

# 半教師あり学習による麻雀の降り局面の判別

我妻 敦<sup>1,a)</sup> 田中 哲朗<sup>2,b)</sup>

**概要:** 強いコンピュータ麻雀プレイヤーを作成するためには、降りるべき局面とそうでない局面を正しく判別することが有用であると考えられる。しかし降りる局面のラベル付けを人手で行うにはコストがかかる。本研究では先行研究で公開してくれているラベル付きデータを用いて、降りる局面の一致率を精度を高めるために半教師あり学習を用いた。半教師あり学習を用いた場合の降りる局面の一致率は 0.586 で、半教師あり学習を用いない場合の一致率である 0.492 と比較し、有意水準 5% で差があるという結果を得た。

## Classification of folding/unfolding mahjong positions using semi-supervised Learning

ATSUSHI WAGATSUMA<sup>1,a)</sup> TETSURO TANAKA<sup>2,b)</sup>

**Abstract:** It is desirable to classify unfolding / folding positions correctly to create a strong computer Mahjong player. Although there are researches to create such classifiers by supervised learning, they require a large quantity of training data made by skilled human agents. In this paper, we propose to use semi-supervised learning to improve the learning accuracy. We compare the two classifiers, the one is obtained by semi-supervised learning and the other is obtained by supervised learning. The matching rate of folding positions with the former achieved 0.586, although that with the latter is 0.492.

### 1. はじめに

2人完全情報ゲームであるチェスや将棋では人間のトッププレイヤー以上の実力を持ったコンピュータプレイヤーが既に実現されている [1] [2]。対して多人数不完全情報ゲームである麻雀のコンピュータプレイヤーの実力はまだ人間のトッププレイヤーには追いついていない。

麻雀では自分が和了ることよりも和了れないことの方が多いため、強いコンピュータプレイヤーを作るには自分の損失を少なくするという「降り」が重要になると考えられる。麻雀における「降り」とは、自分が和了りに向かうことを止めて自分の捨てた牌で相手に和了られてしまうことがないように牌を捨てていくことである。先行研究で

は降りと鳴きを考慮したコンピュータ麻雀プレイヤーを作成し、平均プレイヤーより良い性能を得ている。学習の精度を上げるためには学習し使用するデータの数は多いほうが良いと考えられているが、降りる局面を学習させるために必要な降りる局面のラベル付けを人手で行うにはコストがかかる。

本研究では、先行研究で公開してくれているラベル付きデータを用いて半教師あり学習を行い、ラベル付けされていないデータにラベル付けをし教師データに追加することで全体の一致率を下げることなく降りる局面のラベルの一致率を向上させることを目的とする。

### 2. 関連研究

降りるを特別扱いしない麻雀のコンピュータプレイヤーとして北川らの研究がある [3]。北川らは牌譜の打ち手との一致を目指して3層ニューラルネットワークを用いて評価関数の重み調節を行った。調節の結果、ツモ局面においては約56%、鳴き局面においては約89%の一致率を得ており、麻

<sup>1</sup> 東京大学大学院 総合文化研究科  
Graduate School of Arts and Sciences, The University of Tokyo

<sup>2</sup> 東京大学情報基盤センター  
Information Technology Center, The University of Tokyo

a) wagatsuma@tanaka.ecc.u-tokyo.ac.jp

b) ktanaka@tanaka.ecc.u-tokyo.ac.jp

雀サイト東風荘 [4] にて試合を行ったところレーティング 1318 を得た。同サイトの下位 13 % 程度のプレイヤーのレーティングは 1300 であるためコンピュータ麻雀プレイヤーとしては弱いと言える。

本研究では、降りの判別能力を高めることでより強いプレイヤーを作成することを目標としているが、麻雀の降りを考慮したコンピュータプレイヤーに関する論文として水上らの論文がある [5]。水上らは多人数性を排除した 1 人麻雀のためのプレイヤーを教師あり学習によって実現し、1 人麻雀では考慮されない「鳴き」「降り」を実現する機構を導入することで 4 人麻雀へ適応させた。「降り」や「鳴き」の機能を導入することにより同機能を導入していないベースラインよりも良い成績を残しており、麻雀サイト天鳳 [6] において平均プレイヤーのレーティングである 1500 点を上回るレーティング 1651 点を得ている。水上らの論文において、降りに関する学習で使用しているデータ数は降りる局面が 531、そうでない局面が 5185 である。

