Mastering the Game of Go with Deep Neural Networks and Tree Search

Выполнил:
Мищенко В.А.

2016 г.

Авторы

- David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, Nal Kalchbrenner, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, Demis Hassabis – Google DeepMind.
- John Nham, Ilya Sutskever – Google.

New Methods

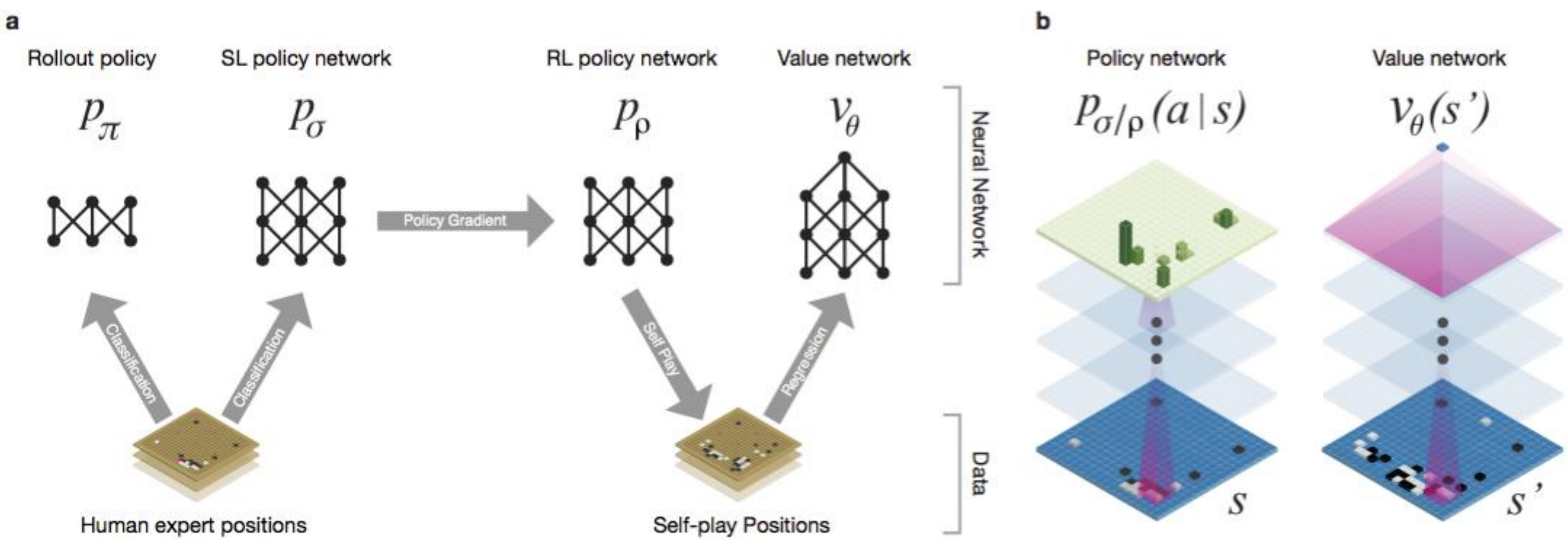
- A new approach to computer Go that uses value networks to evaluate board positions and policy networks to select moves.
- A new search algorithm that combines Monte-Carlo simulation with value and policy networks.

Deep convolutional neural networks

- They employ a similar architecture for the game of Go.
- Monte-Carlo tree search (MCTS) uses Monte-Carlo rollouts to estimate the value of each state in a search tree.
- They use these neural networks to reduce the effective depth and breadth of the search tree:

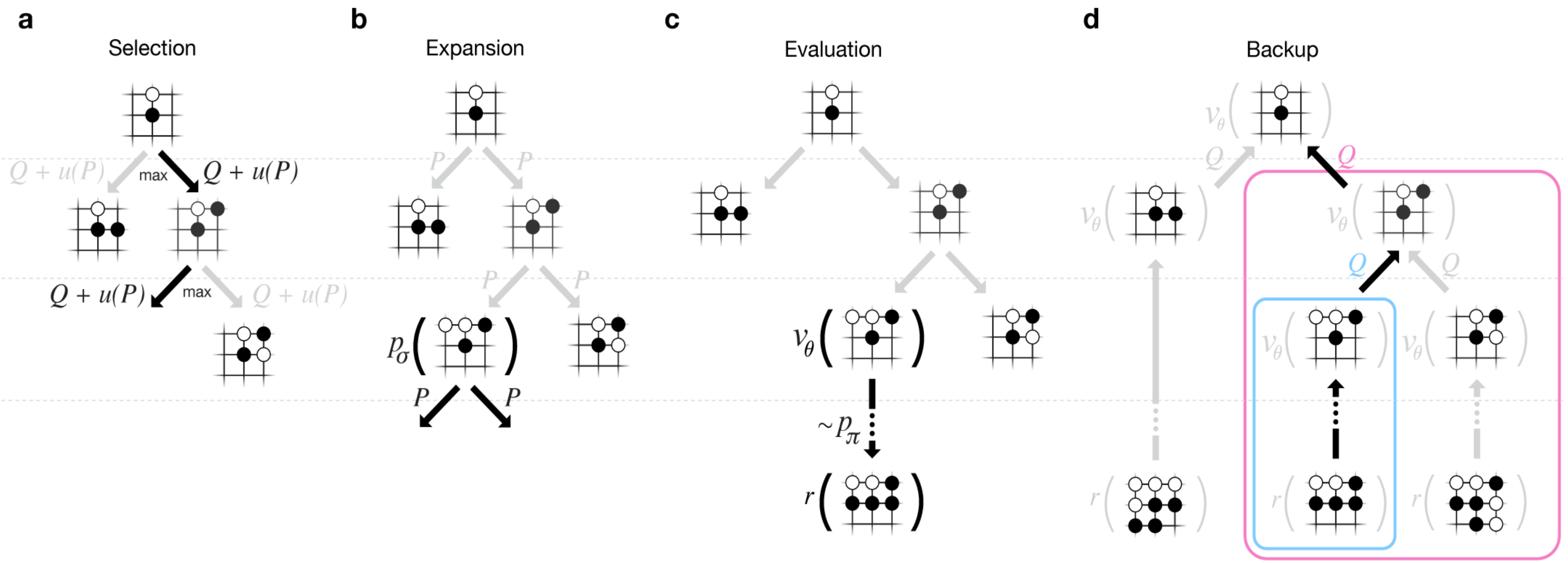
1. evaluating positions using a value network,
2. sampling actions using a policy network.

