Deuro: AI技术白皮书

August 12, 2018

摘要

当我们开发人工智能软件解决方案时,Wo Men Yi Shi Dao Da Yue 10Ci Wai ,Wo Men Huan Jing Li Liao Qi Ye Ke Hu Ti Chu De Xin Ren He An Quan Wen Ti . Zhe Jiu Shi Deuro的用武之地.Deuro是世界上第一个也是唯一一个区块链上的人工智能生态系统。 通过权力下放,Kai Fang Zhi Li ,Wang Luo An Quan He Dai Bi Jing Ji ,Deuro可以帮助AI公司大幅降低计算成本,Suo Duan AI系统开发周期,Ti Gao Sheng Chan Shu Ju Yin Si He An Quan Xing .

由Deuro提供的 人工智能解决方案

1. gStorage: 隐私保护去中心化的人工智能客户端驱动的数据存储系统
    - 1.a. 特征提取(Shu Ju Yu Chu Li)
    - 1.b.去中心化的伪匿名多方密钥加密和解密
    - 1.c. AI数据作为资产
    - 1.d. 企业级别解决方案
    - 1.e. 消费者级别解决方案
    - 1.f. 建模集存储库
    - 1.g. 预测数据存储
    - 1.h. 模型中心

2 gCrawl: 去中心化的高性能算法自动生成的训练数据解决方案
- 2.a. 爬虫任务作为交易
- 2.b. 爬到的数据作为资产
- 2.c. 容错分布式高性能爬网
- 2.d. RAFT一致性共识算法
- 2.e. 去中心化的数据存储
- 2.f. DFSM解码路径
- 2.g. 转移学习以更新模型参数

3. gPredict: 去中心化的分布式梯度学习框架
    - 3.a. 训练任务作为交易
    - 3.b. 训练完备的模型作为资产
    - 3.c. 浮点映射
    - 3.d. 超参数路由
    - 3.e. 结果回归
4. gCompute: 互联网规模的AI dApps解决方案
    - 4.a. AI模型作为资产
    - 4.b. 可扩展性
    - 4.c. 可控制性
    - 4.d. 系统稳定程度
    - 4.e. 可维护性
    - 4.f. 响应性
    - 4.g. 高并发性

关键词

AI dApps: 分散式人工智能应用程序,Qi Hou Duan Tong Chang Shi Qu Kuai Lian

Weak-AI: 弱人工智能/应用人工智能

AGI: 通用人工智能

Brute-force search: 在计算机科学中,Man Li Sou Suo Huo Qiong Ju Sou Suo (Ye Cheng Wei Sheng Cheng He Ce Shi)Shi [?] Chong Fei Chang Pu Bian De Wen Ti Jie Jue Ji Zhu ,Ta Bao Gua Xi Tong Di Mei Ju Jie Jue Fang An De Suo You Ke Neng Hou Xuan Zhe ,Bing Jian Cha Mei Ge Hou Xuan Ren Shi Fou Man Zu Wen Ti De Chen Shu .

AlphaGo(?): AlphaGo是一款玩电脑游戏Go的计算机程序。它由Alphabet Inc.在伦敦的Google DeepMind开发.

GD: 梯度下降是用于找到函数最小值的一阶迭代优化算法。 为了使用梯度下降找到函数的局部最小值,Xu Yao Cai Yong Yu Dang Qian Dian Chu Han Shu De Ti Du (Huo Jin Si Ti Du)De Fu Zhi Cheng Bi Li De Bu Chang . Xiang Fan ,Ru Guo Cai Yong Yu Ti Du De Zheng Zhi Cheng Bi Li De Bu Chang ,Ze Jie Jin Gai Han Shu De Ju Bu Zui Da Zhi ; Ran Hou Jiang Gai Guo Cheng Cheng Wei Ti Du Shang Sheng .

Supervised-Learning: 监督学习是学习函数的机器学习任务,Gai Han Shu Ji Yu Shi Li Shu Ru - Shu Chu Dui Jiang Shu Ru Ying She Dao Shu Chu

Unsupervised-learning: 无监督学习是一种机器学习算法,Yong Yu Cong Mei You Biao Ji Xiang Ying De Shu Ru Shu Ju Zu Cheng De Shu Ju Ji Zhong De Chu Tui Lun .

NBC: 在机器学习中,Po Su Bei Xie Si Fen Lei Qi Shi [?] Xi Lie Jian Dan De “概率分类器”,它基于贝叶斯定理应用特征之间的强(Tian Zhen)Du Li Jia She .

CA: 聚类分析的任务是对一组对象进行分组,Shi De Tong [?] Zu (Cheng Wei Ji Qun)Zhong De Dui Xiang (Zai Mou Chong Yi Yi Shang)Yu Qi Ta Zu (Ji Qun)Zhong De Dui Xiang Geng Xiang Si (Zai Mou Chong Yi Yi Shang)

DMR: 维数减少或降维是通过获得一组主要变量来减少所考虑的随机变量数量[1]的过程。 它可以分为特征选择和特征提取

PCA: 主成分分析(PCA)是一种统计过程,Ta Shi Yong Zheng Jiao Bian Huan Jiang Ke Neng Xiang Guan Bian Liang De [?] Zu Guan Cha Zhi Zhuan Huan Wei Cheng Wei Zhu Cheng Fen De Xian Xing Bu Xiang Guan Bian Liang De [?] Zu Zhi .

K-NN: k-NN是一种基于实例的学习或懒惰学习,Qi Zhong Han Shu Jin Zai Ben Di Jin Si ,Bing Qie Suo You Ji Suan Du Tui Chi Dao Fen Lei . k-NN算法是所有机器学习算法中最简单的算法之一。

SVM: 支持向量机(SVM,也支持向量网络)Shi Dai You Xiang Guan Xue Xi Suan Fa De Jian Du Xue Xi Mo Xing ,Yong Yu Fen Xi Yong Yu Fen Lei He Hui Gui Fen Xi De Shu Ju

RL: 强化学习(RL)是受行为主义心理学启发的机器学习领域[引证需要],关注软件代理应如何在环

境中采取行动以最大化一些累积奖励的概念。

AI-complete: 在人工智能领域,Zui Kun Nan De Wen Ti Shi Fei Zheng Shi Di Cheng Wei AI-complete或AI-hard,这意味着这些计算问题的难度等同于解决中心人工智能问题 - 使计算机像人一样聪明, Huo Qiang Da De AI

NP-complete: 在计算复杂性理论中,NP完全决策问题是属于NP和NP难复杂类的问题。 在这种情况下,NP代表"非确定性多项式时间"。 NP完全问题的集合通常用NP-C或NPC表示

DL: 深度学习(Ye Cheng Wei Shen Du Jie Gou Hua Xue Xi Huo Fen Ceng Xue Xi)Shi Ji Yu Xue Xi Shu Ju Biao Shi De Geng Yan Fan De Ji Qi Xue Xi Fang Fa Xi Lie De [?] Bu Fen ,Er Bu Shi Te Ding Yu Ren Wu De Suan Fa . Xue Xi Ke Yi Shi Jian Du ,Ban Jian Du Huo Wu Jian Du

CNN: 卷积神经网络(CNN,或ConvNet)是一类深度前馈人工神经网络,Yi Cheng Gong Ying Yong Yu Fen Xi Shi Jue Tu Xiang .

RNN: 递归神经网络(RNN)是一类人工神经网络,Qi Zhong Dan Yuan Zhi Jian De Lian Jie Xing Cheng Yan Xu Lie De You Xiang Tu .

LSTM: 长短期记忆(LSTM)单位(Huo Kuai)Shi Di Gui Shen Jing Wang Luo (RNN)的层的构建单元。 由LSTM单元组成的RNN通常称为LSTM网络。 公共LSTM单元由单元,Shu Ru Men ,Shu Chu Men He Wang Ji Men Zu Cheng

Singularity: 技术奇点(Ye Jiu Shi Jian Dan Di Shuo ,Qi Dian)Shi Ren Zao Chao Ji Zhi Neng (ASI)的发明将突然引发技术增长失控的假设,Dao Zhi Ren Lei Wen Ming De Bu Ke Si Yi De Bian Hua . Gen Ju Zhe [?] Jia She ,Ke Sheng Ji De Zhi Neng Dai Li (Li Ru Yun Xing Ji Yu Ruan Jian De Ren Gong Tong Yong Zhi Neng De Ji Suan Ji)Jiang Jin Ru Zi Wo Gai Shan Zhou Qi De "失控反应",每个新的和更智能的一代出现越来越快,Dao Zhi Zhi Neng Bao Zha Bing Chan Sheng Qiang Da De Chao Ji Zhi Neng

GAN: 生成对抗网络(GAN)是一种用于无监督机器学习的人工智能算法,You Liang Ge Shen Jing Wang Luo Xi Tong Zai Ling He You Xi Kuang Jia Zhong Xiang Hu Jing Zheng Shi Xian

SAAS: 软件即服务是一种软件许可和交付模型,Qi Zhong Ruan Jian Shi Ji Yu Ding Yue Xu Ke De ,Bing Qie Shi Ji Zhong Tuo Guan De .