本研究では半教師あり学習のゲームへの適用を扱っているが、これに関する先行研究としては将棋をテーマにした林らの研究がある [10]。この研究では半教師あり学習の 1 つである Self-Training を用いて自己対戦棋譜を学習に使用することで、人間の棋譜では考慮されなかった局面の学習を行うことで評価関数を改善し、熟練者の棋譜のみを用いて学習をおこなったプログラムとの対戦で 56.8% の勝率を得ている。

### 3. Self-Training

Self-Training は半教師あり学習の一種で少量のラベルありデータから学習を行い、その学習結果を用いてラベルなしデータのラベルを予測し、予測したラベルが正しいとみなして学習用のラベルありデータに追加するという手法である [8]。Self-Training のステップを以下に示す。

- (1) ラベルありデータで教師あり学習を行う。
- (2) 上記の学習器を用いてラベルなしデータのラベルを予測する。
- (3) ラベルなしデータと予測結果を基にした擬似正解ラベルをラベルありデータに追加する。
- (4) 再度学習を行う。(1)~(4) を繰り返す

### 4. 提案手法

麻雀の降りる局面に関するラベルのありデータの作成には人の手で一つ一つの局面で調査するため非常にコストがかかる。一方で、ラベルなしデータは麻雀サイトなどで公開されている牌譜データから容易に入手することが可能である。そこで本研究では、水上らの研究で述べられている降りる局面の認識に関する学習の際に半教師あり学習の一種である Self-Training を用いてラベルなしデータを教師データとして用いる手法を提案する。

Self-Training を用いる前提として、分類器が高い確信度で予測した結果は正しいというものがある。本研究においては降りる局面とそうでない局面の分類は高い確信度で予測できるものが多いであろうという仮定の元で提案手法に Self-Training を用いている。

#### 4.1 麻雀への Self-Training の適用

Self-Training を麻雀に適用するにあたり、ラベルありデータとして水上らが公開しているタグ付きの牌譜 [9] から得られたデータを使用し、ラベルなしデータとしてインターネット麻雀サイト天鳳より入手できる、同サイトの上位 0.1% 程度のプレイヤーのみがプレイできる鳳凰卓の牌譜から得られたデータを使用した \*1。牌譜からラベルありデータおよびラベルなしデータを得る際には水上らが公開しているプログラム [9] を使用した。

学習に使用した特徴は水上らの論文 [5] で使用していたものと同じものを使用する (表 1)。相手の状態に関しては 3 人の対戦相手それぞれについて特徴を考えるため特徴の数は合計で 34 である。

表 1 降り判断の特徴量 ( [5] より)

	項目	次元数
自分の状態	聴向数	1
	七対子に必要な牌の枚数	1
	リーチしているか	1
	親かどうか	1
	副露数	1
	副露により晒したドラの数	1
	持っているドラの数	1
	相手の状態	リーチしているか
相手の状態	副露数	1
	副露により見えたドラの数	1
	親かどうか	1
	切りたい牌のそのプレイヤーに対しての安全度	4
	場の状態	山の残りの枚数
カンの数		1
捨て牌にあるドラの数		1

得られたラベルありデータのうち約 8 割分の対局データを学習用ラベルありデータとして使用し、残りをテスト用データとして使用している。学習用ラベルありデータの総局面数は 12979 であり、そのうち降りる局面の数は 1216 であった。テスト用データの総局面数は 3302 であり、そのうち降りる局面の数は 295 であった。また、タグのついていない牌譜から水上らのプログラムを用いて抽出したラベルなしデータの総数は 103177 であった (約 8000 局分)。

本研究で使用した教師あり学習に用いる学習器は gaussian kernel による SVM で実装には LIBSVM [7] を用い

\*1 2011 年 1 月 1 日から 2011 年 12 月 31 日までに行われた 4 人対局のもの、約 8000 局分のデータ

た。ガウシアンカーネルのパラメータであるコスト ( $c$ ) 及びガンマ ( $g$ ) はグリッドサーチによって決定した。値の範囲は  $2^{-10}$  から  $2^{10}$  までを 4 倍刻みで行ったのちにその中で最も良かった値と  $2^{-10}$  から  $2^{10}$  を超えない範囲でその前後  $2^{-3}$  から  $2^3$  倍を 2 倍刻みで行った。

グリッドサーチの具体的な方法として、学習用ラベルありデータで 5 分割交差検定を行い降りる局面に関して適合率と再現率の調和平均 (F 値) の平均が最も良かった組み合わせをパラメータとして採用する。5 分割交差検定の際には全てのデータの中からランダムに 5 分割してしまうと、予測しようとする局面に近い局面を学習しやすいという過学習が生じてしまい正しい結果が得られないと考えたため、1 局を 1 つの塊とみなして全体を 5 分割している。

