2020年度 卒業論文

麻雀における他家の手牌の待ち予測

〇〇年〇〇月〇〇日

電気情報工学科

（学生番号： 1TE16240N）

松田 真治

九州大学 工学部

**概要**

本研究では麻雀において他家の手牌の待ちを予測する手法を提案する。場の情報、他家の捨て牌、他家の副露状況から機械学習を用いて待ち牌を予測する。良い結果は出ていないが人間のセオリーと一致する部分も多く、一定の学習はなされていると考えられる。

**目次**

1. はじめに
   1. 研究背景
   2. 研究目的
   3. 研究手法
2. データ
3. 入力に順序関係の情報をもたせる重要性

3.1 捨て牌読みにおける順序関係の情報の重要性

3.2 手出しとツモ切り

3.3 副露後の打牌

4. 待ち牌の推定

4.1 アルゴリズム

4.2 順序関係の情報を持った捨て牌の表現

4.3 順序関係の情報を持たない捨て牌の表現

4.4 副露（チー）の表現

4.5 副露（ポン）の表現

4.6 副露（カン）の表現

4.7 実装

5. 実験・結果

5.1 牌番号による比較

5.2 順序関係の情報の有無による比較(1)

5.3 順序関係の情報の有無による比較(2)

5.4 捨て牌順序2つと3つの比較

5.5 半荘数による比較

5.6 打牌数による比較

1. **はじめに（研究背景、従来研究、自分の研究）**

**1.1 研究背景**

麻雀とは不完全情報ゲームである。将棋や囲碁のようにすべての情報がプレイヤーに開示されている完全情報ゲーム異なり、限られた情報の中から他家の状況を推察しつつ自分に与えられた材料を用いて期待値を最大化するように選択を行い続けるゲームある。

例えば高い手をもらった場合でも他家にスピード感を感じた場合には妥協して副露したり、安く勝算の低い手をもらった場合でも他家の進行速度が遅いと判断した場合には押さえつけるためにリーチしたりすることがある。このように麻雀では自分都合のみで選択を行うことは少なく、常に他家の状況を推察しながら打牌の選択を行う。

これは現実世界でいうところの押し引きと非常に類似している。したがって機械学習を用いて麻雀のような不完全情報ゲームに関する研究を行うことは現実に起きうる問題の押し引きに役に立つ可能性がある。

**1.2 研究目的**

本研究の目的は押し引きの際に特に重要な要素である他家の待ち牌予測を新たな手法を用いて行うことである。

**1.3 研究手法**

先行研究[1][2]などにおいて待ち牌の予測に用いられる情報はある色に着目したときの捨てられた枚数や牌の種類の枚数などの順序関係の情報を持たない情報であった。

しかし捨て牌から他家の手牌の推測を行う際に相手がどの順で手牌を切ったか、またどのように副露した後にどのような牌を切ったかなどといった順序関係に関連した情報は極めて重要なものだと一般的に考えられている。そこで本研究では入力に順序関係の情報を与える手法について提案する。

**1.4 簡単な麻雀のルール**

麻雀のルールについて簡単に説明する。

まず麻雀とは34種\*4枚の合計136枚の牌を用いて4人で行われるゲームである。









図1. 麻雀牌

麻雀牌は1~9の牌からなる数牌と東南西北白発中からなる字牌から構成される。また数牌には萬子、筒子、索子といった種類が存在する。図1で言えば上から順番に萬子の1~9,筒子の1~9,索子の1~9,字牌となっている。

麻雀の目的は手牌を和了形にすることである。和了の形は一般形、七対子形、国士無双形があるが、ここでは一般形のみ触れる。

一般形とは1雀頭と4面子から構成される形である。雀頭とは同じ牌を2枚集めたものである。面子とは３枚または4枚の牌から構成される形で順子と刻子と槓子の3種類がある。順子とは数牌の種類が同じで数字が連続した3つの牌の集まりのことで、刻子とは同じ3つの牌の集まりのことである。槓子とは同じ4つの牌の集まりのことである。

