機械学習を用いた麻雀 AI の構築

1　はじめに

近年，機械学習や深層学習の手法に基づく，ゲームをプレイする人工知能 (Artificial Intelligence, AI) プログラムが数多く開発されている．本研究では，不完全情報ゲームの 1 種である麻雀を対象とする．

麻雀では，あるプレイヤが捨てた牌の種類や順番などから，プレイヤが現在手持ちにしている牌の状態の予測が可能である．この予測は戦術構築の際に重要であり，一般的に上級者ほど予測精度は高いと考えられる．従来手法 [1] では Recurrent Neural Network (RNN) の一種である Long Short-Term Memory (LSTM) によるニューラルネットワークモデルを用いて，捨てた牌の時系列的な特徴を捉えることで予測精度を向上させていた．

本研究では，従来手法に基づいて，モデルを提案する．

2　麻雀とは

以下に麻雀のゲーム進行と用語について説明する．

麻雀は 4 人のプレイヤによってプレイされるゲームである．1 ゲームは局と呼ばれる単位によって区切られている．局のはじめに，各プレイヤはそれぞれ 13 枚の牌を手持ちにしており，1 枚手持ちに加えると，1 枚手持ちから捨てる行為を繰り返すことで牌を替えていく．牌の組み合わせで三枚の組 4 セットと二枚の組 1 セットという特定の条件を満たすことでアガリとなり，アガったプレイヤが点を得て局が終了する．局を複数回（通常 8 回程度）繰り返すことで 1 ゲームが終了する．終了時に最も得点の多いプレイヤの勝利となる．また，局の最中，プレイヤは他プレイヤがどのような牌を手持ちにしているか直接知ることはできず，この点から麻雀には不完全情報ゲームである．

- 鳴き
  他プレイヤが捨てた牌を利用して自身の手持ち牌をアガリへ近付ける行為．
- テンパイ
  アガリの一つ前の状態．他プレイヤの捨てた牌や鳴いた牌の情報などから手持ち牌がテンパイ状態であるかの予測をテンパイ予測と呼ぶ．
- リーチ
  手持ち牌がテンパイしている場合，リーチをかけることで得点は増える，しかしリーチをかけるとアガリ牌以外の牌は全部捨てないといけない．
- シャンテン数
  テンパイまで必要な有効牌の数はシャンテン数と呼ばれる．例えば，テンパイまで後 1 枚有効牌が必要の時は 1 シャンテンと呼ばれる．

3　要素技術

3.1　LSTM

Long Short Term Memory [2] ネットワークは，通常「LSTM」 と呼ばれる．長期的な依存関係を学習することのできる， RNN の特別な一種である．

すべてのリカレントニューラルネットワークは、ニューラルネットワークのモジュールを繰り返す，鎖状をしている．標準の RNN では，この繰り返しモジュールは，単一の tanh 層という，非常に単純な構造を持っている．

LSTM も鎖のような構造をもっているが，繰り返しモジュールは異なる構造を持っている．単一のニューラルネットワーク層ではなく，図 1 のように特別な方法で相互作用する，4 つの層を持っている．

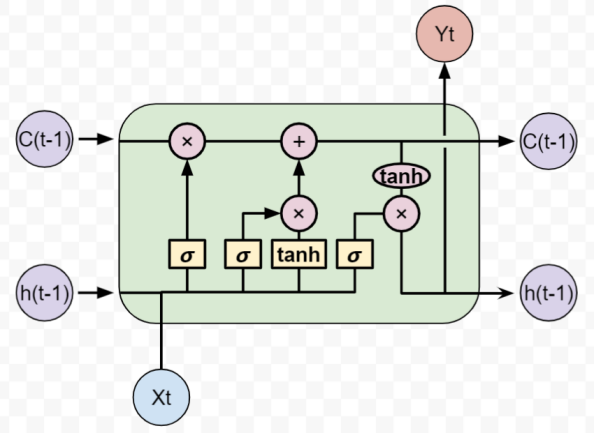

図 1: LSTM の構造

LSTM の中間出力は RNN でも中間情報として出力されていた隠れ状態と LSTM 特有のセル状態の 2 種類

がある．また，LSTM の特徴として忘却ゲート，入力ゲート，出力ゲートの 3 種類のゲートを持つというところがある．ゲートとは、情報をどの程度つぎの時刻に伝達するかを制御するコンポーネントであり、0 から 1 の値をとなる。

LSTM は RNN で不可能だった長期的特徴の学習を可能にしたセルだが、計算コストが大きいという問題点がある．

3.2 GRU

GRU [3] とは，LSTM の忘却ゲートと入力ゲートを単一の更新ゲートにマージし隠れ状態のみ伝達していくニューラルネットワークのモデルである．以下に具体的に説明する．

