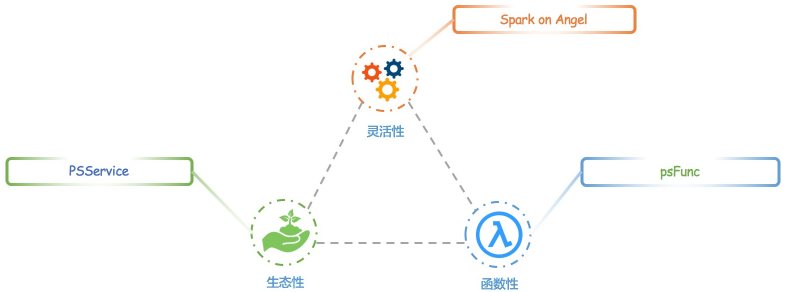
Angel

——A Flexible and Powerful Parameter Server

Tencent
Andymhuang（黄明）

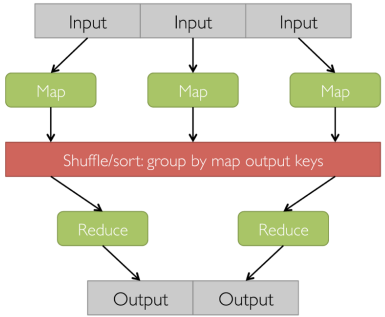
- 一个基于参数服务器（Parameter Server）理念的分布式机器学习平台



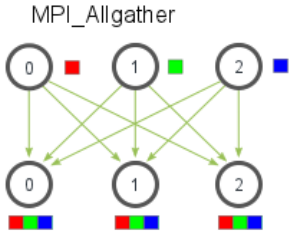
https://github.com/tencent/angel

目录

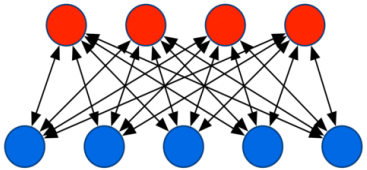
分布式计算的三种范式



MapReduce

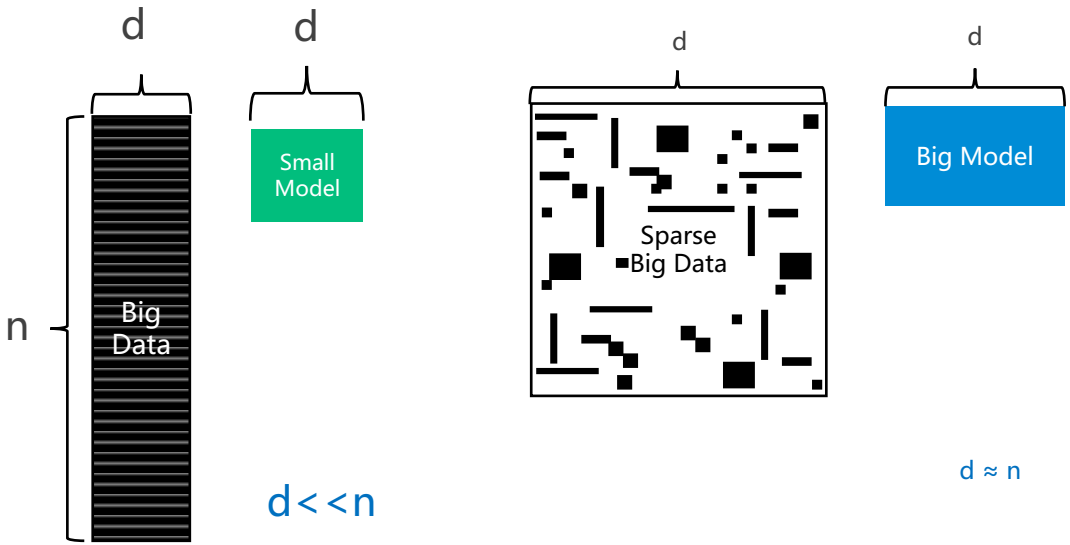


MPI



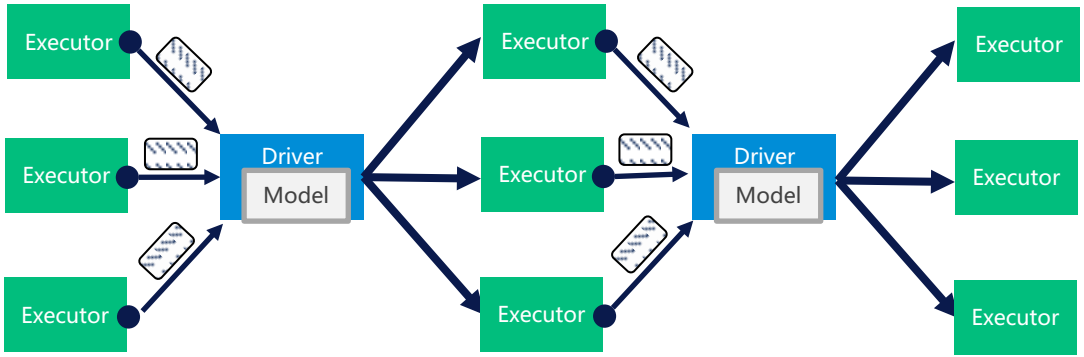
Parameter Server

腾讯的现网需求



寻找满足十亿级维度的工业级的分布式机器学习平台

Spark机器学习的瓶颈



- Driver成为参数汇总的单点瓶颈，难以支撑大规模模型及数据，
- 十亿级维度的模型训练，实际应用中降维处理
- Executor之间相互等待，整体效率不高

其它机器学习平台

GraphLab

苹果收购了

PETUUM

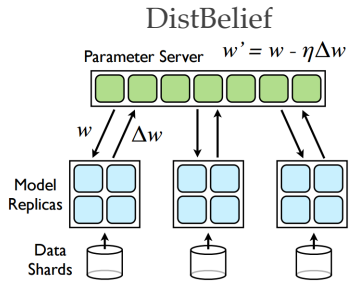
实验室级别，而且融资后不开源了

DMTK

转支持深度学习CNTK

dmlc XGBoost

针对性太强



转深度学习TensorFlow

Angel

开始研发
•2015

投入生产
•2016

正式开源 V1.0.0
•2017

- 能支持十亿级别维度的模型训练
- 基于Matrix/Vector的模型自动切分和管理，兼顾稀疏和稠密两种格式
- 提供多种同步控制机制（BSP/SSP/ASP）