Colo: 托管中心(Ye Cheng Pin Xie Gong Zhi Huo colo)或"运营商酒店"是一种数据中心,Qi Zhong She Bei ,Kong Jian He Dai Kuan Ke Gong Ling Shou Ke Hu Zu Yong .

On-premise distribution: 内部部署软件(You Shi Shi "内部部署"或缩写为"本地内部")安装并使用软件在个人或组织的场所(Jian Zhu Wu Nei)De Ji Suan Ji Shang Yun Xing ,Er Bu Shi Zai Yuan Cheng She Shi Shang Yun Xing . Zuo Wei Fu Wu Qi Chang Huo Yun

DFSM: 确定性有限状态机是一种有限状态机,Ta Jie Shou He Ju Jue Fu Hao Chuan ,Bing Qie Zhi Wei Mei Ge Shu Ru Zi Fu Chuan Chan Sheng Zi Dong Ji De Wei [?] Ji Suan (Huo Yun Xing)

MSE: 均方误差

BFT: 拜占庭容错

Deuro提供的解决方案

在这里,Wo Men Ti Chu Liao Ji Yu Qu Kuai Lian De Chuang Xin AI解决方案,Yi Jie Jue Shang Shu Wen Ti .

gStorage: 对于数据安全和隐私问题,Wo Men Ti Chu Liao gStorage解决方案。 考虑到AI特殊用例和区块链智能合约支持,Wo Men Yin Ru Liao Fen Bu Shi Duan Dao Duan Jia Mi Jie Jue Fang An Lai Cun Chu /Jian Suo Pei Xun Shu Ju ,Sheng Chan Yu Ce Shu Ju He Mo Xing Wen Jian .

gCrawl: 由于缺乏标签数据问题,Wo Men Ti Chu Liao Chuang Xin De gCrawl架构。 gCrawl系统由容错分布式高性能数据爬行系统,You Zhuang Tai NoSQL数据库(Zhong Fu Shu Ju Shan Chu Yi Ji Yuan Shu Ju Cun Chu),Fen San Cun Chu ,Mo Xing Can Shu Chuan Shu Xue Xi Yi Wan Cheng Fan Kui Xun Huan Zu Cheng .

gPredict: 对于算法准确性问题,Wo Men Ti Chu Liao [?] Chong Chuang Xin De Fang Fa Lai Xun Lian /Ping Gu /Ce Shi Qu Kuai Lian Shang De Mo Xing Xing Neng . gPredict包括浮点映射,Chao Can Shu Lu You He Jie Ma Jie Guo Hui Gui .

gCompute: 对于开发,Ce Shi He Bu Shu AI系统时的高硬件和软件成本,Wo Men Yin Ru Liao gCompute模块。 gCompute模块具有可扩展性,Ke Kong Xing ,Nan Du ,Ke Wei Hu Xing ,Xiang Ying Xing He Bing Fa Xing .

1. gStorage: 隐私保护去中心化的人工智能客户端驱动的数据存储系统

为了以无信任的方式安全地存储用户数据,Wo Men Bi Xu Que Bao Ke Hu De Yin Si He An Quan . Zai Zhe Li ,Wo Men Ti Chu Liao [?] Chong Zai Qu Kuai Lian Shang Cun Chu Yong Hu Shu Ju De Chuang Xin Fang Fa . Yi Xia Shi gStorage工作原理的工作流程图。

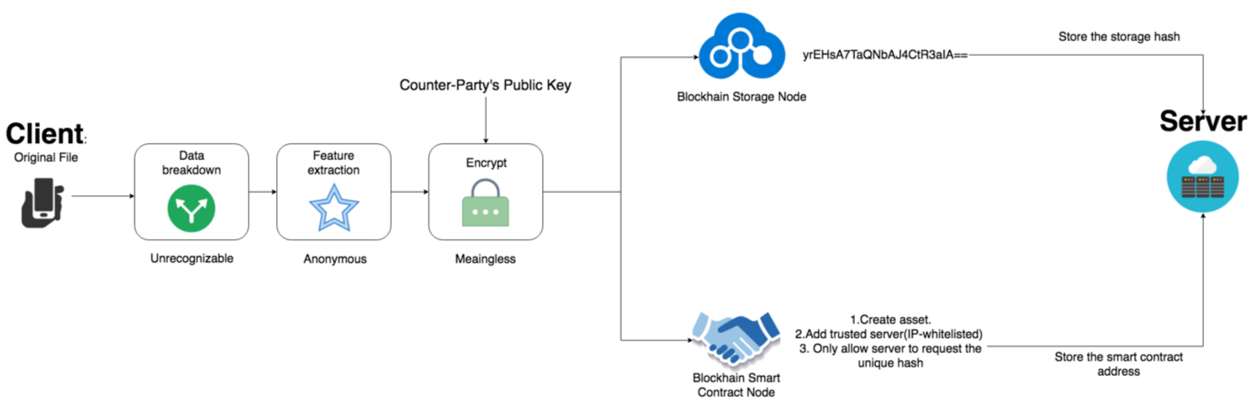


Figure 1: gStorage 工作流程图

1.a. 特征提取(Shu Ju Yu Chu Li)

在这里,Wo Men Zhan Shi Liao Yu Yin Shi Bie He Tu Xiang Shi Bie Zhong Shi Yong De Gong Tong Te Zheng . Yin Pin Gong Neng :Shi Jian Te Zheng ,Guang Pu Te Zheng . Tu Xiang Te Zheng

:Bian Yuan ,Jiao Luo /Xing Qu Dian ,Ban Dian /Gan Xing Qu Qu Yu ,Ji .

时间特征

```
class TF(object):
    def __init__(self, nfilt=40, ncep=13,
                 lowerf=133.3333, upperf=6855.4976, alpha=0.97,
                 samprate=16000, frate=100, wlen=0.0256,
                 nfft=512):
        # Store parameters
        self.lowerf = lowerf
        self.upperf = upperf
        self.nfft = nfft
        self.ncep = ncep
        self.nfilt = nfilt
        self.frate = frate
        self.fshift = float(samprate) / frate

        # Build Hamming window
        self.wlen = int(wlen * samprate)
        self.win = numpy.hamming(self.wlen)

        # Prior sample for pre-emphasis
        self.prior = 0
        self.alpha = alpha

        # Build mel filter matrix
        self.filters = numpy.zeros((nfft/2+1,nfilt), 'd')
        dfreq = float(samprate) / nfft
        if upperf > samprate/2:
            raise(Exception,
                  "Upper frequency %f exceeds Nyquist %f" % (upperf, samprate/2))
        melmax = mel(upperf)
```

```
melmin = mel(lowerf)
dmelbw = (melmax - melmin) / (nfilt + 1)
# Filter edges, in Hz
filt_edge = melinv(melmin + dmelbw * numpy.arange(nfilt + 2, dtype='d'))

for whichfilt in range(0, nfilt):
    # Filter triangles, in DFT points
    leftfr = round(filt_edge[whichfilt] / dfreq)
    centerfr = round(filt_edge[whichfilt + 1] / dfreq)
    rightfr = round(filt_edge[whichfilt + 2] / dfreq)
    # For some reason this is calculated in Hz, though I think
    # it doesn't really matter
    fwidth = (rightfr - leftfr) * dfreq
    height = 2. / fwidth

    if centerfr != leftfr:
        leftslope = height / (centerfr - leftfr)
    else:
        leftslope = 0
    freq = leftfr + 1
    while freq < centerfr:
        self.filters[freq,whichfilt] = (freq - leftfr) * leftslope
        freq = freq + 1
    if freq == centerfr: # This is always true
        self.filters[freq,whichfilt] = height
        freq = freq + 1
    if centerfr != rightfr:
        rightslope = height / (centerfr - rightfr)
    while freq < rightfr:
        self.filters[freq,whichfilt] = (freq - rightfr) * rightslope
        freq = freq + 1
```

```
        # Build DCT matrix
        self.s2dct = s2dctmat(nfilt, ncep, 1./nfilt)
        self.dct = dctmat(nfilt, ncep, numpy.pi/nfilt)

    def sig2s2mfc(self, sig):
        nfr = int(len(sig) / self.fshift + 1)
        mfcc = numpy.zeros((nfr, self.ncep), 'd')
        fr = 0
        while fr < nfr:
            start = round(fr * self.fshift)
            end = min(len(sig), start + self.wlen)
            frame = sig[start:end]
            if len(frame) < self.wlen:
                frame = numpy.resize(frame,self.wlen)
                frame[self.wlen:] = 0
            mfcc[fr] = self.frame2s2mfc(frame)
            fr = fr + 1
        return mfcc

    def sig2logspec(self, sig):
        nfr = int(len(sig) / self.fshift + 1)
        mfcc = numpy.zeros((nfr, self.nfilt), 'd')
        fr = 0
        while fr < nfr:
            start = round(fr * self.fshift)
            end = min(len(sig), start + self.wlen)
            frame = sig[start:end]
            if len(frame) < self.wlen:
                frame = numpy.resize(frame,self.wlen)
                frame[self.wlen:] = 0
```

```
            mfcc[fr] = self.frame2logspec(frame)
            fr = fr + 1
        return mfcc

    def pre_emphasis(self, frame):
        # FIXME: Do this with matrix multiplication
        outfr = numpy.empty(len(frame), 'd')
        outfr[0] = frame[0] - self.alpha * self.prior
        for i in range(1,len(frame)):
            outfr[i] = frame[i] - self.alpha * frame[i-1]
        self.prior = frame[-1]
        return outfr


    def frame2logspec(self, frame):
        frame = self.pre_emphasis(frame) * self.win
        fft = numpy.fft.rfft(frame, self.nfft)
        # Square of absolute value
        power = fft.real * fft.real + fft.imag * fft.imag
        return numpy.log(numpy.dot(power, self.filters).clip(1e-5,numpy.inf))

    def frame2s2mfc(self, frame):
        logspec = self.frame2logspec(frame)
        return numpy.dot(logspec, self.s2dct.T) / self.nfilt

def s2dctmat(nfilt,ncep,freqstep):
    """Return the 'legacy' not-quite-DCT matrix used by Sphinx"""
    melcos = numpy.empty((ncep, nfilt), 'double')
    for i in range(0,ncep):
        freq = numpy.pi * float(i) / nfilt
        melcos[i] = numpy.cos(freq * numpy.arange(0.5, float(nfilt)+0.5, 1.0, 'double'))
    melcos[:,0] = melcos[:,0] * 0.5
```