The neural networks



Searching with Policy and Value Networks

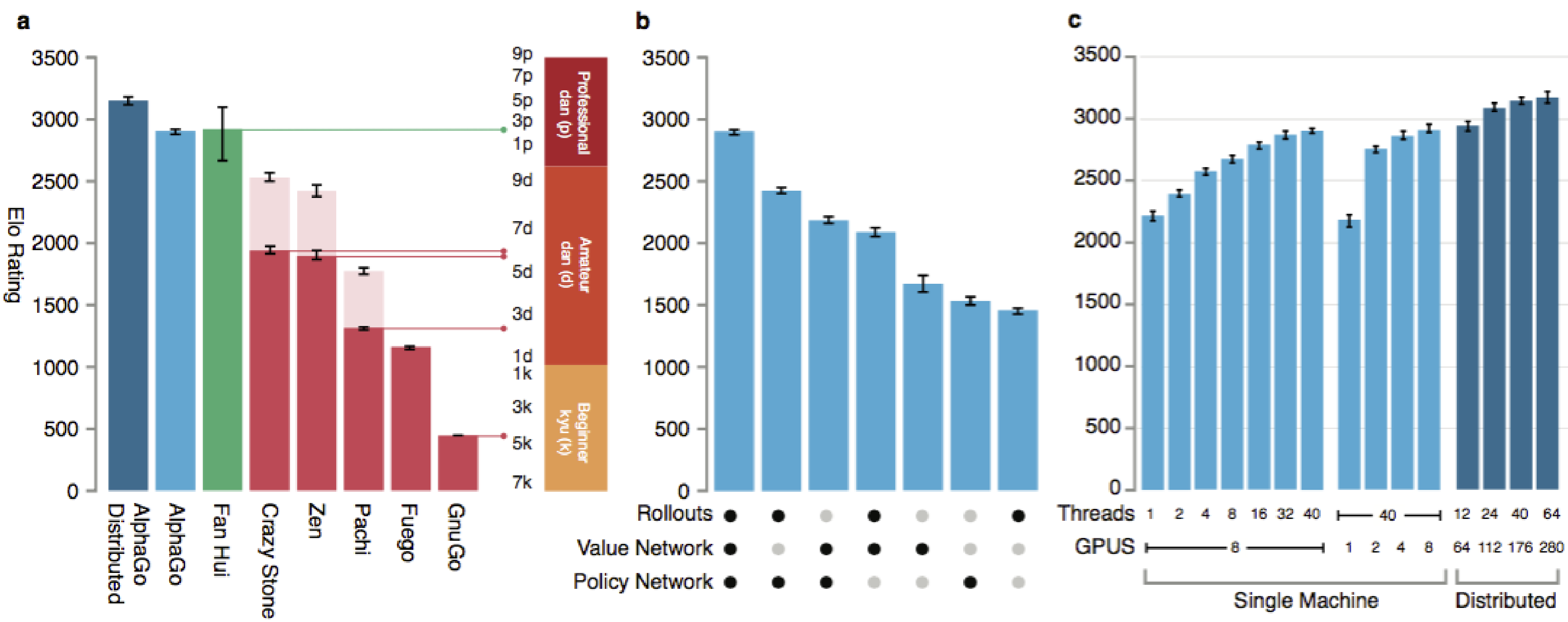
- AlphaGo combines the policy and value networks in an MCTS algorithm that selects actions by lookahead search.



Discussion

- They have developed effective move selection and position evaluation functions for Go, based on deep neural networks that are trained by a novel combination of supervised and reinforcement learning.
- They have introduced a new search algorithm that successfully combines neural network evaluations with Monte-Carlo rollouts
- AlphaGo won the human European champion (5-0)
- AlphaGo is more effective than other Go programs

Evaluating the Playing Strength of AlphaGo

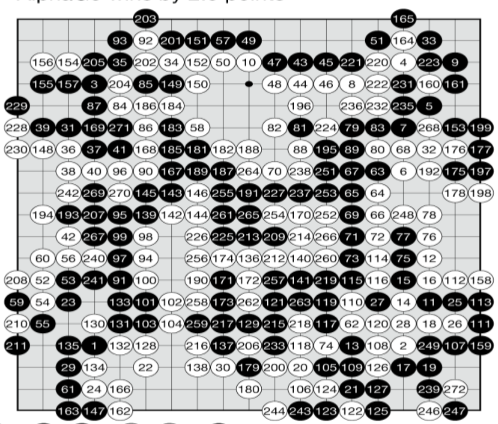


AlphaGo vs Fan Hui

Game 1

Fan Hui (Black), AlphaGo (White)

AlphaGo wins by 2.5 points

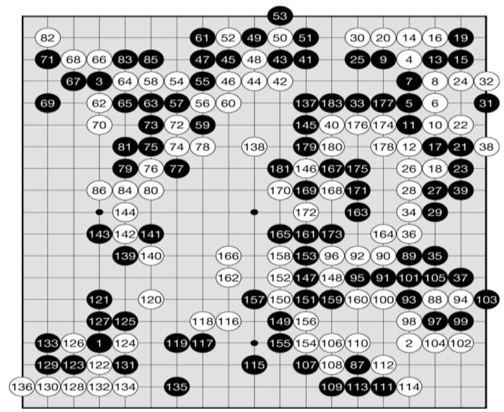


234 at 179 245 at 122 250 at 59

Game 2

AlphaGo (Black), Fan Hui (White)

