

## ACM 【程式=演算法+資料結構】@蜡笔小轩V

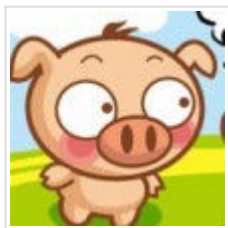
“这么漂亮的女生追求你，你为什么把她拒了？”“妹纸再漂亮，也没有算法漂亮。走了，哥要通宵刷题去了。”

目录视图

摘要视图

RSS 订阅

### 个人资料



Dinosoft

访问：337572次

积分：4275

等级：BLOG > 5

排名：第7499名

原创：94篇 转载：11篇

译文：1篇 评论：182条

### 關於博主

囡轩，男。  
不是信科，只是软院。  
不是校队，只是伪ACMer。  
不是牛人，只是鸟人。

异步赠书：9月重磅新书升级，本本经典

程序员9月书讯

每周荐书：ES6、虚拟现实、物联网（评论送书）

## 蒙特卡罗树搜索+深度学习 -- AlphaGo原版论文阅读笔记

标签：AlphaGo MCTS 深度学习 蒙特卡罗树搜索

2016-03-23 01:01

14008人阅读

评论

分类：机器学习（19）

版权声明：本文为博主原创文章，转载请注明出处。

目录(?)

[+]

原版论文是《Mastering the game of Go with deep neural networks and tree search》

间的可以看看我这篇笔记凑活一下。网上有一些分析AlphaGo的文章，但最经典的有正还是原义，还是踏踏实实搞懂AlphaGo的基本原理我们再来吹牛逼吧。

## 需要的一些背景

对围棋不了解的，其实也不怎么影响，因为只有feature engineering用了点围棋的知识。这里有一篇《九张图告诉你围棋到底怎么下》可以简单看看。

时运不济，命途多舛。  
沒文化，学历低，没钱读研。  
苦逼中的2B青年一枚，2B中的戰鬥機

PS：哥的id是Dinosoft，是dinosaur的谐音，恐龙是也。

## 軌跡

2011.7.13-2011.10.8 深圳企鵝公司实习

2012.7.9- 杭州某工地搬砖,边搬砖边自学机器学习ing。。

## 文章搜索

## 文章分类

ACM题解 (50)

算法学习资料 (6)

web (6)

Linux/C/C++ (11)

搞IT (7)

hadoop (2)

其他 (10)

机器学习 (20)

## 阅读排行

一般筛法求素数+快速线 (34241)

cs229 斯坦福机器学习笔 (21799)

给钓鱼网站骗子一点颜色 (16704)

蒙特卡罗树搜索+深度学

对深度学习不怎么了解的，可以简单当作一个黑盒算法。但机器学习的基础知识还是必备的。没机器学习基础的估计看不太懂。

- 1 “深度学习是机器学习的一种，它是一台精密的流水线，整头猪从这边赶进去，香肠从那边出来就可以了。”

# 蒙特卡罗方法

- 1 蒙特卡罗算法：采样越多，越近似最优解；
- 2 拉斯维加斯算法：采样越多，越有机会找到最优解；
- 3 举个例子，假如筐里有100个苹果，让我每次闭眼拿1个，挑出最大的。于是我随机拿1个，再随机
- 4
- 5 作者：苏椰
- 6 链接：<https://www.zhihu.com/question/20254139/answer/33572009>