工业级别可用的参数服务器

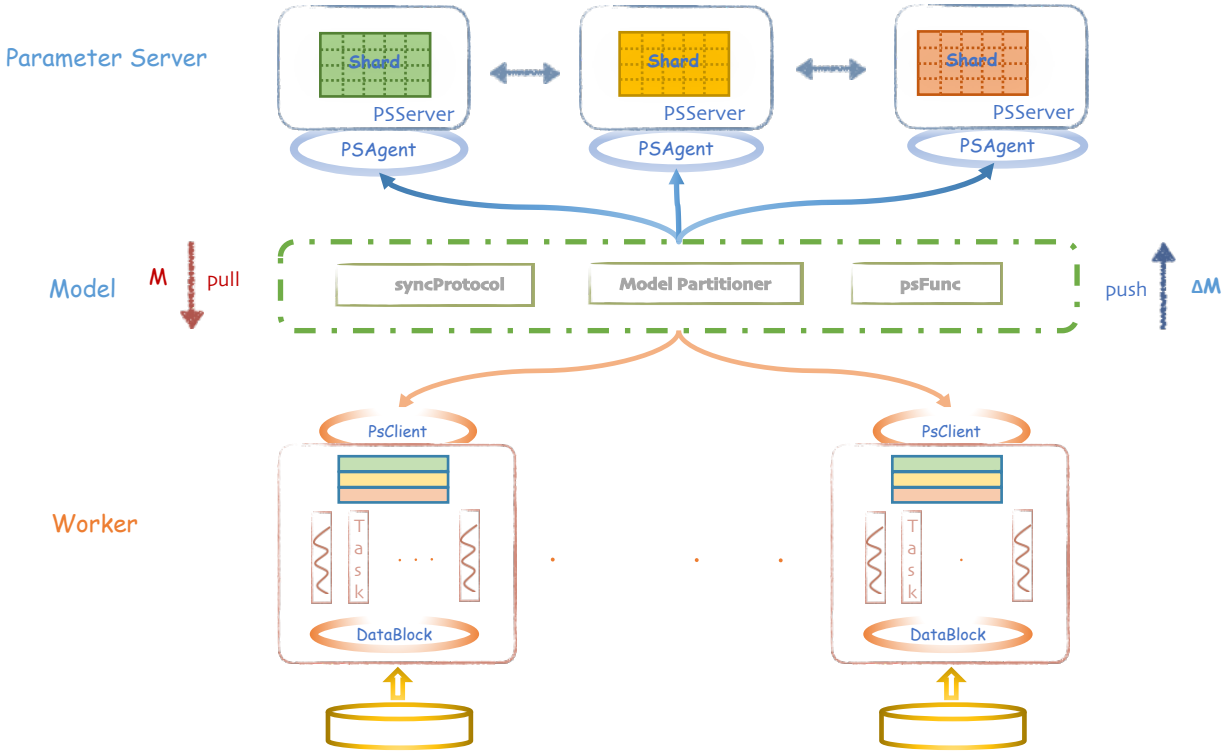
丰富的机器学习及数学计算库

- 集成LR（ADMM-LR），SVM，KMeans，LDA，MF，GBDT等机器学习算法
- 多种优化方法，包括ADMM，OWLQN， LBFGS和GD
- 支持多种损失函数、评估指标，包含L1、L2正则项算法

- 基于PSModel的机器学习友好接口，以Model为核心编程
- 支持Spark on Angel，Spark代码小量改动就可以运行Angel之上
- 灵活的psFunc，便于复杂算法的开发，实现模型并行

友好的用户编程接口

Angel的系统架构


Parameter Server
Shard
PSServer
PSAgent
Shard
PSServer
PSAgent
Shard
PSServer
PSAgent
Model
M
pull
syncProtocol
Model Partitioner
psFunc
push
ΔM
Worker
PsClient
Task
DataBlock
PsClient
Task
DataBlock

