2M-05

# Improved Data Structure and Deep Convolutional Network Design for Haifu Data Learning in the Game of Mahjong

Shiqi Gao† Fuminori Okuya† Naoki Mizukami‡ Yoshihiro Kawahara† Yoshimasa Tsuruoka†

Graduate School of Information Science and Technology, The University of Tokyo†
Graduate School of Engineering, The University of Tokyo‡

## 1 Introduction

The research of artificial intelligence (AI) on perfect information games has been very much advanced and already reached the top level that no human beings can run neck and neck even in the game of Go by the AI AlphaGo series. However for imperfect information ones, people still have a long way to go. It is still very difficult for AI to adopt the best strategy when facing imperfect situations. Mahjong is the most classical imperfect game example for its huge social popularity (more than 800 million) and its strong complexity (about 140 tiles in all with four players). Therefore, we choose Japanese Mahjong as an research tool for its standard rules and huge past game record data which is called 'haifu'. The accordance rate of haifu data is recognized as a benchmark for estimating the machine's learning ability. Traditional mahjong learning methods were mainly by human artificially extracting features and designing function blocks. Although there has been related researches in deep learning [1] and CNN [2] these two years, they still cannot exceed traditional methods' result due to their methods' limits. In this paper, we introduce a new data structure in order for containing more information. For the deep neural network structure, we elaborately separate the information gained into different input parts and make the merge after feature extraction and uses several new techniques for training. We show our result reaching the accordance rate of 68.0%, much better than previous deep learning methods, and also exceeds state-of-art traditional result(62.0%) on this task [3].

## 2 Related Work

### 2.1 Basic mahjong rules

Japanese Mahjong has 136 tiles with 34 types and four same tile in each type. The 34 types are 1m(man)-9m, 1s(sou)-9s, 1p(pin)-9p, and 東南西北白発中. Each player has 14 tiles in hand, and the basic winning tile combination is $x$(AAA) + $y$(ABC) + DD while $x + y = 4$, where AAA represents for triplets(three same tiles), ABC represents for sequences(three tiles with sequential numbers) and DD represents for two same tiles. Here are another two important terms:

Riichi
A player can declare riichi showing he is waiting to win, just lacking one last tile. Once declaring riichi, one's tile cannot be changed anymore, but he will own huge chances of winning larger scores.

Dora
Hand patterns and the number of dora tiles in hand are two important factors for a winning a large score. In each game, there will be one or several types of dora tiles according to dora indicators.

### 2.2 Past research

Traditional mahjong AI usually has two function blocks, the offense part and the defense part. For offense part, the tile efficiency is the main thing considered in order for faster winning and larger scores, ignoring other players' actions. For the defense part, how to avoid ron(which means defeated) by others is the task. The training is to get a balance between offense and defense and decide output strategies in different situations [3] [4]. The state-of-art accordance rate accuracy for haifu learning during test is 62% [4].

Deep learning has been hot these several years and is featured for automatic feature extraction ability which needs no artificial extraction. However how to design the data structure and how to build the network is still a difficult task. Tsukiji made a 2-layer full-connected network with 1653-dimension input, receiving a test accuracy rate of 43% [1]. Also in Tsukiji's new article, he designed a 5 by 34 by 5 data structure like an image to contain the tile information, 5 planes for own tiles and discarded tiles, and the 34 by 5 structure for each plane, achieving a test accuracy at 53.98% [2].

## 3 Data Structure Design

| | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 1m | 0 | 0 | 0 | 0 |
| 2m | 1 | 0 | 0 | 0 |
| 3m | 1 | 1 | 1 | 0 |
| 4m | 1 | 0 | 0 | 0 |
| ⋮ | | | | |
| 9m | | | | |
| 1s | 1 | 0 | 0 | 0 |
| ⋮ | | | | |
| 9s | 0 | 0 | 0 | 0 |
| 1p | 0 | 0 | 0 | 0 |
| ⋮ | | | | |
| 9p | 1 | 1 | 1 | 0 |
| 東 | 0 | 0 | 0 | 0 |
| ⋮ | | | | |
| 白 | 0 | 0 | 0 | 0 |
| ⋮ | | | | |
| 中 | 1 | 1 | 1 | 0 |

Figure 1: Data Plane Structure

{gaosq0604, okuya23, kawahara}@akg.t.u-tokyo.ac.jp, {mizukami, tsuruoka}@logos.t.u-tokyo.ac.jp

Different from previous CNN structure, we design the 34 by 4 plane structure shown in Figure.1, therefore 136 dimensions of features for each convolutional plane. We believe data in such kinds of structures is easy for training. The 34 rows representing 34 types of tiles, while the 4 columns representing the existing number of each tile type in given situation. We adopt 54 planes for representing information which is shown in Table. 1. For current situation, we use 13 planes to represent, including 1 plane for representing present tiles in hand with 1 plane marking aka dora five tiles behind, 4 planes for 4 players' discarded tiles, 4 planes for 4 players' naki tiles, 1 plane for dora indicators, 1 plane for round wind and 1 plane for own wind. Besides, past actions and situations also have great influence on present choice making. After experiments, we include last 4 rounds' information of hand tiles, discarded tiles and naki tiles. For latest action, we contain one more dora plane.

