方法として、先読み段階における探索ノード数や探索速度の制限、手の決定段階における特定の指し手の偏重などをあげている。

一般に、機械学習を行う際には教師データの量が重要となる。将棋の評価関数に用いられる特徴量の数は膨大である。少数の教師データによる機械学習を行った場合、教師データ中に十分現れない特徴も多く現れ、それらについて適切な学習を行うことが困難になる。また、少数の教師

データに対する指し手の一致率を高めるように学習した結果、教師データに対する過学習を起し教師データ以外のデータに対しての精度が落ちるという可能性もある。実際、Bonanza の場合では 1 万を越える評価関数パラメータの学習のためにプロ棋士および将棋倶楽部 24 から集めた棋譜計 60,000 局を用いている。この手法においては仮に同一人物による大量の棋譜が存在した場合には、それを用いることで「特定の個人の特徴を反映した評価関数」の作成が可能である。しかし実際に入手可能な棋譜は、最も対局数の多いと考えられるトッププロ棋士ですら 2,000 局程度であり、このデータのみから学習を行うことで「あらゆる局面において」対象人物の着手を推定できるような着手推定モデルを作成することは先述の問題から困難であることが予想できる。ここに示したような関連研究でも、条件を満たす棋譜データが少数しかない場合にはそれのみを利用した学習をするのではなく大量のデータによる学習を別に行って利用しており、この問題が存在することが伺える。

3. 提案手法

3.1 特定の個人の小規模データセットによる評価関数の調整

本研究では、評価関数の学習において文献 [3] で用いられた手法と同様に、事前に大量の棋譜データから評価関数のパラメータを学習し、その後改めて特定の個人の小規模データセットを用いてパラメータを調整する、という手法を用いる。関連研究において、小規模なデータセットを用いた学習には過学習のリスクがあることが示されている。しかし大規模なデータセットと組み合わせることで一定の学習成果が得られることも期待できる。また、着手推定というタスクにおいてこの学習手法と他の学習手法との比較を行うための実験 (4.3 節に示した) を行ったところこれを裏付けるような傾向が得られたため、本実験での評価関数の学習にはこの手法を採用することとした。

3.2 探索段階における個人の特徴の反映

3.1 節で示した手法に加え、個人の特徴を反映してこれを探索のパラメータとする手法を組み合わせる最終的な着手推定モデルを作成する。ここで個人の特徴とは「読みの深さ」、「得意な戦形」、「読み抜けししやすい場面」といった評価関数には直接反映されないが個人によって異なるような情報である。実際のプレイヤーは、盤面の評価の段階のみならず、プログラムで言うところの「ゲーム木の探索」にあたる読みの過程にも個性を持っている。そこで、それらの情報を探索過程に反映することでより実際のプレイヤーに近い探索を実現することができ、結果としてよりよい着手推定の正解率が得られると考えられる。そこで、これら探索に関する特徴を探索のパラメータとして数値化し、着手推定の正解率が最大となるように調整して探索を行うこと

で、既存の探索手法と比較してよりよい一致率を得ることが期待できる。

本稿では、この個人の特徴としては「読みの深さ」を用いる。実際に作成した評価関数を用いて、様々な探索深さによる着手推定を行いその正解率を比較することで、「読みの深さ」を探索に反映することが着手推定の正解率の向上に繋がることを確認する。

4. 特定の個人の着手推定を行う実験

4.1 実験の概要

本章では、提案手法を用いて実際に行った実験とその結果について述べる。まず 4.2 節において実験に用いた将棋ソフトである技巧の解説を行う。その後、4.3 節においていくつかの評価関数の学習手法を比較した実験について述べ、4.4 節では読みの深さを探索に反映させた場合の着手推定の精度の変化に関する実験について述べる。