# 蒙特卡罗树搜索(MCTS)

网上的文章要不拿蒙特卡罗方法忽悠过去；要不笼统提一下，不提细节；要不就以为只是树形的随机谈。但MCTS对于理解AlphaGo还是挺关键的。

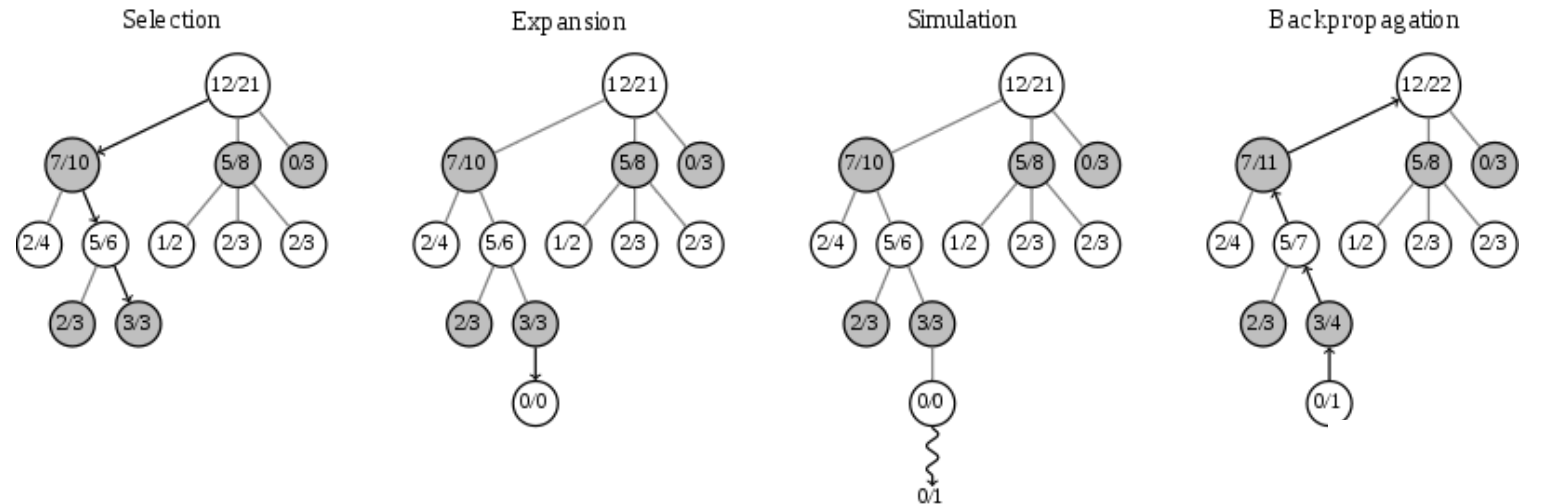
MCTS这里的采样，是指一次从根节点到游戏结束的路径访问。只要采样次数够多，我们可以近似知道走那条路径比较好。貌似就是普通的蒙特卡罗方法？但对于树型结构，解空间太大，不可能完全随机去采样，有额外一些细节问题要解决：分支节点怎么选（宽度优化）？不选比较有效的分支会浪费大量的无谓搜索。评估节点是否一定要走到到底得到游戏最终结果（深度优化）？怎么走？随机走？

基本的MCTS有4个步骤Selection,Expansion,Simulation,Backpropagation（论文里是backup，还以为是备份的意思），论文里state，action，r(reward)，Q函数都是MCTS的术语。

关闭

深度学习斯坦福cs231n i	(13981)
cs229 斯坦福机器学习笔	(10554)
auc指标含义的理解	(8292)
[kaggle实战] Digit Recog	(8012)
UFLDL_Tutorial 笔记 ( d	(7158)
介绍一个可玩性蛮高的东	(7132)
	(7062)

最新评论	
[转] 搞ACM的你伤不起 by Roba Dinosoft: @emoheithree:当年走了狗屎运，误入算法岗。要是现在在投，估计简历都过不了。	
[转] 搞ACM的你伤不起 by Roba 朔北冥: 不知道大佬现在是什么情况?找工作时算法岗位吗？	
一般筛法求素数+快速线性筛法求 msatergz: @sinat_18667599:我又返回去看了一下他的签名	
使用opencv实现通过摄像头自动: Dinosoft: @nuniuqi6442:https://github.com/di	au...
使用opencv实现通过摄像头自动: nuniuqi6442: 你好楼主 你的代码放到github了吗	
[kaggle实战] Digit Recognizer -- ggclml: @qq_25533853:请问解决了吗？我跑出来也都是2	
[kaggle实战] Digit Recognizer -- ggclml: @qq_25533853:请问 解决了吗？我跑出来也都是2	
[kaggle实战] Digit Recognizer -- ggclml: 楼主，我用LR，用你建议的参数，提交结果准确率只有0.85071，请问是什么原因？	
《Wide & Deep Learning for Rec	



