The Ga

争上游 (ZhengShangYou, or “Con
Chinese card game that is part st
player is dealt ~18 cards, get rid c
patterns, and win by getting rid of
aimed to train an RL agent to hav
against humans in a 2-player vers

3 Main Challe

Battle Royale Exploratio

Leon Lin (leonl@stan

me

ıpetition Upstream") is a rategy, part luck. Each of cards by matching all their cards first. I e an above 50% win rate sion of this game.

enges

Round Robin to Ba

Battle Royale Algorithm

Init buffer with 100,000 random games
Init A with 24 agents, 12 with duplicate models
for e in NumEpochs **do**
  $\epsilon = \frac{1}{e+2} * \sqrt{\frac{|A_0|}{|A|}}$
  Kill_Threshold $= 0.5 * (1 - .75^{e * \frac{|A|}{|A_0|}})$
TRAIN
  for i in range(500) **do**
    sample = random 4096 points from buffer
    for Agent ag in A **do**
      Train Agent ag sample
  for i in range($|A|$) **do**

The PvP n different r other, and training cc

Two initial the param

1) To det

n for Card Game ZSY

ford.edu)

attle Royale

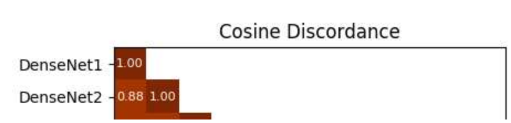
ature of the game allows for
nodels to be tested against each
gradually selected/reduced as
ontinues.

experiments were run to determine
neters for the final algorithm:

ermine the size of the buffer, 5

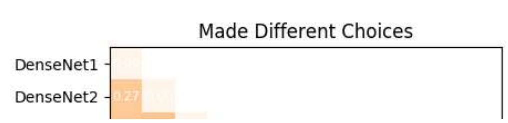
Discordance

The ConvNet Agents all had
different network structures
Using the models from the
round robin was run: all 12
action to find the average c
across and percentage of ti



and Aggregation

similar performances despite
s. Did they learn the same policy?
Round Robin experiment, one final
models evaluated each state and
osine similarities for their Q values
mes they chose different actions



• *Stochastic* + *large discrete state*
each dealt 18 cards, there are ~1
• *Partially Observed*: A player only
• *Train vs Test*: It cannot be traine
data required, but it will be testec

I have previously attempted this b
40% win rate against a human pla

The Data

Balancing the need to preserve the complexity of the game with the need to save memory, all ‘hands’ of cards are represented by stacks of one-hot encodings of how many there are of each

1x15 array

space: With 2 players
.51 trillion initial states
see their own cards.
d against humans for the
l against them

ut only achieved about a
ayer (me).

A hand of cards

```
1, 2, 0, 0, 0, 2, 2, 2, 1, 2, 2, 1, 1, 0]], dtype=int8)
```

representing the counts of each value of card

```
andToExpanded(h)
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
, 1, 0, 0, 0, 1, 1, 1, 0, 1, 1, 0, 0, 0],
, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0],
```