4.2 技巧

本研究においては、実験環境としてオープンソースの将棋ソフトである技巧 [9] を用い、実験のために一部変更を加えた。学習の初期値として、技巧では Bonanza 6.0 で学習された駒割を用いている。駒割以外の特微量の初期値は 0 である。学習はミニバッチ法を用いており、ミニバッチサイズを k として、一度に k 局面のデータを利用して勾配計算とパラメータ更新を行う。このパラメータ更新 1 回を 1 イテレーションとして、これを n イテレーション繰り返す。技巧は損失関数として棋譜との一致率、勝率予測と勝敗との負の対数尤度、浅い探索結果と深い探索結果との誤差の 3 つを用いた確率的勾配降下法を用いて学習を行う。最終的な学習結果としては各イテレーションごとの学習値の指数移動平均値を用いる。本実験では、棋譜との一致率を高めることが目的であるので、単純化のためこれらのうち棋譜との一致率のみを用いることとした。学習時の探索深さは、事前に設定した最小値から最大値の間で離散一樣分布に基づきランダムに選択される。技巧での初期値は最小値が 1、最大値が 2 となっている。本実験ではこれを「学習時の探索深さは 1~2 手」等と表現している。また、学習率など学習に関する各種パラメータは、特別に記載がない限り技巧の初期値のままである。

技巧には標準で、学習を行うと学習結果をテストデータに適用し、着手の一致率を求める機能が備わっている。これは、学習時に用いた探索深さが x 手である場合に、テストデータ中の各局面について、学習中の特徴パラメータを用いて α - β 探索の手法を基本とする深さ $x+1$ 手の探索を行い、得られた着手が実際の着手と一致するかどうかを確かめ、テストデータ全体に対する一致率を求める機能である。本予備実験においてはこの機能に一部変更を加え、任意の探索深さにおいて着手推定の正解率を求めることが可

能となるようにした。また、テストデータのみならず学習データに対しても着手推定の正解率を求めるようにした。

4.3 特定の個人の小規模データセットによる評価関数の学習の実験

4.3.1 実験の設定

大規模データセットでの学習を行った後に特定の個人の小規模データセットによる追加学習を行う手法が、特定の個人の小規模データセットのみを利用して学習する手法に比べて優れていることを確認するために比較を行った。実験のために、「加藤一二九段の指した棋譜 1,100 局」ならびに「プロ棋士による 7 大棋戦+順位戦の棋譜 31,000 局」を用意した。本実験では前者を「加藤の棋譜」、後者を「7 棋戦の棋譜」と呼ぶ。「加藤の棋譜」のうち 100 局、「7 棋戦の棋譜」のうち 30,000 局を学習データとし、残りはテストデータとする。学習データの量はテストデータの量に比べ多いことが一般的であるが、本研究は数百件程度の、学習に使うにはきわめて少数の棋譜しか持たないプレイヤーに対して着手推定モデルを構築することを目標とするため、「加藤の棋譜」の学習データの量は意図的に少なくしている。また、「加藤の棋譜」に関しては棋譜の指し手のうち「加藤一二九段が指した手」のみを学習およびテストの対象としている。これらの学習データで調整した評価関数を用いて、「加藤の棋譜」のテストデータに対する着手推定を行いその正解率を求めた。本実験においては学習時の探索深さは 1~2 手、着手推定時の探索深さは 3 手とした。また、本実験において比較を行った学習条件は以下のとおりである。

7 棋戦 「7 棋戦の棋譜」の学習データで学習したもの。

(ミニバッチサイズ 8,000、10,000 イテレーション)

加藤 「加藤の棋譜」の学習データで学習したもの。(ミ

ニバッチサイズ 1,000、1,000 イテレーション)

7 棋戦+加藤 「7 棋戦の棋譜」の学習データで学習(ミニ

バッチサイズ 8,000、10,000 イテレーション)した後、

その結果を初期値として「加藤の棋譜」の学習データ

で学習(ミニバッチサイズ 1,000、1,000 イテレーシ

ョン)したもの。

4.3.2 実験結果

各学習条件によって得られた着手推定の正解率を表 1 に示した。なお、表 1 中の括弧内の数字は、イテレーション回数とその数値のときに表 1 に示した着手推定の正解率を得たことを表す。全ての学習条件において $\frac{n}{10}$ イテレーションごとに着手正答率の記録を行い、最も良かった記録をその学習条件における記録として掲載している。また、図 1 には、「加藤」の学習条件と「7 棋戦+加藤」の学習条件における 100 イテレーションごとの着手推定の正解率の変化を示した。加えて図 1 には「7 棋戦」の学習条件における最終的な着手推定の正解率 (44.33%) を参考として示した。