return melcos

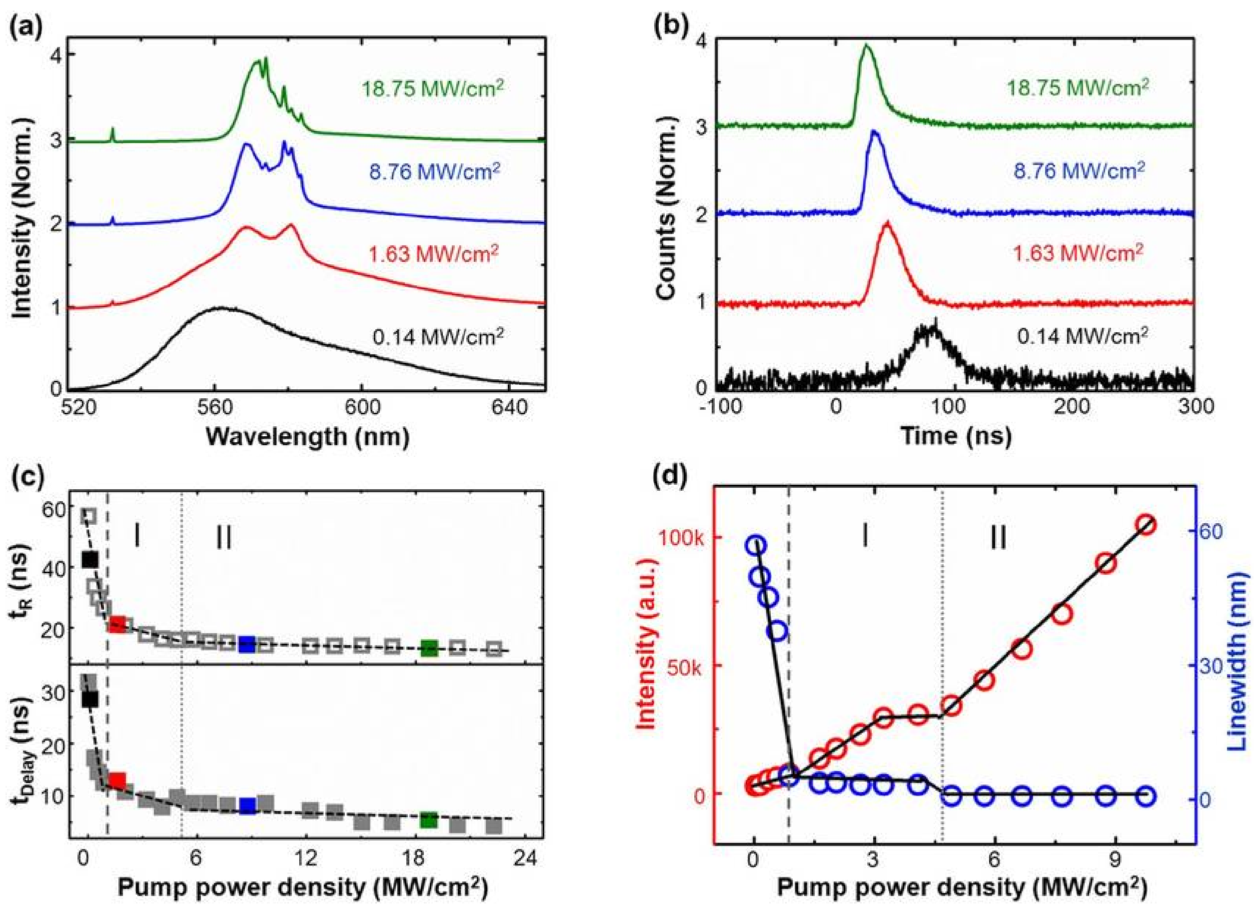


Figure 2: 时间特征

边缘 Edge

边缘是两个图片区域之间存在限制(Huo Bian Yuan)De Jiao Dian . Tong Chang ,Bian Yuan Ke Yi Ju You Xiang Dui Zi You De Xing Zhuang ,Bing Qie Ke Yi Bao Han Jiao Cha Dian . Tong Guo Bian Yuan ,Bian Yuan Tong Chang Bei Biao Zheng Wei Tu Xiang Zhong Ju You Li Ti Jiao Du Fan Wei De Jiao Dian Zu . Ci Wai ,[?] Xie Chang Gui Ji Suan Ran Hou Jiang Gao Qing Xie Du Lian Jie Zai [?] Qi Zhao Xiang Kuang Jia Geng Wan Zheng Di Miao Hui Bian Yuan . Zhe Xie Ji Suan Tong Chang Hui Dui Bian Yuan De Shu Xing Shi Jia [?] Xie Xian Zhi ,Li Ru Xing Zhuang ,Ping Hua Du He Qing Xiang Du . Zai Ju Bu ,Bian Yuan Ju You [?] Wei Jie Gou .

角落/兴趣点 Corners

术语角和关注焦点在某种程度上相反地被利用并且暗示图片中的点状高光,Qi Ju You Lin Yu Er Wei Jie Gou . You Yu Zao Qi Ji Suan Zui Chu Jin Xing Liao Bian Yuan Fa Xian ,Yin Ci Chu Xian Liao “角落”这个名称,Ran Hou Zai Bian Yuan Fa Sheng Gu Zhang Yi Fa Xian Lu Xian (Jiao Luo)De Kuai Su Bian Hua . Ran Hou ,Zhe Xie Ji Suan De Mu Biao Shi Bu Zai Xu Yao Biao Da Bian

Yuan Shi Bie ,Li Ru Tong Guo Zai Tu Xiang Qing Xie Zhong Sou Suo Yi Chang De Xing Zhuang Liang . Ran Hou Ke Yi Kan Dao ,Sheng Cheng De Jiao Luo Tong Yang Bei Shi Bie Zai Tu Pian De Bu Fen Qu Yu Shang ,Zhe Xie Bu Fen Bu Shi Chuan Tong Yi Yi Shang De Jiao Luo (Li Ru ,Zai Hun An De Ji Chu Shang Ke Neng Hui Fa Xian [?] Ge Xiao Liang Dian). Zhe Xie Jiao Dian [?] Xia Zi Bei Cheng Wei Guan Zhu Jiao Dian ,Ran Er Biao Da "角落"被自定义使用。

团 /感兴趣的区域点 Bolb

团给出了图片结构的相关描述,Jiu Xiang Yu Yan Huan Jing Er Yan ,Er Bu Shi Geng Xiang Dian De Jiao Luo . Tong Guo Tuan Miao Shu Fu Ke Yi Ding Qi Di Bao Han You Xuan Dian (Guan Li Yuan Fan Ying De Zui Ji Duan De Lin Yu Huo Zhong Dian),Zhe Yi Wei Zhao Xu Duo Tuan Biao Shi Fu Tong Yang Ke Yi Bei Shi Wei Xing Qu Dian Guan Li Yuan . Ban Dian Cha Zhao Qi Ke Yi Qu Fen Tu Pian Zhong De Qu Yu ,Zhe Xie Qu Yu Guo Yu Ping Hua ,Wu Fa Yi Ren He Fang Shi Bei Jiao Luo Zhi Shi Qi Shi Bie .

