

畳み込み機能を強化した麻雀プレイヤーの行動モデルの学習

水口 琳南[†] 藤田 悟[†]
法政大学 情報科学部[†]

1. まえがき

現在、囲碁や将棋などの完全情報ゲームにおいて、AI プレイヤーはプロに勝利できるまでの成長を遂げている。一方、ポーカーや麻雀、人狼などの不完全情報ゲームにおいては、AI が人間に勝利することは比較的難しいとされている。中でも、麻雀は隠された内容が圧倒的に多く、ゲームの流れを予測することが難しい。開発されている麻雀 AI で最も強いといわれているものは、マイクロソフトの研究開発機関が開発した Suphx(Super Phoenix) であり、この AI プレイヤーは捨て牌と3つの副露牌、立直の計5つのネットワークを学習している[1].

本研究では、Suphx が先行研究に挙げているネットワークをベースモデルとし、入力に与える特徴量を牌の種類ごとに分割し、ネットワークに与えることで予測精度の向上を図る[2]. また、特徴量方向の畳み込みを行うことでより強い麻雀 AI を作成する。

2. 提案手法

2.1. 入力データの構造

ベースモデルは34×4の配列を複数用いて表現し、その入力に対して畳み込みをしていた。ここで、34 は牌の種類数の総数を表す。しかし、この方法では、34 種類の牌の両端に位置する牌の情報が、ほかの牌の情報と比較して参照される回数が少ない。また、2 種類の数牌をまたがる部分の無駄な畳み込みも生じる。そこで、入力を牌の種類ごとに分割することでこれらの問題の解決を試みた。

ネットワークには、対局中の情報を萬子、筒子、索子、字牌に分けて表現したもの 4 つを入力として与えた。数牌の情報は9×4の配列を複数個用いて表現し、字牌の情報は7×4の配列を複数個用いて表現する。配列のサイズの4は各牌の最大枚数であり、枚数に合わせて左詰めで1を与える、7と9は字牌と数牌それぞれの種類を意味している。手牌情報を9×4と7×4の配列で表現したときの各配列の様子を図1に示す。

対局中の各局面の情報は「手牌、4人の捨て牌、4人の副露牌、ドラ表示牌、自分以外の3人の立直状況、場風、自風、1巡前の13の特徴量、2から4巡前の9の特徴量」の計55の特徴量で表現し、萬子と筒子、索子に関する入力構造はそれぞれ55×9×4、字牌に関する入力構造は55×7×4とする。捨て牌と副露牌は1局を通して累積の情報であり、立直状況は相手が立直しているときは配列のすべての要素を1に、していないときは全ての要素を0にすることで表現する。また、場風と自風は、その牌と対応する行のすべての要素を1とし、それ以外の要素を0で埋めることで表現する。1巡前の特徴量は、手牌と4人の捨て牌、4人の副露牌、ドラ表示牌、自分以外の3人の立直状況を含み、2から4巡前の特徴量は、手牌と4人の捨て牌、4人の副露牌を含む。どちらにも場風と自風の情報を含まない理由は、これらは1局を通して変化しないからである。また、誰かが立直した直後の1巡は対局

中において最も警戒すべき1巡である。さらに、誰かが槓をした直後、すなわちドラ表示牌が増えることもゲームの流れに大きな変化を与えるため、1巡前の特徴量は他と比較して多くの情報を含んでいる。数巡前の特徴量を入力に含めることで、捨て牌や副露牌の順序を考慮した学習が可能になる。

捨て牌ネットワークの正解データは長さ34の配列で表現する。34は牌の種類を意味し、捨てた牌と対応する要素を1に、それ以外の要素は0で埋める。ポンネットワークと立直ネットワークの正解データは長さ2の配列で表現し、ポンや立直をしなかった場合はインデックス0の要素を1とし、した場合はインデックス1の要素を1とする。また、チーネットワークの正解データは長さ4の配列で表現する。チーしなかった場合はインデックス0の要素を1とする。チーした場合は、順子のうち最小の数牌を鳴いたときはインデックス1の要素を1に、中央の数牌を鳴いたときはインデックス2の要素を1に、最大の数牌を鳴いたときはインデックス3の要素を1とする。また、ポンと立直、チーネットワークに与えるデータは、該当行為の選択可能局面のみであり、行為が不可能な局面は学習データから除外した。