図2. 雀頭の例

図3. 順子の例

図4. 刻子の例

図5. 槓子の例





図6. 和了一般形の例

4人のプレイヤーはゲーム開始時にそれぞれ13枚ずつ牌を配られ、親から順に山から一枚牌を取り、手牌から一枚牌を捨てるという動作を繰り返していく。この過程の中で上記のような和了形が完成すると和了となり、役に基づいた点数を獲得できる。

役とは和了形が特定の条件を満たしていた場合に付加されるボーナスのようなものである。例えば混全帯么九という役がある。この役の条件は雀頭と全ての面子が1,9,または字牌を含むことである。従って図6の和了は混全帯么九の条件を満たしており、獲得できる点数が上昇する。

また麻雀には和了を目指す方法として副露というものがある。副露とは自分の手牌にあと一つ牌が揃えば面子が完成するという状況で、かつ他人がその足りない一つの牌を捨てた場合に、出来上がった面子を他家に晒すことと引き換えにその牌を拾って、面子を完成させることである。

副露にはチーとポンとカンが有る。チーとは副露して順子の形を作ることであり、これは自分の左側からしか牌を拾うことができない。ポンとは副露して刻子の形を作ることであり、誰からでも牌を拾うことができる。カンとは副露して槓子の形を作ることであり、これも誰からでも拾うことができる。またカンは他家の牌を拾わずとも、自分で四枚同じ牌を集めた時点で、晒すことができる。

麻雀とはこのように、副露などを駆使しつつうまく手牌を役と絡めて和了を目指し、得た点数の多寡を競い合うゲームである。

**2.データ**

**2.1 概略**

牌譜はオンライン麻雀サイト天鳳の最上位卓である鳳凰卓のものを利用した。牌譜は天鳳公式サイト[3]にて無償で公開されている。期間は2017年度のもの全てで、ルールは四人打東南戦赤あり喰い断ありである。

また天鳳の牌譜はXMLファイルによって特殊な形式で表現されているため、機械学習で用いるためには解析をする必要がある。解析については小林聡氏のブログ[4][5][6][7]を参考にした。

本研究ではツモ情報と打牌情報と和了情報のみを利用した。

**2.2 牌譜解析**

天鳳の牌譜はXMLファイルであるが、仕様に準拠した表記はされておらず、実質、天鳳の独自構文であると考えて良い。

具体的にはゲーム中に起こった事象をタグとその属性で表現し、それが起こった順に並べられているだけである。閉じタグなども存在していない。

天鳳の牌譜は1ファイルが1半荘を表す。

<INIT>タグは局の開始を表現し、<AGARI>タグ、<RYUKYOKU>タグはそれぞれ、和了があった場合の情報、和了がなかった場合の情報が表現されている。<AGARI>タグも<RYUKYOKU>タグもともに局の終了を表現するタグである。すなわち<INIT>タグから<AGARI>タグまたは<RYUKYOKU>タグまでで一局を表現している。1つのXMLファイルにこの<INIT>タグから<AGARI>タグまたは<RYUKYOKU>タグまでが局数分書いてある。また様々なタグを用いて<INIT>タグから<AGARI>タグまたは<RYUKYOKU>タグタグの間に、どのような打牌があったか、どのような副露があったか、リーチはいつ行われたかなどの情報が表記されている。本研究では和了の情報のみを抽出したため<INIT>から<AGARI>までの情報のみを利用した。

本研究で利用したタグについて簡潔に説明する。

まず前提として、天鳳の牌譜では牌に0~135までの牌番号が振られている。

ツモ情報は<T(牌番号)>, <U(牌番号)>, <V(牌番号)>, <W(牌番号)>の形で表記されている。T,U,V,Wは誰がツモしたかを表現している。また、打牌情報は<D(牌番号)>, <E(牌番号)>, <F(牌番号)>, <G(牌番号)>の形で表記されている。D,E,F,Gは誰が打牌したのかを表現している。

和了情報は<AGARI>タグで表現されている。AGARIタグは複数の属性を持つが、本研究で利用したのは誰が和了ったのかを表すwho属性と和了者の副露状況を表すm属性である。m属性は独特な表記がなされているため、上記の小林氏のブログを参考にしていただきたい。また副露情報には誰から副露したかなどの情報も入っているが、今回は入力次元数の削減のため、削除した。