表 1 各学習条件による加藤テストデータでの着手正答率の最大値

	加藤テストデータに対する 着手推定の正解率 (イテレーション回数)
7 棋戦	44.33% (10,000)
加藤	42.20% (100)
7 棋戦+加藤	44.72% (300)

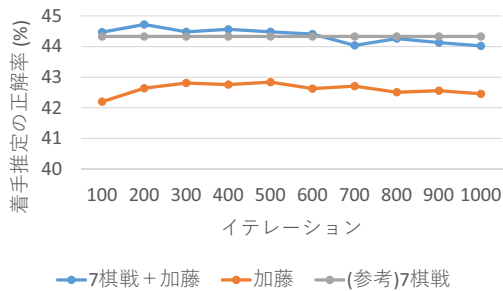


図 1 各学習条件による加藤テストデータでの着手推定率の変化

この結果によれば、「7 棋戦の棋譜」のみから学習した結果を用いて求めた着手推定結果 (44.33%) に比べ、「加藤の棋譜」のみから学習した結果を用いて求めた着手推定結果 (42.20%) が下回っている。「7 棋戦の棋譜」と「加藤の棋譜」を併用した場合 (44.72%) は、「加藤の棋譜」のみから学習した結果 (42.20%) に比べればややよい結果が得られており、「7 棋戦の棋譜」のみから学習した結果 (44.33%) と同程度である。「加藤の棋譜」の指し手である加藤一二三九段と、「7 棋戦の棋譜」の指し手であるトッププロとの実力の近さが「7 棋戦の棋譜」と「加藤の棋譜」を併用した場合 (44.72%) と「7 棋戦の棋譜」のみから学習した結果 (44.33%) の結果の近さに影響していると考えられる。また、図 1 に示した学習結果の推移を見ると、教師データ数の少ない「加藤の棋譜」による学習を用いる「加藤」の学習条件、「7 棋戦+加藤」の学習条件のいずれも、イテレーション数を重ねるごとに着手推定の正解率が低下傾向にある。これは、文献 [3][6] にも触れられている、教師データの少なさに起因する過学習に陥っていると考えられる。従って、学習が収束するのを待たずに途中で学習を打ち切ることによって、よりよい精度を実現することが可能である。この結果を踏まえて、4.4 節の実験においては「7 棋戦の棋譜」と「加藤の棋譜」を併用して学習を行うこととした。

4.4 探索深さを变化させた場合の着手推定精度の実験

4.4.1 実験の設定

個人の特徴として「読みの深さ」を考えると、これに対応する探索のパラメータとして「探索深さ」が挙げられる。そこで本実験では探索深さを様々に变化させた場合の着手推定の正解率の変化を調べた。評価関数の学習には、まず「7 棋戦+順位戦」の棋譜のみを用いて技巧の初期設定によ