- 時間依存の状態数を減らす
  LSTM では C と h の二つが状態を保持している．これを一つにまとめ，記憶セルという構造はなくなる．
- ゲートコントローラ数を減らす
  LSTM では入力ゲート、忘却ゲート、出力ゲートに 1 つずつゲートコントローラが必要であるが，そこで，忘却ゲートと入力ゲートの操作を一つのコントローラで操作するように変更する．

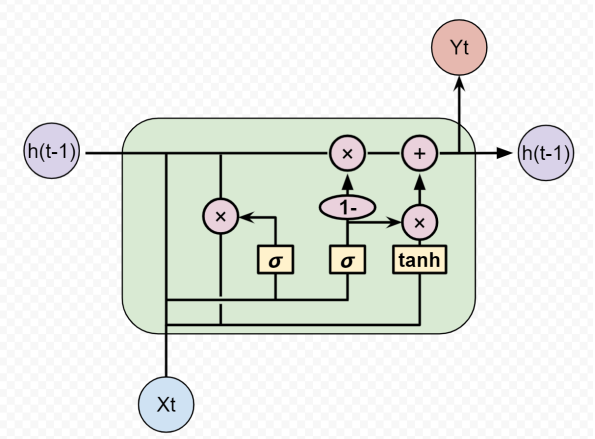

図 2: GRU の構造

3.3 Optimizer

Optimizer とは深層学習において重みパラメータを最適化するアルゴリズムである．本研究では以下の三つの最適化アルゴリズムを用いた．

- SGD Momentum
  SGD とは確率的勾配降下法というアルゴリズムである [4]．学習率が一定であり，収束結果が安定していることから今でも用いられている．そして Momentum SGD では，一つ前の勾配情報を用いて慣性を加える，安定した収束が得られる SGD の拡張手法である．
- Adagrad
  Adagrad とは各次元ごとに学習率を調整していくという手法である [5]．勾配が緩やかな次元での収束に時間かかるという問題を解決した．
- Adam
  現在最も使われている最適化アルゴリズムの一つである．Adam の正体は移動平均で振動を抑制するモーメンタムと学習率を調整して振動を抑制する RMSProp を組み合わせたアルゴリズムである [6].

4 提案手法の概要

4.1 従来手法

図 3 に従来手法のモデルを示す．従来手法ではいくつの問題がある．

- リーチは必ずテンパイしているという情報を利用していない
- リーチ後の捨て牌は選択できないため情報の価値が低い
- 目標の捨て牌と鳴き牌だけ使用している
- 1 シャンテンという重要な状態を考慮していない

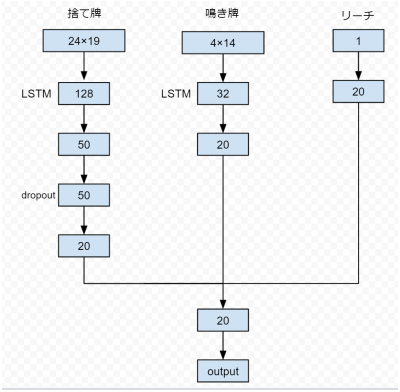

図 3: 従来手法のモデル

4.2 提案手法

図 4 に提案手法のモデルを示す．提案手法では従来手法のモデルの改善と拡張をした．

- 改善
  従来手法のリーチの入力部分とリーチ状態が正のデータを取り除いた．
- 拡張
  目標の捨て牌だけじゃなく，あたり牌予測モデルでも使われている他の人の捨て牌と自分の手牌も入力として使った．

  麻雀のプレイヤにとって，1 シャンテンも大事な情報である，相手がテンパイする一つ前に危険な牌を捨てることができる．そこで，モデルの出力を二値表現から三つの種類（テンパイ，1 シャンテン，それ以外）に変更した．

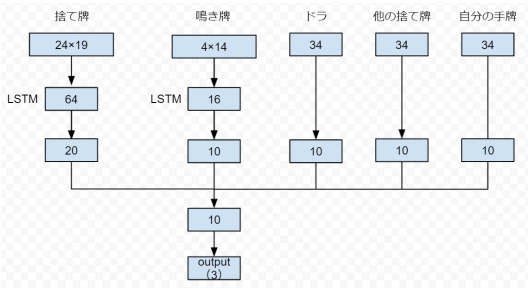

図 4: 提案手法のモデル

5 実験

本研究では，LSTM モデルを基づいて，モデルの改善と拡張をした,．データセットは天鳳の 2012 年から 2015 年までの鳳凰卓の牌譜を使っている．

5.1 実験 1

従来手法のモデルの LSTM の部分を GRU に変えて学習した．学習にかかる時間と学習後モデルの精度を比較し，二つモデルの性能を考察した．表 1 に実験条件を示す．