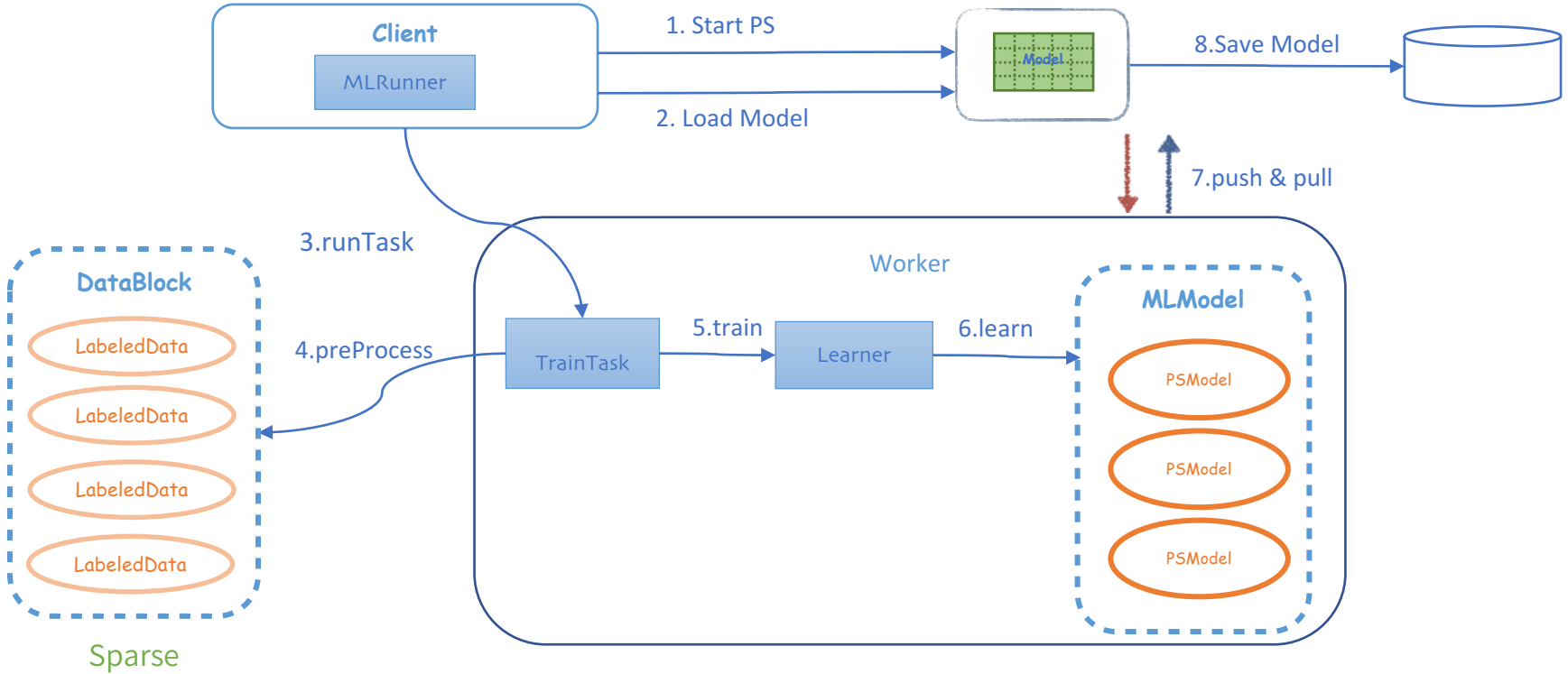
核心抽象

Mapper
Reducer

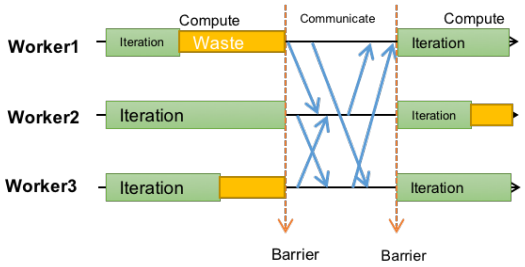
RDD

PSModel

Angel的运行机制

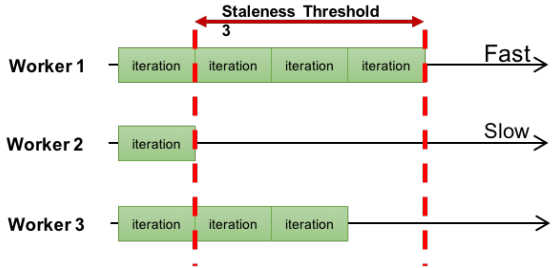

Client
MLRunner
1. Start PS
2. Load Model
Model
8.Save Model
7.push & pull
3.runTask
DataBlock
LabeledData
LabeledData
LabeledData
LabeledData
Sparse
4.preProcess
Worker
TrainTask
5.train
Learner
6.learn
MLModel
PSModel
PSModel
PSModel

同步控制（Sync Controller）



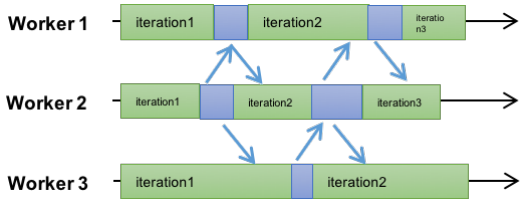
BSP

适用范围广，但等待时间长
angel.staleness = 0



SSP

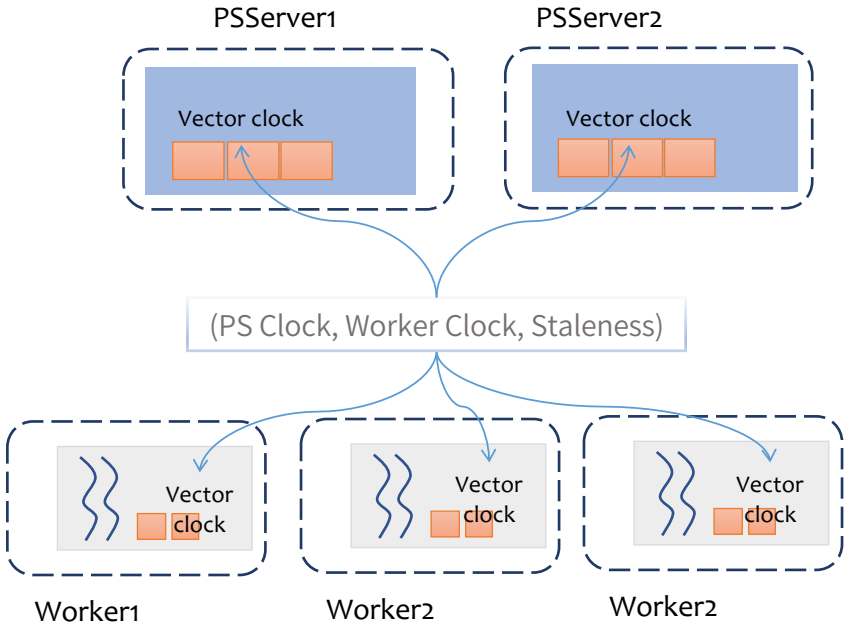
等待时间较短，但需要更多迭代
angel.staleness=N



ASP

无等待时间较短，收敛无保证
angel.staleness=-1

同步控制（Sync Controller）



向量时钟

- 在Server端为每个分区维护一个向量时钟，记录每个worker在该分区的时钟信息
- 在Worker端维护一个后台同步线程，用于同步所有分区的时钟信息
- Task在对PSModel进行Get或其他读取操作时，根据本地时钟信息和staleness进行判断，选择是否进行等待操作
- 每次迭代完，调用Clock方法，更新向量时钟

