Multi-Labelled Value Networks for Computer Go

Ti-Rong Wu , I-Chen Wu , Senior Member, IEEE, Guan-Wun Chen , Ting-han Wei , Tung-Yi Lai , Hung-Chun Wu , Li-Cheng Lan

Submitted to IEEE TCIAIG on May 30, 2017

組員: 312551114 朱立民, 312551061 陳盈圖, 312553024 江尚軒

Outline

- Introduction
- Architecture
- Experiment

Introduction

Motivation

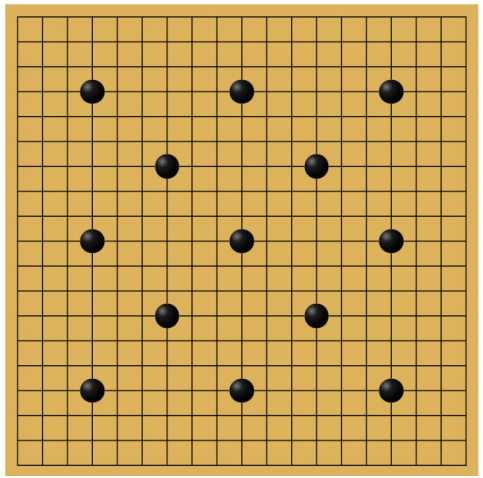
- Komi may have different values in different places.
  - Customarily set to 7.5 according to Chinese rules, and 6.5 in Japanese rules.
- Find useful approach to playing handicap games.
- Dynamic komi strategy while playing game.

Komi

To compensate for the initiative black has by playing first, a certain number of points, the so-called komi, are added for white (the second player), balancing the game.

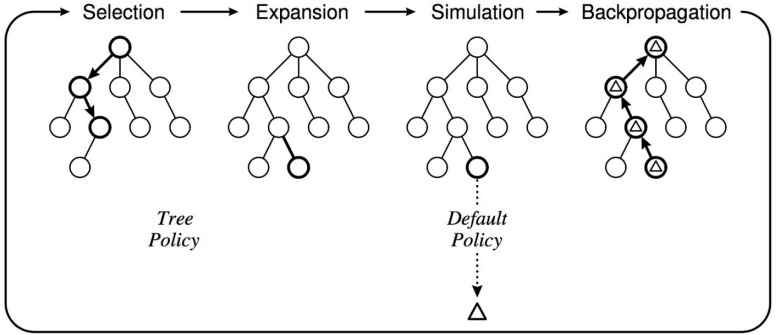
Handicap

It is common for players with different strengths to play h-stone handicap games, where the weaker player, usually designated to play as black, is allowed to place h-stones first with a komi of 0.5 before white makes the first move.



Background

- MCTS:



- DCNNs:
  - Three DCNNs: SL(Supervised Learning), RL(Reinforcement Learning), VN(Value Network)

Background

- Board Evaluation Network:
  - indicating the probability that the point belongs to black by the endgame
  - minimize the MSE between predicted probabilities and the actual endgame ownership of each board point
- Dynamic komi:
  - make the program play more aggressively
  - further strengthen one’s advantage when winning or to catch up and minimize one’s losses when losing
  - Previous method determine komi value
    - score-based situational (SS)
    - value-based situational (VS)

Score-Based Situational

- Adjusts the komi based on $E[score]$, the expected score of rollouts over a specific amount of Monte Carlo simulations.
- The new komi is set to be $k + \alpha E[score]$, where α is komi rate.
- The komi rate is set to be higher near the beginning of a game, and lower near the end.

