Comparison Training of Shogi Evaluation Functions with Self-Generated Training Positions and Moves

Akira Ura*, Makoto Miwa**,
Yoshimasa Tsuruoka*, Takashi Chikayama*

*Graduate School of Engineering, The University of Tokyo
**School of Computer Science, The University of Manchester

13/Aug/2013

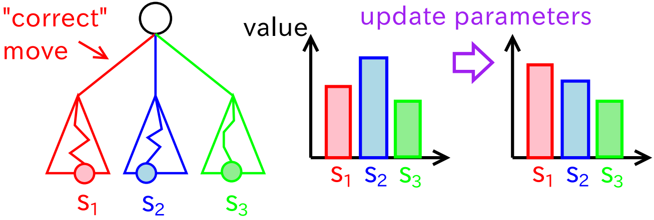
Introduction

- Strong computer game players need accurate evaluation functions.
- The number of parameters of an evaluation function can be very large.
- Automatic tuning of the parameters has been studied.

Comparison Training [Tesauro,2001]

- a supervised learning method applied successfully in Shogi [Hoki+,2011][Kaneko+,2011]
  - Moves played by human experts are regarded as “correct” moves.
  - A large number of game records of experts are needed.

The number of game records of experts is limited!



Using moves played by computer players

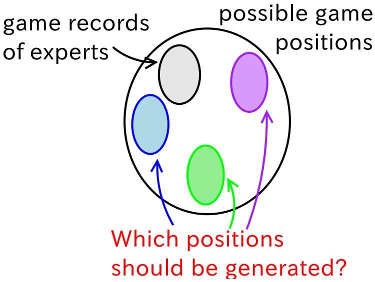
- Moves played by computer players may be used for comparison training.

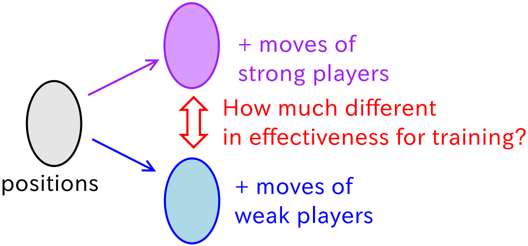
$$\Uparrow$$

- Computer players are already as strong as human experts.
  - Deep Blue defeated The World Chess Champion in 1997.
  - Some Shogi programs defeated professional players in 2013.
- The process of generating moves can be easily parallelized.

Goal

- To improve the strength of a computer player using moves played by the player
  - Which positions are effective for training?
  - How do the strengths of players affect the resulting players?



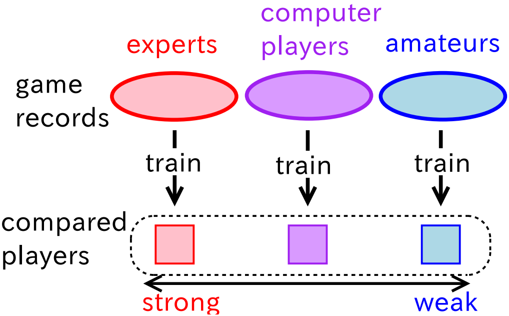


Study on influence of quality of game records [Kaneko,2012]

- Three sets of game records were used for comparison training.

$\rightarrow$ Game records of experts were the most effective.

$\rightarrow$ Game records of computer players were more effective than those of amateurs.



Study on using search results of computer players

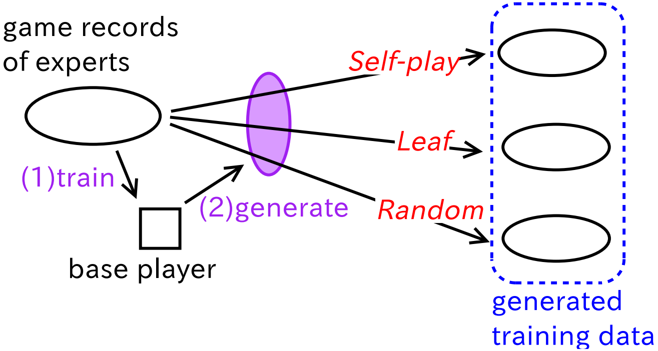
- To assign scores [Lee+,1988][Buro,1999]
- Reinforcement learning [Beal+,2001][Veness+,2009]
- Evolutionary computation [Fogel+,2004][Bǒsković+,2010]

$$\Downarrow$$

These studies were not for comparison training.