模型分区（Model Partitioner）

定义

- 将模型切分成多个部分，存储在不同PS节点上

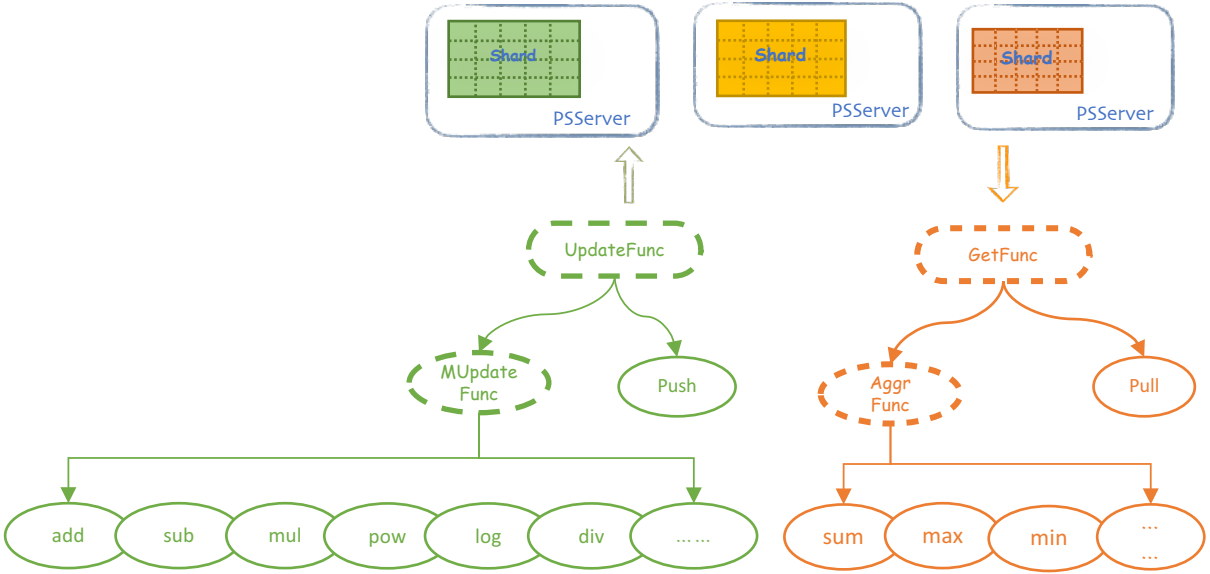
优点

- 保证PS负载均衡
- 降低PS单点性能瓶颈
- 关联的数据在同一个PS上

Angel的模型分区

- 默认将模型分成大小相等的块
- 可以指定分区块大小
- 支持横切和纵切
- 自定义矩阵分区，量身定制区块分布方式

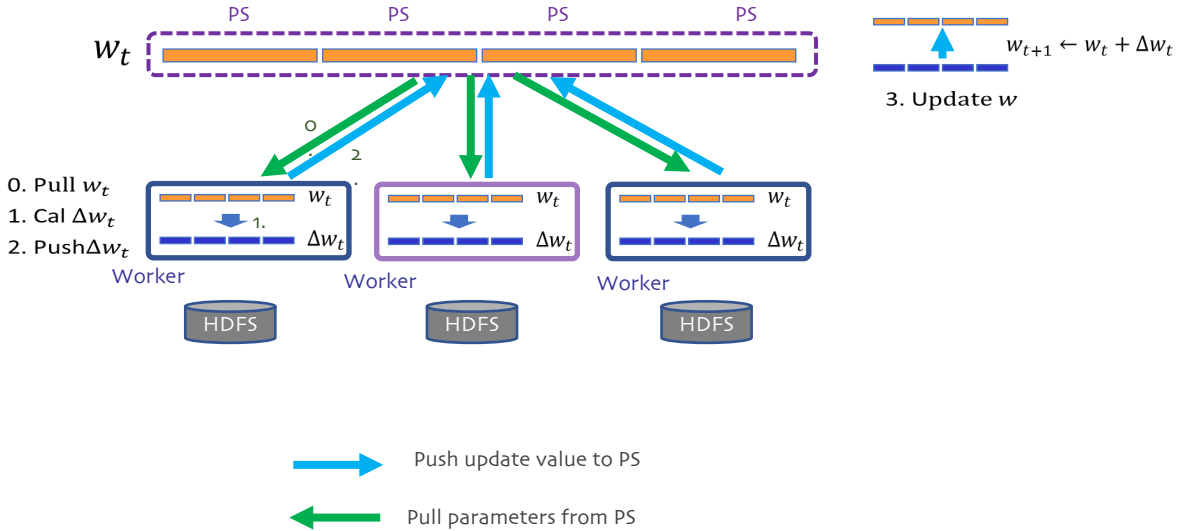
自定义函数（psFunc）



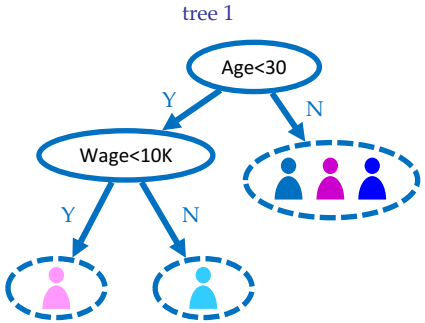
Algorithms on Angel

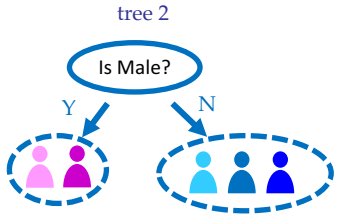
LR on Angel

Step 0: Worker从PS获得参数W_t
Step 1: Worker计算参数的更新值ΔW_t
Step 2: Worker把ΔW_t推送给PS
Step 3: PS更新参数（$W_{t+1} \leftarrow W_t + \Delta W_t$）