グリッドサーチの結果、降りる局面における F 値が最も良かったのは  $c=2^{10}$ ,  $g=2^{-10}$  の時で 0.619 であった。以降の実験ではガウシアンカーネルのパラメータにはこの値を使用している。

一般的に Self-Training はデータの追加を複数回行うが、本研究では半教師あり学習が降りる局面を学習させる際に適用することが有用なのかどうかを調べることを目的としているため、データの追加は 1 度しか行っていない。

#### 4.2 閾値の調整

ラベルなしデータにラベルをつける際には誤ったラベルがつけられる可能性がある。LIBSVM ではそのラベルがどのくらいの確かさでつけられたものなのかを確率として出力することができる (以後推定確率と呼ぶ)。そこで推定確率に閾値を設定し、閾値以上のデータのみを学習データとして追加したほうが学習の精度が上がると予想した。また元の学習用ラベルありデータは降りる局面数がそうでない局面数より非常に少ないため、一律で閾値を決めるのではなく降りる局面とそうでない局面で別々に閾値を設定する方が良くとも予想した。そこで降りる局面についての推定確率の閾値を 0.9 0.1 まで 0.1 刻みの 9 通り、そうでない局面の推定確率の閾値を 0.999, 0.99, 0.95 の 3 通りに設定し、それぞれを組み合わせで半教師あり学習を行わせた。それぞれの閾値の組み合わせについて半教師あり学習を行ったモデルによる予測結果とテストデータの一一致率を表 2 から表 4 に示す。また降りる局面についての F 値を表 5 に示す。

また半教師あり学習を行わない場合のモデルによる予測結果とテストデータの一一致率及び F 値をベースラインとする。この時の値は降りる局面に関してが 0.492, そうでない局面に関してが 0.981, 全体では 0.938 で降りる局面の F 値は 0.585 であった。表 2 から表 5 においてベースライン以上の値だったものは太字で示している。

各閾値において、ラベルなしデータから学習用ラベルありデータに追加されたデータ数について、表 6 と表 7 に

示す。

表 2 降りる局面の一一致率

		そうでない局面の閾値		
		0.999	0.99	0.95
降りる局面の 閾値	0.9	<b>0.492</b>	0.424	0.329
	0.8	<b>0.532</b>	0.458	0.349
	0.7	<b>0.556</b>	0.485	0.380
	0.6	<b>0.583</b>	0.489	0.380
	0.5	<b>0.586</b>	<b>0.501</b>	0.397
	0.4	<b>0.610</b>	<b>0.522</b>	0.431
	0.3	<b>0.634</b>	<b>0.549</b>	0.444
	0.2	<b>0.658</b>	<b>0.590</b>	0.478
	0.1	<b>0.719</b>	<b>0.651</b>	<b>0.522</b>

表 3 そうでない局面の一一致率

		そうでない局面の閾値		
		0.999	0.99	0.95
降りる局面の 閾値	0.9	<b>0.983</b>	<b>0.987</b>	<b>0.993</b>
	0.8	0.978	<b>0.987</b>	<b>0.991</b>
	0.7	0.976	<b>0.984</b>	<b>0.991</b>
	0.6	0.974	<b>0.983</b>	<b>0.991</b>
	0.5	0.973	<b>0.981</b>	<b>0.989</b>
	0.4	0.969	0.980	<b>0.989</b>
	0.3	0.966	0.976	<b>0.988</b>
	0.2	0.961	0.972	<b>0.984</b>
	0.1	0.943	0.960	0.977

表 4 全体の一一致率

		そうでない局面の閾値		
		0.999	0.99	0.95
降りる局面の 閾値	0.9	<b>0.939</b>	0.936	0.934
	0.8	<b>0.939</b>	<b>0.939</b>	0.934
	0.7	<b>0.939</b>	<b>0.940</b>	0.936
	0.6	<b>0.939</b>	<b>0.939</b>	0.936
	0.5	<b>0.938</b>	<b>0.939</b>	0.936
	0.4	0.937	<b>0.939</b>	<b>0.939</b>
	0.3	0.936	<b>0.938</b>	<b>0.939</b>
	0.2	0.934	<b>0.938</b>	<b>0.939</b>
	0.1	0.923	0.932	0.937

