

対局経験を利用した相手の手の予測システム

大島 安簡、Reijer Grimbergen

山形大学

概要

ゲーム研究において、主な探索方法では相手が常に最善手を打つことを仮定しており、対局経験があったとしても同様に仮定して探索している。しかし、人間プレイヤーでは相手の手を予測する際に以前の対局経験を利用して探索する。本稿ではより人間らしく相手の手を予測しながら探索するシステムを提案する、その手を予測するための記憶装置として 3 層ニューラルネットワークを用い、対局した棋譜を学習させた。そして、そのネットワークを使用し、未学習パターンの出力と、それを用いた探索を行った。結果として、学習誤差と学習データ数により未学習パターンに対する正解率は高くなることを確認できたが、探索に用いるには学習データに改良の必要性が出てきた。

A System for Using Past Experience to Predict Opponent Play

Yasuhiro Oshima, Reijer Grimbergen

Yamagata University

Abstract

The general assumption when performing search in game-playing research is that the opponent will play the best move. Even when there is experience against the same opponent, this is still being assumed. However, human players use their experience against the same opponent to make a prediction about which move the opponent will play. In this research, we will propose a searching system with a more human-like approach, trying to predict the moves of the opponent. In order to make such prediction, a three-layer neural network will be used as memory, which is being trained by the games played against the same opponent. Furthermore, this network was used to predict the move in the case of unknown patterns, as well as guiding a search algorithm. As a result, it was shown that the system could predict moves in unknown patterns, but that improvements are needed to actually lead to improved playing strength.

1. はじめに

ゲーム研究において、主な探索方法では相手が常に最善の手を打つことを仮定して先読みをしている。この時、相手の手を決める際に相手の情報は関与しておらず、一度対局した相手でも最善の手を打つことを仮定している。人間プレイヤーでは一度対局した相手ならば、相手の手を予測する際に以前の対局経験を利用して探索するはずである。

このような人間に近い先読みを実現すると、同じ時間内に深く探索ができる。また、相手に合う最善手を選択することが出来るため、その対局をすぐに終わらすことができる。

そのためには以前対局した相手の情報を記憶し、探索する時に局面に応じて相手が指す可能性のある手を予測して手を展開できるシステムを作成する必要がある。

今回の研究では、対局した情報から対局時に出現していない局面でも手を予測できるように、3 層ニューラルネットワークを使用し、先手・後手時別に相手の手と局面を学習した。そして、そのニューラルネットを探索内で先手・後手でニューラルネットワークを使い分けながら予測し、対局を行うことでニューラルネットワークの評価を行った。

2. 関連研究

ニューラルネットワークによる研究は数多くあり、ゲーム研究に詰め碁の解を求める研究がある。9×9 の範囲で 3 層ニューラルネットワークを使用し、数百回学習させた後で既学習パターンを完全に答えることができる。また、2000 パターンの定石手を学習させたネットワークでは、未学習パターンに対しても 6 割程度の正解を答えることができるなど、データベースとは違い未知のパターンに対しても予測が出来る[1]。その研究の結論として強い囲碁プログラムでは今後ニューラルネットワークが使用されることを示唆しているが、実際に探索に用いるプロセスは示されていないかった。そのため、今回行う研究はニューラルネットワークを用いて予測システムを提案し、囲碁と同じような盤を構成する五目並べを用いて探索で使用することで、ニューラルネットワークを探索に用いるモデルを示す。

3. ニューラルネットワーク

ニューラルネットワークとは、ニューロンがどのように結合しているかを意味しており、ゲーム研究では階層結合型ネットワークを使用する。このネットワークは、入力層・隠れ層・出力層の構成となっており、各層の結合部分には重みがあり、入力層から出力層へ信号が伝わる際に重みによって変換され伝播される。入力パターンに対して出力されるためには重みの調節が必要となり、以下がその重みを調節する学習式である。

$$W_{i,j}^{n,n-1}(t+1) = W_{i,j}^{n,n-1}(t) + \Delta W_{i,j}^{n,n-1}(t) \quad (1)$$

$$\Delta W_{i,j}^{n,n-1}(t) = \eta \delta_i^n X_j^{n-1} + \alpha \Delta W_{i,j}^{n,n-1}(t-1) \quad (2)$$