GBDT：树模型+Boosting





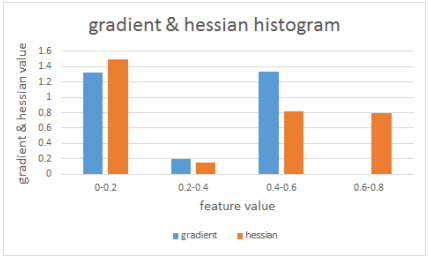
A predict( ) 5+0.5=5.5

B predict( ) 10+1.5=11.5

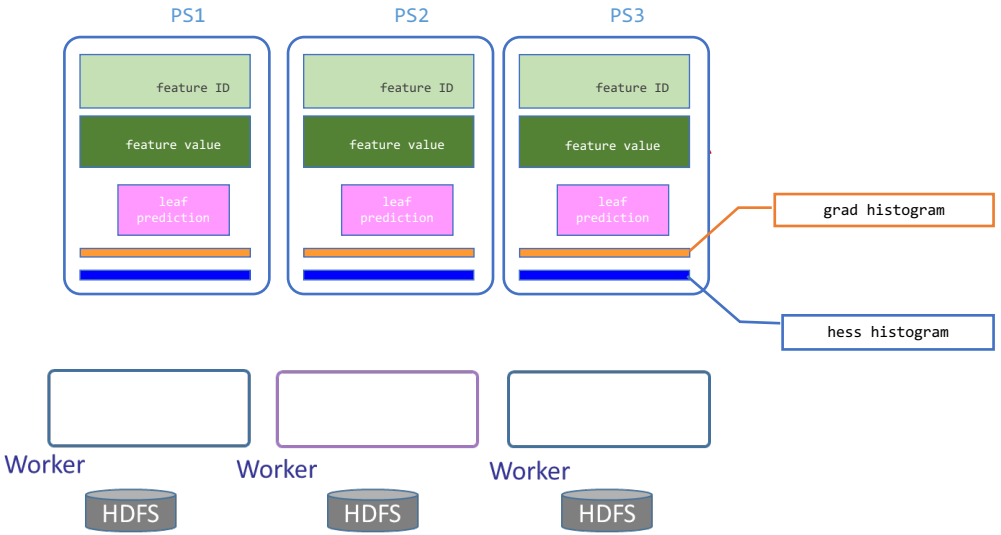
C predict( ) 1+1.5=2.5

D predict( ) 1+0.5=1.5

E predict( ) 1+1.5=2.5

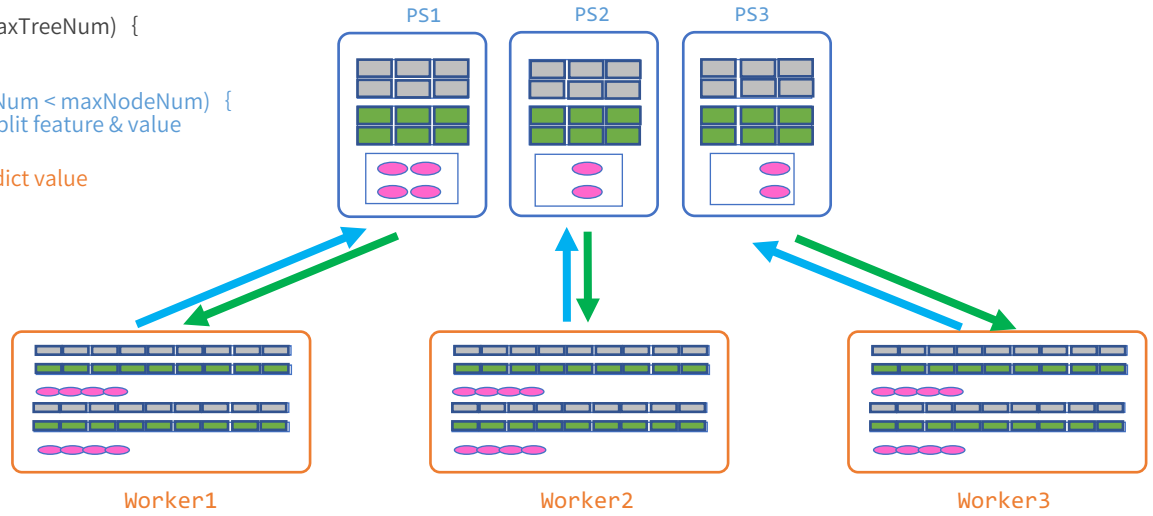


GBDT on Angel: 模型存储



GBDT on Angel（1）：构建森林

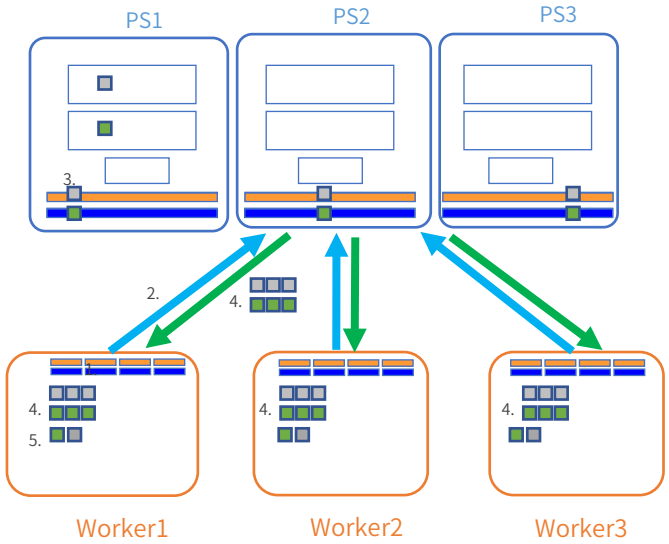
```
while (treeNum < maxTreeNum)  {
   create a new tree
          while (nodeNum < maxNodeNum)  {
                 find split feature & value
          }
    calculate leaf predict value
   finish a tree
}
```