Table 1: Input features for neural networks

| Feature | # of planes |
|---|---|
| Own hand tiles | 2 |
| Discarded tiles | 4 |
| Naki tiles | 4 |
| Dora indicators | 1 |
| Round wind | 1 |
| Own wind | 1 |
| Past 1 situation | 11 |
| Past 2 situation | 10 |
| Past 3 situation | 10 |
| Past 4 situation | 10 |

# 4 Neural Network Structure Design

We propose our filter design for the network. Instead of using several different filters in order for capturing different features, we aim to use several layers of small filters to cover the whole layer. However our data structure is too narrow with only 4 in width, hard for gaining very deep layers like other convolutional networks, therefore we adopt 3 convolutional layers with 6 by 2 in filter size and 144 in filter height. We put 2 full-connected layers after convolutional ones, with 500 neurons in the front one and 100 neurons in the behind one which is combined with an 14-dimension array which represents information about ranking, score, existing riichi stick number and honba information. We use relu as the activation function, Adam as the optimizer method with an initial learning rate at 0.001, with a batch normalization layer and a dropout layer at 0.5 after each layer.

# 5 Experimental and Simulation Results

## 5.1 Game records for training

Our neural network is trained on NVIDIA Tesla K10 GPU with 32GB memory size. We solve this tile prediction problem as a multi-classification problem with 34-class softmax layer. We adopt the supervised machine learning method for prediction model and use game records from the 'Houou' table at the online mahjong site 'Tenhou' in the year of 2015 as the training data. The 'Houou' table is only open for the top 0.1% mahjong players so the game records can be considered as good quality. During each game, we just follow one player' s game record, and make sampling before that player' s riichi declaration since once riichied, the player' s tiles cannot be changed.

## 5.2 Network training and results

We sampled 600000 situations in all for network training, with 10% for validation set. The final validation test accuracy arrives around 68.5%. For authority, we randomly select 10000 rounds of situations picking from the data of another year in order to eliminate correlation among data, and achieves an accordance rate of over 68.0%, exceeding the state-of-art 62.0% accuracy result [4].

## 5.3 Two interesting findings

Pooling is recognized as a very efficient and useful downsampling tool for convolutional network training. However during our experiment, we find the pooling layer not only not contributing to the accuracy but also making demerits. For analysis, we think the image recognition problem usually has the movement and rotation invariance, that's why pooling can make parameters fewer and the network more robust. However for our task, the data plane structure is very elaborate and concise, so the space invariance cannot work well.

From the test cases, we also find this convolution structure perfectly understands the information of in hand tiles and what are the dora tiles seeing from the indicators. We find the output of the network from test cases are all among the legal tile actions in hand. Besides, according to the mahjong rule, dora tile and dora indicator does not have simple linear relationship. However the network perfectly learns that, and with dora indicator information instead of pure dora information, the accuracy even raises about 1%.

# 6 Conclusion and Future Work

In this paper, we proposed a strong haifu learning method for game of mahjong with elaborately designed data and network structure, exceeding state-of-art learning accuracy. Here we simplify mahjong learning into discarded tile learning as a multi-classification problem. Besides, deeper network can be used for training in future in order for higher accuracy such as residual networks. Aka dora tile information might be better represented, together with a more proper way of adding the information such as ranking, score etc. into training.

# Acknowledgement

This work was supported by JST ERATO Grant Number JPMJER1501, Japan.

# Reference

[1] 築地毅, 柴原一友: ディープラーニング麻雀-オートエンコーダとドロップアウトの有効性, ゲームプログラミングワークショップ 2015 論文集, pp.136-142(2015).
[2] 築地毅, 柴原一友: CNN 麻雀-麻雀向け CNN 構成の有効性, ゲームプログラミングワークショップ 2017 論文集, pp.163-170(2017).
[3] Naoki, M. and Yoshimasa, T.: Building a computer Mahjong player based on Monte Carlo simulation and opponent models, IEEE Conference on Computational Intelligence and Games (CIG), pp.275-283(2015).
[4] 水上直紀, 鶴岡慶雅: 期待最終順位に基づくコンピュータ麻雀プレイヤの構築, ゲームプログラミングワークショップ 2015 論文集, pp.179-186(2015).