(1) は重い修正の式で、 ΔW は重みの修正量を示しており、(2) がその式の概要となる。 α は安定化定数で、前回の重み修正量を使用し、学習収束時の振動を抑える効果がある。 η は学習率で、収束の早さに関係する値である。一般的に安定化定数は小さい値をとることで振動幅を小さく、学習率は大きな値をとることで学習速度を早める効果がある。

4. ニューラルネットワークの仕様と教師値

本研究でのニューラルネットワークの仕様と教師値の説明する。入力層は各ユニットに座標を割当て、ユニットには 0 と 1 を使った 2bit 入力で座標状態を表す。黒石・白石・空の 3 種類の状態があり、入力状態を明確にするため各入力は 10, 01, 00 とし、各座標に対して 2 値の入力が必要になる。また、出力には各ユニットに座標を割当て、相手が指す予測値を出力するため各座標には 1 値の出力になるため以下の仕様になる。

表 1：入力範囲を 15×15 にしたシステムの仕様

入力層ユニット数	入力座標数の 2 倍(450)
出力層ユニット数	入力座標数に対応(225)
隠れ層ユニット数	複数(試行錯誤的方法で決定)
学習法	誤差逆伝播法
教師値	局面に対する手の値

教師値として入力局面に対する相手の指す座標には 0.9, それ以外は 0.1 の値を与えるようにする. 学習が十分に行ったものは, 入力に対する出力で値が 0.9 に近い値の座標が相手の指す座標として出力される.

5. 相手の手の学習データ

学習データは実際の対局から相手の局面と手を使用する. 今回着目する手は, 一般的な 5 目並べの必須な手であるオープン 3 を止める, オープン 4・クローズ 4 を止めるといった必須手以外の手である. このような必須手は, 探索を通さずに算出しているため, 相手が探索を行う必要がある局面とその手を学習することで, 予測しにくい相手の思考を学習できることになる.

また, 学習データは進行度順に並べ, 出現数に応じて並べる. 進行度順に並べることで, 前後の学習であまり差がでないようにし, 学習速度を早めるようにする. また, 出現数に応じて並べるのは, 同じ局面で複数の手がある場合に, 出現数の多いものを高い出力として学習させるために行う.

そして, 自分が先手・後手のときで 2 種類のデータを用意し, 2 種類のネットワークを作成する. 先手・後手では黒と白の石の数が違い, 先手後手の局面を混合させるとどちらを視点の手を予測するのかが分かりづらい. そのため, 先手後手で 2 種類用意し, 探索では先手・後手で使い分ける.

6. 相手の手の学習実験

実際に先手・後手で各 10~50 対局した棋譜から学習データを作成し, 相手の手を学習できるかを調べる. また, 対局数による学習データ数の違いから未知の入力に対する出力を確かめる. ニューラルネットワークの各仕様と学習終了条件は以下のようにした.