Our method

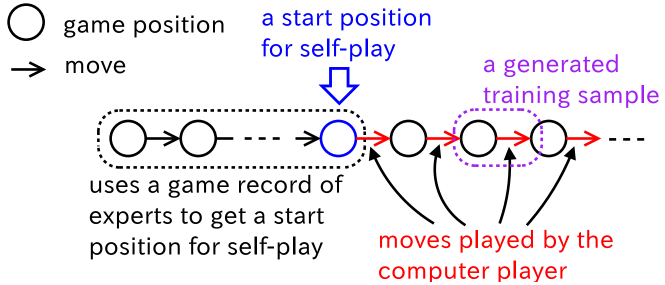
- We compare three sets of self-generated data as additional training data.



Self-play data

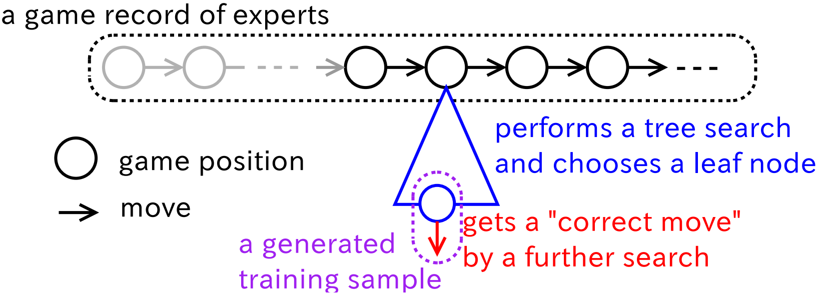
- Positions that appear in self-play of a computer player are used as the training positions.

→ These positions are expected to appear also in actual games.



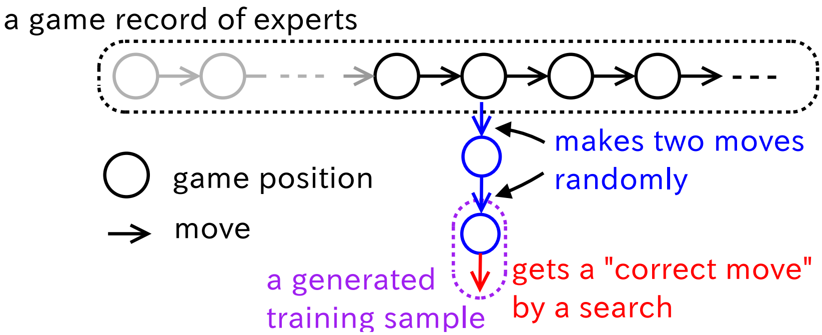
Leaf data

- Leaf positions in search trees are used as the training positions.
- Training positions should share the same characteristics as positions evaluated in actual game-tree searches.
  - The idea has been previously proposed by Buro (1999).



Random data

- Positions created by playing two legal moves randomly.
- These positions may not have characteristics similar to positions appearing in actual games.

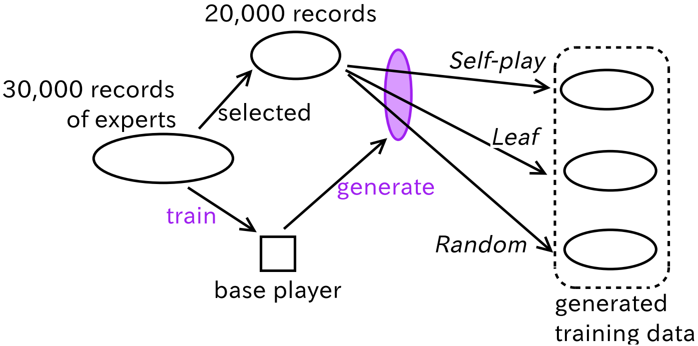


Evaluation

- Performance evaluation
  → Using both game records of experts and self-generated training data
- Analyses on effects of different training data
  → Using only one training data set at a time

Experimental settings

- We used a strong Shogi program, Gekisashi.
- We used 30,000 records of experts for training.
- We calculated winning rates from 7,000 matches.
  - The number of nodes searched was 300,000.



Performance evaluation

We evaluated the strengths of players using both the game records of experts and the generated data.