山脊 Ridges

对于拉伸物品,Bian Yuan De Si Xiang Shi [?] Chong Te Zheng Zhuang Zhi . Cong Hun An Ji Bie Tu Pian Chu Li De Bian Yuan Miao Shu Fu Ke Bei Shi Wei Ping Jun Shu Zhou De Tui Ce . Cong Shi Yong De Jiao Du Lai Kan ,Bian Yuan Ke Yi Bei Ren Wei Shi Yu Dui Cheng Zhong Xin Dui Hua De [?] Wei Wan Qu ,Bing Qie Geng Ju You Yu Mei Ge Bian Yuan Dian Xiang Guan De Lin Yu Bian Yuan Kuan Du De Zhi Liang . Ke Bei De Shi ,Jin Guan Ru Ci ,Zai Bian Yuan Gao Guang Yu [?] Ban Lei Bie De An Ji Tu Xiang Zhi Jian Jiang Bian Yuan Gao Guang Yu Bian Yuan ,Jiao Luo Huo Ban Dian Gao Guang Qu Fen Kai Lai Zai Suan Fa Shang Geng Nan . Tong Guo Bian Yuan Miao Shu Fu ,Jin Ke Neng Duo Di Yong Yu Fei Xing Tu Pian Zhong De Jie Dao Ti Qu Yi Ji Yong Yu Qu Chu Hui Fu Tu Pian Zhong De Jing Mai - Can Jian Bian Yuan Shi Bie .

1.b. 去中心化的伪匿名多方密钥加密和解密

在这里,Wo Men Jiang Tao Lun gStorage的核心。 去中心化的伪匿名多方密钥加密和解密算法。

1. 它在某种意义上是去中心化的,Wo Men Jiang Shi Yong Qu Kuai Lian Zuo Wei Hou Duan ,Bing Jiang Qian Ming De Jiao Yi Yan Bo Dao Zheng Ge Wang Luo ,Er Bu Hui Yi Lai Zai Ke Xin De Di San Fang Zhi Shang ,Yin Wei Hou Zhe Ke Neng Hui Dao Zhi Dan Dian Gu Zhang .
2. 然后,Ta Zai Mou Chong Yi Yi Shang Shi Wei Ni Ming De ,Wo Men Zhi Bao Hu Ti Qu De Gong Neng Bing Jiang Jia Mi De Ha Xi Bao Lu Gei Zheng Ge Wang Luo . Jiu Xiang Bi Te Bi De Gong Zuo Yuan Li [?] Yang (Yong Hu Shi Yong Si Yao Zai Ben Di Dui Jiao Yi Jin Xing Qian Ming ,Ran Hou Jiang Jiao Yi Ha Xi Yan Bo Dao Zheng Ge Wang Luo Jin Xing Yan Zheng).
3. 它涉及多方,Zai Mou Chong Yi Yi Shang Wo Men Xu Yao Ti Qian Zhi Dao Dui Fang Gong Yao ,Yi Bian Dui Fang (Fu Wu Qi ,Wu Lian Wang She Bei ,Wang Zhan ,IP地址,Zhi Neng He Yue)Ke Yi Zai Yi Hou Shi Yong Ta Men Zi Ji De Si Yao Lai Dui Jia Mi De Ha Xi Jie Mi .

保护提取的特征

一旦我们为AI模型提取了特征表示,Wo Men Jiu Xu Yao Bao Hu Ta . Wo Men Tong Guo Shi Yong Dui Fang De Gong Yao Jiang Ti Qu De Te Zheng Jia Mi Wei Mi Wen Lai Bao Hu Ta . Zhe Li Wo

Men Jiang Shi Yong Xing Ye Biao Zhun De Gong Yao Jia Mi Suan Fa .

1. 每一方都有一个Pair(K,K-1)密钥:K是公钥,K-1是密钥,Zhe Yang $D_{K\text{-}1}$ [E_K [M]] = M
2. 因为知道公钥和密码,Ji Suan Si Yao Zai Ji Suan Shang Shi Bu Ke Xing De Yin Ci Gong Gong Mi Yao Jia Mi Xi Tong Yin Ci Bei Cheng Wei Fei Dui Cheng Jia Mi Xi Tong
3. 公钥K可以公开提供,Li Ru ,Zai Gong Gong Ke Yong Suo Yin Zhi Zhong
4. 许多人可以加密,Zhi You [?] Ge Ren Ke Yi Jie Mi

密钥生成:

选择大小相同,p和q的2个大素数。

计算n = pq,并且Φ(n)=(q-1)(p-1)。

选择随机整数e,1 <e <Φ(n),s.t。 gcd(e,Φ(n))= 1。

计算d,1 <d <Φ(n)s.t. e^d = 1 modΦ(n)。

公钥:(e,n)密钥:d

加密:

给定消息M,0 <M <n使用公钥(e,n),计算C(密文) = M mod n

解密:

M ϵ Zn- {0} CϵZn-{0}.

给出密文C,使用私钥(d)解密.

计算Cd modn=(Me modn)d modn=Med mod n = M

广播加密的哈希

一旦我们将提取的特征加密,Wo Men Jiang Jin Ru p2p分散数据库。 然后将加密特征的唯一散列注册到区块链中,Bing Jiang Ci Jia Mi San Lie Yan Bo Dao Zheng Ge Wang Luo Zhong . Kuang Gong Jiang Zhi Dao Te Zheng De Jia Mi Ha Xi ,Ke Yi Zhao Dao Shu Yu Dan Ge Yong Hu De Suo You Te Zheng Shu Ju ,Dan Kuang Gong Wu Fa Jie Mi Ta (Yin Wei Kuang Gong Mei You Jia Mi Shi De Si Yao). Jiu Xiang Suo You Bi Te Bi Kuang Gong Du Ke Yi Zhao Dao Shu Yu Dan Ge Yong Hu De Jiao Yi [?] Yang ,Dan Ta Men Wu Fa Jiang Zi Jin Zhuan Ru Kuang Gong Zi Ji De Zhang Hu ,Yin Wei Kuang Gong Mei You Dui Fang Zhang Hu De Si Yao .

将资产所有权转让给对方

一旦我们在区块链上注册了特征资产,Fu Wu Qi Jiu Ke Yi Qing Qiu Shu Yu Te Ding Yong Hu De Wei [?] Ha Xi Te Zheng . Ran Hou ,Fu Wu Qi Ke Yi Cong p2p分散数据库中检索加密的密文,Bing Shi Yong Zi Ji De Si Yao Jie Mi Te Zheng De Mi Wen .

———-

Solidity ETH access_control_template()

———-

pragma solidity ^0.4.23;

```
contract gStorageAccessControl {
    address admin;
    address server_address;

    modifier onlyByServer { if (msg.sender != server_address) throw; _ }
    modifier onlyByAdmin { if (msg.sender != admin) throw; _ }
     address[] public features;
    event FindTheUniqueHash(address[] unique_hash);

    function gStorageAccessControl(address server_address_ ) {
        admin = msg.sender;
        server_address = server_address_;
    }
    function registerFeatures(address unique_hash) onlyByAdmin{
        features.push(unique_hash);
    }
    function transfer() onlyByServer {
        //Retrive data from the storage
        FindTheUniqueHash(features);
    }
}
```

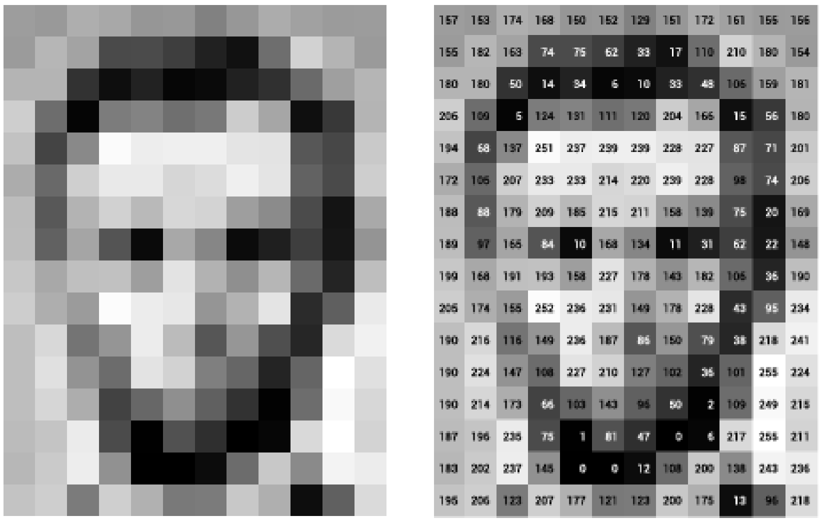
保护隐私的伪匿名数据共享

通过这种方式,Wo Men Tong Guo Ying Yong Te Zheng Ti Qu Lai Bao Hu Ke Hu De Yin Si . Wo Men Shi Yong Xing Ye Biao Zhun Gong Yao Jia Mi Lai Bao Hu Shu Ju De An Quan Xing . Wo Men Huan Tong Guo Jiang Fu Wu Qi De Gong Yao Chuan Di Gei Zhi Neng He Yue Lai Qiang Zhi Zhi Xing Quan Xian Kong Zhi .