**3.入力に順序関係の情報を持たせる重要性**

**3.1 捨て牌読みにおける順序関係の情報の重要性**

他家の捨て牌から相手の待ち牌を読む場合に、牌を切る順番などの順序関係の情報が重要になる理由について一つの例を用いて簡潔に説明する。

まず前提として麻雀において数牌は端に近づけば近づくほど利用価値が低くなる。例えば1で構成できる面子は123,111の2パターンしか無いが、5で構成できる面子は345,456,567,555の4パターンある。

したがって基本的に牌の利用価値は1,9<2,8<3-7である。仮に不要な牌1と5の両方を持っていて、いずれ二枚とも捨てるとしても、捨てる順序は1,5の順になるのがセオリーである。しかし実際にゲームをやってみると往々にして5,1の順で捨てるパターンが存在する。この順番に捨てる理由は様々ではあるが例えば端牌に関連する手役である混全帯么九などを狙っていることが多い。したがって1,5の順で切られている場合に比べると5,1の順で切られている場合の方がわずかに端牌の危険度が高く、中張牌の危険度は低くなると考えるのがセオリーである。

このように捨て牌の情報に順序関係をもたせると推測できる要素が増えるため、入力に順序関係の情報を与えることは重要だと考えられる。

**3.2手出しとツモ切り**

麻雀では牌の切り方に2つのパターンが存在する。手出しとツモ切りである。手出しとは牌を山からツモした後、もともと手牌にあった牌を切ることで、ツモ切りとは牌を山からツモした後そのツモした牌をそのまま切ることである。

この２つは似ているようで大きく意味が異なる。

手出しされた牌とは、何らかの理由があって手牌に残していたが、ツモってきた牌の価値がより高かったために切られてしまった牌である。言い換えれば捨て牌にある牌よりは価値が高く、手牌にある牌よりは価値が低い牌である。

一方ツモ切りされた牌とは手牌にある牌よりは価値が低いが、捨て牌にある牌より価値が高いとは必ずしも言えない牌である。

3.1では5,1の順で牌を切った場合、端牌の危険度が増すと述べたが、実は1がツモ切りだった場合、この推測は成立しない。あくまでもこの推測は1と5が比較されたことが前提のためである。

したがって捨て牌に順序関係の情報を持たせる際には、切られた牌が手出しであるかツモ切りであるかという情報を盛り込むことによってより待ち牌推測の性能が向上すると推測される。

**4.機械学習のためのデータの表現**

**4.1 アルゴリズム**

以下のようなパラメータを用いて機械学習を行った。

・順序関係の情報を持った捨て牌

・順序関係の情報を持たない捨て牌

・副露

**4.2 順序関係の情報を持った捨て牌の表現**

順序関係の情報を持った捨て牌の作り方について説明する。

簡潔に言えば任意の二枚の牌を捨てる順序関係について一意に数字を割り当てて管理すれば良い。順序関係に数字を割り当てる方法について説明する。

まず以下のようにして予め牌と番号との対応を決定する。

|  |  |
| --- | --- |
| 牌 | 番号 |
| ツモ切り一萬〜九萬 | 0~8 |
| ツモ切り一筒〜九筒 | 9~17 |
| ツモ切り一索〜九索 | 18~26 |
| ツモ切り東、南、西、北 | 27~30 |
| ツモ切り白、発、中 | 31~33 |
| ツモ切り五萬、五筒、五索 | 34~36 |
| 手出し一萬〜九萬 | 37~45 |
| 手出し一筒〜九筒 | 46~54 |
| 手出し一索〜九索 | 55~63 |
| 手出し東、南、西、北 | 64~67 |
| 手出し白、発、中 | 68~70 |
| 手出し五萬、五筒、五索 | 71~73 |

表1. 牌と番号の対応付け

ツモ切りと手出しで牌の種類が違うとみなすと、牌の種類は合計74種類と考えることができる。74\*74=5476であるから二枚の牌の順序関係は0~5475で表現することができる。

二枚のパイを捨てた順序関係に対応する数字の割当は自由であるが、本研究では(先に捨てた牌の牌番号) \* 74 + (後に捨てた牌の牌番号)という割り当てを行った。