GBDT on Angel（2）：分裂树节点

find split feature & value

1. worker计算梯度直方图（一阶&二阶）
2. worker推送梯度直方图到PS
3. PS计算局部最佳分裂点
4. worker从PS拉取P个局部最佳分裂点
5. 计算出全局最佳分裂点，创建树节点



细节文档和代码

中文

- https://github.com/tencent/angel/blob/master/docs/algo/gbdt_on_angel.md

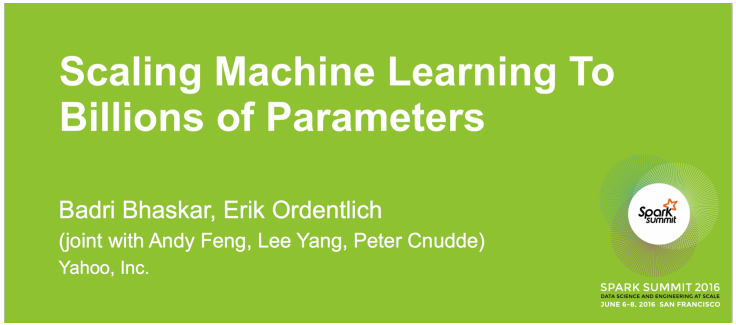
English

- https://github.com/tencent/angel/blob/master/docs/algo/gbdt_on_angel_en.md

GBDTLearner.scala

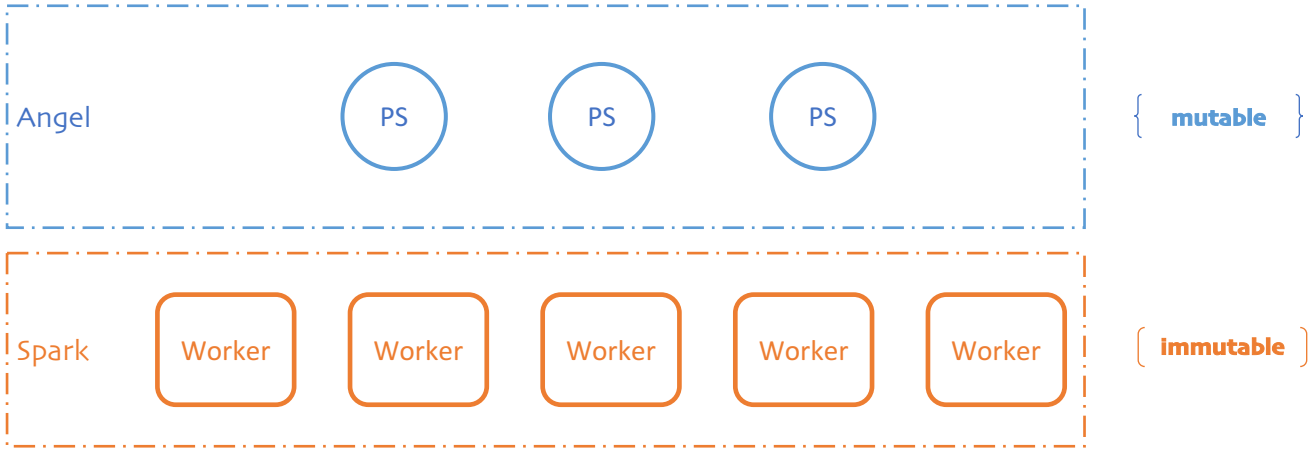
Spark on Angel

Spark on PS的回顾



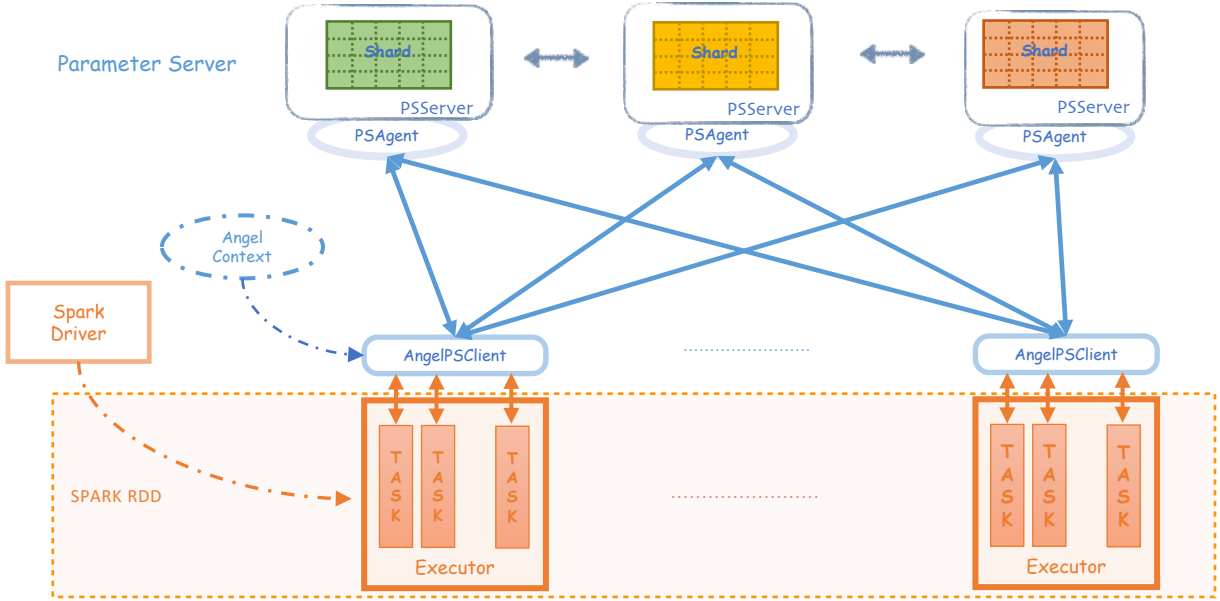
https://issues.apache.org/jira/browse/SPARK-6932