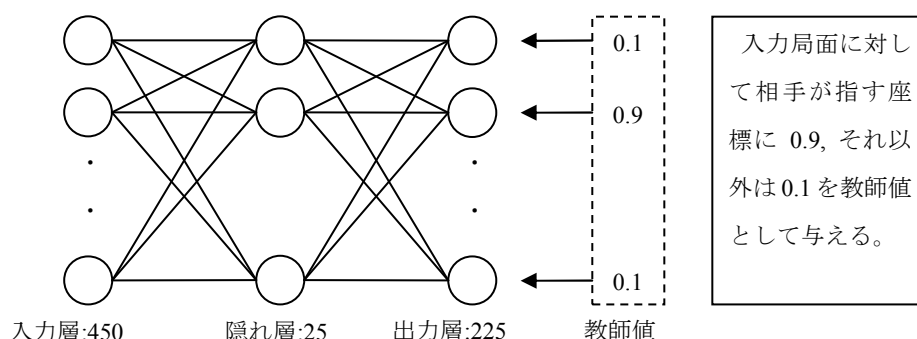


図 1 ニューラルネットワークの仕様

隠れ層のユニット数は 25, 学習値を 0, 安定化定数を 0.8 と固定し, 二乗平均誤差が 0.08 以下になるか, 1000 回学習するまで学習させた. また, 隠れ層ユニットの数は, 試行錯誤的方法[2]で各対局数別に学習回数 50 回までの収束推移で決定した. 以下がそのグラフである.

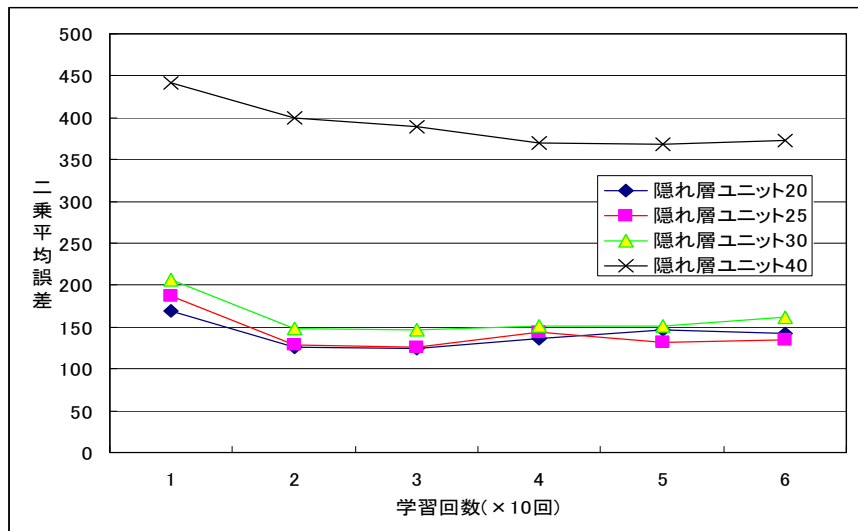


図2 各隠れユニット数の学習推移

隠れ層ユニット数 25 が、学習誤差の減少率大きい。他のユニット数では学習誤差がこれ以上少なくなるらないため、隠れ層ユニット数を 25 として学習を行うことにした。

7. 予測局面と手の学習収束結果

自分が先手・後手から見た局面の予測手の学習を行った。

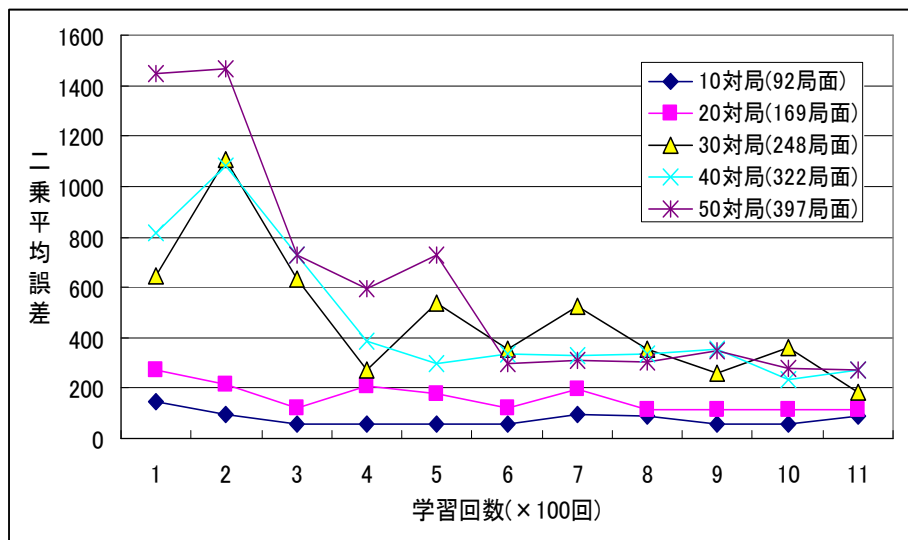


図3 各対局数による学習推移(先手局面学習)

