

Intuition from Class Activation Mapping using Convolutional Neural Networks

Samuel Huber

Group 13

Problem Statement

How can we identify the best strategy for a given position?



“Tactics is knowing what to do when there is something to do. Strategy is knowing what to do when there is nothing to do.”

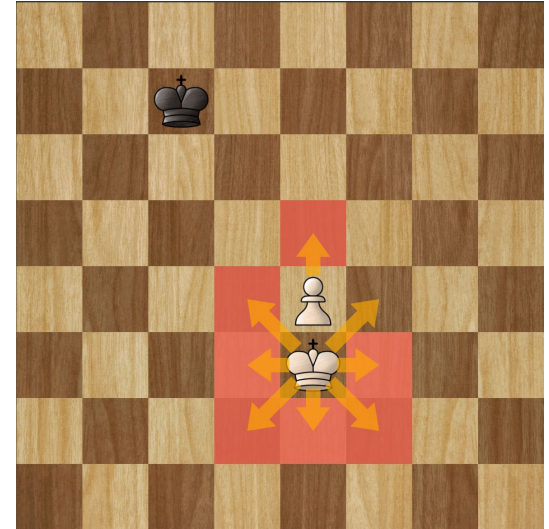
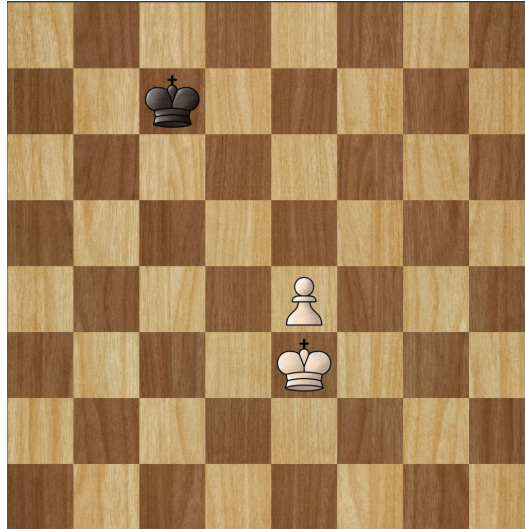
— Savielly Tartakower

Tactical Sequences



Positions Requiring Strategy

More specifically, how can we visually represent the winning idea?



Most Well-Known Computers/Players

Stockfish:

- Minimax Algorithm with Alpha-Beta Pruning using heuristics for evaluation
- Evaluates close to hundreds of millions of positions per second
- Around 3400 ELO

AlphaZero:

- Monte Carlo Tree Search and Neural Network
- Evaluates tens of thousands of positions per second
- 1000 Game Match vs. Stockfish - 155 W / 6 L / 839 D

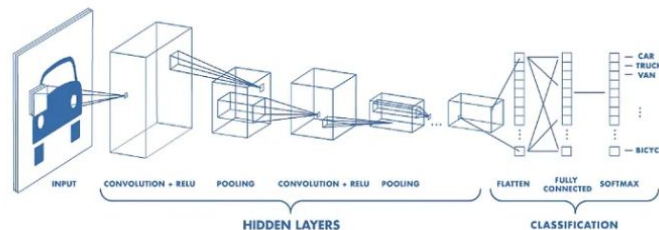
Magnus Carlsen:

- Human Brain (Biological Neural Network)
- Highest Rating Ever - 2882 ELO



Convolutional Neural Networks (CNNs)

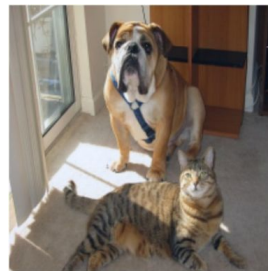
- Mainly used for image classification/object identification
- Perform convolution to extract important features from input
- Train neural network on features



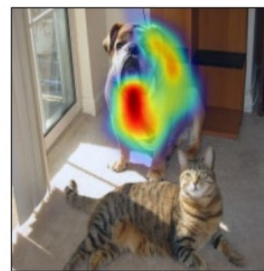
<https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939>

Class Activation Maps (CAMs)

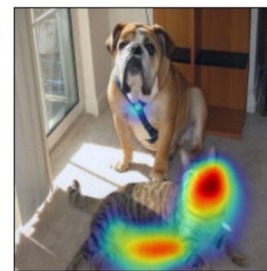
- Run input through trained CNN
- Get the gradient with respect to the model loss
- Scale to image size, color as heatmap, overlay onto original image



Original Image



Identified Dog

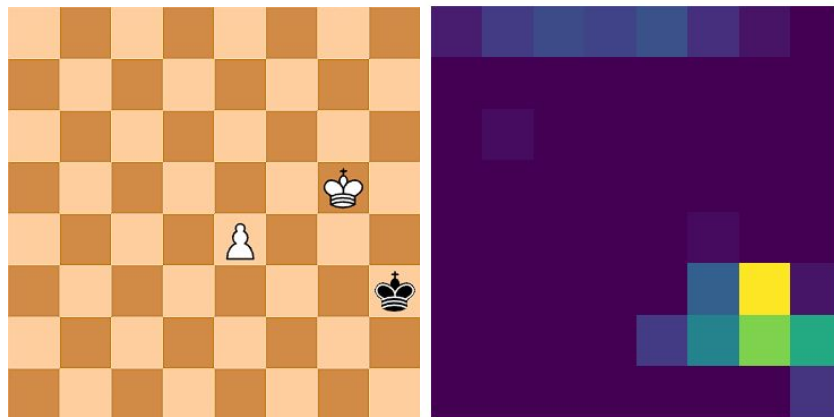
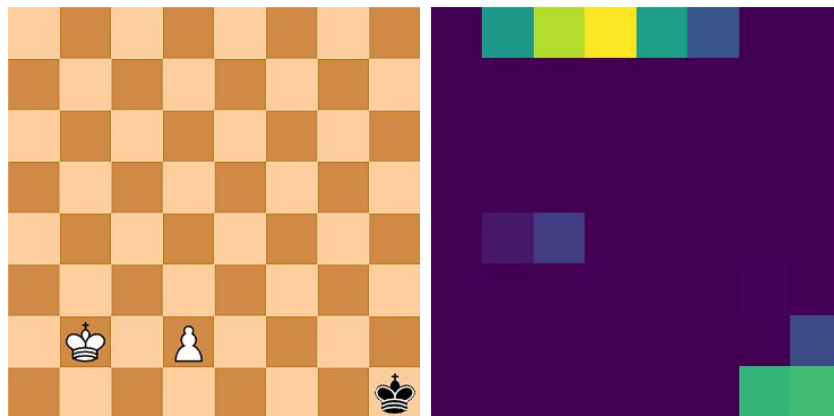


Identified Cat

arXiv:1610.02391

Process and Results

- Generated nearly 2,000,000 K PvK endgame positions
- Obtained scores using Syzygy endgame tablebase
- Trained CNN (95%+ accuracy)
- Insert random position
- Obtain Grad-CAM
- Plot!



Difficulties and Lessons Learned

- Hyperparameter Tuning
- Best Convolution/Hidden Layers for Feature Extraction
- Generating Positions (took more than 9 hours)
- Storage space for tablebase (endgames with 7 pieces requires 18 TB)

Future Work and Other Use-Cases

Future Work:

- Search for optimal hyperparameters
- Expand number of pieces
- Use engine for heuristic evaluation

Other Use-Cases:

- Assisting in learning for other board games (checkers, shogi, go, etc.)
- Help humans develop deeper understanding in geopolitical events or territorial conflict in combat/warfare

Questions?