1.c. AI数据(Tu Xiang ,Yu Ju)Zuo Wei Zi Chan

图像数据有时有令人困惑的各种形式。 可以表示声音的基本方式的数量实际上相当小。 各种图像文件类型是由于存在很多压缩图像数据的方法和许多不同的数据打包方式。 我们可以将图像抽象为区块链上的资产。 资产也可以是状态机,Qi Zhong Zhuang Tai Zhuan Huan Zai Yuan Shu Ju Zhong Biao Shi . Mei Ci Ji Qi Geng Gai Qi Zhuang Tai Shi ,Du Hui Hong Fa Shi Wu Yi Jiang Yuan Shu Ju Geng Xin Wei Xin Zhuang Tai (Shi Yong WebSocket或HTTP rest方式监听它的状态

变化). Zai Wo Men De Li Zi Zhong ,Wo Men Ke Yi Zai Zhe Chong Zi Chan Zhuang Tai Ji Zhuan Huan Fang Shi Zhong Shi Xian timestamped_private_key。



157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

Figure 3: 图像矩阵表示形式

1.d. 企业级别解决方案

对于企业客户,Wo Men Jiang Wei Ta Men Bu Shu Kai Xiang Ji Yong De Jie Jue Fang An . Ke Hu Xu Yao Gou Mai Wo Men De Tong Zheng Cai Neng Jie Shou Wo Men De Fu Wu . Wan Cheng Fu Wu Hou ,Wo Men Jiang Xiao Hui Ta Men Gou Mai De Tong Zheng .

1.e. 消费者级别解决方案

对于面向消费者的应用程序,Kai Fa Ren Yuan Ke Yi Tong Guo Jian Dan Di An Zhuang Wo Men De SDK快速让gStorage处理他们的应用程序。 我们将实施免费增值模式,[?] Dan Da Dao Mou Xie Shi Yong Ci Shu ,Wo Men Jiang Yao Qiu Ta Men Gou Mai [?] Ding Shu Liang De Tong Zheng Zuo Wei Chu Bei . Wo Men Jiang Gen Ju API调用销毁通证。

1.f. 建模集存储库

考虑到上述架构,Wo Men Ke Yi Qing Song Di Jiang Suo You Pei Xun Shu Ju Cun Chu Dao Qu Kuai Lian Dui Deng Shu Ju Ku Zhong . Dui Yu Yin Pin Xun Lian Shu Ju ,Wo Men Ke Yi Cun Chu Yuan Shi De Wei Ya Suo Huo Wu Sun Ya Suo De wav文件和相应的标准答案。

1.g. 预测数据存储

对于AI dApps生成的所有生产数据,Ta Ye Ke Yi Tong Guo Wo Men Xian You De gStorage管道。并允许用户安全地检索以及发送文件。

1.h. 模型中心

对于流行的深度学习框架(Ru Tensor-flow,Keras,Pytorch,Kaldi,Caffee,CNTK等)Sheng Cheng De Mo Xing ,Wo Men Ye Ke Yi Shi Yong Shang Shu Fang Fa Cong Qu Kuai Lian Shu Ju Ku Zhong

An Quan Di Cun Chu He Huo Qu Ta Men .

2. gCrawl: 分布式高性能算法自动生成的训练数据解决方案

我们展示了一种名为“gCrawl”的全新机器学习架构,Yong Yu Zai Jian Du Fen Lei Ren Wu Zhong Shi Yong Wei Biao Ji De Xin Xi . Wo Men Bu Jia She Wei Biao Ji De Shu Ju Cai Yong Yu Biao Ji Shu Ju Lei Si De Lei Ming Huo Sheng Cheng Miao Shu . Yin Ci ,Wo Men Ke Neng Xiang Yao Li Yong Cong Yin Te Wang Ren Yi Xia Zai De Da Liang Wei Biao Ji Tu Pian (Huo Sheng Yin Shi Li Huo Nei Rong Ji Lu)Lai Zeng Qiang Dui Gei Ding Tu Pian (Huo Sheng Yin Huo Nei Rong)Ren Wu De Zhi Xing . Wo Men Miao Shu Liao [?] Chong gCrawl的方法,Gai Fang Fa Shi Yong Jian Hua De Jie Ma Lu Jing He Chuan Di Xue Xi Lai Cong Wei Biao Ji De Shu Ju Die Dai Mo Xing .

Xia Mian Shi gCrawl计算集群的体系结构,Ta Shi Shui Ping Ke Kuo Zhan De . Zai Shi Jian Zhong ,Wo Men You Neng Li Chu Li **上千万个爬虫任务!**

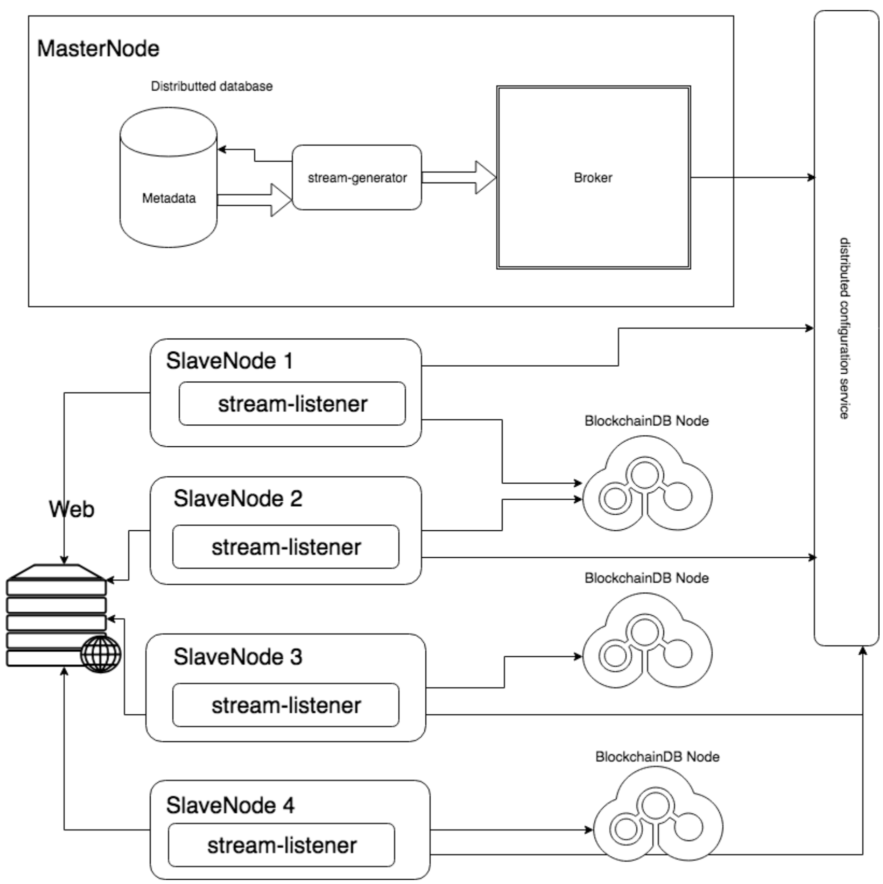


Figure 4: gCrawl爬虫系统

2.a. 将爬虫任务作为交易

我们将爬行任务抽象为区块链分类帐上的交易。

```
contract CrawlingTask {

    struct CrawlingMachine {
        bool crawled;
        address location;
    }

    struct Task {
        string url;
    }

    address submiter;
    mapping(address => CrawlingMachine) machines;
    Task[] tasks;

    function crawl() {
        //The crawling business logic
    }
}
```

2.b. 爬到的数据作为资产

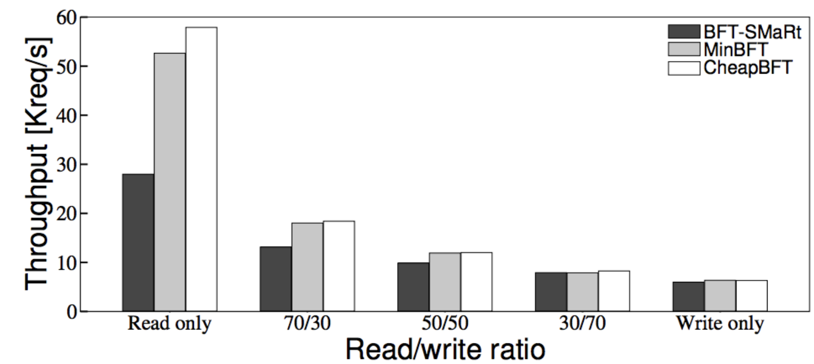
我们可以将已爬到的数据抽象为区块链中的资产

```
contract CrawledData {
    string[] metadata;
    string transcript;
    address audio_url;
}
```