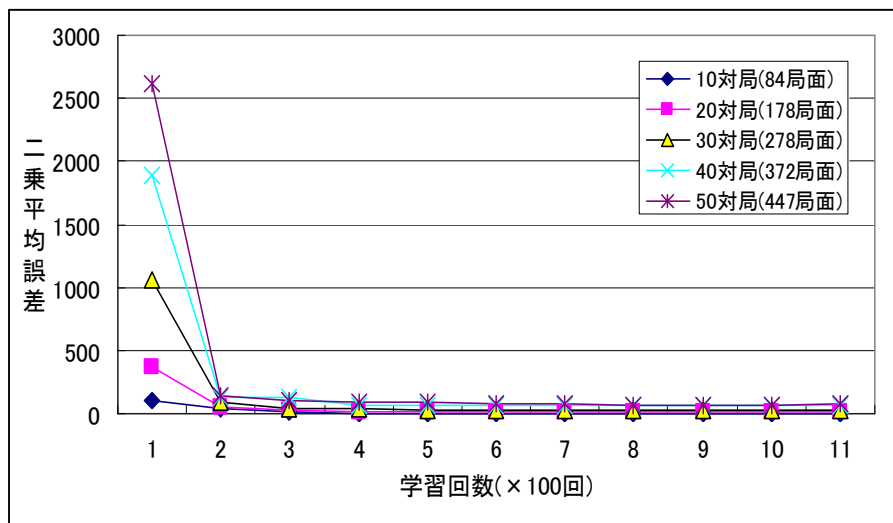


図4 各対局数による学習推移(後手局面学習)

自分が後手の時の局面では十分に学習できたが、先手局面では同じ局面で複数の手が存在する局面が多くあり、その同じ局面の誤差率により、全体的に誤差が大きいのとなってしまった。しかし、学習した局面を予測するには十分な出力を持っている。

8. 未知の局面に対する出力分布

ここでは学習したネットワークに、学習データにない局面である未学習パターンを各手番別に先手局面 223 局面、後手局面 260 局面を入力し、出力値を高い順に並べ、入力局面に対応する相手の手の正解ユニットがある、出力値の順位をカウントした結果を示したものである。

未知パターンに対する出力分布の出力値の順位は、左から出力値が高い順に 15 の範囲で分けられて並べられている。例えば、1-15 番目グラフ値では、出力値が 1-15 番目に高いユニットの座標に対局数ごとに正解ユニットが 18~10 個あったことを示す。他の範囲も同様に正解ユニットをカウントしている。

出力値が座標に相手が指す可能性があることを示唆し、正解がどの出力値に分布しているかが分かる。本来であれば、出力値が高い左の付近に正解のカウントプロットが集まるはずであるが、学習誤差により結果が大きく分かれた。学習誤差の少ない後手局面を学習したネットワークの方が、出力値が高い付近に正解カウントが多く、理想的な状態となっている。また、学習データ数が多い方が順位の高いところに正解ユニットが分布しており、学習数が多い方がいいことがわかる。