图片展示了如何更新节点的胜率，选择胜率大的分支进行搜索（7/10->5/6->3/3），到了3/3叶子节点进个action，然后进行模拟，评估这个action的结果。然后把结果向上回溯到根节点。来自[维基百科](#)

具体的细节，可以参考UCT（Upper Confidence Bound for Trees）algorithm – the most popular algorithm in the MCTS family。从维基百科最下方那篇论文截的图。原文有点长，这里点到为止，足够理解AlphaGO即可。N数，控制exploitation vs. exploration。免得一直搜那个最好的分支，错过边上其他次优分支上的好机会。

```

Algorithm 2 The UCT algorithm.
function UCTSEARCH( $s_0$ )
    create root node  $v_0$  with state  $s_0$ 
    while within computational budget do
         $v_l \leftarrow \text{TREEPOLICY}(v_0)$ 
         $\Delta \leftarrow \text{DEFAULTPOLICY}(s(v_l))$ 
        BACKUP( $v_l, \Delta$ )
    return  $a(\text{BESTCHILD}(v_0, 0))$ 

function TREEPOLICY( $v$ )
    while  $v$  is nonterminal do
        if  $v$  not fully expanded then
            return EXPAND( $v$ )
        else
             $v \leftarrow \text{BESTCHILD}(v, C_p)$ 
    return  $v$ 
    
```

关闭

**aizenggege:** 您好，源码在哪问什么我找不到啊，谢谢

**[kaggle实战] Digit Recognizer -- Dinosoft:**

@u013001422:anaconda集成的python2环境。DL的环境自己另外调试好。

## 文章存档

2017年10月 (3)

2017年09月 (2)

2017年07月 (1)

2017年03月 (1)

2016年12月 (2)

展开

## 评论排行

给钓鱼网站骗子一点颜色 (52)

what if 让我来安排大学本 (25)

[转] 搞ACM的你伤不起 b (15)

[kaggle实战] Digit Recog (12)

一般筛法求素数+快速线l (9)

用位运算巧解元素出现次 (4)

从几何角度看SVD (4)

1137. 河床 (4)

Sicily 1798. Alice and Bc (4)

cs229 斯坦福机器学习笔 (4)

## 博客推荐

酷壳 (RSS)

```
function EXPAND( $v$ )
  choose  $a \in \text{untried actions from } A(s(v))$ 
  add a new child  $v'$  to  $v$ 
    with  $s(v') = f(s(v), a)$ 
    and  $a(v') = a$ 
  return  $v'$ 
```

```
function BESTCHILD( $v, c$ )
  return  $\arg \max_{v' \in \text{children of } v} \frac{Q(v')}{N(v')} + c \sqrt{\frac{2 \ln N(v)}{N(v)}}$ 
```

```
function DEFAULTPOLICY( $s$ )
  while  $s$  is non-terminal do
    choose  $a \in A(s)$  uniformly at random
     $s \leftarrow f(s, a)$ 
  return reward for state  $s$ 
```

```
function BACKUP( $v, \Delta$ )
  while  $v$  is not null do
     $N(v) \leftarrow N(v) + 1$ 
     $Q(v) \leftarrow Q(v) + \Delta(v, p)$ 
     $v \leftarrow \text{parent of } v$ 
```