## 5. 考察

降りる局面に関する F 値について、グリッドサーチの際最も良かった F 値は 0.619, 本研究で行った半教師あり学習を用いた手法において、降りる局面の閾値 0.5, そうでない局面の閾値 0.999 の時に全体の一一致率を下げずに最も F 値が良くなり 0.631 であった。全体の一一致率を考慮しない場合最も良い F 値は 0.641 であった、しかし水上らの論文ではグリッドサーチの結果から得られた最も良い F 値は 0.73 であると述べられていた [5]。著者に確認してみた

表 5 降りる局面の F 値

		そうでない局面の閾値		
		0.999	0.99	0.95
降りる局面の 閾値	0.9	0.574	0.543	0.470
	0.8	<b>0.607</b>	0.565	0.486
	0.7	<b>0.619</b>	<b>0.590</b>	0.516
	0.6	<b>0.631</b>	<b>0.588</b>	0.515
	0.5	<b>0.629</b>	<b>0.598</b>	0.525
	0.4	<b>0.633</b>	<b>0.605</b>	0.557
	0.3	<b>0.640</b>	<b>0.612</b>	0.566
	0.2	<b>0.641</b>	<b>0.628</b>	0.581
	0.1	<b>0.625</b>	<b>0.632</b>	<b>0.596</b>

表 6 降りる局面として追加されたデータ数

閾値	追加されたデータ数
0.9	909
0.8	1167
0.7	1412
0.6	1619
0.5	1839
0.4	2083
0.3	2388
0.2	2786
0.1	3676

表 7 そうでない局面として追加されたデータ数

閾値	追加されたデータ数
0.999	2997
0.99	9379
0.95	19378

ところ、LIBSVM の“-v” オプションを使用して交差検定を行ったという回答で、さらに調べたところこのオプションで行われる交差検定は全てのデータからランダムに分割して交差検定するものであった。今回使用したラベル付き学習データで LIBSVM の“-v” オプションを用いて 5 分割交差検定を行ったところ最も良い F 値は 0.746 で水上らの論文の値より良かったが、用いたデータ数が異なるため明確に有意かどうかはわからなかった。LIBSVM の“-v” オプションを用いた交差検定の手法と本論文の交差検定の手法とで F 値に大きな差が出たため、4 章 1 節で述べたような過学習が生じている可能性は非常に高いと考えられる。

半教師あり学習を用いて麻雀の降りる局面の学習を行わせたところ、大きな一致率の低下は見られなかった。降りる局面の一致率とそうでない局面の一致率に関しては、降りる局面の閾値を下げた追加する局面数を増やすと降りる局面の一致率は向上し、そうでない局面の一致率は低下した。反対に降りる局面の閾値を上げて追加する局面数を増やすと降りる局面の一致率が向上し、降りる局面の一致率は低下した。このことから降りる局面の一致率とそうでない局面の一致率に関してはトレードオフの関係にあると考えられる。

本研究では全体の一致率を下げることなく降りる局面の一致率を向上させるという目的で実験を行っていたが、実験の結果から降りる局面の閾値が 0.5、そうでない局面の閾値が 0.999 の時が全体の一致率は低下していないという条件のもとで降りる局面の一致率が 0.586 で最も良かった。半教師あり学習を行わなかった場合の降りる局面の一致率は 0.492 であり、カイ二乗検定により降りる局面の一致率に関して有意水準 5% で差があるとわかった。このことからラベルありデータが少ない現状では麻雀の降りる局面の認識の学習に半教師あり学習を用いることは有用であると考えられる。

降りる局面の閾値を小さくしていくとあるところから全体の一致率が低下しているが、これは閾値が小さくなると推定確率が低いものが学習に使われることになるため、誤ってラベルをつけられたデータを学習しているために生じたのではないかと考えられる。そのため、推定確率に閾値を設定し閾値以上のデータのみを学習データとして追加したほうが学習の精度が上がるという予想は正しいと考えられる。また本研究では降りる局面とそうでない局面の一律で閾値を決めるのではなく別々に閾値を設定する方が良いと予想した。降りる局面の学習用ラベルありデータの数はそうでない局面のラベルありデータと比べて非常に少なかったため、降りる局面の閾値は大きいものから小さいものまで幅広く設定し、そうでない局面の閾値は大きめに設定し幅を狭くしている。実験の結果では降りる局面の閾値が 0.5、そうでない局面の閾値が 0.999 の時が、全体の一致率は低下していないという条件のもとで降りる局面の一致率が最も良かったので、降りる局面とそうでない局面の数に差がある場合、一律で閾値を決めるのではなく別々に閾値を設定する方が良いと予想は正しいと考えられる。