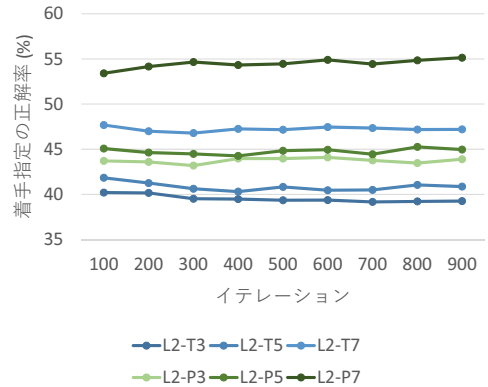


図 2 学習時探索深さ 1~2 手の場合の着手推定の正解率

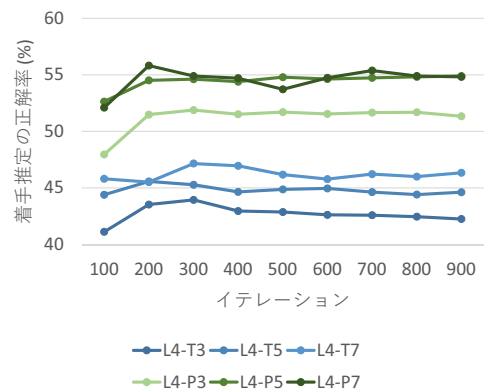


図 3 学習時探索深さ 3~4 手の場合の着手推定の正解率

る学習を行った。さらにその結果を初期値として、「加藤の棋譜」の学習データを用いて探索深さを 1~2 手、3~4 手、5~6 手の 3 通りにわけ、各々でミニバッチサイズ 1,000 の学習を行った。学習は 100 イテレーションごとに結果を保存し、最大 900 イテレーションまで行って合計で 27 通りの評価関数を得た。この 27 通りの評価関数について、「加藤の棋譜」の学習データ、テストデータそれぞれに対する着手推定の正解率を求めた。着手推定時の探索深さは 3 手、5 手、7 手の 3 通りとした。

4.4.2 実験結果

各条件における実験結果を図 2、図 3、図 4 に示す。なお、図中では学習時探索深さ $x-1 \sim x$ 手、テストデータに対する着手推定時探索深さ y 手の設定を $Lx-Ty$ としている。同様に学習データに対する着手推定の場合は $Lx-Py$ である。

全体として、900 イテレーション程度の学習でも概ね学習は収束傾向にある。図 2 から図 4 のいずれを見てもテストデータに対する着手推定の正解率は、着手推定時の探索深さ 3 手の場合およそ 40%~43%、5 手の場合 40%~45%、7 手の場合 45%~47% の間に分布している。テストデータに対する着手推定の正解率は着手推定時の探索深さを深く

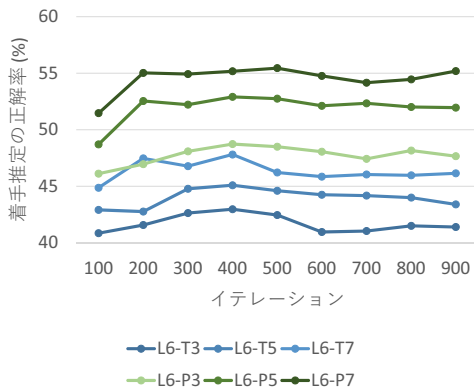


図4 学習時探索深さ5~6手の場合の着手推定の正解率

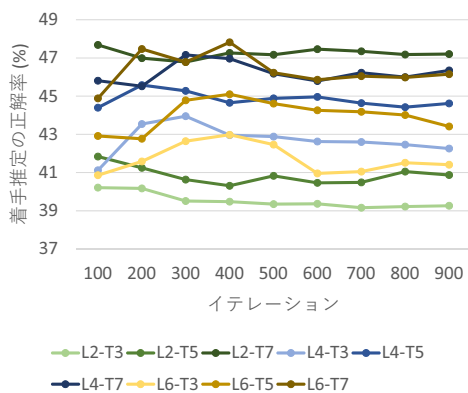


図5 テストデータに対する着手推定の正解率

表2 各探索深さによる加藤テストデータでの着手正答率の最大値

探索深さ	加藤テストデータに対する 着手推定の正解率の最大値
学習1~2手、探索3手	40.21%
学習1~2手、探索5手	41.84%
学習1~2手、探索7手	47.68%
学習3~4手、探索3手	43.95%
学習3~4手、探索5手	45.58%
学習3~4手、探索7手	47.16%
学習5~6手、探索3手	42.98%
学習5~6手、探索5手	45.10%
学習5~6手、探索7手	47.82%

するにつれ向上する傾向にあることがわかる。実験で着手推定モデルを作成する対象とした加藤一二三九段は、実力的には「7棋戦の棋譜」で指し手となっているようなトッププロにも近いプレイヤーであるため、プレイ中の先読みもかなり深くまで行っていると推測できる。読みの深さを深くすることで着手推定の正解率が向上するという傾向はこの推測とも一致する。このことから、探索時に読みの深さを適切に設定することで着手推定の正解率を向上させることが可能であるといえる。

