AlphaGo Zero

Shahzeb Aamir

Content:

1. What is Go?
2. What is AlphaGo?
3. Monte Carlo Tree Search
4. NN Architecture

Go

19x19 grid

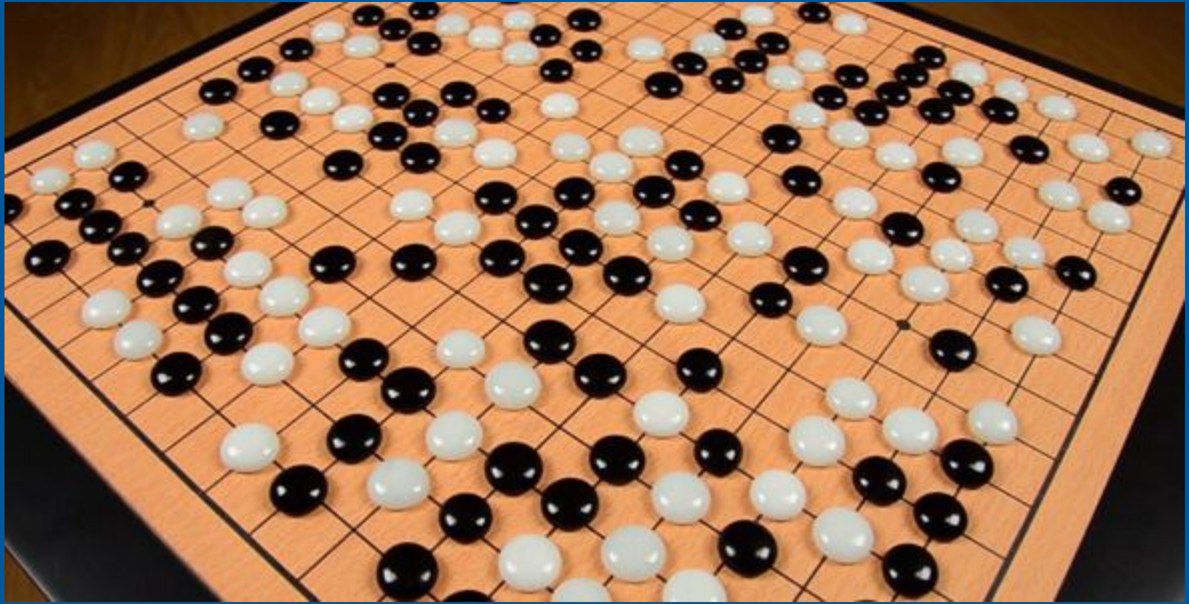
Turn-Based, Two players game

Goal:

Surround and capture opponents

stones, or strategically create

spaces of territory.



Go

19x19 grid

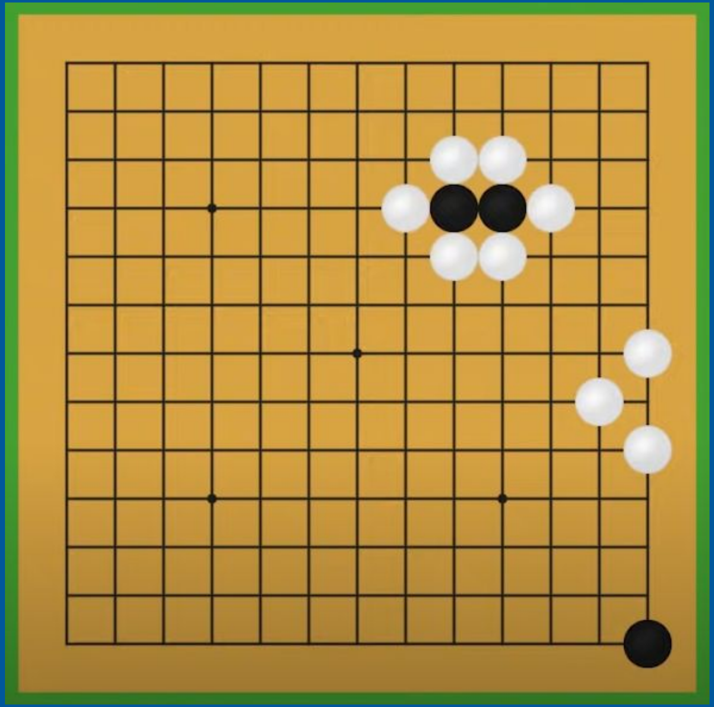
Turn-Based, Two players game

Goal:

Surround and capture opponents

stones, or strategically create

spaces of territory.



