Reproducing and Enhancing MuZero

Madhu Sivaraj and David Tian

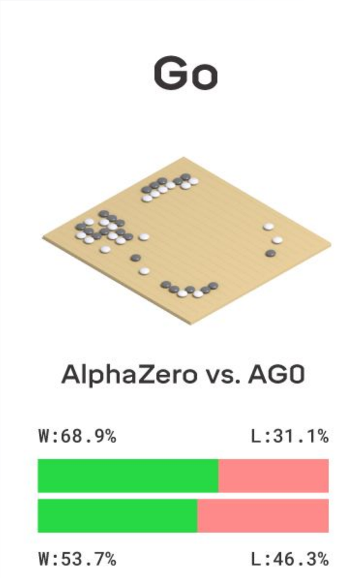
Professor Sungjin Ahn

Introduction

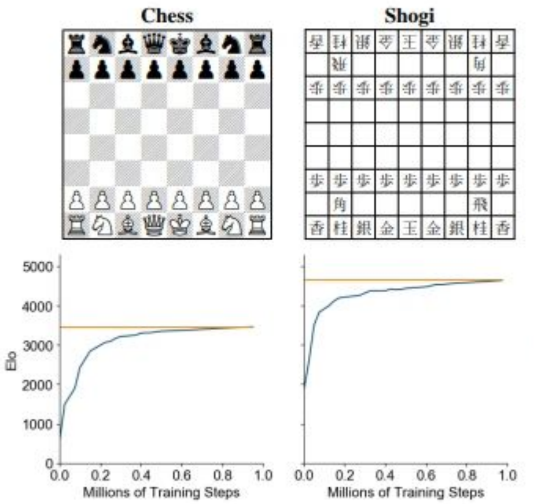
- MuZero is a relatively new reinforcement learning algorithm created by engineers at Deepmind (Schrittwieser et al., 2019)
  - Latest state-of-the-art design in planning algorithms
  - Outperform humans in Go, chess, shogi, and Atari arcade games
  - Combines a tree-based search with a learned model
  - Able to learn to make decisions without any knowledge of underlying dynamics

History

AlphaGo (2014)
Monte Carlo
Tree Search



AlphaZero (2017)
Learns without
human strategy



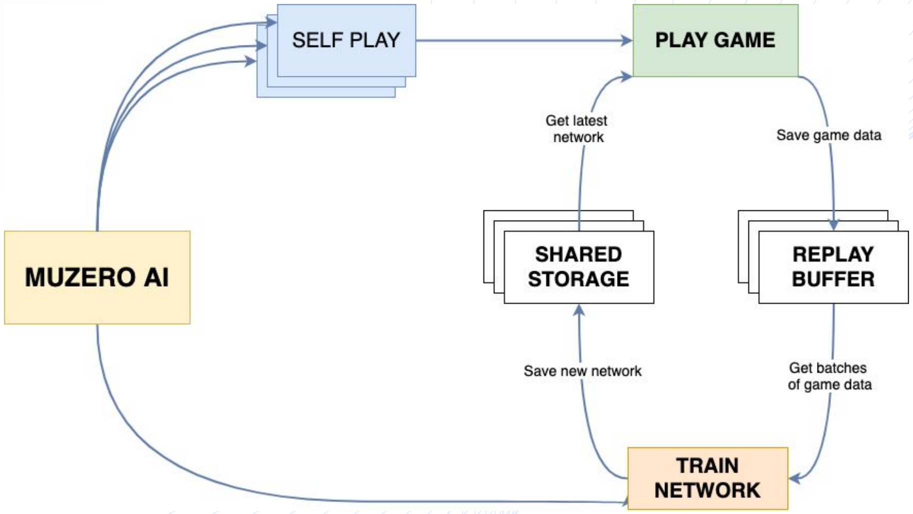
MuZero (2019)
Learns without
game model

Problem Statement

- Improve the current training rates or final results of the MuZero algorithm
- A common problem in reinforcement learning is the balance between exploration and exploitation.
- The challenge here is to build upon the current algorithm in an intuitive way that allows it to make more informed decisions when choosing which actions to explore.

Architecture

- MuZero plays multiple games against itself
- Keeps data from those games and uses it to train three networks
- At every turn of the game, start a new tree from observations (h)
    - Traverse to leaf node with highest Upper Confidence Bound
    - Compute the predicted reward and new hidden state from parent (g)
    - Compute the policy and values for the new hidden state (f)
    - Backpropagate values

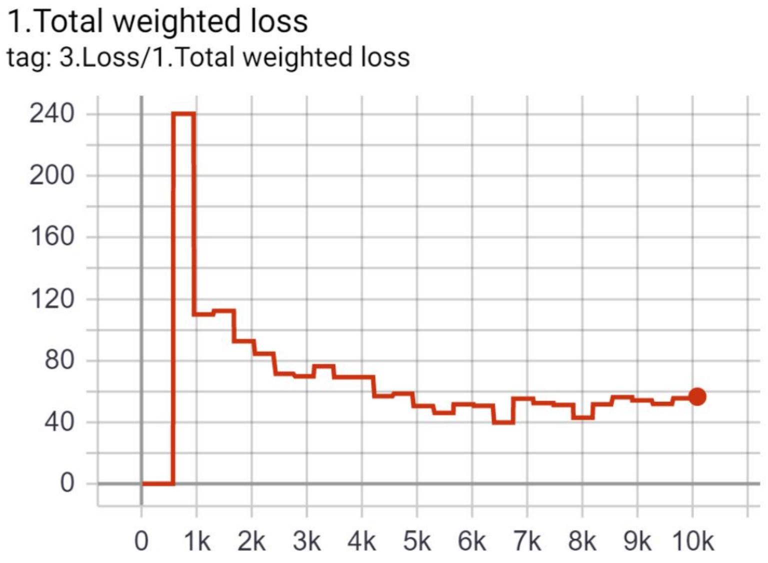


Approach

- Applied simulated annealing, in which the algorithm is allowed to make “mistakes” to discover potentially better strategies.
- Modified the algorithm to select a node based on a probability distribution computed from a softmax of all the UBS scores (instead of always traversing the action tree by selecting the node with the highest UCB)
- Increased the maximum number of moves that the algorithm would simulate to counteract the effect of exploring a wider area of the tree.

Results

Total loss of the original algorithm (left) compared to that of our modified algorithm (right)



Conclusion

Experimented with strategies to help improve the performance of MuZero without much success

- Hyperparameter tuning
- Continuous action space
- Modified MCTS shows somewhat promising results, but could use some more work MuZero is a step ahead of its predecessors, still not quite suitable for real-life scenarios

References

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap: "Mastering Atari, Go, Chess and Shogi by Planning with a Learned Model", 2019; arXiv:1911.08265. David Foster. (2019).