# AlphaGo

关闭

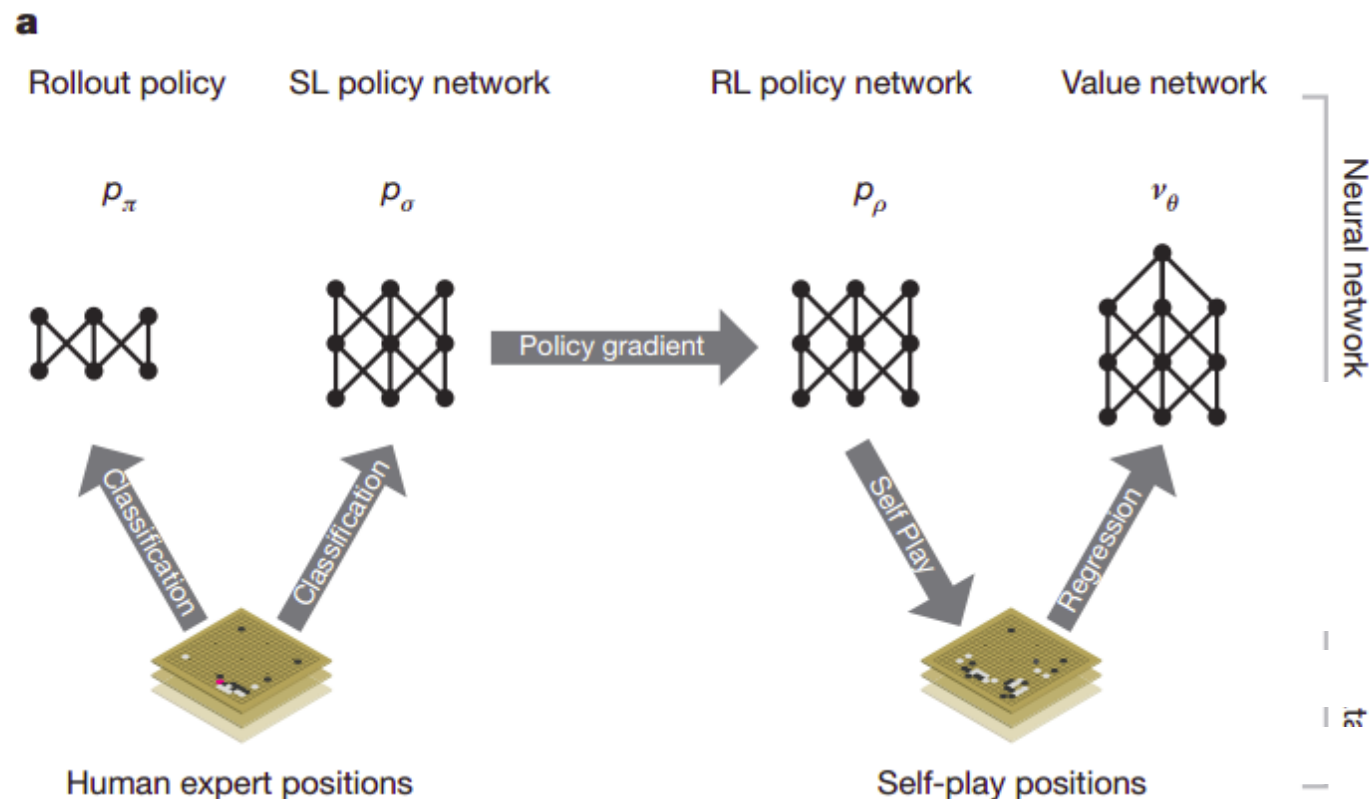
NotOnlySuccess

Void (RSS)

matrix67 (RSS)

## 推荐文章

- \* CSDN新版博客feed流内测用户征集令
- \* Android检查更新下载安装
- \* 动手打造史上最简单的Recycleview 侧滑菜单
- \* TCP网络通讯如何解决分包粘包问题
- \* SDCC 2017之大数据技术实战线上峰会
- \* 快速集成一个视频直播功能



四大组件。最后只直接用了其中3个，间接用其中1个。

## Policy Network ( $P\sigma$ )

Supervised learning (SL) 学的 **objective** 是高手在当前棋面 (state) 选择的下法 (action)。  $P\sigma = (a|s)$

要点

1. 从棋局中**随机**抽取棋面 (state/position)
2. 30 million positions from the KGS Go Server (KGS是一个围棋网站)。数据可以说是核心，所以说AI战胜人类还为时尚早，AlphaGo目前还是站在人类expert的肩膀上前进。
3. 棋盘当作19\*19的黑白二值图像，然后用卷积层（13层）。比图像还好处理。**rectifier nonlinearities**