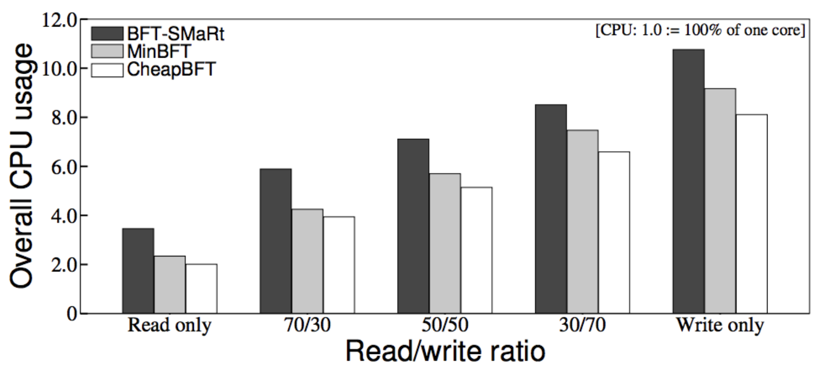
2.c. 高容错分布式高性能爬虫系统

ZooKeeper是一种容错协调服务,Yong Zuo Da Xing Fen Bu Shi Kuang Jia De [?] Bu Fen ,Yong Yu Zhi Xing Xian Jue Ce ,Tong Bu He Gu Zhang Shi Bie Deng Ji Ben Ren Wu . Ci Bu Fen Xian Shi Liao Dui Lei Si ZooKeeper的BFT收益的评估,Gai Shou Yi Yi Lai Yu BFT-SMaRt,MinBFT和CheapBFT分别用于请求传播。 ZooKeeper使客户能够在信息中心中存储和恢复(Tong Chang Hen Shao)Shu Ju Kuai ,Zhe Xie Xin Xi Zhong Xin Zai Ge Chong Ji Bie De Shu Jie Gou Zhong Shou Dao Jian Kong . Wo Men Ping Gu Liao Ge Chong Du Qu He Zhuan Xie Ren Wu Hun He De San Chong Zhi Xing Fang Shi . Zai Suo You Qing Kuang Xia ,1,000个客户不止一次地访问各种信息中心,Zi Xi Yue Du Bing Zhuan Xie [?] Ge Zi Jie Dao Liang Qian Zi Jie Zhi Jian Ren Yi Da Xiao De Xin Xi . Jie Guo Biao Ming ,Zai Zhi Xing Jie Duan (Ji ZooKeeper应用程序)Zhi Xing Zhen Zheng De Gong Zuo (Er Bu Jin Jin Shi Fa Song Da An),Zu He Yue Ding Dui Kuang Jia Zhi Xing De Ying Xiang Jian Ruo Liao . Jie Guo ,San Ge ZooKeeper使用中的每一个都提供了比较吞吐量来组成压倒性的

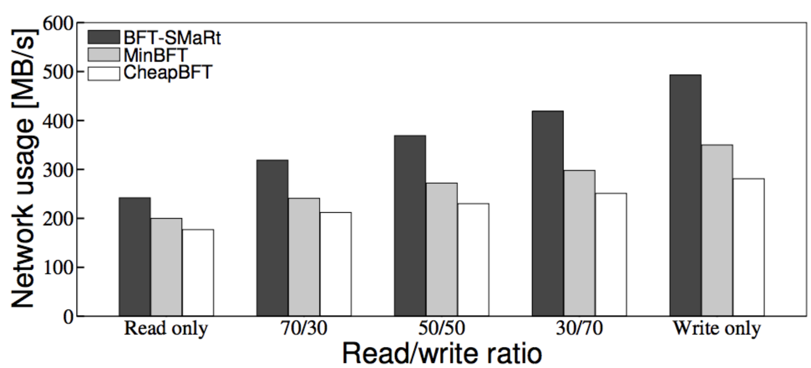
工作负载。尽管如此,Zi Chan Yin Xiang Zai Bian Hua Zhi Jian Cun Zai Gen Ben Dui Bi :Yu Ji Yu MinBFT的ZooKeeper相比,Ji Yu CheapBFT的变体中的复制品可以节省7-12BFT-SMaRt执行相比,Ji Yu CheapBFT的ZooKeeper的资产资金利用23-42



(a) Realized throughput for 1,000 clients.



(b) CPU usage per 10 Kreq/s normalized by throughput.



(c) Network transfer volume per 10 Kreq/s normalized by throughput.

Figure 5: 我们的ZooKeeper服务的不同BFT变体的执行和资源利用结果包括工作负载,Bao Gua Du Te De Du Xie Cao Zuo Zu He .

2.d. RAFT一致性算法

Raft是许多NoSQL数据库中使用的一致性算法。 我们使用NoSQL数据库来存储爬网任务的所有元数据信息。 以下是RAFT一致性算法的领导者选举的伪代码

```
# follower not receiving heartbeats from leader
if election_timeout?
  increment_term()
  change_state_to_candidate()
  vote_myself()
  send_request_vote_to_all_partners()
  while candidate?
    wait_feedback(random_time)
    process_feedback()
    foreach entry in append_entries_received()
      if self.term() < entry.term()
        mark_current_election_as_lost()
      end
    end
    if i_won_election?
      change_state_to_leader()
      start_heartbeat_signals()
    else if partner_won_election?
      change_state_to_follower()
    else if no_one_won_election?
      increment_term()
    end
  end
end

# continue as leader or follower...
```

2.e. 去中心化的数据存储

我们将使用不同的分散式数据区块链存储解决方案来存储已爬到的数据。 我们已经在第1节中详细介绍存储部分。

2.f. DFSM解码路径

对于每个类别,Wo Men Shou Xian Shi Yong Guan Jian Zi Sou Suo (Tong Chang ~106-108张图片)Shou Ji Da Liang Qian Zai Tu Pian . Ran Hou Zai Mou Ge Shi Ke ,Wo Men Zhong Fu Qing Qiu Ge Ren Biao Ji [?] Ge Xiao Zi Ji ,Zai Gai Zi Ji Shang Zhun Bei Fen Lei Qi ,Yao Qiu Fen Lei Qi Yu Jian Biao Ji He Zhi Xin Du ,Ran Hou Xuan Ze [?] Ge Xiao De Shen Qing Ren Ji Jin [?] Bu Kao Lu .

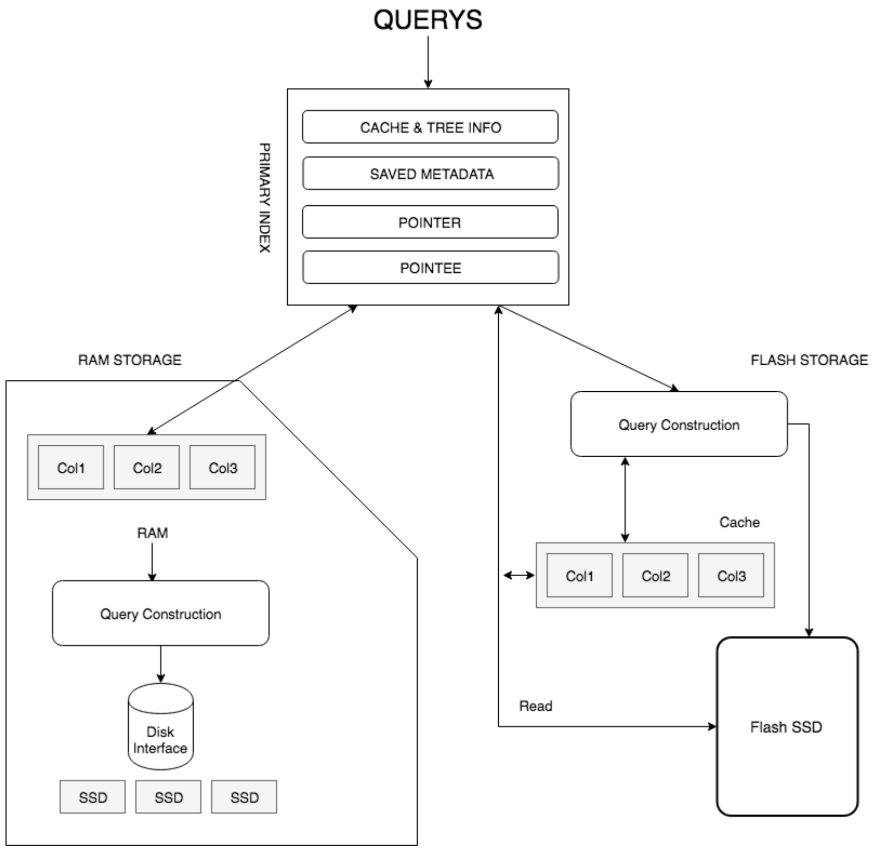


Figure 6: 分布式数据库存储系统

2.g. 转移学习以更新模型参数

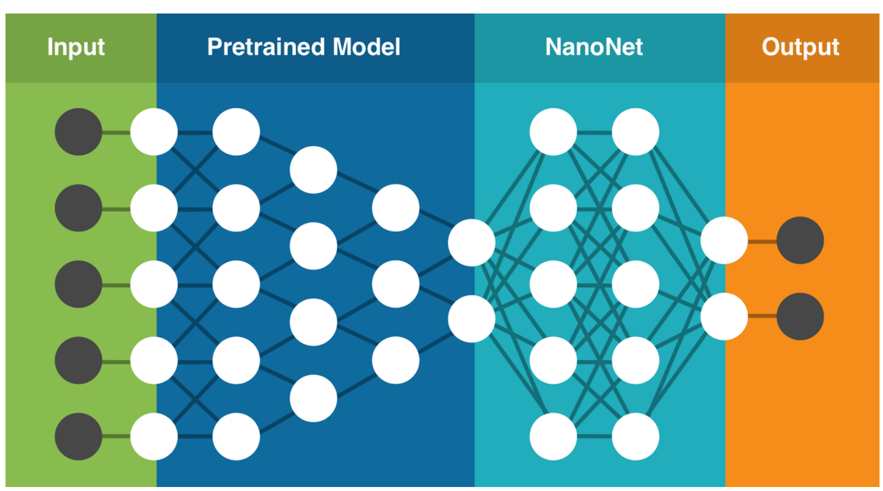


Figure 8: 转移学习图

(转移学习)Gei Ding Yuan Yu DS和学习任务TS,目标域DT和学习任务TT,转移学习旨在使用DS和TS中

的知识帮助改进DT中目标预测函数fT(·)的学习, Qi Zhong DS!= DT,或TS!= TT。 在上面的定义中,Yu Shi [?] Dui D = {X,P(X)}。 因此,Tiao Jian DS!= DT表示XS!= XT或PS(X)!= PT(X)。