AlphaGo wins by resignation

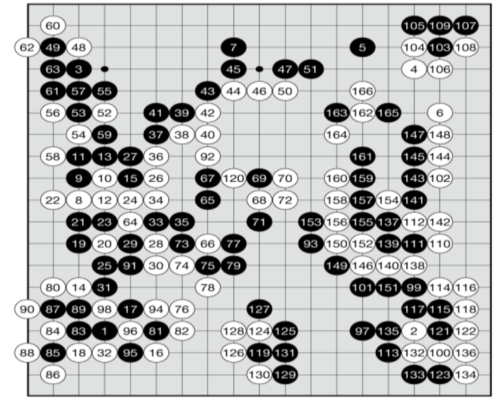


182 at 169

Game 3

Fan Hui (Black), AlphaGo (White)

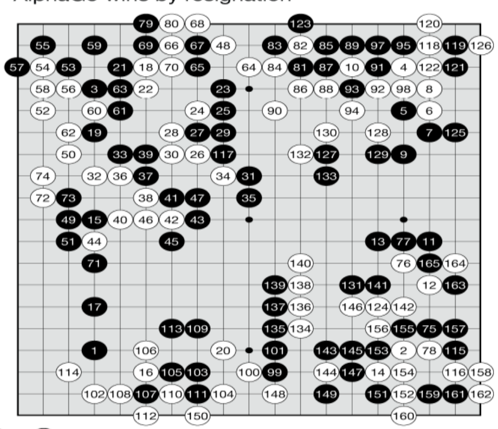
AlphaGo wins by resignation



Game 4

AlphaGo (Black), Fan Hui (White)

AlphaGo wins by resignation

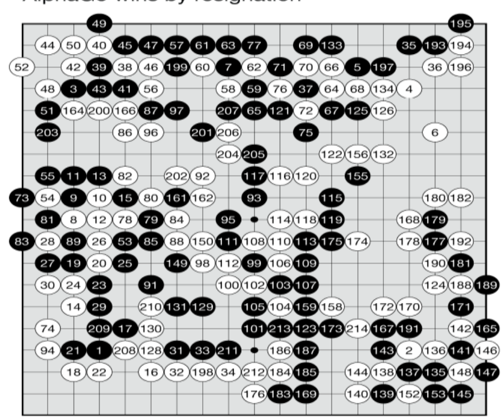


96 at 10

Game 5

Fan Hui (Black), AlphaGo (White)

AlphaGo wins by resignation



90 at 15 127 at 37 151 at 141 154 at 148 157 at 141 160 at 148

163 at 141

- Спасибо за внимание!

Functions at each step

- Supervised Learning of Policy Networks

$$\Delta\sigma \propto \frac{\partial \log p_\sigma(a|s)}{\partial \sigma}$$

- Reinforcement Learning of Policy Networks

$$\Delta\rho \propto \frac{\partial \log p_\rho(a_t|s_t)}{\partial \rho} z_t$$

- Reinforcement Learning of Value Networks

$$v^p(s) = \mathbb{E}\left[z_t \mid s_t = s, a_{t\ldots T} \sim p\right] \qquad \Delta\theta \propto \frac{\partial v_\theta(s)}{\partial \theta}(z - v_\theta(s))$$

Monte-Carlo tree search in AlphaGo.

- Each edge (s; a) of the search tree stores an action value Q(s; a), visit count N(s; a), and prior probability P(s; a).

$$a_t = \operatorname*{argmax}_a \left(Q(s_t, a) + u(s_t, a) \right)$$

bonus $u(s, a) \propto \frac{P(s,a)}{1+N(s,a)}$

$$N(s, a) = \sum_{i=1}^{n} \mathbf{1}(s, a, i)$$

$$V(s_L) = (1 - \lambda) v_\theta(s_L) + \lambda z_L \, .$$

$$Q(s, a) = \frac{1}{N(s, a)} \sum_{i=1}^{n} \mathbf{1}(s, a, i) V(s_L^i)$$