例として以下の捨て牌からデータを抽出する。



図2. 捨て牌例（すべてツモ切りとする）

はじめに捨て牌からすべての順序関係を抜き出す。上図の場合は（五萬,三筒）,（五萬,六索）,（五萬,東）,(三筒, 六索), (三筒, 東),(六索,東)である。これを牌番号に置き換える。置き換えたものが(4,11),(4,23),(4,27),(11,23),(11,27),(23,27)である。今度は組み合わせを(x,y)としたとき74\*x+yを全ての組み合わせについて取るとそれぞれ307,319,323,837,841,1729となる。

よって入力として図○のような5476列の表を用意し、列307,319,323,837,841,1729の値を1にする。

このようにして入力に順序関係の情報をもたせることが可能になる。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | ・・・ | 307 | ・・・ | 5474 | 5475 | 5476 |
| 0 | 0 | 0 | ・・・ | 1 | ・・・ | 0 | 0 | 0 |

表3. 入力

このようにすることで、任意の２つの牌を切った順序を表現することができ、入力に順序関係の情報を与えることができる。

**4.3 順序関係の情報を持たない捨て牌の表現**

要するに何を切ったかという情報である。図2の捨て牌からデータを抽出する事を考える。

列数74、初期値0の表を考える。次に捨て牌のそれぞれの牌番号について対応する列の値をインクリメントしていく。

**4.4 副露情報（チー）**

赤牌を含むかどうかで2パターン存在する。

どこの牌を鳴いたかで3パターン存在する。

副露した牌の中で最小の牌番号をもつ牌の種類で21パターン存在する。

したがって126列あれば全てのチーは表現できるので適宜数字と対応させる。

**4.5 副露情報（ポン）**

赤牌を含むかどうかで2パターン存在する。

どの牌を鳴いたかで34パターン存在する。

したがって68列あれば全てのポンは表現できるので適宜数字と対応させる。

**4.6 副露情報（カン）**

カンの種類（暗カン・明カン・加カン）で3パターン存在する。

どの牌を鳴いたかで34パターン存在する。

したがって102列あれば全てのカンは表現できるので適宜数字と対応させる。

**4.7 実装**

Chainerで３層ニューラルネットを構築して学習を行った。第一層の入力数は入力次元数、出力数は100。第二層の入力数は100、出力数は100。第三層の入力数は100、出力数は2。最適化関数はSGDを利用した。エポック数30、バッチサイズ16で学習を行った。