Spark on PS的基本理念



1. 分离系统中的变和不变
2. 以少博多
3. 降低对Spark Core的侵入性

Spark on Angel的架构

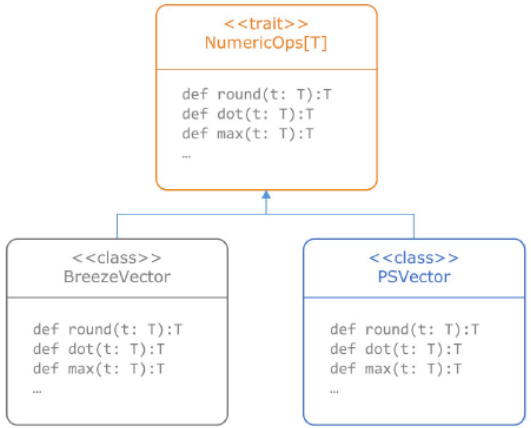


Spark on Angel的基础写法

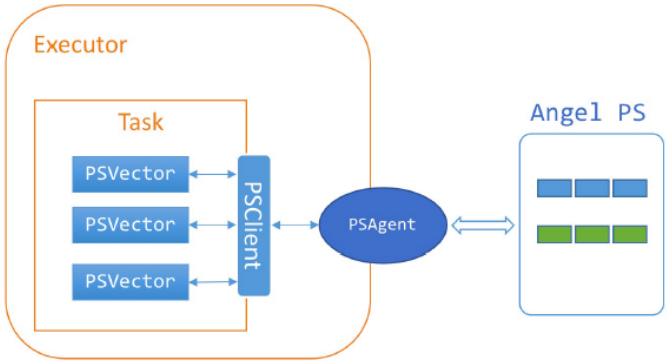
```
val psContext = PSContext.getOrCreate(spark.sparkContext)
val pool = psContext.createModelPool(dim, capacity)
val psVector = pool.createModel(0.0)
rdd.map { case (label , feature) =>
        psVector.increment(feature)
        ...
}
println("feature sum size:" + psVector.mkRemote.size())
```

- 启动SparkSession
  - 初始化PSContext，启动Angel的PSServer
  - 创建PSModelPool, 申请到PSVector
  - **核心调用**：在RDD的运算中，直接调用PSVector，进行模型更新
  - 终止PSContext
- 停止SparkSession

Vector的透明替换



混入相同特征



透明替换

- 将BreezeVector透明替换为PsVector
- 适用于MLLib大部分算法
- 替代成本非常低

Spark on Angel的进阶写法

- Spark

```
def runOWLQN(trainData: RDD[(Vector, Double)], dim: Int, m: Int, maxIter: Int): Unit = {

    val initWeight = new DenseVector[Double](dim)
    val l1reg = 0.0
    val owlqn = new BrzOWLQN[Int, DenseVector[Double]](maxIter, m, 0.0, 1e-5)

    val states = owlqn.iterations(CostFunc(trainData), initWeight)
    ……

}
```

- Spark on Angel

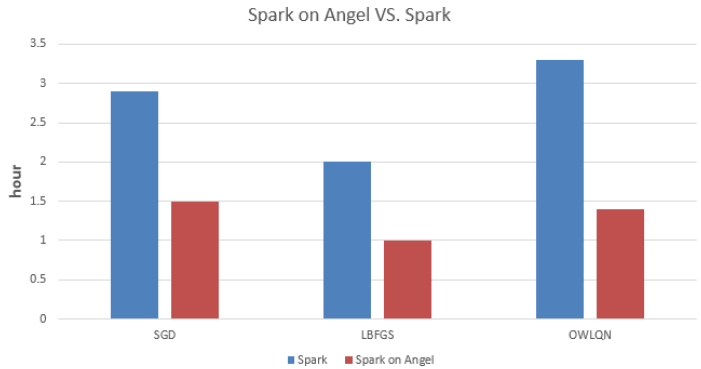
```
def runOWLQN(trainData: RDD[(Vector, Double)], dim: Int, m: Int, maxIter: Int): Unit = {

    val pool = PSContext.createModelPool(dim, 20)

    val initWeightPS = pool.createZero().mkBreeze()
    val l1regPS =  pool.createZero().mkBreeze()

    val owlqn = new OWLQN(maxIter, m, l1regPS, tol)
    val states = owlqn.iterations(CostFunc(trainData), initWeightPS)
    ……

}
```

性能比对

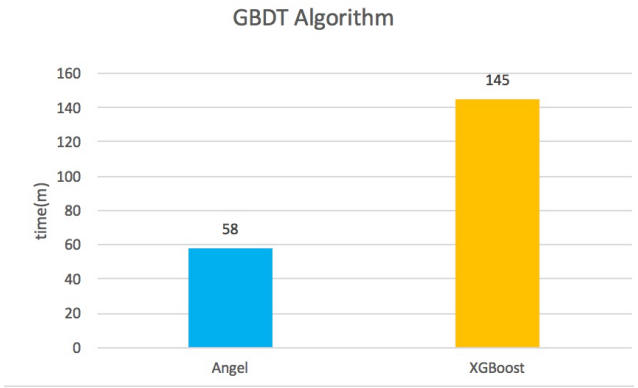
——生产数据，现网环境，尽量公平

Spark on Angel vs Spark —— LR

	Spark	Spark on Angel	加速比例
SGD LR (stepSize=0.05,maxIter=100)	2.9 hour	1.5 hour	48.3%
L-BFGS LR (m=10, maxIter=50)	2 hour	1 hour	50.0%
OWL-QN LR (m=10, maxIter=50)	3.3 hour	1.4 hour	57.6%