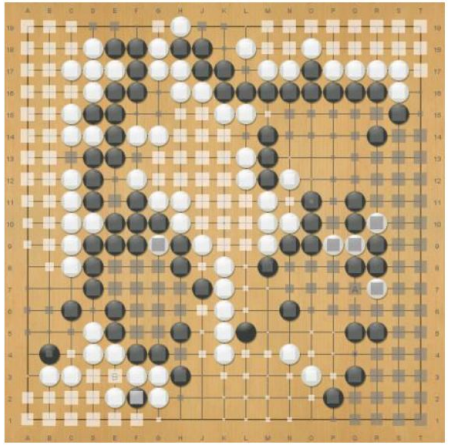
Value-Based Situational

- Adjusts the komi dynamically so that over a specific amount of Monte Carlo simulations, the win rate of the root of the tree, v, falls into a predefined interval (l, u) (e.g. (45%, 50%)), or at least as close to the interval as possible.
- If v is higher than u, increase komi by one, and if lower than l, decrease by one.

Architecture

Architecture

- SL and RL policy networks follow AlphaGo's with some slight modifications
  - Add one more input channel for the ko feature
  - Trained the SL policy network with 3-step prediction
  - RL policy network only went through one round of training
    - No baseline value is used
- Use our RL policy networks to generate 30M self-play games
  - Game ends when the ownerships of all points are determined
    - Using to know the number of winning points (BV)



Architecture

- Multi-labelled value network (ML)
  - AlphaGo’s value network only output the win rate of a position with 7.5 komi
  - The value network includes a set of outputs v□
    - Indicating the value of the position for k-komi games
  - Similar to AlphaGo’s value network which has one more input channel to indicate the player color

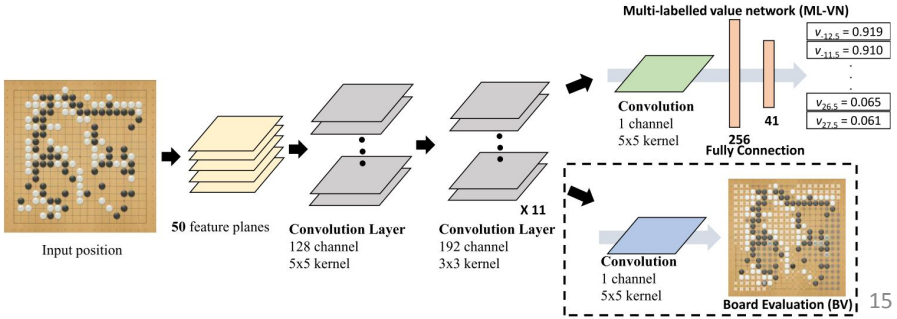
Architecture

- Multi-labelled value network training date

k (komi)	Label on v_k	v_k (win rate)
-3.5	1	0.800943
-2.5	1	0.748812
-1.5	1	0.748309
-0.5	1	0.680036
0.5	1	0.678897
1.5	1	0.599618
2.5	1	0.599108
3.5	-1	0.512413
4.5	-1	0.511263
5.5	-1	0.423886
6.5	-1	0.423626
7.5	-1	0.339738
8.5	-1	0.339353
9.5	-1	0.265258
10.5	-1	0.264586
11.5	-1	0.20581
12.5	-1	0.204716

Architecture

- Board Evaluation(BV) and Multi-Labelled(ML) value network
  - Input is the position on the board
  - There are two outputs
    - ML-VN outputs the multi-labelled value v□
    - BV outputs the ownership of each point for the given position
  - The MCTS batch size is different from the AlphaGo's mini-batch size of 1
    - They use a large batch size of 16 for speeding up performance
  - ML-Based Dynamic Komi (ML-DK)



Architecture

- ML-Based Dynamic Komi (ML-DK)
  - New approach to supporting dynamic komi on value networks by leveraging multi-labelling
  - Mixed win rate for dynamic komi k: $w_k = (1 - \lambda)r_k + \lambda v_k$
    - We want w_k fall into or at least close to the predefine interval (l, u)
      - Let k0 be the game komi
        - if w_{k_0} is in the interval, k0 is still the next dynamic komi
        - if $w_{k_0} > u$, locate the closest komi k such that $w_k \leq u$, the next dynamic komi is set to $k0 + \alpha(k - k0)$ where α is the komi rate