さらに降りる局面の一致率を高くする方法として、半教師あり学習でラベルを予測する際に使用する特徴に手を加えるという方法がある。人間が降りる局面のタグ付けを行う場合には牌譜から実際に選ばれた手を見てそれをもとにその局面にタグをつけるかどうかの判断をしている [5]。しかし本研究でのラベルなしデータの予測の際には水上らの論文 [5] の降りる局面の学習に使用している特徴と同じものを使用しており、実際に選ばれた手に関する特徴は一切考慮されていない。実際に選ばれた手に関する特徴を用いてラベルなしデータの予測を行うことで、間違ったラベルのデータが学習用ラベルありデータに混じってしまうことが少なくなり結果として全体の一致率や降りる局面の一致率の向上につながると考えられる。

今回は半教師あり学習が降りる局面を学習させる際に適用することが有用なのかどうかを調べることを目的としてたため半教師あり学習によるデータの追加を 1 度しか行っていないが、通常半教師あり学習によるデータの追加は複

数回行われる。データの追加を複数回行った場合の一致率の変化を調査することは今後の課題である。

## 6. おわりに

本研究では、麻雀の降りる局面の学習の半教師あり学習を適用する手法について提案した。提案手法では降りる局面の一致率について有意水準 5%で向上しているということができた。だが天鳳の上位 10%のプレイヤーに降りる局面かどうかの判断をさせた場合の一致率は全 106 局面と少ないものの 0.783 となっていたため、上級者プレイヤーの降り判断には及ばない結果になってしまった。半教師あり学習の際に使用する特徴の変更や、半教師あり学習を行う回数増加などで一致率の向上ができると考える。

降りる局面の学習についての一致率は向上させることができたが、これを麻雀プレイヤーに組み込み実際に対局させた場合にどの程度成績が上がるかということにはわかっていない。コンピュータ麻雀プレイヤー同士で対局を行わせる、あるいはコンピュータ麻雀プレイヤーと人間で対局を行うという環境があまり整っていないために現状ではデータを取ることも難しい状態である。今後コンピュータ麻雀プレイヤーの研究を進展させていくにあたって、対局する環境の整備が急がれる。

## 謝辞

本研究を進めるにあたり、タグ付きの牌譜データや牌譜データからラベル付きデータを出力するためのプログラムを公開していただき、また疑問に思った点に関する質問に回答して下さった東京大学の水上 直紀氏に心より感謝申し上げます。

## 参考文献

- [1] Murray Campbell, A. Joseph Hoane Jr., Feng-hsiung Hsu. Deep Blue. 2002. Artificial Intelligence(134). pp.57-83
- [2] 金子 知適, 田中 哲朗. 現役プロ棋士に勝ち越したコンピュータ将棋～第 2 回電王戦, 第 23 回世界コンピュータ将棋選手権速報～: 2. 多数の計算機を活用したゲーム木探索技術の進歩 -三浦弘行八段と GPS 将棋との対局を振り返って-. 2013. 情報処理 54(9). pp.914-922
- [3] 北川竜平, 三輪 誠, 近山 隆. 麻雀の牌譜からの打ち手評価関数の学習. 2007. ゲームプログラミングワークショップ 2007 論文集. pp.76-83.
- [4] インターネット雀荘「東風荘」のホームページ. <http://mj.giganet.net/>
- [5] 水上 直紀, 中張 遼太郎, 浦 晃, 三輪 誠, 鶴岡 慶雅, 近山 隆. 多人数性を分割した教師付き学習による 4 人麻雀プログラムの実現. 2014. 情報処理学会論文誌 55(11).
- [6] オンライン対戦麻雀ゲーム 天鳳. <http://tenhou.net/>
- [7] Chih-Chung Chang, Chih-Jen Lin(2105). LIBSVM - A Library for Support Vector Machines. <https://www.csie.ntu.edu.tw/~cjlin/libsvm/>
- [8] Xiaojin Zhu. Semi-Supervised Learning Literature Survey. 2008.
- [9] 水上 直紀 (2015). Naoki Mizukami. <http://www.logos.t.u-tokyo.ac.jp/~mizukami/> (参照:2015-7-1).
- [10] 林 伸也, 浦 晃, 三輪 誠, 田浦 健次朗, 近山 隆. 自己対戦棋譜を利用した半教師あり学習による将棋の評価関数の学習. ゲームプログラミングワークショップ 2011 論文集, pp.143-149