Go

19x19 grid

Turn-Based, Two players game

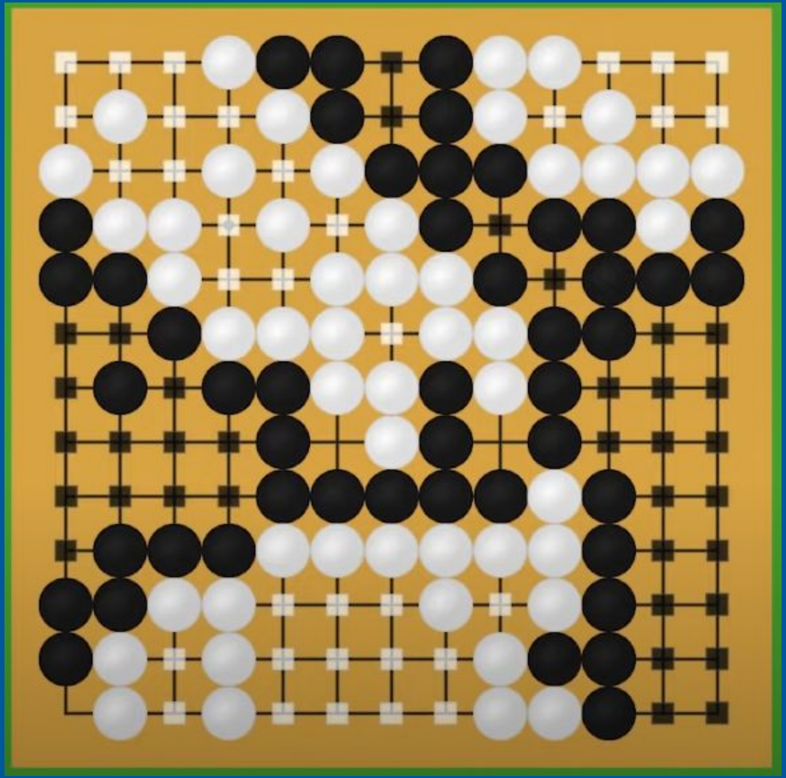
Goal:

Surround and capture opponents

stones, or strategically create

spaces of territory.

Highest points of “empty spaces” win



Go

Simple rules, right?

However, it has 10^{170} possible board

Configurations

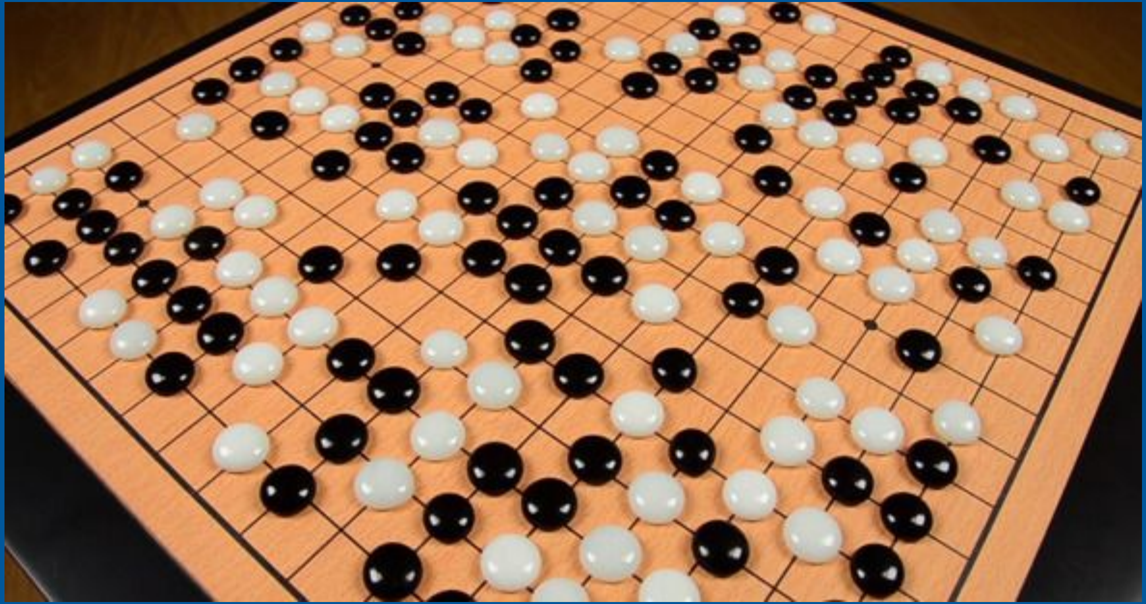
More than the atoms in this universe

YES, you read right. More than the

atoms in this known universe

It's a googol times more complex than

chess



What is AlphaGo

AlphaZero, a single system that taught itself from scratch how to master the games of chess, shogi, and Go, beating a world-champion computer program in each case.

What Problems does it solve?

Turn-based, fully observable positions with definite sets of rules. The opponent goal is to prevent us from winning

Not so intelligent Approach

Brute force method. Search all possible moves and its subsequent branches to evaluate and select the best move

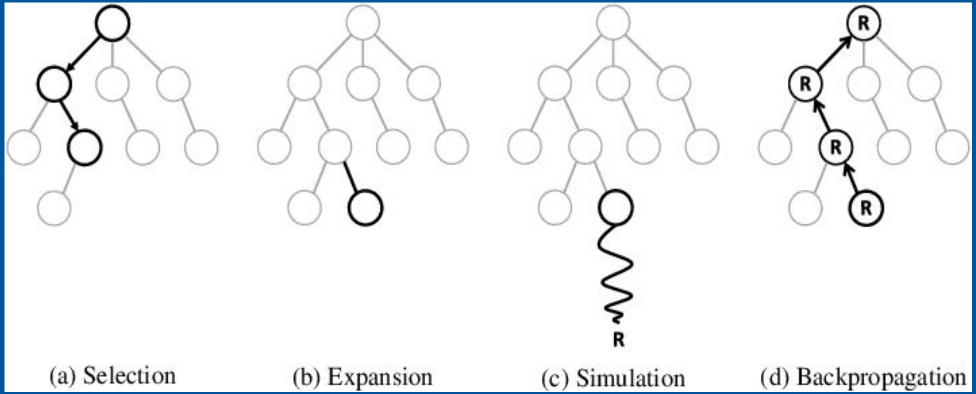
A better Approach, and the idea behind AlphaGo

