Alpha Zero For Connect4

December 14, 2019

1 AlphaZero

1.1 Monte-Carlo Tree Search (MCTS)

1.1.1 Upper Confidence Bound

$U(s,a) = Q(s,a) + \sqrt{\frac{2\ln\sum_b N(s,b)}{1+N(s,a)}}$

$U(s,a)$ is the upper confidence bound for the current state s and action a
$Q(s,a)$ is the expected reward by taking action a in state s
$N(s,a)$ is the number of times we took action a from state s
$\sum_b N(s,b)$ is the total number of plays from state s

1.1.2 Upper Confidence Bound Alpha Zero

$U(s,a) = Q(s,a) + c_{puct}P(s,a)\frac{\sqrt{\sum_b N(s,b)}}{1+N(s,a)}$

$U(s,a)$ is the upper confidence bound for the current state s and action a.
$Q(s,a)$ is the expected reward by taking action a in state s.
c_{puct} is a constant that controls the amount exploration
$P(s,a)$ probability to take action a in state s as predicted by the neural network
$N(s,a)$ is the number of times we took action a from state s
$\sum_b N(s,b)$ is the total number of plays from state s

1.1.3 Training Loss

$l = (z-v)^2 - \pi^T \log p$

z is the outcome of the game -1, 0, 1 for the current player
v is the value prediction of the value
π is the policy form the MCTS
p is the network prediction of the policy

1.1.4 Alpha Zero Algorithm

while current iteration $<$ iterations **do**
  for episode 1, M **do**
    while !game terminated **do**
      while current simulation $<$ mctssimulations **do**
        while !s leaf node **do**
          if s root node **then**
            $p(s) = (1-\epsilon)p(s) + \epsilon\eta_d(\alpha)$
          play move $a = \text{argmax}_a \left(Q(a,s) + c_{puct} p(s,a) \frac{\sqrt{(N(s))}}{1+N(s,a)} \right)$
          $N(s) \leftarrow N(s) + 1$
        if s terminal game state **then**
          $v \leftarrow z$
        else
          evaluate s with the network to get $v(s) and p(s)$
          $v \leftarrow v(s)$
          if player BLACK **then**
            $v \leftarrow -v$
        for all state-action pairs (s,a) **do**
          if player BLACK **then**
            $v \leftarrow -v$
          $Q(s,a) \leftarrow \frac{N(s,a)Q(s,a)+v}{N(s,a)+1}$
          $N(s,a) \leftarrow N(s,a) + 1$
    $p(s,a) = \left(\frac{N(s,a)}{N(s)}\right)^{1/\tau}$
    sample from $p(s)$ to play next self-play move a
    add training example $(s, p(s), v')$ to experience buffer
    get the true outcome z of the game
    for all training examples of game **do**
      if player WHITE **then**
        $v' \leftarrow z$
      else
        $v' \leftarrow -z$
  train the neural network with the training examples from the experience buffer