データは49%を訓練データセットとして、21%を検証データセットとして、30%をテストデータセットとして利用する。

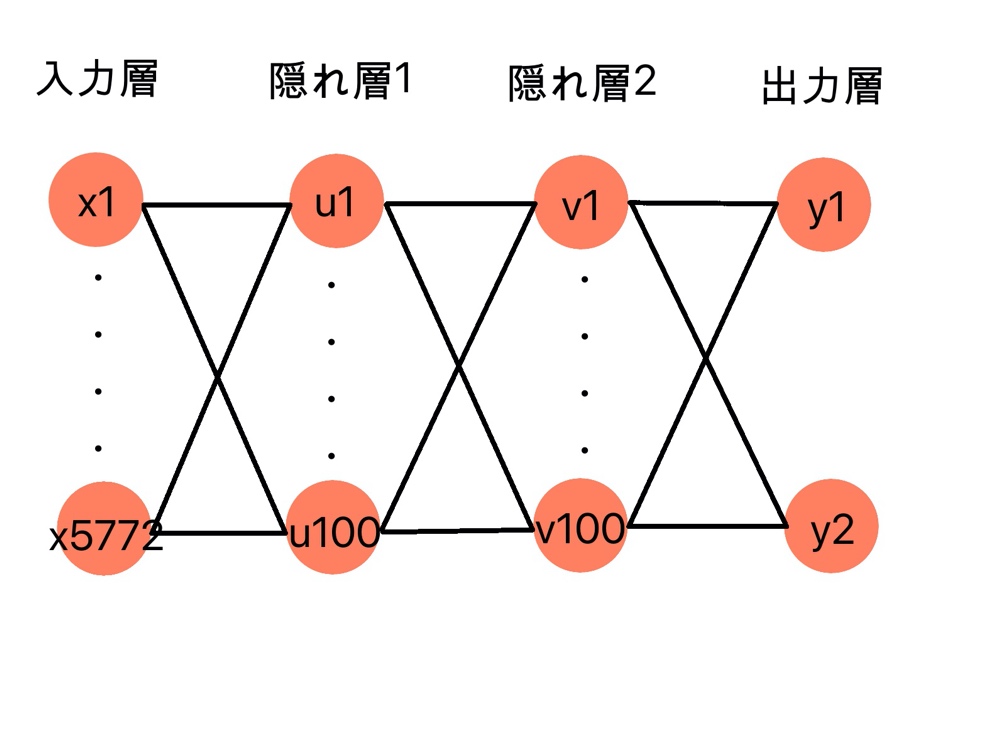


図. NNイメージ図

**5.実験・結果**

**5.0 サイコロを振った場合**

参考までにサイコロで当たり牌か当たり牌でないかを決めた場合の結果を示す。

確率aで当たり牌だと決めるとすると

precision = tp / (tp + fp) = 1 / 2

recall = tp / (tp + fn) = a

f1score = 2 \* recall \* precision / (recall + pre) = a / (a + 1 / 2)

となる。したがって確率a = 1とすると、f1scoreは最大値2 / 3を取る。

**5.1 牌番号による比較**

使用したパラメータ：順序関係の情報を持った捨て牌、副露

和了数：251128

全ての牌について待ち予測を行なった。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 牌 | Acc | Acc | Pr0 | Pr1 | Re0 | Re1 | Fs0 | Fs1 |
| 一萬 | 0.999 | 0.944 | 0.956 | 0.117 | 0.987 | 0.038 | 0.971 | 0.057 |
| 二萬 | 0.999 | 0.915 | 0.938 | 0.124 | 0.973 | 0.057 | 0.955 | 0.078 |
| 三萬 | 0.999 | 0.895 | 0.926 | 0.129 | 0.964 | 0.065 | 0.944 | 0.087 |
| 四萬 | 0.999 | 0.867 | 0.910 | 0.139 | 0.947 | 0.085 | 0.928 | 0.105 |
| 五萬 | 0.999 | 0.861 | 0.905 | 0.160 | 0.946 | 0.094 | 0.925 | 0.119 |
| 六萬 | 0.999 | 0.871 | 0.912 | 0.138 | 0.949 | 0.081 | 0.930 | 0.102 |
| 七萬 | 0.999 | 0.897 | 0.927 | 0.120 | 0.965 | 0.059 | 0.945 | 0.079 |
| 八萬 | 0.999 | 0.913 | 0.936 | 0.120 | 0.973 | 0.052 | 0.954 | 0.073 |
| 九萬 | 0.999 | 0.944 | 0.956 | 0.092 | 0.986 | 0.029 | 0.971 | 0.045 |
| 一筒 | 0.999 | 0.946 | 0.957 | 0.125 | 0.988 | 0.038 | 0.972 | 0.058 |
| 二筒 | 0.999 | 0.915 | 0.937 | 0.121 | 0.975 | 0.050 | 0.956 | 0.070 |
| 三筒 | 0.999 | 0.897 | 0.927 | 0.112 | 0.965 | 0.055 | 0.946 | 0.074 |
| 四筒 | 0.999 | 0.871 | 0.913 | 0.142 | 0.949 | 0.086 | 0.931 | 0.108 |
| 五筒 | 0.999 | 0.864 | 0.905 | 0.150 | 0.949 | 0.082 | 0.927 | 0.106 |
| 六筒 | 0.999 | 0.871 | 0.911 | 0.143 | 0.952 | 0.080 | 0.931 | 0.102 |
| 七筒 | 0.999 | 0.899 | 0.930 | 0.122 | 0.965 | 0.063 | 0.947 | 0.083 |
| 八筒 | 0.999 | 0.916 | 0.937 | 0.113 | 0.976 | 0.045 | 0.956 | 0.064 |
| 九筒 | 0.999 | 0.945 | 0.955 | 0.092 | 0.988 | 0.026 | 0.971 | 0.040 |
| 一索 | 0.999 | 0.946 | 0.957 | 0.102 | 0.989 | 0.028 | 0.972 | 0.045 |
| 二索 | 0.999 | 0.913 | 0.937 | 0.129 | 0.972 | 0.060 | 0.954 | 0.082 |
| 三索 | 0.999 | 0.902 | 0.930 | 0.126 | 0.968 | 0.060 | 0.948 | 0.081 |
| 四索 | 0.999 | 0.872 | 0.914 | 0.145 | 0.949 | 0.089 | 0.931 | 0.110 |
| 五索 | 0.991 | 0.846 | 0.907 | 0.140 | 0.924 | 0.114 | 0.915 | 0.126 |
| 六索 | 0.999 | 0.871 | 0.912 | 0.140 | 0.950 | 0.083 | 0.931 | 0.104 |
| 七索 | 0.999 | 0.900 | 0.928 | 0.125 | 0.967 | 0.060 | 0.947 | 0.081 |
| 八索 | 0.999 | 0.916 | 0.939 | 0.126 | 0.974 | 0.057 | 0.956 | 0.078 |
| 九索 | 0.999 | 0.947 | 0.958 | 0.100 | 0.988 | 0.031 | 0.973 | 0.047 |
| 東 | 0.999 | 0.984 | 0.984 | 0.029 | 0.999 | 0.001 | 0.992 | 0.002 |
| 南 | 0.999 | 0.985 | 0.986 | 0.064 | 0.999 | 0.004 | 0.992 | 0.007 |
| 西 | 0.999 | 0.987 | 0.988 | 0.021 | 0.999 | 0.002 | 0.994 | 0.003 |
| 北 | 0.999 | 0.987 | 0.988 | 0.056 | 0.999 | 0.003 | 0.993 | 0.006 |
| 白 | 0.999 | 0.982 | 0.984 | 0.034 | 0.999 | 0.002 | 0.991 | 0.004 |
| 発 | 0.999 | 0.983 | 0.984 | 0.057 | 0.999 | 0.004 | 0.992 | 0.007 |
| 中 | 0.999 | 0.982 | 0.983 | 0.068 | 0.998 | 0.008 | 0.991 | 0.014 |

