CrazyAra

Deep Learning for Crazyhouse Chess

TECHNISCHE UNIVERSITÄT DARMSTADT

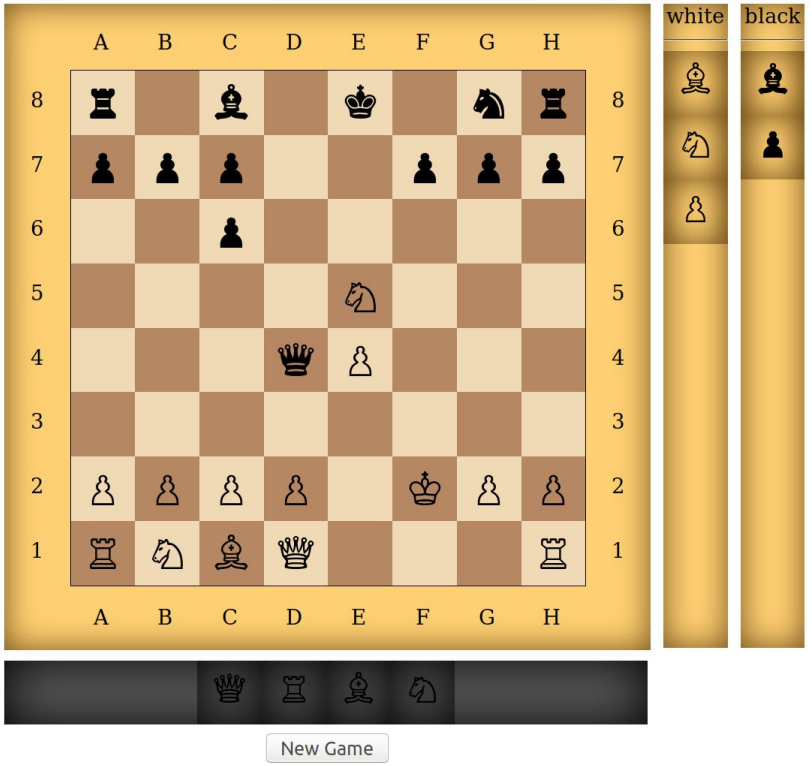
Johannes Czech, Alena Beyer, Moritz Willig

Crazyhouse?! - Rules Overview

Normal chess rules

- Introduction of pockets
- Introduction of piece drops!

- Complex situations
- More focused on tactics
- Higher number of critical situations
- Usually faster games
- Less drawing (< 1%)



Boardstate of a Crazyhouse game

CrazyAra - Milestones

Adapt Alpha(Zero)Chess to Crazyhouse

$$(\boldsymbol{p}, v) = f_\theta(s) \text{ and } l = (z - v)^2 - \boldsymbol{\pi}^{\mathrm{T}} \log \boldsymbol{p} + c\|\theta\|^2$$

- Add pockets & piece drops

Training procedure

- Supervised learning on human games
- Reinforcement learning using self-play

Evaluate performance

- 1,000 test- and 1,000 validation-games (lichess.org)
- Mate-in-one-Benchmark test
- Test against humans

Description:

l: Loss term

s: Game state

p: Predicted policy

π: True policy

v: Predicted value

z: True value

c: Regularization constant

θ: *Weights*

CrazyAra - Milestones

Adapt Alpha(Zero)Chess to Crazyhouse

$$(\boldsymbol{p}, v) = f_\theta(s) \text{ and } l = (z - v)^2 - \boldsymbol{\pi}^{\mathrm{T}} \log \boldsymbol{p} + c\|\theta\|^2$$

- Add pockets & piece drops ✓

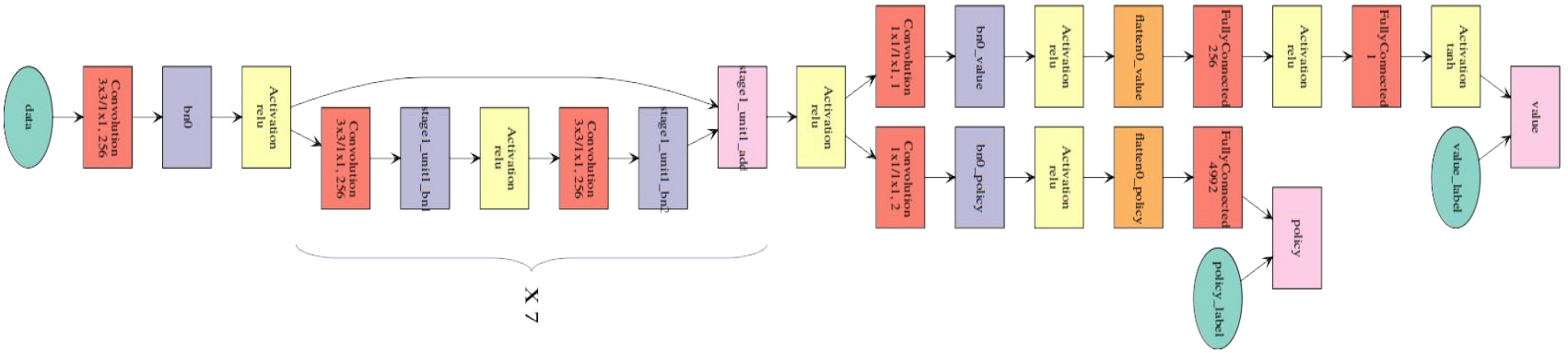
Training procedure

- Supervised learning on human games ✓
- Reinforcement learning using self-play ☐