2.2. ネットワーク構造

特徴量方向の畳み込みを行うため、55×9×4、または55×7×4の入力を1×9×220、または1×7×220に変形しフィルタ数512を持つ2次元畳み込み層に与える。その出力に対して、牌の種類軸と、フィルタ数512の軸に対して、フィルタ数50を持つ2次元の畳み込みを2回行う。その出力を平坦化したものを300ニューロンの全結合層に入れ、最終的な出力を得る。各畳み込み層と全結合層のあとにはバッチ正規化層とドロップアウト層を追加することで過学習を抑え、活性化関数はReLU関数を採用する。1層目の畳み込みには、サイズが220×1、ストライド1のカーネルを用いる。また、2層目と3層目の畳み込みに関しては、数牌の入力を与える層にはサイズが3×3、ストライド1のカーネルを、字牌の入力を与える層にはサイズが3×1、ストライド1のカーネルを用いる。

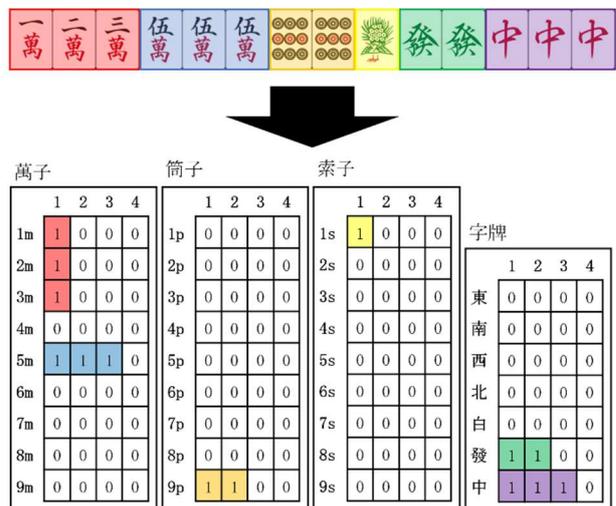


図1：手牌の表現方法

Learning behavioral models of mahjong players with enhanced convolution

[†]Rinna Mizuguchi and Satoru Fujita, Computer and Information Sciences, Hosei University

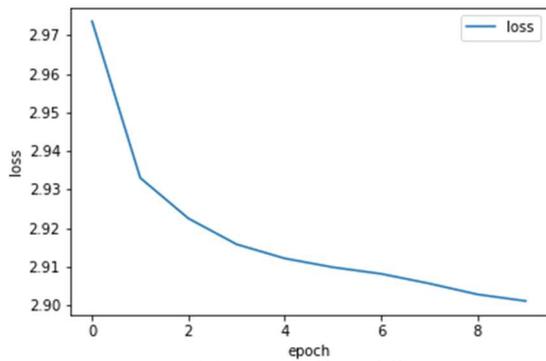


図 2：ロス値の推移

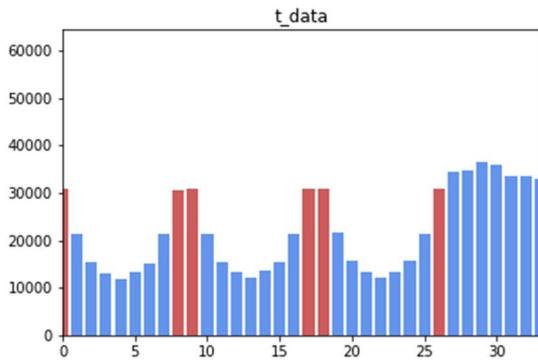


図 3：テストデータの正解データの分布

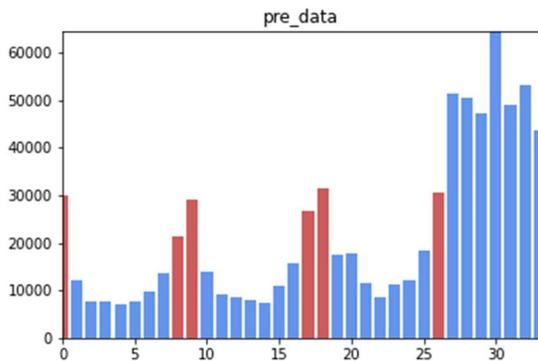


図 4：捨て牌ネットワークの予測分布

構築したネットワークは捨て牌ネットワークとチーネットワーク、ポンネットワーク、立直ネットワークの 4 つであり、出力層を除いてすべて同様の構造を用いる。また、捨て牌予測は 34 クラス分類、チー予測は 4 クラス分類、ポンと立直予測は 2 クラス分類問題として扱う。そのため、捨て牌予測とチー予測のネットワークは出力層に Softmax 関数を用い、ポン予測と立直予測のネットワークは出力層に Sigmoid 関数を用いる。