表 1: LSTM と GRU 比較実験の実験条件

最適化手法	Adagrad
損失関数	binary crossentropy
学習率	0.0001
エポック数	50
データ数	500000

5.2 実験 2

実験 2 では従来手法のモデルの改善と拡張した．それに全体的にモデルのパラメータ数を減らした．従来手法のモデルをテンパイ予測モデルと，本研究のモデルをシャンテン数予測モデルと呼ぶ．二つのモデルを同一データセットで学習し，精度を比較した．

5.3 実験 3

実験 2 で使ったテンパイ予測モデルとシャンテン数予測モデルをを使い，三つの最適化アルゴリズムを用いて学習した．最適化アルゴリズムごとに学習率を調整し，各アルゴリズムの性能を比較した．表 2 と表 3 に各アルゴリズムの学習率を示す．

表 2: テンパイ予測モデルの学習率

アルゴリズム	学習率
sgd	0.0002
adagra	0.001
adam 1	0.00001
adam 2	0.000001

表 3: シャンテン数予測モデルの学習率

アルゴリズム	学習率
sgd	0.001
adagrad	0.001
adam	0.00001

6 結果

6.1 実験 1 の結果

表 4 に実験 1 の結果を示す学習時間を少し減少したが，精度の劣化は見られなかった．

表 4: 実験 1 の結果

モデル	学習時間	正解率	0 の F 値	1 の F 値
LSTM	7763	0.9159	0.9539	0.5215
GRU	7400	0.9161	0.9540	0.5230

6.2 実験 2 の結果

表 5 にテンパイ予測モデルモデルの実験結果を示す，0 は「テンパイしていない」を，1 は「テンパイしている」を表す．表 6 にシャンテン数モデルの実験結果を示す，0 は 0 シャンテンいわゆる「テンパイしている」を，1 は「 1 シャンテン」を，2 はそれ以外の状態を表す．二つのモデルの共通の部分 (テンパイ予測モデルの 1 とシャンテン数予測モデルの 0) を比べると，シャンテン数予測モデルの精度はが上がったことはわかる．

表 5: テンパイ予測モデルの実験 2 結果

	precision	recall	F 値	support
0	0.9364	0.9811	0.9582	83912
1	0.6953	0.3935	0.5026	9214
accuracy			0.9229	93126

表 6: シャンテン数予測モデルの実験 2 結果

	precision	recall	F 値	support
0	0.6792	0.4795	0.5621	9214
1	0.5424	0.4381	0.4847	26759
2	0.7795	0.8866	0.8296	57153
accuracy			0.7174	93126

6.3 実験 3 の結果

表 7 と 表 8 に実験 3 の結果を示す．三つのアルゴリズムを試したが，精度を上げることができなかった．もっと複雑なモデルであれば効果が出る可能性があると考えられる．また麻雀は不完全情報ゲームなので，精度の限界はあると考えられる．

7 まとめと今後の課題

本研究は三つの実験によって，モデルの精度を向上を試したが，結果は不十分であった．本研究のモデルに対し，GRU の学習時間は LSTM より少し減少したが精度はほぼ同一であった，最適化アルゴリズムはモデルの精度に影響はなかった．一方でモデルの改善と拡張により，精度が少し上がった．

表 7: テンパイ予測モデルの実験 3 結果

アルゴリズム	正解率	0 の F 値	1 の F 値
sgd	0.9172	0.9546	0.5218
adagrad	0.9159	0.9539	0.5215
adam 1	0.9157	0.9538	0.5149
adam 2	0.9163	0.9541	0.5218

表 8: シャンテン数予測モデルの実験 3 結果

アルゴリズム	精度	0 の F 値	1 の F 値	2 の F 値
sgd	0.7173	0.5619	0.4906	0.8293
adagrad	0.7105	0.5649	0.4811	0.8246
adam	0.7131	0.5599	0.4784	0.8271

今後の課題として，Transformer のモデルを使って，麻雀 AI のモデルを構築することが考えられる．

参考文献

[1] 青野義樹. 機械学習を用いた麻雀戦術における状況予測手法の提案. 大阪府立大学卒業論文, 2018.

[2] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, Vol. 9, No. 8, pp. 1735–1780, 1997.

[3] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014.

[4] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, Vol. 323, pp. 533–536, 1986.

[5] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, Vol. 12, pp. 2121–2159, 2011.

[6] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.