表4. 実験5.1の結果

上の結果から数牌は真ん中に近いほど推測の精度が上がり、端に近づくほど推測の精度が下がることがわかる。これは人間のセオリーとも合致する。

また字牌の推測の精度が数牌の推定の精度と比べると低い。これも人間のセオリーに合致する。

したがってある一定の学習はなされていると考えられる。

**5.2 順序関係の情報の有無による比較(1)**

使用したパラメータ：順序関係の情報を持たない捨て牌、順序関係の情報を持った捨て牌、副露

和了数：251128

予測した牌：五萬

一方は順序関係の情報をもった捨て牌＋副露で、他方は順序関係の情報を持たない捨て牌で待ち予測を行った。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 順序 | Acc | Acc | Pr0 | Pr1 | Re0 | Re1 | Fs0 | Fs1 |
| 無 | 0.901 | 0.900 | 0.900 | 0.363 | 0.999 | 0.001 | 0.947 | 0.002 |
| 有 | 0.999 | 0.866 | 0.903 | 0.159 | 0.953 | 0.080 | 0.927 | 0.106 |

表5. 実験5.2の結果

**5.3 順序関係の情報の有無による比較(2)**

使用したパラメータ：順序関係の情報を持たない捨て牌、順序関係の情報を持った捨て牌、副露

和了数:334630

予測した牌：五萬

一方は順序関係の情報をもった捨て牌＋副露で、他方は順序関係の情報を持たない捨て牌で待ち予測を行った。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 順序 | Acc | Acc | Pr0 | Pr1 | Re0 | Re1 | Fs0 | Fs1 |
| 無 | 0.902 | 0.901 | 0.901 | 0.0 | 0.999 | 0. | 0.948 | nan |
| 有 | 0.992 | 0.856 | 0.905 | 0.149 | 0.939 | 0.0979 | 0.921 | 0.118 |

