

決定木を用いた将棋の局面評価

渡邊聡 小谷善行

東京農工大学大学院工学研究科

{watanabe, kotani}@fairy.ei.tuat.ac.jp

概要

今までに、将棋の静的評価関数の学習は様々な方法で行われているが、決定木を用いた局面や指し手の分類学習はあまり行われていない。

本研究では、ある局面での指し手を方針別に分類し、決定木を用いてプロの棋譜から最善手を学習する実験を行った。属性を、指し手が含む方針と局面の情報として、二分木の決定木を生成した。

その結果、平均 75% の正解率で最善手を分類し、1局面毎では、可能手の 30% (平均 33.3 手) を最善手と判定し、そのうちに実際の最善手が含まれている割合は 72% だった。

また、決定木を用いた短手数数の詰み・不詰みの判定では、詰み 92.5%、不詰み 82.4% という正解率を得た。

Position Evaluation Using Decision-Tree in Shogi

Satoshi Watanabe Yoshiyuki Kotani

Tokyo University of Agriculture and Technology

{watanabe, kotani}@fairy.ei.tuat.ac.jp

Abstract

By now there are various methods for learning of the static evaluation function in shogi, but it was not much classified learning the position and the move using decision-tree..

We classified a move in a position by the policy, and experiment of learning the best move from professional game record using the decision-tree by this research. The attribute was the policy which a move contains and feature on the position. Decision-tree was generated binary tree.

As a result, the best move can be classified at 75% of the average correct answer rates, in each 1 position, it judged a best move for 30% in the candidate move (average 33.3 move), it contains the best move at 72%.

Moreover, correct answer rates was checkmate 92.5%, no mate 82.4% in the judged that there is mate or no mate using decision tree.

1 概要

本稿では、コンピュータ将棋において、決定木を用いて局面評価を行う方法について述べる。棋譜を学習データとして決定木学習を行い、ある局面の最善手を判定することを目的とする。

今までに将棋の静的評価関数の学習はTD法やニューラルネットなど様々な方法で行われているが、決定木を使った局面や指し手の分類学習はあまり行われていない。チェスでは決定木を終盤譜の学習に用いて、2手・3手詰めを分類した研究がある[1]。

決定木学習はニューラルネットワークやその他の手法に比べて、単純なアルゴリズムで計算量が少なく、判断の理由が明確に示されるため、学習過程が知識として理解しやすいという特徴がある。

実験は、指し手を方針別に分類し、プロの指し手が満たす方針とその局面を表す属性から決定木学習を行った。また、決定木を用いた短手数の詰み・不詰み判定についても実験を行った。

(2) 決定木学習アルゴリズム (ID3)

決定木学習 (ID3) は、対称群を2つのクラス (たとえば勝ちと負けなど) に分類するときの規則を発見するための手法である。それぞれの対象は、ある決まった属性集合によって記述され、それぞれの属性は、その中で取り得る属性値からなる集合をもっている。

集合Cのすべての要素が同一のクラスに属しているときの決定木は、そのクラスを出力する葉となる。また、Cが複数のクラスを代表する対象を含んでいるときには、適当な属性を選択することによって集合Cをそれぞれに互いに素な部分集合C1, C2, ..., Cnに分類する。ここで、CiはCの要素のなかで選択されたi番目の属性値をもつ要素を含んだ部分集合である。部分集合のそれぞれに同様の規則形成手順を繰り返すことによって処理が行われ、最終的な結果は木構造となる。それぞれの葉部分はクラスを出力し、各ノードでは、テストされるべき属性を規定しており、そこからの枝分かれはその属性が取り得る値に対応している。