SIMULATE

for j in range(i, $|A|$) **do**

Simulate $50000/\frac{|A|*|A|+1}{2}$ games between A_i and A_j

Add games to buffer

TEST

for Agent ag in A **do**

Update ag's "VS" winrate

Test Agent ag for 200 games against the 3 static agents

Update Agent ag's winrate against the 3 static agents

KILL

if ag's VS winrate < Kill_Threshold **then**

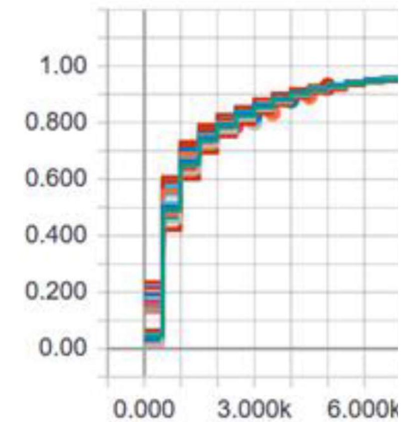
Remove ag from A

agents we
random ga
to see whe
set the bu
start with

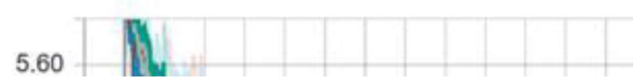
2) To figu
much to e
algorithm

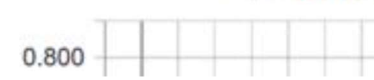
From this, I decided the exploration an

Then, the models from the second exp
with 2 instances), and the Battle Royal



vs Rand



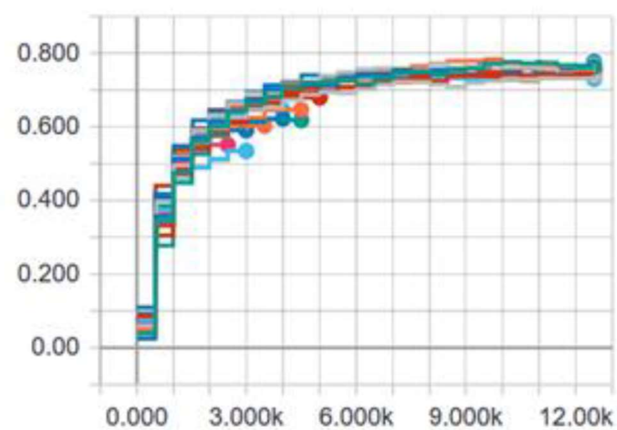


ꞌre trained on a large amount of
ames, and tested at various intervals
en winrates stopped increasing. This
ffer size to max out at 250k and
100k

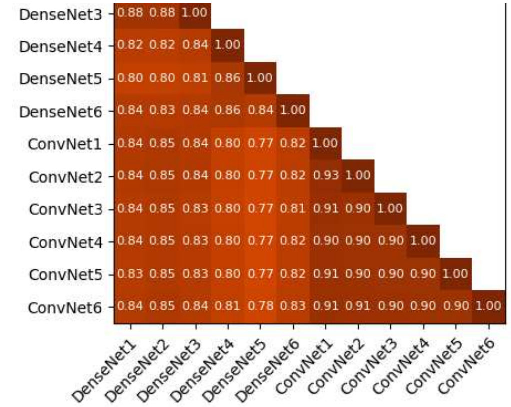
ure out when to kill models and how
xplore, a shorter version of the
with 12 agents and no removal.
d kill threshold formulae.

eriment were doubled (each one
e was run.

om



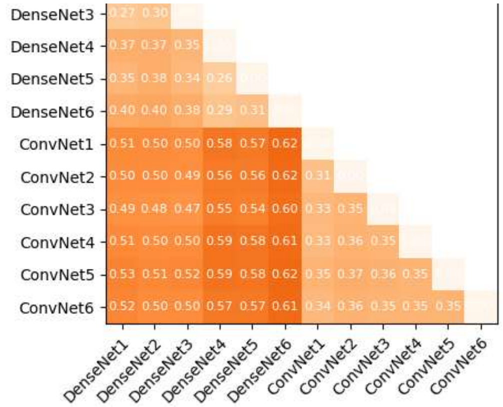
vs Greedy



Despite being trained on th
quite different. This suggest
might improve performance
average, minimum, or maxi
or 9 models. The agg-
regation agents did
significantly better
than any individual
agent.