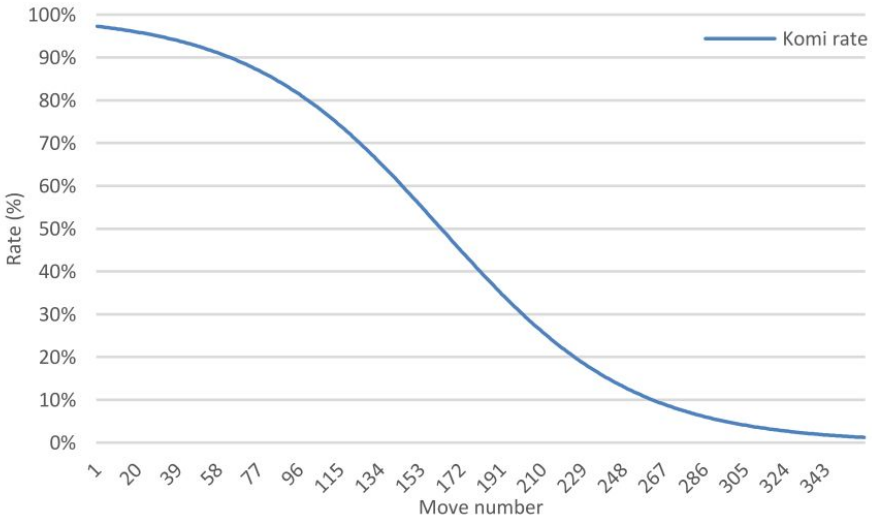
Architecture

- ML-Based Dynamic Komi (ML-DK)

procedure ML-BASED DYNAMIC KOMI

1. *Require:*
2. $\boldsymbol{i}$: *the ordinal number of the current move to play;*
3. $\boldsymbol{w_k}$: *the mixed win rate of the root for all komi k;*
4. $\boldsymbol{k0}$: *the real komi for this game;*
5. $\boldsymbol{B}$: *the total points of the whole board, 361 in 19 × 19 Go games;*
6. $\boldsymbol{c}$ *and* $\boldsymbol{s}$: *parameters to decide different komi rate;*
7. $\boldsymbol{u}$ *and* $\boldsymbol{l}$: *the contending interval (u,l);*
8. **if** $Value < l$ **then**
9. *Locate a komi k such that* $\boldsymbol{w}_k \geq l$ *and* $\boldsymbol{w}_{k-1} < l$.
10. **else if** $Value > u$ **then**
11. *Locate a komi k such that* $\boldsymbol{w}_k \leq u$ *and* $\boldsymbol{w}_{k-1} > u$.
12. **else**
13. *Locate komi k as the* $\boldsymbol{k0}$;
14. **end if**
15. $KomiDiff \leftarrow k - k0$
16. $GamePhase \leftarrow \boldsymbol{i} \,/\, \boldsymbol{B} - s$
17. $KomiRate \leftarrow (1 + exp(c \cdot GamePhase))^{-1}$
18. $DynamicKomi \leftarrow Komi + KomiDiff \cdot KomiRate$

end procedure



Training Flow

- Train SL policy network
- Train RL policy network one round with policy gradient
- Train BV-ML value network with MCTS & RL policy network
  - Use RL policy network to generate 30 million self-play games

Experiments

Experiments environment

- GTX 980Ti GPUs * 4
- Intel Xeon E5-2690s * 2
- 128 GB memory
- Run on Linux

Different Value Networks

- VN: just the value network
- BV-VN: the value network with BV
- ML-VN: the value network with ML
- BV-ML-VN: the value network with BV + ML

Metrics: MSE (Mean-Square Error)

$$\frac{1}{2}\sum_{i=1}^{n}\left(z_i - v(s_{ij})\right)^2$$

Different Value Networks

The reason is that the BV output only provides the probability of each point's ownership at the end of the game. It has nothing to do with the win rates, nor the MSE of the win rates.