Evaluate performance

- 1,000 test- and 1,000 validation-games (lichess.org) ✓
- Mate-in-one-Benchmark test ✓
- Test against humans ✓

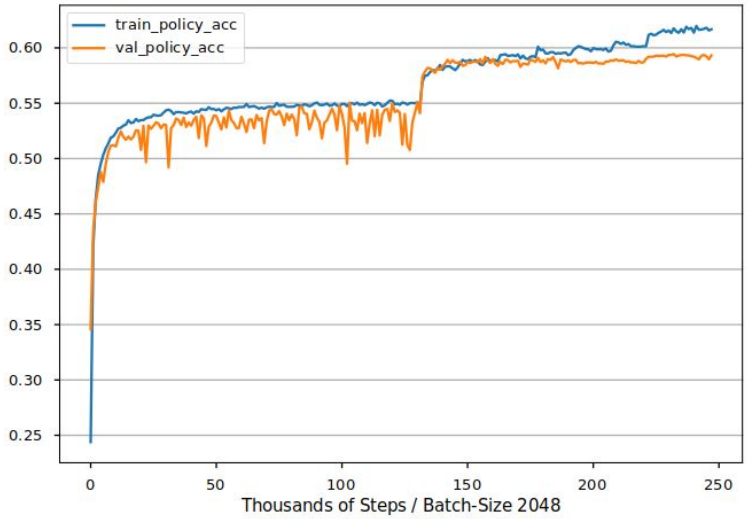
Model-Architecture - Overview



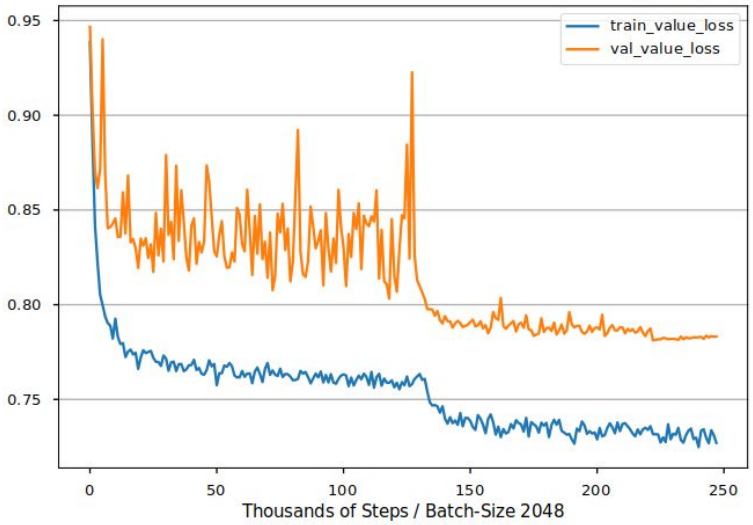
- **Input dimension:** 34x8x8 (no board history!)
- **Output dimension**: [1, 4992] (value, policy)
- **Structure:** Input - 1 CNN-Layer - 7 residual blocks - value head & policy head
- **Reweighting** of the gradients (to avoid overfitting - Mastering the game of Go without human knowledge, David Silver et al.)
  - 0.01 value loss gradient
  - 0.99 policy loss gradient
- **Normalization** of the inputs to the numerical range [0.0f, 1.0f]

Training Results - Summary

Prediction on professional moves (%)



MSE of professional game outcomes



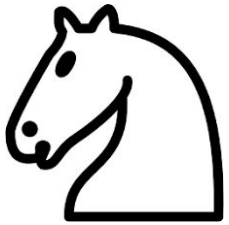
Trained for ~40 hours on 569,537 human games
- All time controls (including Bullet) with elo >= 2,000

Move prediction accuracy
- 61.2% train - 59.3% validation - 58.9% test

Learning rate schedule using NAG (Nesterov Accelerated Gradient)
- Dropping of lr two times - [0.1, 0.01, 0.001]

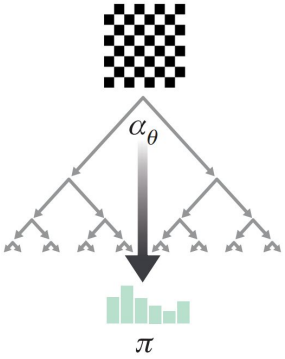
Performance on 1,000 mate-in-one positions
- 93.9% top-1-Acc - 99.7% top-5-Acc:

Achievements - Summary



Integration to **lichess.**org
BOT CrazyAra: https://lichess.org/@/CrazyAra

- Played > 100 games against humans
- Final rating: **2298.44** +/- 79.3.
- Better than 98.2% of Crazyhouse players.
- The source code will be hosted at:
  - https://github.com/QueensGambit/CrazyAra



Integration of **M**onte **C**arlo **T**ree **S**earch

References

- Mastering the game of Go with deep neural networks and tree search, David Silver et al.
- Mastering the game of Go without human knowledge, David Silver et al.
- Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm, David Silver et al.

- https://github.com/careless25/lichess-bot
- https://github.com/Zeta36/chess-alpha-zero
- https://github.com/benediamond/chess-alpha-zero
- https://github.com/glinscott/leela-chess

Live Match with Audience - Volunteer?

Thank you!

https://wallpapercave.com/w/LPXTLA7