Hur

I build the game in Unity for
PC and Mac and got these



e same data, their policies seemed
ted that an aggregation model
. 9 kinds were tested: taking the
mum Q value from the top 3, 6, or

Agent vs	Random	Greedy	Old DQN
Min6Combo	97.8%	77.7%	75.4%
ConvNet2	96.6%	72.6%	62.8%
DenseNet9	96.2%	61.7%	24.4%

nan Tests

Player	Ratio	Winrate	σ

5x15 array
whe

card (suit doesn't matter in ZSY). 5x15 arrays represent the player's

hand, action, cards it has played
opponent has played. These 4 ar
5x60 array to represent state-ac

Simulated games are stored in a
by Mnih et al. (2013). Many diffe
initialized together, battle each c
single, collective replay buffer fo
exploration, and hyperparameter

Github Repo

PC build of gam

0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1]], dtype=uint8)

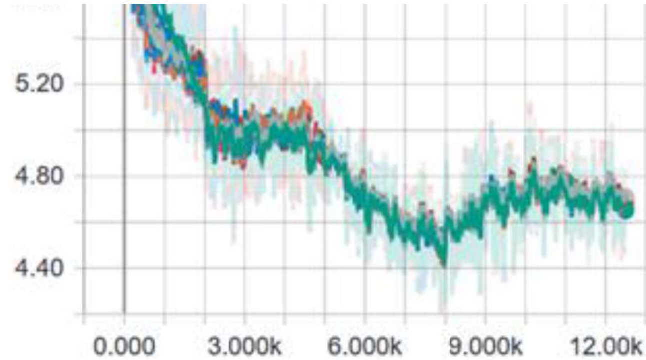
with each column being a one-hot encoding of
ther there are 0, 1, 2, 3, or 4 of a card

, and cards the
re concatenated into a
tion pairs.

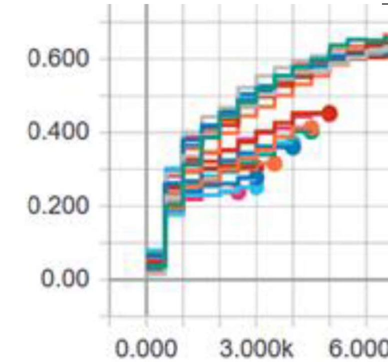
replay buffer inspired
erent models are
other to contribute to a
r data efficiency,
r search.

e

Mac build of game



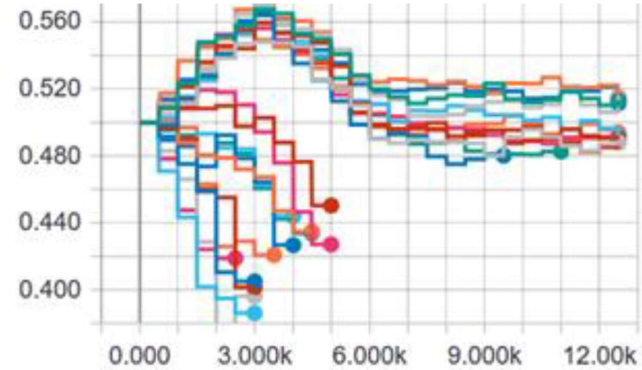
log loss



vs Old D

Each state-action pair is assigned a value
losing games. DenseNet Agents flatten th
4 fully connected layers while ConvNet Ag
of 3x3, 1x3, or 1x1 valid convolutions bef
Activations are all ReLu, LeakyReLu, or Si
the configs.csv files on the Github repo fo
Testing was done against a Random ager
from the previous project.

ConvNet agents performed significantly b
end 10 of the 24 initial agents remained,



QN

vs Each Other

e of 1 for winning games and 0 for
ne 5x60 array and pass it through 3-
gents first pass it through 2-4 layers
fore 2 fully connected layers.
gmoid (for the output layer). See
or full definitions of each agent.
nt, a Greedy agent, and the agent

etter than DenseNet agents.By the
all of which were CNNs.

PC and Mac and got these test results back from human players.

The data is too small to conclusively say that human level performance was achieved, but the initial results are promising: on average, no human beat the agent.

N

With the game built for hun
real human player data and
states where the agent cho
models such as RNNs can
normalization may be helpf

1	62:66	48.4%	4.4%
2	27:32	45.8%	6.5%
3	16:24	40.0%	7.5%
4	12:20	37.5%	8.6%
5	20:36	35.7%	6.4%

n

e

ext Steps

nan players, it is possible to collect
d to better visualize different game
ose badly. Additionally, different
be attempted and gradient
ful for longer term training.