TTIC 31230, Fundamentals of Deep Learning

David McAllester, Autumn 2021

MuZero

Playing Blind

Mastering Atari, Go, chess and shogi by planning with a learned model, Schrittweiser et al., Nature 2020.

In alpha zero we have a state (the board) and procedure for updating the state for each action.

A representation of the board is fed to the value and policy networks.

In mu zero no state representation is provided. The system is only given an action set and rewards for acting.

The Algorithm

The algorithm is very similar to alpha zero but with the addition of a state transition RNN g_Φ satisfying

$$r^k, s^k = g_\Phi(s^{k-1}, a^k)$$

Given the state model we can perform Monte Carlo tree search (MCTS) as in alpha zero.

Alpha zero is modified to allow intermediate reward, to use discounted reward, and to do long roll outs in addition to shallow tree searches.

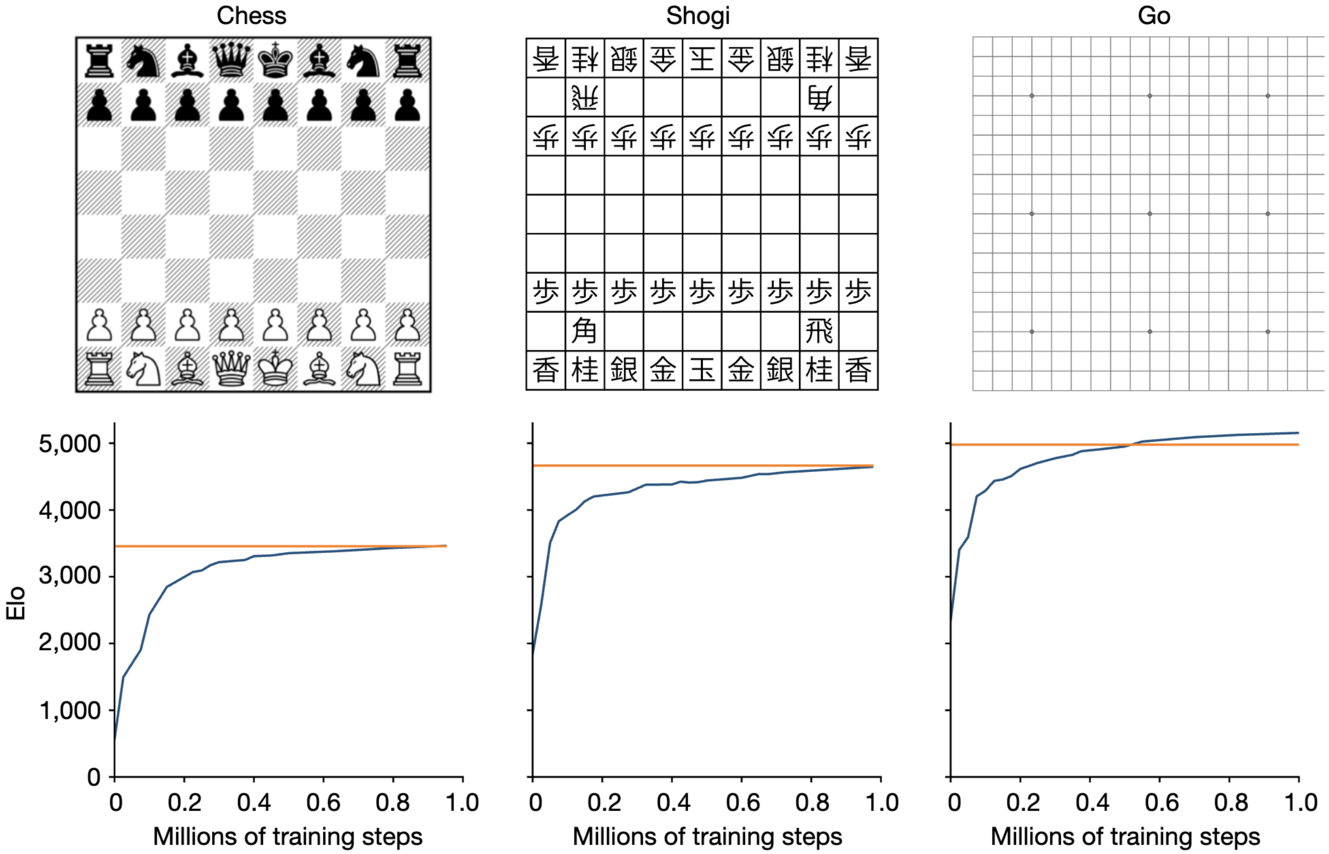
The loss function

$$\Phi^* = \operatorname*{argmin}_{\Phi} \sum_k \mathcal{L}^\pi + \mathcal{L}^V + \mathcal{L}^R + c||\Phi||^2$$

$\mathcal{L}^\pi$ and $\mathcal{L}$ are essentially the loss functions for the policy and value respectively of alpha zero.

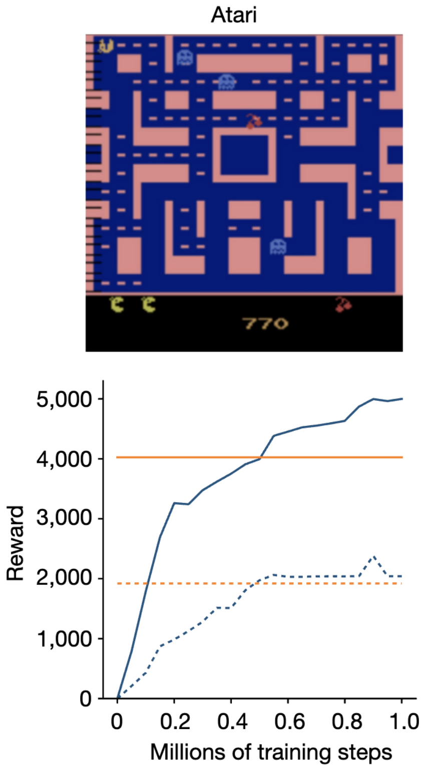
The loss function function $\mathcal{L}^R$ trans the model RNN g_Φ to predict the rewards. Rewards are still provided to μ Zero.

Results



The orange line is alphazero.

Results



These are human normalized scores averaged over all 57 Atari games. The orange line is the previous state of the art system. Solid lines are average scores and dashed lines are median scores.

END