図5は実験結果のうちテストデータに対する着手推定の

正解率のみを1つのグラフにまとめたものである。また、各条件において得られた着手推定の正解率の最大値については表2に示した。これを見ても、着手推定時の探索深さを深くしていくと着手推定の正解率が向上していく傾向にあることがわかる。その一方、学習時の探索深さと着手推定の正解率については一定の傾向があるわけではない。例えば、学習時の探索深さが3~4手の場合(青系の色で示した)についてみると、着手推定時の探索深さ3手および5手では着手推定の正解率は最もよい傾向にある。実際最大値で見ても、最も高い値を示している。一方、着手推定時の探索深さ7手では一転して最も悪い結果を残している。これも最大値で見たときにも最下位となっている。また、学習時と着手推定時双方の探索深さが近い場合に結果が良くなる可能性も考えられたが、これについても結果からは否定される。すなわち、学習時の探索深さに関わらず、着手推定時は3手の場合(薄い色で示した)が最も結果が悪く、7手の場合(濃い色で示した)に最も結果がよくなるという結果であった。以上のことから、学習時の探索深さと着手推定の結果について、なんらかの一定の傾向は確認できなかった。

5. まとめ

今回の実験によって、特定の個人の着手推定を行うにあたり、大規模データと特定の個人の小規模データを組み合わせ学習した評価関数が、特定の個人の小規模データのみで学習した評価関数よりもよい結果を出すことを示した。さらに、その学習法を用いて様々な探索深さでの着手推定を行ったところ、探索深さを変化させることで着手推定の正解率が向上することも確認できた。今回は探索深さが深いほど着手推定の正解率が高まるという結果が得られたが、モデル作成の対象とした加藤一二三九段の実力を考慮すれば妥当な結果であるといえる。

本実験においては着手推定の正解率を着手推定モデルの完成度の指標として用いたが、実際にモデルを用いて対局を行う場合は着手推定に失敗した場合でもそのプレイヤーの実力的に妥当な手を返すことが望ましい。今回作成したモデルがそのような条件を満たしているかについては実際に対局を行うなどして確認すべきである。また、今後は探索過程で用いる個人の特徴の種類を増やして更なる着手推定の正解率向上を目指すことを目指す。加えて、様々なプレイヤーに対して本手法を適用し、本手法が様々なプレイヤーに対し普遍的に使えるものであることも示す必要がある。

参考文献

- [1] 保木邦仁. 局面評価の学習を目指した探索結果の最適制御. ゲームプログラミングワークショップ2006論文集, pp. 78-83, 2006.
- [2] Kunihito Hoki and Tomoyuki Kaneko. Large-scale optimization for evaluation functions with minimax search. *Journal of*

- Artificial Intelligence Research*, Vol. 49, pp. 527–568, 2014.
- [3] 生井智司, 伊藤毅志. 将棋における棋風を感じさせる A I の試作. 情報処理学会研究報告, Vol. 2010-GI-24, No. 3, pp. 1–7, 2010.
 - [4] 保木邦仁. Bonanza - the computer shogi program. http://www.geocities.jp/bonanza_shogi/.
 - [5] 鶴岡慶雅, 横山大作, 丸山孝志, 近山隆. 局面の実現確率に基づくゲーム木探索アルゴリズム. ゲームプログラミングワークショップ 2001 論文集, 第 2001 巻, pp. 17–24, oct 2001.
 - [6] 滝瀬竜司, 田中哲朗ほか. 入玉指向の将棋プログラムの作成. 情報処理学会論文誌, Vol. 53, No. 11, pp. 2544–2551, 2012.
 - [7] 大森翔太郎, 金子知適. 将棋での少数の棋譜からの評価関数の学習における拘束条件の研究. ゲームプログラミングワークショップ 2014 論文集, pp. 41–47, 2014.
 - [8] 伊藤毅志, 杵渕哲彦, 藤井叙人ほか. ゲームにおけるヒューマンエラー - 将棋における考察 -. ゲームプログラミングワークショップ 2014 論文集, Vol. 2014, pp. 196–201, 2014.
 - [9] 出村洋介. 将棋ソフト「技巧」. <https://github.com/gikou-official/Gikou/>.