A deep neural network estimates the most promising set of moves in a search tree

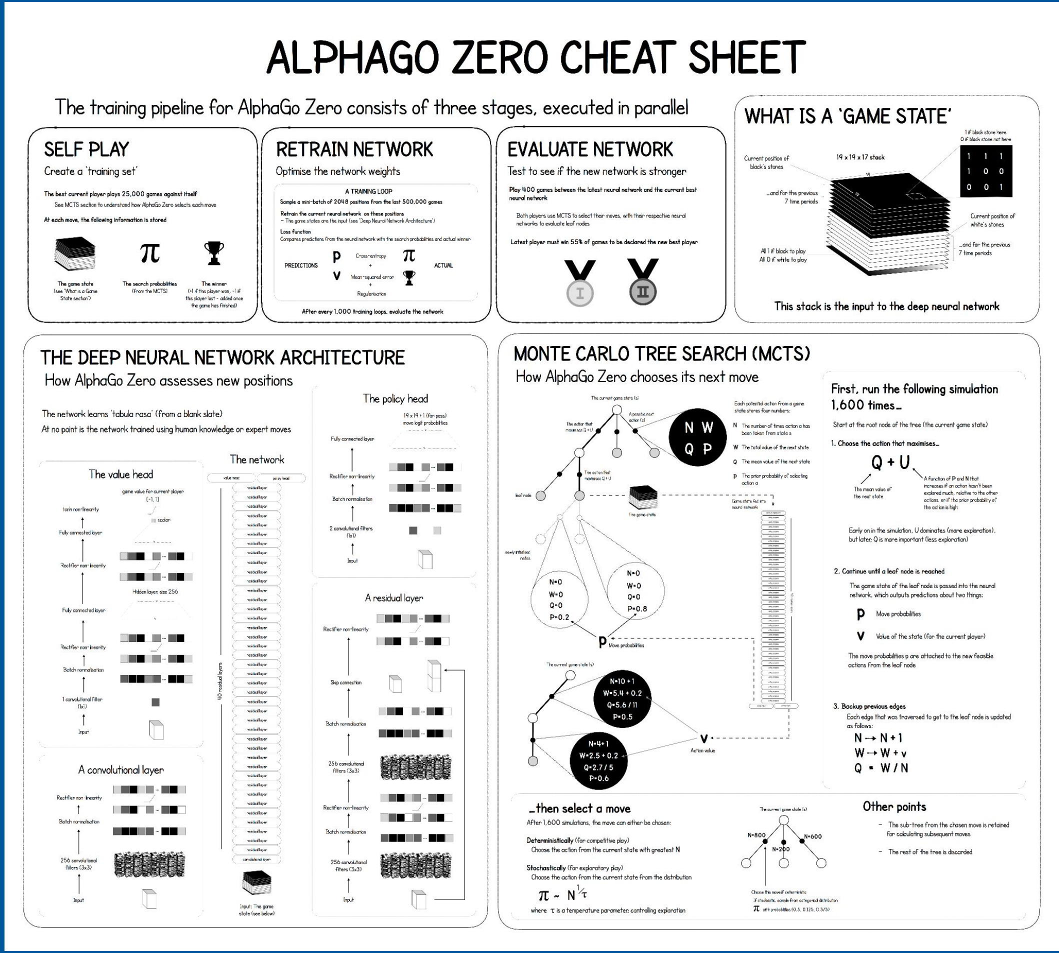
How does it work

Game State as Input → DNN → Estimate the value and proposed policy

Algorithm that performs intelligent search for possible moves based on the suggestion of DNN: Monte Carlo Tree Search