3. 実験と結果

3.1. 実験

教師あり学習を用いて、チー、ポン、立直、捨て牌の予測を行ったが、本稿では、捨て牌ネットワークの学習についての結果だけを示して議論する。入力データは、オンライン麻雀サイト天鳳にて鳳凰位を獲得した、10 人の 4 人麻雀の牌譜を用いて作成した[3]。8 人分のデータを訓練データとし、残りの 2 人分のデータをそれぞれ検証用データとテストデータとした。ネットワークの評価指数には、テストデータにおける予測精度を用いた。ま

表 1：捨て牌ネットワークの学習結果

	Rank1	Rank2	Rank3
提案手法	0.601	0.777	0.847
ベースモデル	0.688	0.936	0.965
Suphx	0.767	-	-

た、ベースモデルの予測精度と Suphx の教師あり学習完了時の予測精度との比較も行った。

3.2. 学習結果

バッチサイズを 64 とし、61011 組の訓練データと 15046 組の検証用データ、11966 組のテストデータを用いて、学習を行った。エポック数 10 で学習したときの訓練データにおける最終的な予測精度は 0.674 に達し、テストデータにおける予測精度は 0.601 となった。また、予測を第 3 候補までひろげたときの予測精度は第 2 候補で 0.777、第 3 候補で 0.847 となり、ベースモデルや Suphx の予測精度を下回る結果を得た。学習データにおけるロス値の推移を図 2 に示し、提案手法の実験結果とベースモデル、Suphx の予測精度を表 1 に示す。論文内に記載がないものは「-」で示した。また、テストデータの正解データの分布と予測結果の分布を図 3、4 に示す。この分布から、本ネットワークは実際よりも中張牌を手牌に残し、字牌を頻繁に捨てるよう学習したことがわかる。

4. 考察

本論文のネットワーク構造とベースモデルの構造、Suphx のネットワーク構造とで大きく異なる点は、特徴量方向の畳み込みの有無と、牌の種類が異なる部分の畳み込みの有無の 2 点である。実験の結果、ベースモデルと Suphx の予測精度よりも低い結果となった。しかし学習時のロス値の推移から、値が収束しておらずネットワークの学習が途中であることがわかる。このことから学習回数を増やすことによる予測精度の向上が予測できる。また、麻雀サイトでは 23 人分の牌譜が公開されているため、学習データ量を増加させることによる予測精度の向上も期待できる。

5. むすび

本論文では、教師あり学習を用いて麻雀の牌譜を学習することで、麻雀 AI プレイヤを作成した。ニューラルネットワークの構造は先行研究をベースとし、ネットワークを構築し学習した。牌の種類で入力を分割すること、それにより牌の種類をまたがる部分の畳み込みを削除すること、また特徴量方向の畳み込みを行うことで予測精度の向上を試みた。学習の結果、テストデータにおけるそれぞれの予測精度は、捨て牌ネットワークで 0.601 となり、ベースモデルと Suphx を下回る予測精度となった。しかし、学習回数を増やすことや学習データを増やすことで予測精度の向上を期待できる。

文 献

- [1] Li, J., Koyamada, S., Ye, Q., Liu, G., Wang, C., Yang, R., Zhao, L., Qin, T., Liu, T.-Y. and Hon, H.-W.: Suphx: Mastering Mahjong with Deep Reinforcement Learning, arXiv preprint arXiv:2003.13590 (2020).
- [2] S.Gao, F.Okuya, Y.Kawahara, and Y.Tsuruoka: Supervised learning of imperfect information data in the game of Mahjong via deep convolutional neural networks, The 23rd Game Programming Workshop of the Information Processing Society of Japan, 2018
- [3] Tenhou: <https://tenhou.net/>. (Online; accessed 28-August 2023).