部分集合への分割には情報量の期待値を用いる。情報量は大きいほど複雑であることを示していて、データの数が少ないほど、また、データ中のクラスが偏っているほど小さくなる。よってテストすべき属性は情報量が最も小さくなるものが選択される。集合Cをある属性でクラスに分割したときの属性値の取る確率をP+, P-とすると、情報量の期待値M(C)は

$$M(C) = -P^+ \log_2 P^+ - P^- \log_2 P^-$$

で表され、テストすべき属性としてAを選択した場合の情報量の期待値B(C, A)は

$$B(C, A) = \text{属性Aが} A_i \text{なる値を取る確率} \times M(C_i)$$

となる。よって、次にテストすべき属性は最も多くの情報量を獲得した属性であり、

$$M(C_i) - B(C, A)$$

を最大にするような属性Aがその時点での候補となる。

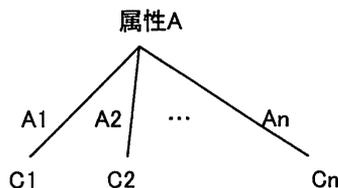


図1 属性の選択

3 決定木による局面評価

決定木による局面評価は、局面がもつ諸属性に注目してその局面を過去の事例によって分類する方法である。局面を適切に分類でき、また計算量が少なくすむような属性を発見できれば、決定木による局面評価は有効であると考えられる。

予備実験として、単純な属性でどの程度局面を勝敗で分類できるかどうかの実験を行った。指し手の分類クラスを最終的な勝ち・負けとして最終手から 50 手前までを学習データとして実験を行ったところ、表 1 の結果となった。

結果から、単純に局面の属性から勝敗を判定することは困難であるといえる。これは、プロの対局は終局直前までほとんど互角に進行し、また、終局した局面でも圧倒的な差はついていない場合が多いためである。

そこで、プロの手を勝敗によって分類するのではなく、指し手の方針別に分類し、プロの手とそれ以外の手を判定する決定木学習を行った。ここでは、プロの指し手は結果を問わず最善手とみなし、また、学習する局面は中盤以降（40 手以降）とした。さらに、決定木を用いた詰み・不詰みの判定についても実験を行った。

表 1 予備実験の結果

	訓練データ		テストデータ	
	データ数	正解数	データ数	正解数
最終 50 手	20000	19705(98.5%)	5000	2492(49.8%)
最終 40 手	16000	15744(98.4%)	4000	2033(50.8%)
最終 30 手	12000	11858(98.8%)	3000	1671(55.7%)
最終 20 手	8000	7924(99.1%)	2000	1168(58.4%)
最終 10 手	4000	3927(98.2%)	1000	635(63.5%)

4 決定木の属性

決定木の属性と分類クラスは表 2 である。また、最善手の判定では決定木はすべて二分木とした。

表 2 決定木の属性と分類クラス

	最善手（プロの手）の判定	詰み・不詰み判定
属性	指し手の方針別分類+局面	局面
分類クラス	1：最善手 0：それ以外の指し手	1：詰み 0：不詰み

4-1 指し手の方針別分類

指し手の方針を文献[1][2]をもとに、主に歩の手筋とその他の先読みなしの手で 25 種類設定し、指し手をどの方針を含むかによって分類した。

実験に用いた棋譜 500 局の中盤以降の平均可能手数には 116.5 手、最善手が 1~27 の方針に含まれる割合は、0.998 (38332/38401) だった。指し手の種類を表 3、選択率の上位 5 つの方針を表 4 に示す。選択率は、選択数（最善手がその方針を含んでいる局面数）を、出現数（その方針を含む手が存在する局面数）で割ったものである。

表3 指し手の方針

	方針
1	歩の交換
2	歩の突き捨て
3	歩の成り捨て
4	垂れ歩
5	焦点の歩
6	単打の歩
7	合わせの歩
8	底歩
9	紐歩
10	王手
11	直前に動いた駒を取る
12	直前に動いた駒以外の駒を取る
13	ただ捨て
14	駒が成る
15	指した駒が次手で成る
16	相手の駒に利きを付ける
17	取られる駒を逃がすまたは防ぐ
18	自分の利きの数より相手の利きが多いマスに駒を動かす
19	相手陣への打ち込み
20	大駒の利きを遮る駒を動かす
21	端に利きを付ける
22	お互いに利きのあるマスに利きを加える
23	盤上の駒を動かす
24	自玉周辺の（自分の駒数－相手の駒数）を増やす
25	自玉周辺の（自分の利き－相手の利き）を増やす
26	相手玉周辺の（自分の駒数－相手の駒数）を増やす
27	相手玉周辺の（自分の利き－相手の利き）を増やす
28	その他（上記のどれにも当てはまらなかった手）