Network architecture (abbr.)	Mean squared error (MSE)
VN	0.35388
ML-VN	0.346082
BV-VN	0.35366
BV-ML-VN	0.348138

Table 2. The average MSE of different value networks in 7.5-komi games.

Different Value Networks

Network	VN	ML-VN	BV-VN	BV-ML-VN
VN	-	39.60% (±4.29%)	39.40% (±4.29%)	32.40% (±4.11%)
ML-VN	60.40% (±4.29%)	-	49.20% (±4.39%)	47.20%
BV-VN	66.60% (±4.29%)	50.80% (±4.39%)	-	47.20% (±4.38%)
BV-ML-VN	67.60% (±4.11%)	52.80% (±4.38%)	52.80% (±4.38%)	-

Table 3. Cross-table of win rates with 95% confidence between different value networks in 7.5-komi games.

Multi-labelled Value Network in Different Komi Games

BV-ML-VN can accurately estimate the situation based on the output for $v_{0.5}$.

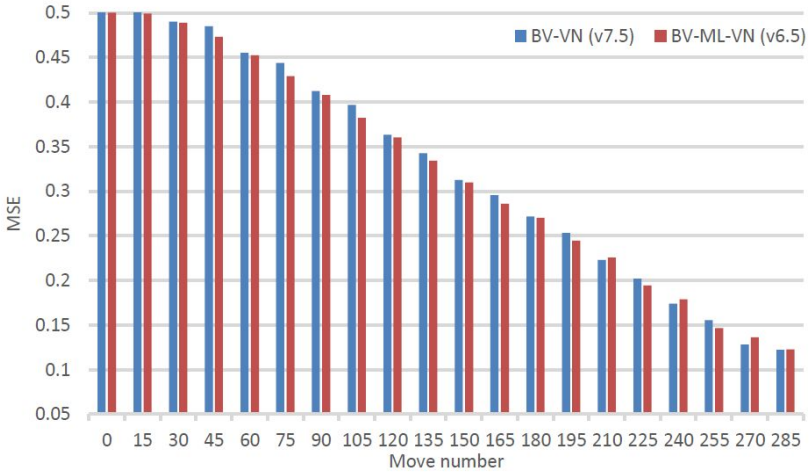


Figure 5. MSE for different value networks in 6.5-komi games.

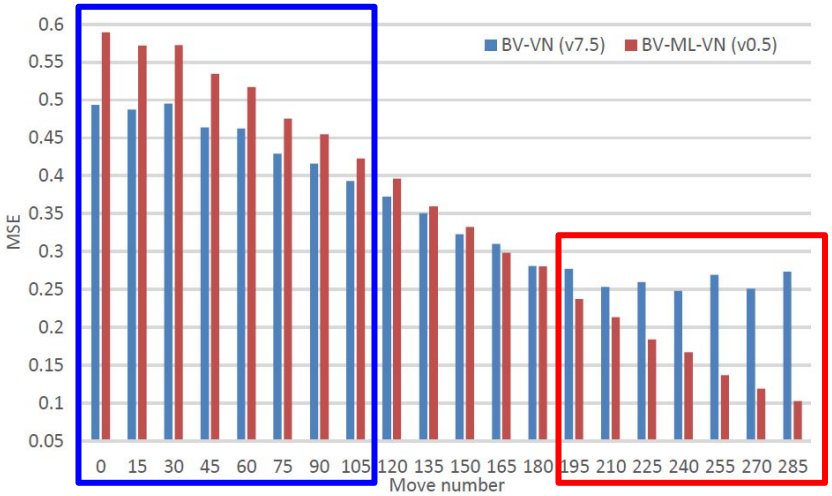


Figure 6. MSE for different value networks in 0.5-komi games.