3. output all legal moves

4. raw input的准确率：55.7%。**all input features**：57.0%。后面methods有提到具体什么特征。需要一点围棋知识，比如liberties是气的意思

## Fast Rollout Policy ( $P\pi$ )

linear softmax + small pattern features。对比前面Policy Network，

- 非线性 -> 线性
- 局部特征 -> 全棋盘

准确率降到24.2%，但是时间3ms-> 2 $\mu$ s。前面MCTS提到评估的时候需要走到底，速度快的优势了。

## Reinforcement Learning of Policy Networks ( $P\rho$ )

要点

- 前面policy networks的结果作为初始值 $\rho = \mu$
- **随机**选前面某一轮的policy network来对决，降低过拟合。
- $z_t = \pm 1$ 是最后的胜负。决出胜负之后，作为前面每一步的梯度优化方向，赢棋就增大预测的P，输棋就减少P。
- 校正最终**objective**是赢棋，而原始的SL Policy Networks预测的是跟expert走法一致的准确率。所以对决结果80%+胜出SL。

跟Pachi对决，胜率从原来当初SL Policy Networks的11%上升到85%，提升还是非常大的。

## Reinforcement Learning of Value Networks ( $v_\theta$ )

关闭

判断一个棋面，黑或白赢的概率各是多少。所以参数只有s。当然，你枚举一下a也能得到p(a|s)。不同就是能知道双方胜率的相对值

- using policy  $p$  for both players (区别RL Policy Network：前面随机的一个P和最新的P对决)
- $v_{\theta}(s) \approx v^{P_{\rho}}(s) \approx v^*(s)$ 。  $v^*(s)$  是理论上**最优**下法得到的分数。显然不可能得到，只能用我们目前最强的  $P_{\rho}$  算法来近似。但这个要走到完才知道，只好再用 Value Network  $v_{\theta}(s)$ 来学习一下了。

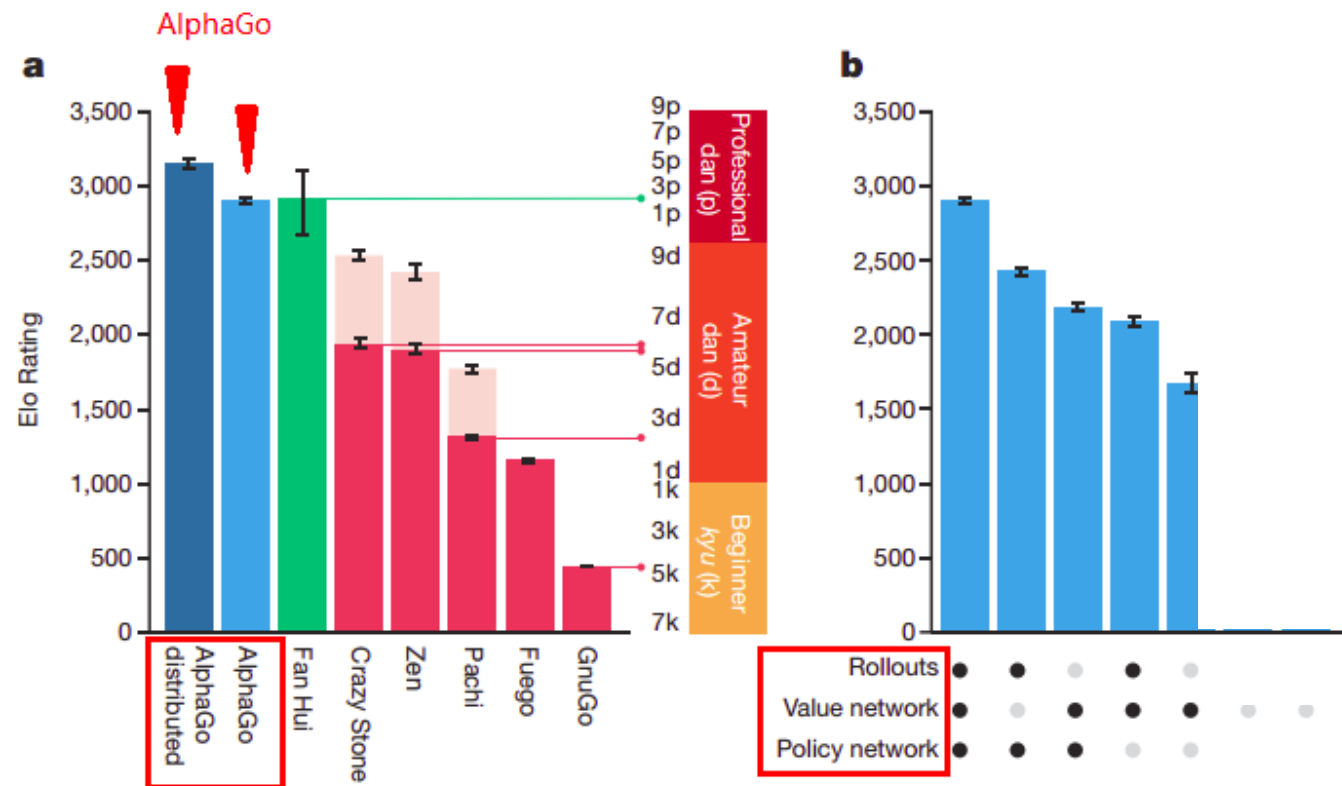
$$\Delta\theta \propto \frac{\partial v_{\theta}(s)}{\partial \theta} (z - v_{\theta}(s))$$