表4 選択率の上位5方針

方針	選択数	出現数	選択率
23	28347	38401	0.738
16	22909	38401	0.597
25	18409	36964	0.498
22	18835	38395	0.490
27	13875	30333	0.457

4-2 局面を表す属性

局面は表5の64個の属性で表す。詰め・不詰めの判定では、詰められる側を相手玉とする。最善手の判定では決定木は二分木なのでそれぞれの属性は属性値が閾値nより大きいか小さいかで分割する。nは最も情報量が減少するような値である。分類精度を上げるには、属性をもっと細かく分割する必要があるが、分割する点をすべて調べるのは時間がかかる。例として、持ち駒の枚数(歩)の分割を考えると、2分割では18通りの分割点があるが、3分割する点は、 ${}_{18}C_2=153$ 通りである。よって属性の数が多い最善手の判定では、2分割とした。

表5 局面を表す属性

番号	属性	最善手の判定	詰み判定
1~7	自分の各持ち駒の枚数	○	○
8~14	相手の各持ち駒の枚数	○	○
15	自分の大駒が成っている	○	×
16	相手の大駒が成っている	○	×
17	自陣の自分の駒の枚数	○	×
18	自陣の相手の駒の枚数	○	×
19	相手陣の自分の駒の枚数	○	×
20	相手陣の相手の駒の枚数	○	×
21	自玉8近傍自由度 (相手の利きのないマスの数)	○	×
22	自玉24近傍自由度	○	×
23	相手玉8近傍自由度	○	○
24	相手玉24近傍自由度	○	○
25~28	自玉8近傍の自分の利きの数 a,b,c,d	○	×
29~32	自玉8近傍の相手の利きの数 a,b,c,d	○	×
33~36	相手玉8近傍の自分の利きの数 a,b,c,d	○	○
37~40	相手玉8近傍の相手の利きの数 a,b,c,d	○	○
41~44	自玉24近傍の自分の利きの数 a,b,c,d	○	×
45~48	自玉24近傍の相手の利きの数 a,b,c,d	○	×
49~52	相手玉24近傍の自分の利きの数 a,b,c,d	○	○
53~56	相手玉24近傍の相手の利きの数 a,b,c,d	○	○
57~64	相手玉8近傍の各マスの利き状態	×	○

王近傍の利きの数は次のような4種類とした。

- a : 重複を許さない利きの数 (=利きのあるマスの数)
- b : 重複を許した利きの数
- c : 相手よりも利きが多いマスの数
- d : 相手の利きがなく、自分の利きがあるマスの数

これらは王自身の利きは含んでいない。

相手玉8近傍の各マスの利き状態は詰み判定で用い、それぞれのマスが(自分の利きだけがある・相手の利きだけがある・お互いの利きがある・どちらの利きもない)の4つの状態を表す。

5 学習データ

最善手の判定では、学習データにプロの棋譜500譜を用いた。中盤以降の1局面ごとにプロの手とそれ以外の手1手を取り出し、76674個のデータを生成した。訓練データを450譜、テストデータを50譜とし、1局面毎についての判定も行った。