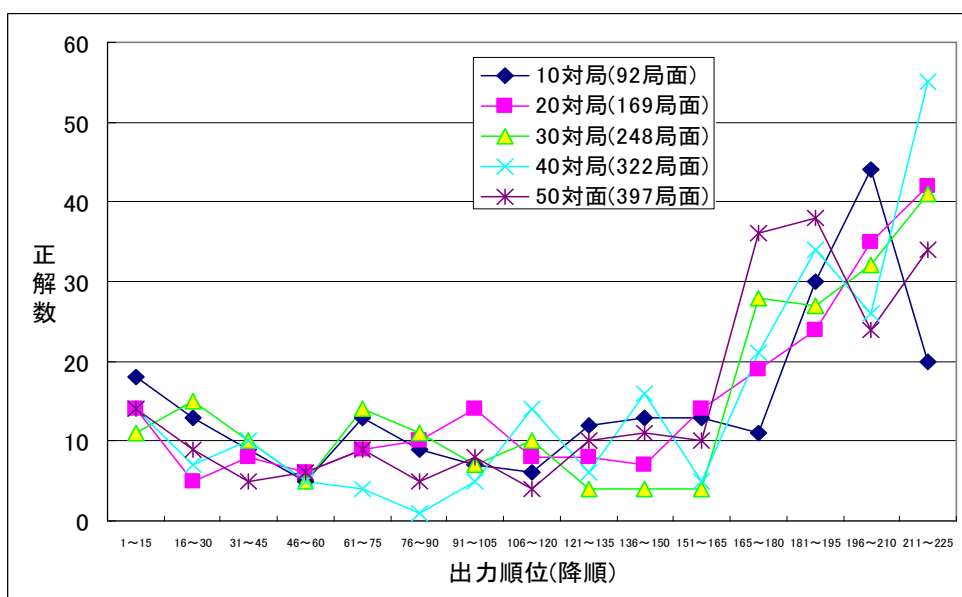


図5 各学習データ数における未知パターンに対する出力分布(先手局面学習)

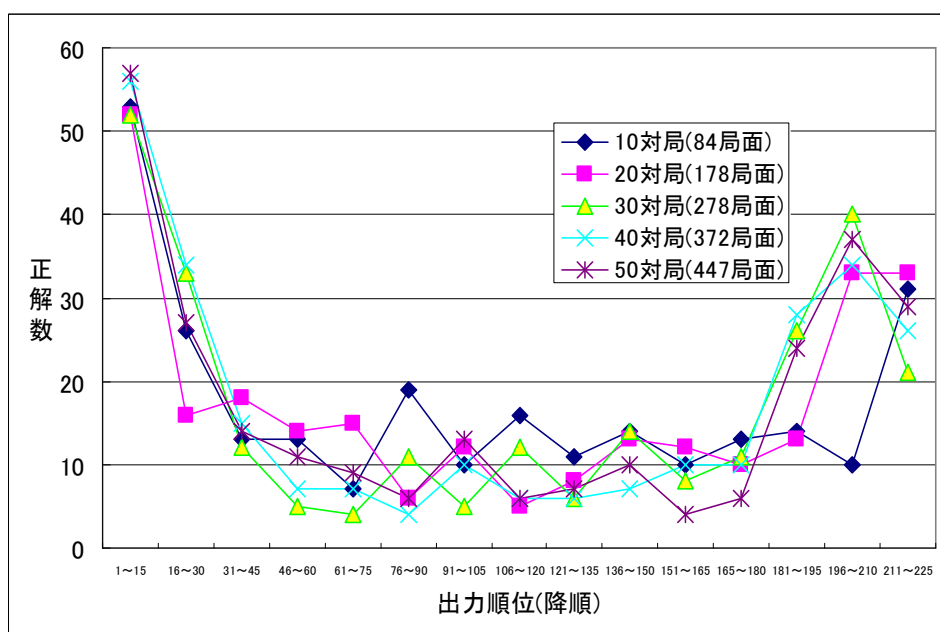


図6 各学習データ数における未知パターンに対する出力分布(後手局面学習)