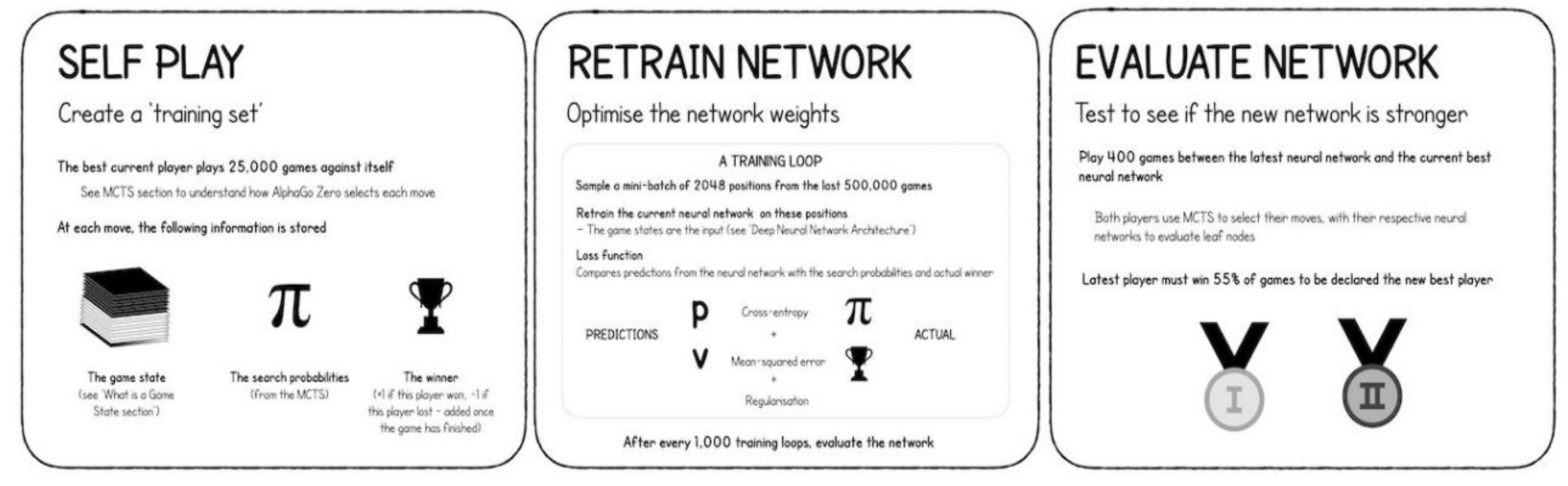
AlphaGo Cheat Sheet



How does it work

https://medium.com/applied-data-science/alphago-zero-explained-in-one-diagram-365f5abf67e0

Three stages executed in parallel



Monte Carlo Tree Search:

The Phases of MCTS:

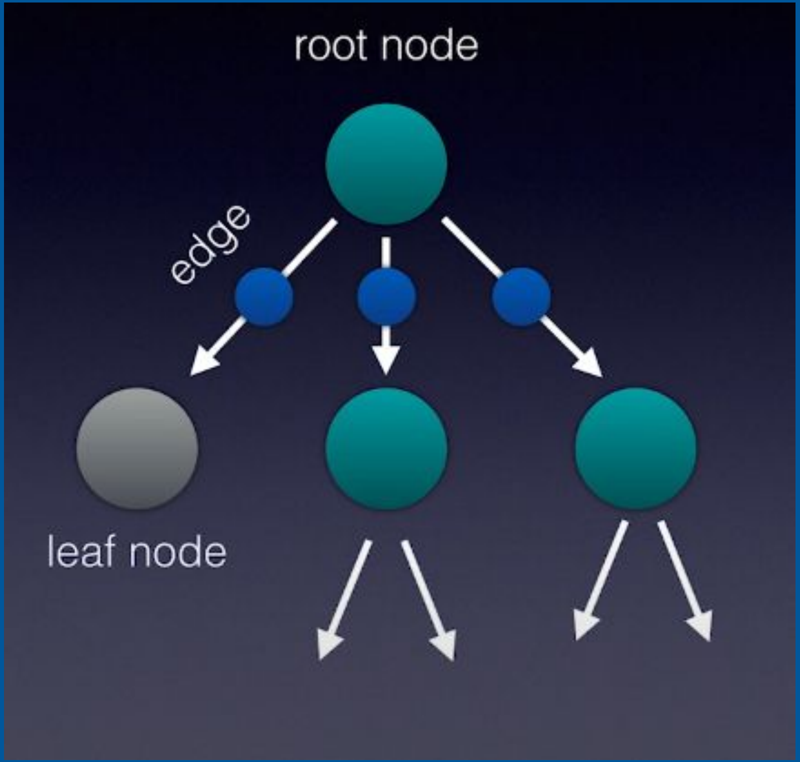
SELECT (Root to Leaf that is most promising)

EXPAND (by using one more move)

BACK UP (and update all edges traversed using statistics)