詰み・不詰み判定の学習データは、3手の全複探索プログラム同士の対局させ、次のようにして1局で1つの詰み又は不詰み局面を生成した。

- ・詰み局面：探索中に最初に詰みを見つけた局面
- ・不詰み局面：詰み局面から2・4・6・8・10手をランダムに戻した局面

ここでは不詰み局面は、厳密に不詰みが成立する局面ではなく、少なくとも3手では詰まない局面をとした。同様に詰み局面は3手詰み局面である。また、厳密には序盤や中盤の局面も不詰み局面ではあるが、実際に判定すべきなのは、詰みがある局面とその直前の局面であるため、不詰み局面は終盤の局面から生成した。学習データを8953局面（詰み5311局面、不詰み3642局面）生成し、10%（895局）をテストデータとして実験を行った。

6 実験結果

(1) 最善手の判定結果

表6 最善手の判定結果

	訓練データ				テストデータ			
	1	0	総数	正解率	1	0	総数	正解率
最善手	31762	2622	34384	92.4%	2939	1014	3953	74.3%
その他の手	2668	31716	34384	92.2%	957	2996	3953	75.8%

表7 1局面毎の判定結果

平均可能手数 (A)	最善手と判定した手の平均 (B)	最善手と判定する割合 (B) / (A)
116.2	33.34	0.297

表6. 7から、生成した決定木は未知の局面で平均75%の正解率で最善手を分類する。また、1局面毎の判定では、全可能手の29.7%を最善手と判定し、そのうちに本当の最善手が含まれていた確率は0.720だった。また、決定木の分類深さ3までで使われた属性は

- ・ 取られる駒を逃がすまたは防ぐ
- ・ 駒を取る
- ・ 直前に動いた駒を取る
- ・ 自軍の大駒がなっている

であった。

結果から、この決定木の判定を使って枝刈りができるほどにはよい結果とはいえない。最善手を含む確率が0.720であるため、3手先では最善手が残っている確率は $0.720^3=0.373$ 程度になってしまう。しかし、候補手の仮評価や優先順位の決定などには使えると考えられる

(2) 詰み・不詰み判定の結果

表8 詰み・不詰み判定の結果

	訓練データ				テストデータ			
	1	0	総数	正解率	1	0	総数	正解率
詰み局面	4618	168	4780	96.6%	491	40	531	92.5%
不詰み局面	103	3175	3278	96.9%	64	300	364	82.4%

表8から、生成した決定木は未知データに対して、詰みなら92.5%、不詰みなら82.4%の正解率で局面を分類できる。また、分類深さ2までで使われた属性は

- ・ 自分の持駒（金）
- ・ 後手王8近傍の自分の利きの数 $a \cdot d$

だった。この決定木は、盤上の駒の状態を全く考慮していないが、短手数での詰みならば、周囲の利きの状態から詰み・不詰みをある程度判定することができるといえる。

まとめ

本稿では、指し手を方針別に分類してプロの棋譜から決定木学習を行い、最善手とそれ以外の手の判定を行った。また、決定木を用いた詰み・不詰み判定の実験を行った。その結果、詰み・不詰みの判定では平均87.5%の正解率となったが、最善手の判定はあまりよい結果は得られなかった。

決定木学習では単純に属性を増やしても未知データに対する精度は上がらない。細かすぎる分類は特異なデータに引きずられる場合があるためである。よって、多くの事例と関連性のある属性を抽出する必要がある。

今後の課題として、

- ・ より正確に指し手、局面を分類できる属性を抽出する
- ・ 属性の分割を増やす
- ・ 決定木で分類した結果を探索に用いる（候補手の順番・延長・枝刈りなど）

といったことが挙げられる。

参考文献

- [1]小谷善行、飯田弘之、なにを刈るべきかー指し手の分類と指した手の割合ー、ゲームプログラミングワークショップ95、pp148-156、1995
- [2]香山健太郎、井上博允、将棋における指し手の絞り込みの学習、ゲームプログラミングワークショップ96、pp104-111、1996
- [3]J.Ross Quinlan、効率的分類手法の学習とチェス終盤譜への応用、知識獲得入門、pp149-173、共立出版、1988
- [4]日本将棋連盟、将棋年鑑「平成十一年度版」、1999