Dynamic Komi and Handicap Games

Method	Even Game
No Dynamic komi	80.80% (±4.89%)
SS-R	80.00% (±4.97%)
SS-B	78.00% (±5.15%)
SS-M	79.60% (±5.01%)
VS-M	82.00% (±4.77%)
ML-DK	83.20% (±4.64%)

Table 4. Performance of dynamic komi in even Games.

Dynamic Komi and Handicap Games

ML-DK appears to perform slightly better than SS-R.

Method	H1	H2	H3	H4
No dynamic komi	74.00% (±5.45%)	50.00% (±6.21%)	30.40% (±5.71%)	10.80% (±3.86%)
SS-R	76.00% (±5.30%)	58.00% (±6.13%)	38.00% (±6.03%)	22.00% (±5.15%)
SS-B	77.60% (±5.18%)	54.00% (±6.19%)	31.60% (±5.77%)	20.40% (±5.01%)
SS-M	74.40% (±5.42%)	49.20% (±6.21%)	34.00% (±5.88%)	12.80% (±4.15%)
VS-M	71.60% (±5.60%)	46.80% (±6.20%)	30.40% (±5.71%)	9.20% (±3.59%)
ML-DK	80.00% (±4.97%)	57.20% (±6.15%)	41.60% (±6.12%)	18.40% (±4.81%)

Table 5. Performance of dynamic komi in handicap games.

The Correlation between BV and VN

The figure shows that both are correlated in general, as indicated by a regressed line (in black), centered around (7.5, 50%).

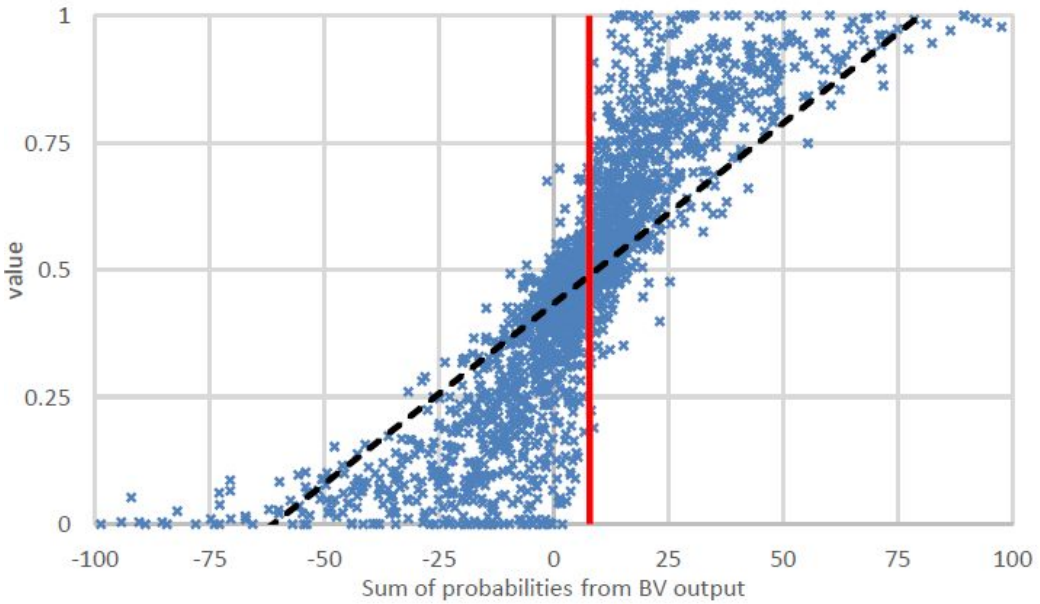


Figure 9. The correlation between BV and VN in 7.5-komi games.

Conclusions

Conclusions

- This paper proposes a new approach for a value network architecture in the game of Go, called a **multi-labelled (ML)** value network.
- **Board evaluation (BV)** is also incorporated into the ML value network to help slightly improve the game-playing strength.
- New **dynamic komi** methods are then designed to support the ML value network, which then improved the game-playing strength

Q & A

Thanks for listening!