(上面式子应该是求  $\min(z - v_{\theta}(s))^2$ ，转成max就可以去掉求导的负号) 因为前序下法是强关联，有一个棋子不同，z是最后结果，一直不变，所以直接这么算会overfitting。变成直接记住结果。只抽取game中的position，居然生成了30 million distinct positions。那就是有这么多局game了。

MSE	training set	test set
before	0.19	0.37
after	0.226	0.234

关闭

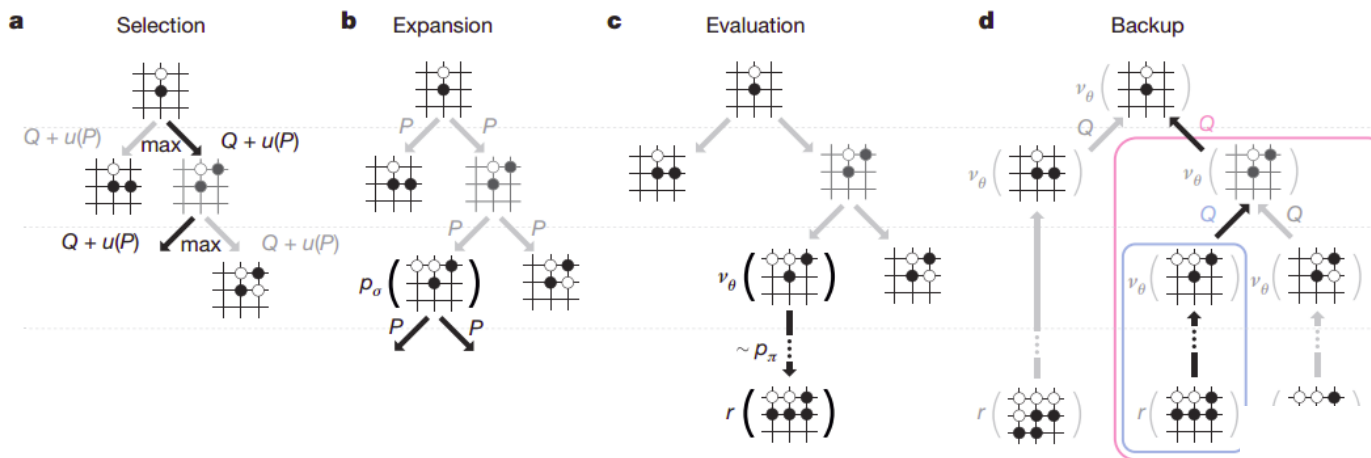
AlphaGo与其他程序的对比。AlphaGo上面提到的几个组件之间对比。这几个组件单独都可以用来当AI，用MCTS组装起来威力更强。(kyu:级，dan:段)