3. gPredict: 去中心化的梯度学习框架

为了提高AI算法的性能,Wo Men Xu Yao Ti Gao Jing Du /Zhao Hui Lu ,Jiang Di MSE。 在本节中,Wo Men Ti Chu Liao [?] Xi Lie Jian Shao Qu Kuai Lian Shang AI模型培训周转时间的方法。

Estimator: 只是一个简单的函数来模拟数据样本的一些有意义的特征。 例如, $Y = g(S)$, $S = (x(1),,,,x(m))$, 其中x(i)是从分布中得出的随机变量 D, i.e. $x(i) \sim D$.

Estimator Bias 衡量我们的估算器在估算实数方面有多好. $Bias(Y_S, Y) = E_{S\sim D^m}[Y_S] - Y$

Estimator Variance 衡量我们的估算器对采样的“跳跃”程度. $Var(Y) = Var_{S\sim D^m}[Y]$. **Bias-variance decomposition for estimators:** 将这两个属性组合在一个公式中。$MSE = E[(Y_S - Y)^2] = Bias^2(Y_S, Y) + Var(Y_S)$ 对S随机变量的期望值。

上述结论的正式数学证明:

$$E[(Y_S - Y)^2] = E[Y_S^2] + Y^2 - 2E[Y_S]Y$$

$$Bias^2(Y_S, Y) = (E[Y_S] - Y)^2$$

$$= E^2[Y_S] + Y - 2E[Y_S]Y$$

$$Var(Y_S) = E[Y_S^2] - E^2[Y_S]$$

3.a. 训练任务作为交易

```
contract TrainingTask {
    TrainedModel previousModel;
    function training();
    TrainedModel trainedModel;
}
```

3.b. 训练完备的模型作为资产

```
contract TrainedModel {
    string[][] configuration;
    double[] weights;
    state[] optimizerState;
}
```

3.c. 浮点映射

根据我们对低精度定点环境中有限数精度的研究,Wo Men Fa Xian Tong Guo Zai Shi Yong Sui Ji Ti Du Xia Jiang Shi Jin Shi Yong 16位表示来控制舍入方案,Wo Men Ke Yi Geng Kuai Di Xun

Lian Mo Xing Er Bu Hui Sun Hai Ren He Jing Du .

Limited Precision Arithmetic 有限精度算术: 基于反向传播的深度学习系统的标准执行通常利用实数的32位浮点表示来进行数据存储和控制。 我们通过表示[IN.FR]来抽象这个问题,Qi Zhong IN和FR分别与数字的整数和小数部分相关。 数字位数(NB)加上小数位数(FB)产生用于表示整数的总位数。 NB + FB的总和称为字长WL。

Stochastic rounding 随机四舍五入: 给定数字x和目标定点表示 <NB, FB>. 我们定义 $\lfloor x \rfloor$ 作为最大的整数倍 $\theta\left(= 2^{-FB}\right)$ <= x. S随机舍入是一种无偏舍入方案,Bing Ju You Yu Qi De She Ru Wu Chai Wei Ling De Li Xiang Te Xing , i.e. E(Round(x, <NB,FB>)) = x

所以, P(rounding x to $\lfloor x \rfloor$) 相对于x的近似值 $\lfloor x \rfloor$:

$$Round(x, < NB, FB >) = \begin{cases} \lfloor x \rfloor, & amp; \text{if } 1 - \dfrac{x - \lfloor x \rfloor}{\theta}, \\ \lfloor x \rfloor + \theta, & amp; \dfrac{x - \lfloor x \rfloor}{\theta}. \end{cases}$$

然后,

$$Convert(x, < NB, FB >) = \begin{cases} -2^{NB-1}, & amp; \text{if } x <= -2^{NB-1}, \\ 2^{NB-1} - 2^{-FB}, & amp; \text{if } x >= 2^{NB-1} - 2^{-FB}. \\ Round(x, < NB, FB >), & amp; otherwise. \end{cases}$$

Multiply and accumulate operation 乘以和累加运算: 考虑两个二维的向量 **d** 和 **h** 这样每个向量都以固定的定点表示 <NB,FB>, 然后定义 **e** = d.h 作为d和h的内积空间. 然后我们可以将计算分成以下步骤:

$z = \sum_{i=1}^{d} a_i b_i$

$c_0 = Convert(z, < NB, FB >)$

3.d. 超参数路由

gPredict的活动是找到一个最接近的标量,Ke Neng Shi Sui Ji Han Shu De Zui Jia Gu Ji ,Er Bu Shi Dui Gai Han Shu De Ke Neng Can Shu De Pai Lie . Sui Ran Xu Duo Kun Bang Bao Qi Wang Zhe Xie Shu Ju Yuan Shi Cong Xiang Liang Kong Jian Zhong Ti Qu De ,Dan gPredict的独特之处在于它促使开发人员更详细地描绘搜索空间。 通过提供有关函数表征位置的更多数据以及最佳质量的位置,gPredict可以更有效地搜索最佳参数。

3.e. 结果回归

最后借助浮点映射和超参数路由,Wo Men Ke Yi Tong Guo Jian Shao Fen Bu Shi Xi Tong De Qu Kuai Lian Lei Xing Lai Huo De Jie Guo .

Reducer algorithm

```
class REDUCER
    def REDUCE(string t, pairs [(s1,c1),(s2,c2)...])
        sum <-- 0
        cnt <-- 0
        for all pair (s,c) belongs to pairs:
```

```
        sum += s
        cnt += c
    rAvg = sum / cnt
    EMIT(string t, integer rAvg)
```

4. gCompute: 互联网规模的AI dApps解决方案

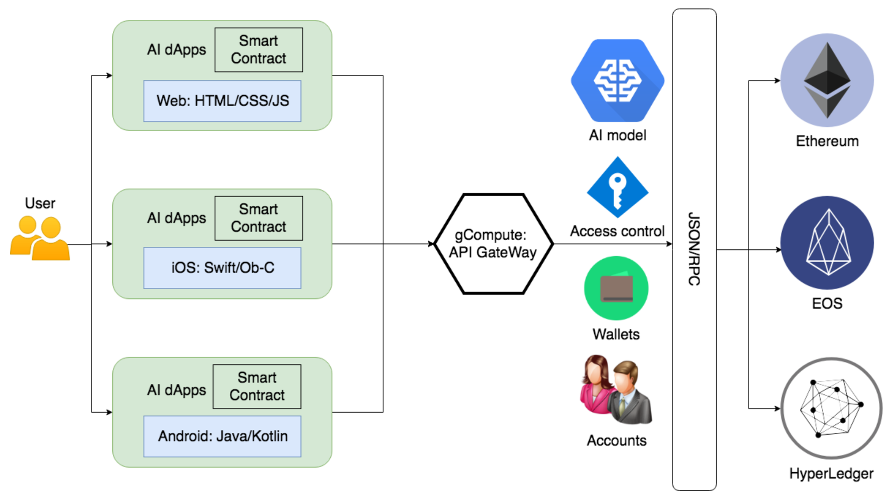


Figure 9: gCompute API 网关

4.a. AI模型作为资产

与我们上面对训练完备模型的定义类似,Wo Men Ke Yi Jiang AI模型表示为区块链上的资产。

4.b. 可扩展性

技术可扩展性: 为了与谷歌,Facebook,阿里巴巴,Teng Xun Deng Chuan Tong Hu Lian Wang Ju Tou Jing Zheng ,Wo Men De Di Ceng Ping Tai Qu Kuai Lian Ji Zhu Bi Xu Heng Xiang Kuo Zhan . Ru Guo Wo Men You AI dApp注册并使用我们的平台轻松集成到所有现有的区块链后端,Wo Men De Xi Tong Neng Gou Chu Li Shu Qian Mo Ji Bie Ri Chang Huo Yue Yong Hu . Dui Yu Zhe Chong Lei Xing De Gong Neng Xu Qiu ,Wo Men Jiang Bu Shu Duo Ge Shi Li ,Jiang He Xin Ye Wu Luo Ji Chou Xiang Wei docker,并将负载均衡器放在我们的实例前面,Yi Que Bao Mei Ge API / JSON调用顺利进行。

商业模式的可扩展性: 传统的互联网公司向我们证明,Zui Hao De Shang Ye Mo Shi Shi Shou Xian Ti Gong Mian Fei Fu Wu ,Ran Hou Ti Gong Ge Chong Shou Ru Mo Shi (Mian Fei Zeng Zhi ,Gao Ji Ban ,Ding Yue Ban). Yin Ci ,Wo Men De Di Ceng Qu Kuai Lian Ji Zhu Ping Tai Ying Gai Ju You Lei Si De Ling Huo Xing .