9. 学習したネットワークを使用した対局実験

ここでは、学習したネットワークを利用して、相手の手を派生する際に、手を予測し展開しながら探索を行うようにした探索木とそうでないものの対局を行う。先手・後手各 50 戦し、計 100 対局を行う。相手は乱数を用いた評価関数を用い、同じような手はあまり指さない相手とする。

使用するネットワークは、先手局面・後手局面ともに、50 対局した時の学習データを 50000 回学習させたネットワークを使用し、手の予測をしたときに、入力した局面が未知パターンであれば出力結果が高い順に並べ替えて展開し、既知パターンであれば出力値が上位 10 座標だけを展開、つまり候補主のみを展開する前向き枝刈りを

行う。

既知パターンと未知パターンの判別は、盤情報を入力し正方出力した各出力ユニットの合計値によって判別を行う。学習を行ったネットワークは、教師値のユニットには 0.9 をそれ以外のユニットには 0.1 を出力するようにになっている。学習誤差が多少あったとしても、図 8 のようにある程度一定の値に分布している。各ユニット出力値の合計値の違いにより既知パターンと未知パターンの判別を行う。

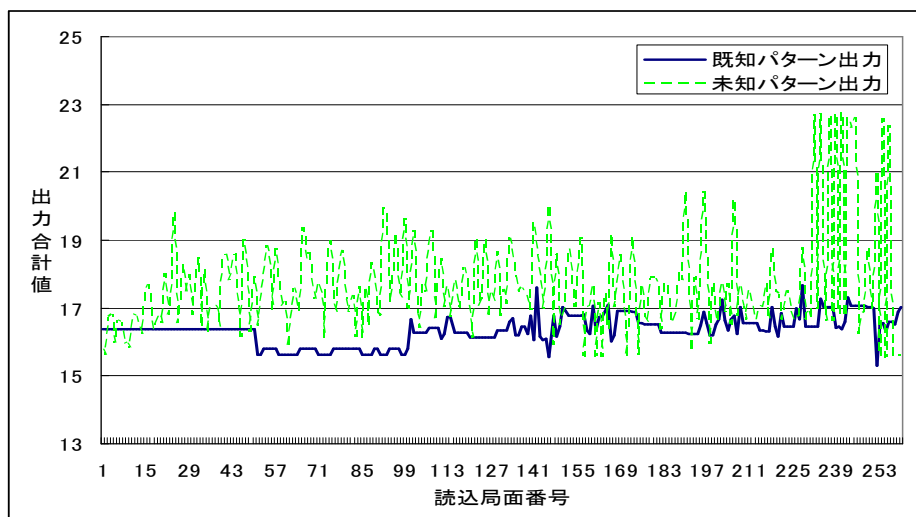


図 7 既知パターンと未知パターンの出力ユニットの合計値の分布

表 2 相手の手の予測をしながら探索する方法と元の探索法との対局結果

対局	結果
相手の手を予測し探索 — 元の探索法	51 – 48 (1 引き分け)

対局結果は、相手の手を予測する探索をしたものと元の探索法の差が 3 差しかなく、相手の手を予測しての効果はほとんど現れなかった。恐らく、学習データにない未知パターンの局面が多く、前向き枝刈りをする事ができなかったと思われる。

10. まとめ

本論文では、以上の結果から、学習データ数と学習誤差により未知パターンの予測結果に影響がでることが確認できた。しかし、学習データで同じ局面で複数の予測手がある場合に、それが学習誤差を大きくする原因となっている。このため、複数の予測手がある局面では、それらをまとめて一つの学習局面としてまとめる必要がある。今後は、局面データの作成方法と学習データ数の増加を行うことが課題となる。

11. 参考文献

- [1] Nobusuke Sasaki, Yasuji Sawada, and Jin Yoshimura, “A Neural Network Program of Tsume-Go”, Computers and Games, pp168-182,1988
- [2] Bahman Kermashahi, ニューラルネットワークの設計と応用, 昭晃堂, 東京, 1999.