MCTS 组装起来前面的组件

关闭





结构跟标准的MCTS类似。

每次MCTS simulation选择

$$\begin{aligned}
 a_t &= \arg \max_a (Q(s_t, a) + u(s_t, a)) \\
 &= \arg \max_a \left( Q(s_t, a) + C * \frac{P\sigma}{1 + \text{搜索次数}N(s, a)} \right)
 \end{aligned}$$

我自己补了个常数C，写到一起容易看点。

$V(\theta_L)$ 是叶子节点的评估值，Q是多次模拟后的期望 $V(\theta_L)$ 。**有趣的是实验结果 $\lambda = 0.5$ 是最好的**

$$V(\theta_L) = (1 - \lambda)v_\theta(s_L) + \lambda z_T$$

- value network  $v_\theta$
- fast rollout走到结束的结果 $z_L$

最开始还没expand Q是0，那SL的 $P\sigma$ 就是prior probabilities。 $P\sigma$ 还能起到减少搜索宽度的作用，普通点得分很低。比较难被select到。**有趣的结论是，比较得出这里用SL比RL的要好！！**模仿人类走棋的SL结果更适合MCTS搜索，因为人类选择的是 a diverse beam of promising moves。而RL学的是最优的下法（whereas RL optimizes

关闭

for the single best move)。所以人类在这一点暂时获胜！不过另一方面，RL学出来的value networks在评估方面效果好。所以各有所长。

搜索次数N一多会扣分，鼓励exploration其他分支。

## summary

整体看完，感觉AlphaGo实力还是挺强的。在机器学习系统设计和应用方面有很大的参考意义。各个组件的设计也挺有意思。

了解了AlphaGo之后，再去看别人的分析就比较有感觉了，比如fb同样弄围棋的 @田渊栋 的 [AlphaGo轶事 - 知乎专栏](#)

顶 4  
踩 0

关闭

上一篇 [\[kaggle实战\] Digit Recognizer -- 从KNN,LR,SVM,RF到深度学习](#)

下一篇 [深度学习斯坦福cs231n 课程笔记](#)

### 相关文章推荐

- [那么蒙特卡洛树搜索\(Monte Carlo Tree Search, M...](#)
- [蒙特卡罗树搜索+深度学习 -- AlphaGo原版论文阅...](#)
- [Python全栈工程师特训班--韦玮](#)
- [机器学习案例实战--欺诈检测](#)

- AlphaGo背后的搜索算法：蒙特卡罗树搜索 && alp...
- Blink在阿里集团的应用实践--陈守元
- 蒙特卡洛树搜索介绍
- Vue2.x知识点面面通
- 蒙特卡洛树算法 (MCTS)
- 大型Web构架设计案例解析
- 深度学习斯坦福cs231n 课程笔记
- Android开发实战30分钟集成第三方SDK
- Introduction to Monte Carlo Tree Search
- Sicily 1790. Single Round Match
- Sicily 1792. Hengheng's Problem
- 蒙特卡罗树搜索+深度学习 -- AlphaGo原版论文阅...

### 查看评论

2楼 [sinat\\_27995349](#) 2017-01-18 15:41发表



不错，比电视台上吹牛的好多了

1楼 [Dream1992a](#) 2016-07-31 16:32发表



请问博主是否知道蒙特卡洛树搜索是如何实现的？是否可以用MATLAB？

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

\* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

关闭

[公司简介](#) | [招贤纳士](#) | [广告服务](#) | [联系方式](#) | [版权声明](#) | [法律顾问](#) | [问题报告](#) | [合作伙伴](#) | [论坛反馈](#)

网站客服

杂志客服

微博客服

[webmaster@csdn.net](mailto:webmaster@csdn.net)

400-660-0108

| 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 | 江苏乐知网络技术有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved