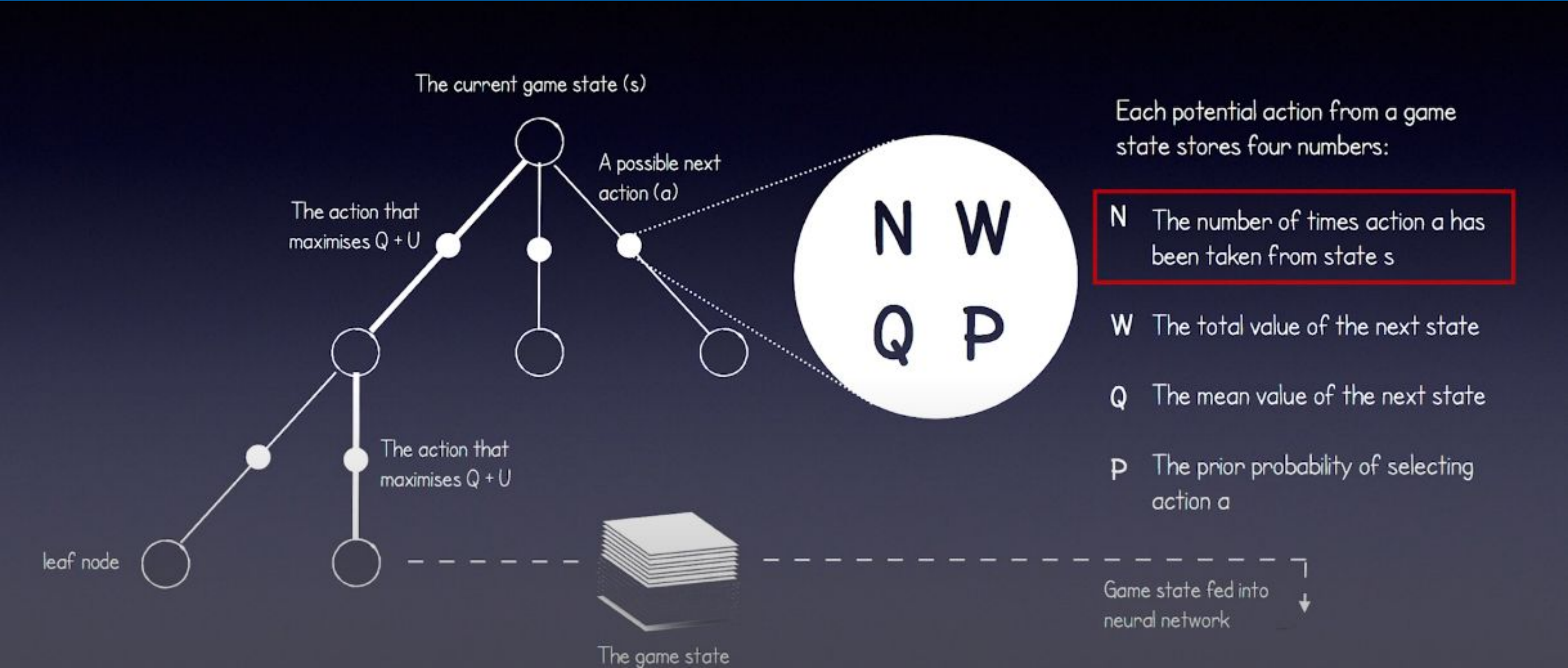
X 1600 times

PLAY



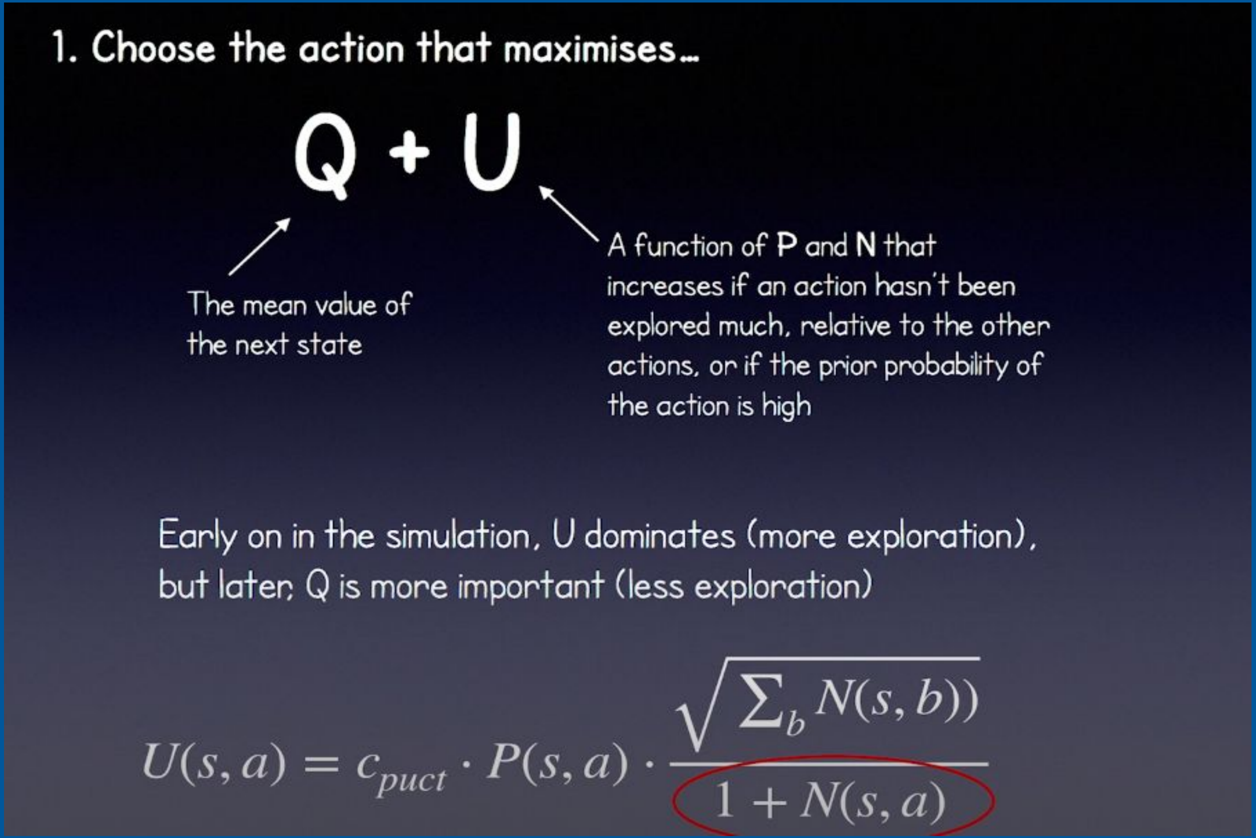
Monte Carlo Tree Search:

Statistics:



Monte Carlo Tree Search:

Details:



Monte Carlo Tree Search:

Details:

2. Continue until a leaf node is reached

The game state of the leaf node is passed into the neural network, which outputs predictions about two things:

p Move probabilities

v Value of the state (for the current player)

The move probabilities p are attached to the new feasible actions from the leaf node

Monte Carlo Tree Search:

Details:

3. Backup previous edges

Each edge that was traversed to get to the leaf node is updated as follows:

$$N \rightarrow N + 1$$

$$W \rightarrow W + v$$

$$Q = W / N$$

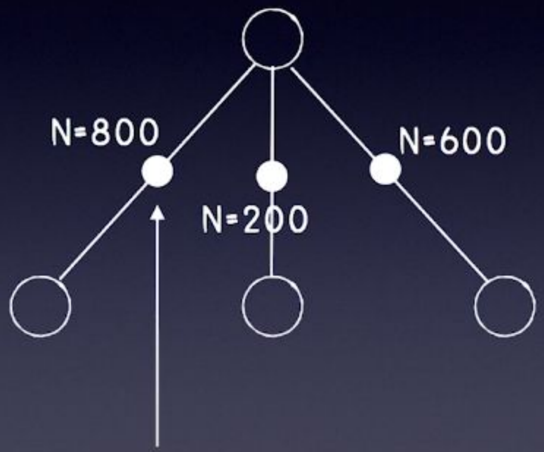
Monte Carlo Tree Search:

Details: For Playing (Selecting the move)

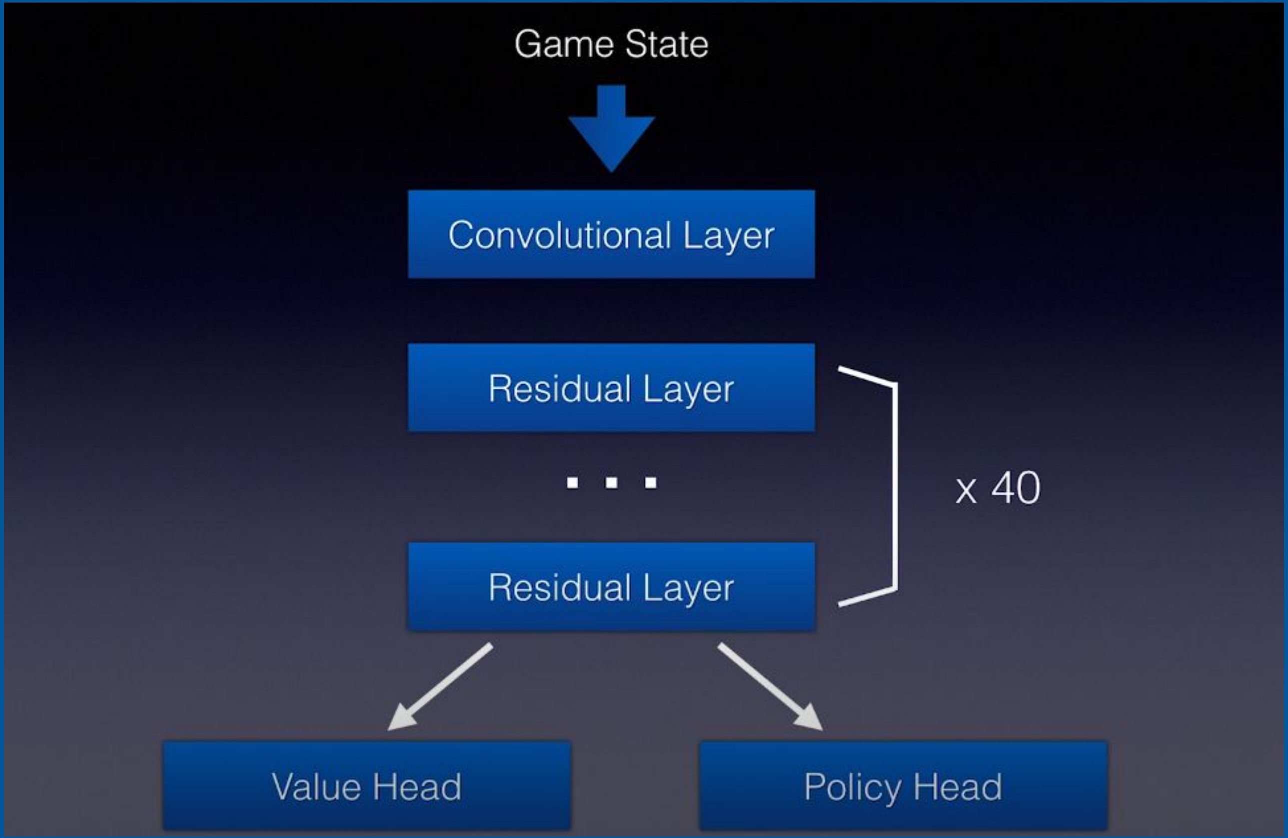
- **Deterministically** (for competitive play) choose the action with the greatest N
- **Stochastically** (for training) sample randomly from the probability distribution…

$$\pi(a \mid s) = \frac{N(s,a)^{1/\tau}}{\sum_b N(s,b)^{1/\tau}}$$

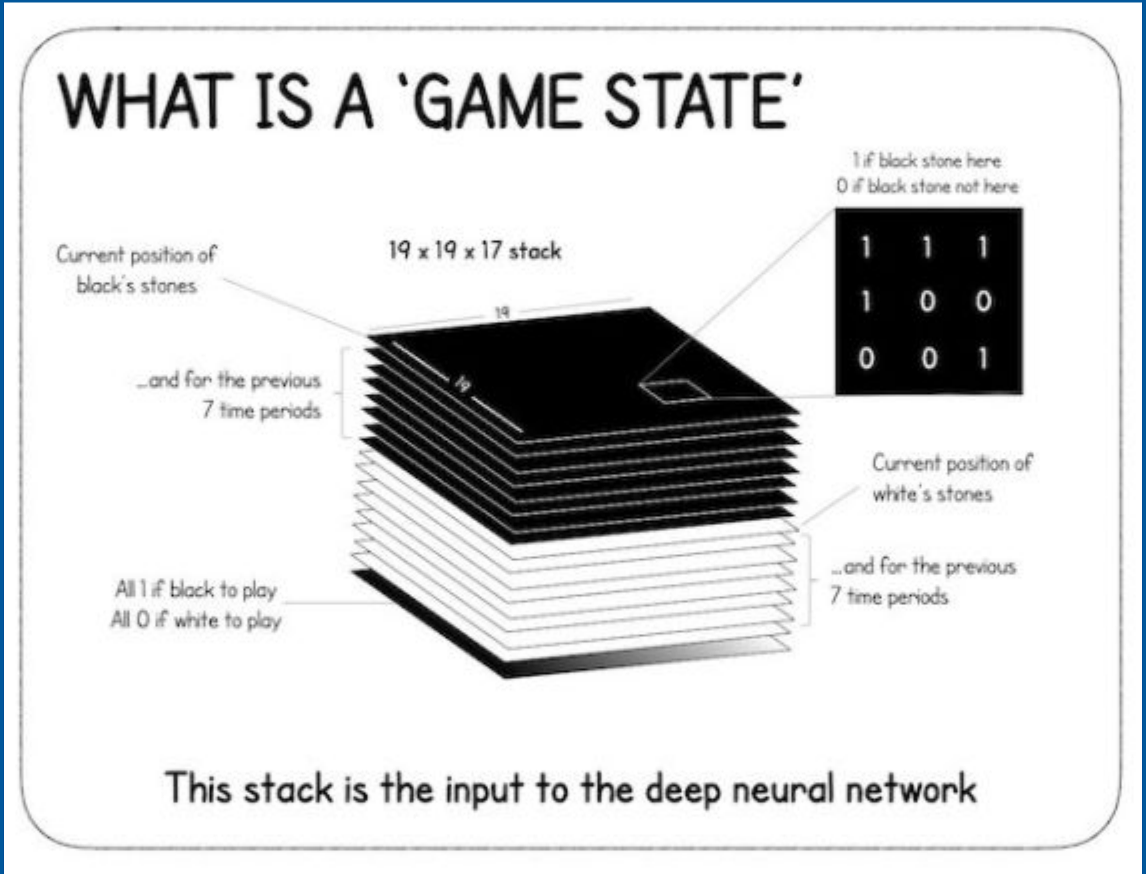
Where τ is a temperature parameter, controlling exploration.



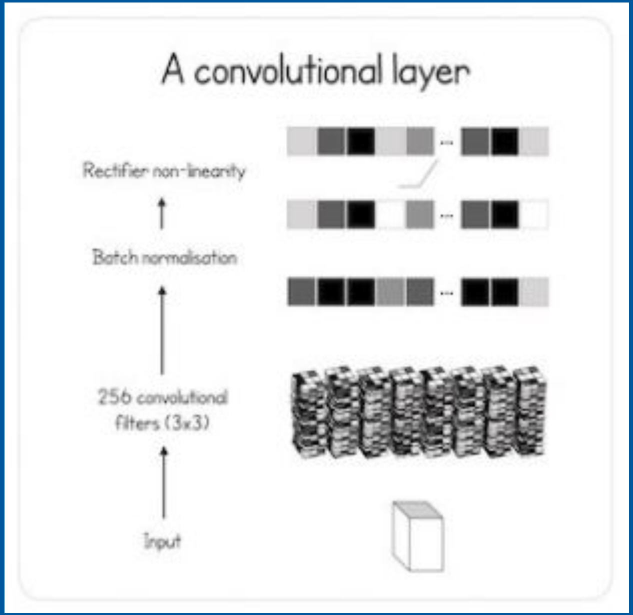
NN Architecture:

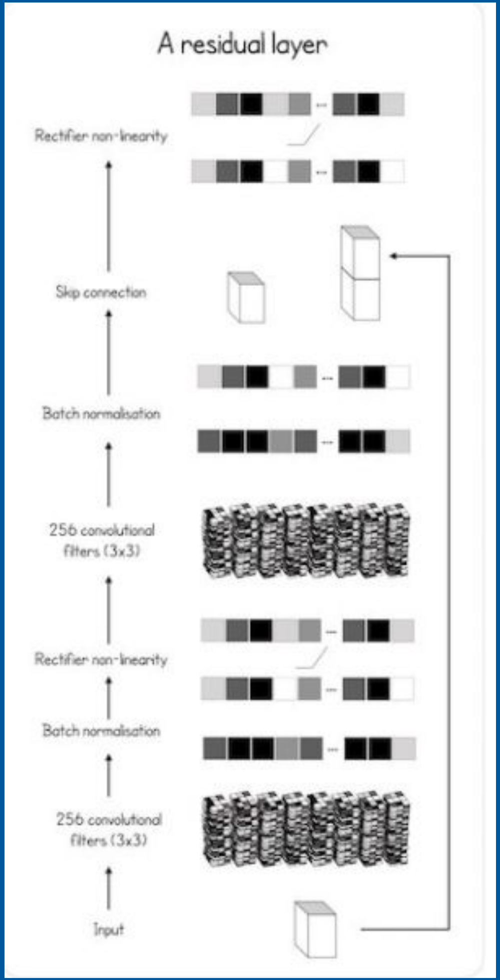


Game State:

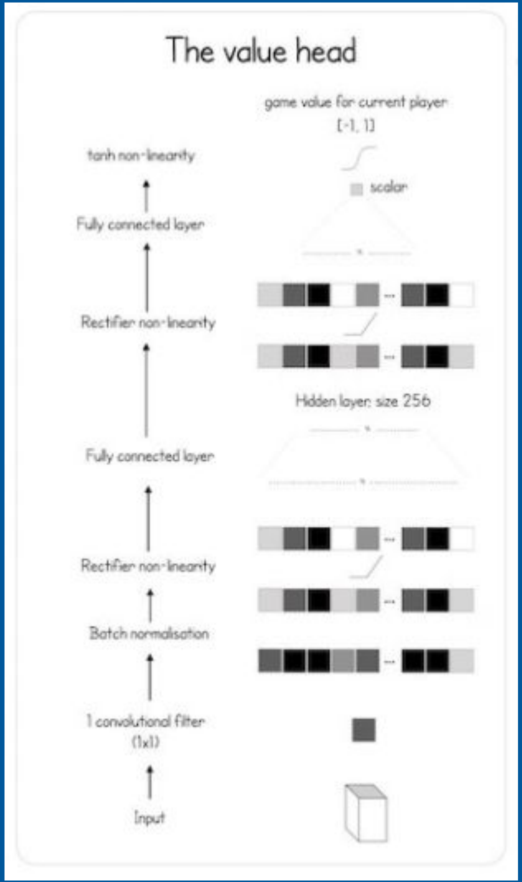

WHAT IS A 'GAME STATE'
1 if black stone here
0 if black stone not here
Current position of
black's stones
19 x 19 x 17 stack
19
19
1 1 1
1 0 0
0 0 1
...and for the previous
7 time periods
Current position of
white's stones
...and for the previous
7 time periods
All 1 if black to play
All 0 if white to play
This stack is the input to the deep neural network

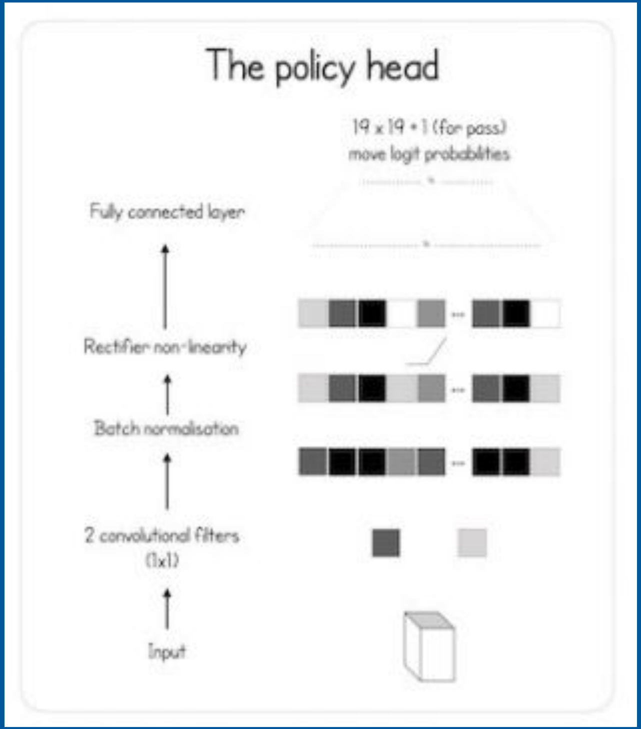
NN Architecture:





NN Architecture:





$$L = (z - v)^2 - \pi \cdot \log(p) + c \cdot ||\theta||^2$$

AlphaGo beats Lee Sedol:

Skoltech

Skolkovo Institute of Science and Technology

Thank You!