- The number of nodes searched for a "correct" move was 30,000,000.
  - about 120 seconds on Xeon E5530 (2.40GHz)
- We used two opponent players
  - a player trained with 30,000 records of experts
  - a player trained with 40,000 records of experts

Results

The *Leaf* data is the most effective.

additional training data	vs 30,000 records (%)	vs 40,000 records (%)
10,000 game records (40,000 records in total)	***52.76	50.00
Self-play	49.92	***47.12
Leaf	*51.51	48.85
Random	49.04	**48.33
Self-play + *Leaf*	*51.27	48.91
Self-play + *Random*	50.56	48.89
Leaf + *Random*	50.48	50.16
Self-play + *Leaf* + *Random*	50.50	50.29

* for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$

When more leaf data is used

Increase in *Leaf* data did not lead to any further improvement.

additional training data	vs 30,000 records (%)	vs 40,000 records (%)
Leaf from 20,000 game records	51.51	48.85
Leaf from 30,000 game records	51.52	49.68
Leaf from 40,000 game records	49.79	49.31
Leaf from 50,000 game records	50.49	49.01

More than one leaf nodes were selected from a position in some records because we prepared only 30,000 game records of experts.

Analyses on effects of different training data

We aimed to identify potential underlying causes of the results by comparing training data sets.

- Quality of moves
  - How do the strength of players affect trained players?
- Situations where positions were generated
  - Is there any difference of influence on resulting trained players between *Self-play*, *Leaf*, and *Random*?

Quality of moves (human strength)

- We extracted records of high-rated experts and those of low-rated experts.
- Each data set included 1,200,000 positions.

$$\Downarrow$$

training data	vs *all experts* (%)
high-rated experts	50.91
low-rated experts	*48.42

* for $p < 0.05$

Records of strong experts were more effective.

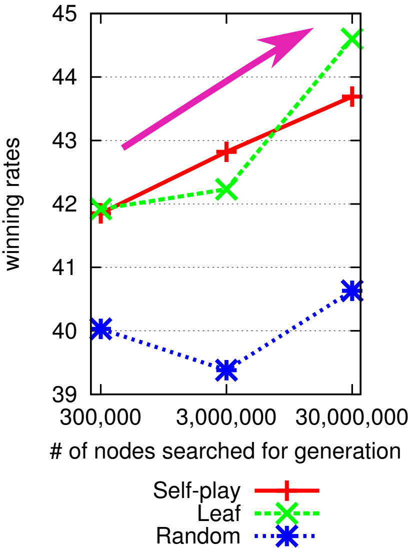
Using self-generated data

- The number of nodes searched for a "correct" move was 300,000, 3,000,000, or 30,000,000.
- Only one set of training data was used for training.
  - 1,200,000 positions in each set of training data
- The opponent player was a player trained with game records of experts.

Quality of moves (search time)

- Moves generated by deep searches led to strong resulting players.
- Game records of experts were significantly more effective than self-generated data.

⇒ Players searching 30,000,000 nodes (120 seconds) may be not so strong as human experts.



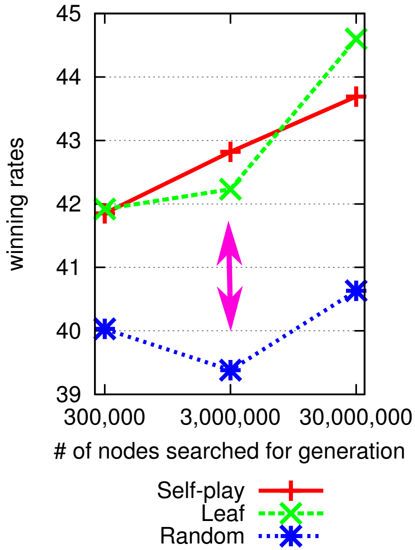
Comparison of situations

- *Random* was less effective than the others.
- *Self-play* and *Leaf* were equally useful.

$\Updownarrow$

- *Leaf* was more useful as additional training data.

$\Rightarrow$ *Leaf* may have a positive influence because positions are different from records of experts.



Effect of self-generation

- We extracted game records of high-rated computer shogi players except Gekisashi from floodgate (shogi server).
- We used only the moves that expended more than five seconds (15 seconds on average).

training data	vs *Experts* (%)
floodgate	44.51
Self-play (120 seconds)	43.69

Self-generation may have a bad effect.

Summary

- We proposed an approach to generating training data by deep searches of a computer player.
- Leaf positions in search trees were useful when they were used as additional training data.
- Experts' moves were significantly more effective than the computer player's moves.
  - $\rightarrow$ The search time may have not been long enough.

Future work

- To explore more effective position selection methods.
  - Selecting leaf nodes important to decide principal variations
  - Introducing win/loss information
- To use other programs along with Gekisashi.
  - Self-generation may have a bad effect.