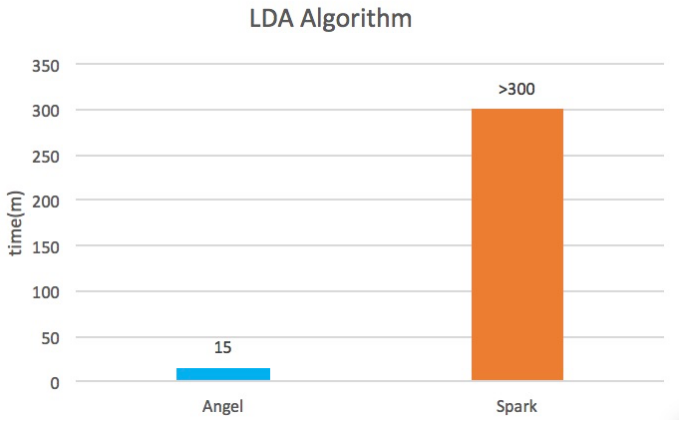
Angel vs XGBoost —— GBDT



框架	Worker	PS	建立20棵树时间
Angel	50 个(内存：10G / Worker)	10个 (内存：10G / PS)	58 min
XGBoost	50个 (内存：10G / Worker)	N/A	2h 25 min

数据：腾讯内部某性别预测数据集，3.3×10^5 特征，1.2×10^8 样本

Angel vs Spark —— LDA



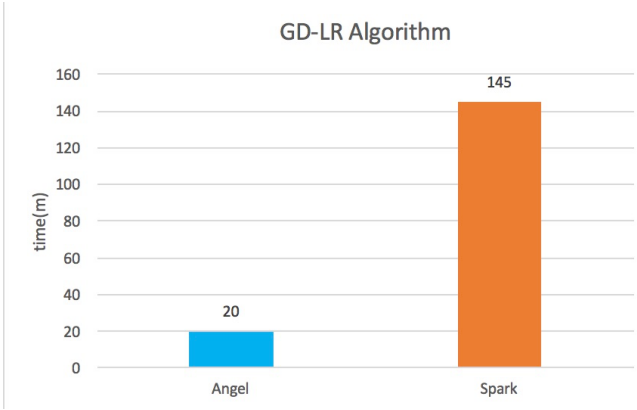
框架	Worker	PS	时间
Angel	20个(内存：8G/Worker)	20个(内存：4G/PS)	15min
Spark	20个(内存：20G/Worker)	N/A	>300min

数据：PubMED

框架	Worker	PS	时间
Angel	50个(内存：10G/Worker)	50个(内存：4G/PS)	1h 7min

DataSet：40G Token： 2 billion
Word：52w Topic： 1000

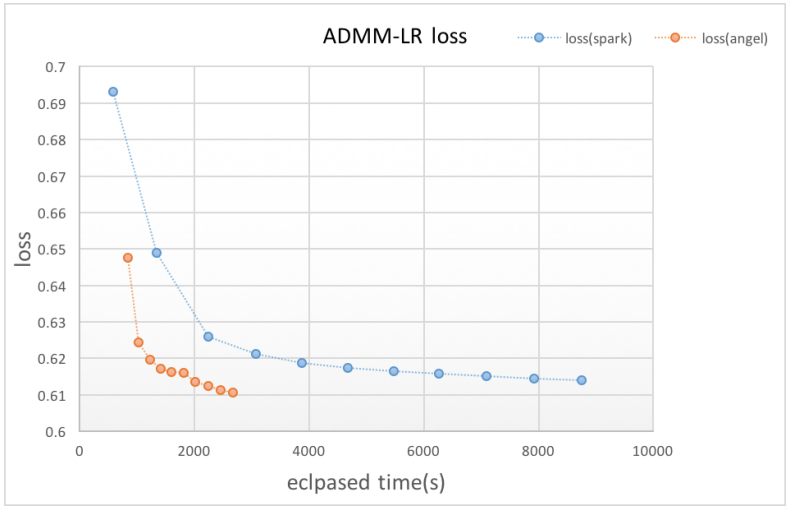
Angel vs Spark —— GD-LR



框架	Worker	PS	迭代100次时间
Angel	50个(内存:10G/Worker)	20个(内存: 5G/PS)	20min
Spark	50个(内存:14G/Worker)	N/A	145min

数据：腾讯内部某推荐数据，5×10^7 特征，8×10^7 样本

Angel vs Spark —— ADMM-LR



框架	Worker	PS	收敛退出
Angel	100个(内存:10G/Worker)	50个(内存：5G/PS)	27 min
Spark	200个(内存:20G/Worker)	N/A	145 min

数据：腾讯内部某推荐数据，5千万特征，1亿样本

开源和展望

OpenSource & Perspective

Angel开源

github:issues

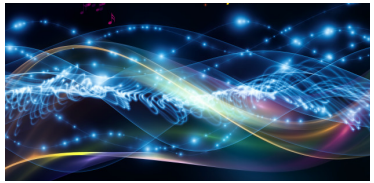
- LightBGM作者：[GBDT] The purposes of using parameter server in GBDT #7
- 海外华人：English translation of documents #95
- 华为工程师：[WIP]Upgrade the netty version of RPC to 4.x #94
- ……

SIGMOD/PODS 2017
Chicago, Illinois, USA, May 14-19

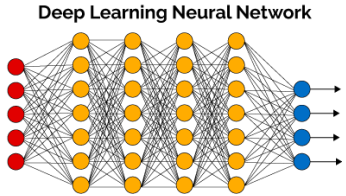
- Heterogeneity-aware Distributed Parameter Servers. SIGMOD, 2017
- ……

下一个版本（What is Next）

Python API



Spark Streaming on Angel



Deep Learning Framework Support

Q & A

微博：@明风

https://github.com/tencent/angel