4.c. 可控制性

不断升级: 无论是大企业还是小型创业公司,Shen Zhi Shi Ge Ti Kai Fa Zhe Du Ying Gai Neng Gou Ling Huo Di Bu Duan Di Shi Yong Xin Gong Neng Sheng Ji Ta Men De Ying Yong Cheng

Xu . Wo Men De Ping Tai Jiang Zhi Chi Mo Xing Ruan Jian He Zhi Neng He Yue Sheng Ji . Ji Shi Gu Ge Chrome,Facebook,腾讯微信,Ping Guo iOS都受到不同的错误或系统故障,Wo Men De Ping Tai He Di Ceng Ping Tai Ying Gai Zu Gou Qiang Da ,Ke Yi Xiu Fu Cuo Wu Bing Zi Dong Sheng Cheng Cheng Xu Diao Yong Dui Zhan .

持续集成: 当开发人员为其现有软件添加新功能时,Ta Men Bu Xiang Zhi Dong Ren He Jiu Gong Neng . Zhe Jiu Shi Chi Xu Ji Cheng Jie Jue Fang An De Yong Wu Zhi Di . Ru Guo Mou Xie Ti Jiao Po Pi Liao Jiu Gong Neng (Wu Fa Tong Guo Mou Xie Ce Shi Yong Li),Dai Ma Tuo Guan Ping Tai Jiang Ju Jue Gai Ti Jiao Bing Sheng Cheng You Guan Cuo Wu De Bao Gao . Wo Men De Xi Tong Yi Ji Di Ceng Qu Kuai Lian Ping Tai Ye Jiang Zhi Chi Zhe Chong Liang Hao Shi Jian .

失败恢复: 最糟糕的情况:Ru Guo Wo Men Yu Dao Wang Luo Gu Zhang ,Xi Tong Gu Zhang ,Di Ceng Qu Kuai Lian Ping Tai DDOS攻击或任何用户密钥凭据被泄露,Wo Men De Ping Tai Yi Ji Di Ceng Qu Kuai Lian Ping Tai Jiang Ju You Gu Zhang Hui Fu Xi Tong . Tong Guo Dui Bu Tong Ban Ben De AI模型以及数据库状态进行版本控制,Wo Men Jiang Neng Gou Wei Zui Zhong Yong Hu He Kai Fa Ren Yuan Ti Gong Ci Lei Gong Neng .

4.d. 系统攻击难度

PoW: 对于公共区块链,PoW是最常采用的一致算法。 比特币是PoW的先驱,Shi Qu Kuai Lian Sheng Tai Xi Tong Zhong Zui Zhao Ming De Gong Shi Suan Fa . PoW难以破解,Xu Yao Chao Guo 50PoW的公共区块链,Bing Zai Xiang Ying De Qu Kuai Lian Shang Wan Cheng Da Bu Fen AI dApp业务逻辑。

PoS: 对于公共区块链,PoS / BFT-DPOS / DPOS是新的挑战者。 PoS被证明更加节能。 在这些算法下,Zai Xiang Ying Qu Kuai Lian Ping Tai Shang Chi You Tong Zheng De Yong Hu Neng Gou Xuan Ze Kuai Sheng Cheng Qi Bing Sheng Cheng Kuai . Wo Men De AI dApp平台将能够安全地与这些公共区块链进行通信,Bing Shi Xian Wo Men Suo You De Ye Wu Luo Ji .

SIEVE: 对于获得许可的私有区块链,Wei Rao Gong Shi Suan Fa You Xu Duo Chuang Xin . Qi Zhong [?] Ge Zui You Qu De Shi SIEVE。 简而言之,SIEVE通过将理论执行和检查阶段包括在内来扩展第一个PBFT计算:1)识别和筛选可能的非确定性请求并建立进入PBFT 3阶段断言协议的交换的确定性,Yi Ji 2)启用协议 尽管经典PBFT提供的信息状态符合协议,Dan Reng Yao Ji Xu Yun Xing Yan Zheng Qi De Shou Yi Tiao Jian . SIEVE分别来自PBFT(由[Aublin et al。,TOCS'15]中描述的思想推动)Tong Guo Zhong Fu Shi Yong PBFT看到改变惯例来降低其多方面的性质并且放弃执行另一个协议惯例的准备。通过性质 允许的私有区块链,Wo Men De Xi Tong Neng Gou An Quan Di Lian Jie Zhe Xie Qu Kuai Lian Yu Shi Shi De Fang Wen Kong Zhi .

Quantum resistance: 后量子密码术,Ye Cheng Wei Liang Zi An Quan Mi Ma Zhu ,Ke Yi Jing Shou Liang Zi PC的攻击。 鉴于特定算术领域中的麻烦问题,Zhe Chong Jia Mi Chuang Xin De Gai Jin Cai Qu Geng Xi Guan De Fang Shi . Tong Guo Diao Cha He Chuang Jian Ji Suan ,Hou Liang Zi An Quan Jia Mi Chuang Xin Ke Yi Zai Xi Tong Zhong Lian Jie ,Geng Zhong Yao De Shi ,Ti Gong Zui Yi Chang De Xin Xi An Quan Xing . Wo Men Qi Dai Kan Dao Zhe Chong Suan Fa De Ji Dong Ren Xin De Yong Li Zai Ge Chong Gong Gong /Si Ren Qu Kuai Lian Shang Fu Zhu

Shi Jian .

4.e. 可维护性

可扩展性: 现代软件的力量是建立在彼此之上的。 通过对第三方库和软件包的充满活力的支持,Kai Fa Ren Yuan Neng Gou Yi Geng Kuai De Su Du Kai Fa Ruan Jian . Wo Men Jiang Ti Gong [?] Zu Ku Yi Lai Guan Xi Guan Li Gong Ju ,Yi Yun Xu Kai Fa Ren Yuan Qing Song Diao Yong He Guan Li Yi Fa Bu De Zhi Neng He Yue Huo Jiang Qi Ta He Tong Zuo Wei Ku Yi Lai Xiang Dao Ru .

可读性: 我们将转换/解析给定区块链上的所有交易,Bing Yun Xu Yong Hu Li Jie Wo Men Suo You De Xi Tong AI dApps。 用户将能够看到谁创建AI智能合约,Zhi Neng He Yue De Zhu Yao Jiao Hu ,Zi Dong Zhi Neng He Yue abi产生等。

可探索性: 我们应该能够探索整个区块链。 我们将提供有关区块链上最近开采的区块的见解。 我们还将提供已经开采并且当前附加到区块链网络的任何块中的任何交易的见解。 用户还可以查看任何公共地址的历史记录并审核余额,Jiao Yi Li Shi Ji Lu Deng .

4.f. 响应速度

快速反应: 卓越的客户体验需要可在几秒钟内获得即时反馈。 如果应用程序需要更长的响应时间,Bing Qie Wu Xiang Ying De AI dApp导致竞争情况低于传统解决方案,Ze Xi Tong Jiang Hui Beng Kui . Wo Men De gCompute平台将尽可能地优化以减少应用程序响应时间。

网络优化: 我们将尽量减少网络中的通信延迟。 网络优化应该有能力保证系统资源的理想使用,Ti Gao Ying Li Neng Li Yi Ji She Qu De You Xiao Xing . Wang Luo You Hua Hui Cha Kan Dao Fu Wu Qi De Ge Ge Gong Zuo Zhan Yi Ji Yu Zhi Xiang Guan De Gong Ju He Lian Jie . Wo Men De Xi Tong Ke Yi Bao Gua Liu Liang Zheng Xing ,Rong Yu Shu Ju Xiao Chu ,Shu Ju Huan Cun He Shu Ju Ya Suo Yi Ji Shu Ju Xie Yi De Jian Hua .

4.g. 高并发性

同步执行: 如果dApp本身需要顺序独立的步骤,Wo Men Ye Ying Gai Zhi Chi Kuai Su Shun Xu Zhi Xing .

异步执行: 即使计算机以顺序模式运行,Bing Shi Yong Suo He Shang Xia Wen Qie Huan De Zu He Lai Shi Xian Wei Yi Bu Zhi Xing ,Xian Dai Ying Yong Cheng Xu Ye Mo Ren Zhi Chi Bing Fa Zhi Xing . Kao Lu Dao Wan Zheng De Hu Lian Wang Gui Mo De AI dApp解决方案,Wo Men Jiang Jin Li You Hua Wo Men De Dai Ma Yi Shi Xian Yi Bu Zhi Xing .

References