表6. 実験5.3の結果

順序関係の情報有りのほうが良い推測の精度が出ている。よって順序関係の情報の有益性が確かめられた。

**5.4 捨て牌の順序2つと3つの組み合わせの比較**

使用したパラメータ：順序関係の情報を持った捨て牌(２つ版)、順序関係の情報を持った捨て牌(3つ版)、副露

和了データ数：6745

予測した牌：五萬

一方は順序関係の情報を持った捨て牌(２つ版)＋副露、他方は順序関係の情報を持った捨て牌(3つ版)+副露で推測を行った。順序関係の情報を持った捨て牌(3つ版)の作り方は4.2の方法を拡張しただけである。ただし3つに拡張すると捨て牌の表現だけで74\*74\*74=405224列になってしまい、メモリが不足してしまうため800半荘分でしか実験を行うことができなかった。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 版 | Acc | Acc | Pr0 | Pr1 | Re0 | Re1 | Fs0 | Fs1 |
| 2 | 0.959 | 0.899 | 0.912 | 0.111 | 0.983 | 0.020 | 0.946 | 0.035 |
| 3 | 0.996 | 0.910 | 0.912 | 0.000 | 0.997 | 0.000 | 0.953 | nan |

表7. 実験5.4の結果

2つ版のほうが高い精度で推測をできている。しかし半荘数があまりにも少なすぎて3つ版の学習が不十分だった可能性が高く、一概に3つ版が2つ版より劣っているとは言い切れない。

**5.5 和了データ数による比較**

使用したパラメータ：順序関係の情報を持った捨て牌、副露

予測した牌：五萬

和了データ数83613、167675, 251128のデータでそれぞれ待ち予測を行った。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 和了数 | Acc | Acc | Pr0 | Pr1 | Re0 | Re1 | Fs0 | Fs1 |
| 83613 | 0.999 | 0.877 | 0.902 | 0.147 | 0.968 | 0.0488 | 0.934 | 0.0734 |
| 167675 | 0.999 | 0.863 | 0.903 | 0.161 | 0.949 | 0.0886 | 0.926 | 0.114 |
| 251128 | 0.998 | 0.858 | 0.904 | 0.157 | 0.942 | 0.0987 | 0.923 | 0.121 |

表8. 実験5.5の結果

半荘数を増やすに連れ推測性能が良くなっていることがわかる。

（もっと半荘数を増やして実験を行う予定）

**5.6 打牌回数による比較**

使用したパラメータ：順序関係の情報を持った捨て牌、副露

和了数：167675

予測した牌：五萬

一方は何も条件を加えずに推測し、他方は打牌が10回以上あった上がりのみを抽出して推測を行っている。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 打牌制限 | Acc | Acc | Pr0 | Pr1 | Re0 | Re1 | Fs0 | Fs1 |
| 無 | 0.999 | 0.863 | 0.903 | 0.161 | 0.949 | 0.0886 | 0.926 | 0.114 |
| 有 | 1.0 | 0.865 | 0.900 | 0.177 | 0.955 | 0.0828 | 0.927 | 0.113 |

表9. 実験5.6の結果

打牌回数を十回以上のものに限定した場合のほうが入力が密になるためより良い推測の精度が出ると想定していたが、実際にはリコールなどにおいて打牌回数を制限していないもののほうが良い推定の精度がでた。

（正直原因はよくわからないのでもうちょっと実験してみます。）

**6．考察**

**7．終わりに**

**参考文献**

[1]我妻敦,原田将旗,森田一,古宮嘉那子,小谷善行. SVRを用いた麻雀における捨て牌危険度の推定.

[2]栗田萌,保木邦仁. 麻雀における他家の手牌と待ちの予測に基づく放銃確率推定.

[3]オンライン対戦麻雀 天鳳 / ログ, <https://tenhou.net/sc/raw/>

[4] 天鳳の牌譜を解析する(1),

<https://blog.kobalab.net/entry/20170225/1488036549>

[5] 天鳳の牌譜を解析する(2),

<https://blog.kobalab.net/entry/20170228/1488294993>

[6] 天鳳の牌譜を解析する(3),

<https://blog.kobalab.net/entry/20170312/1489315432>

[7] 天鳳の牌譜を解析する(4),

<https://blog.kobalab.net/entry/20170720/1500479235>