阿里巴巴大数据竞赛分享

Zakklars
清水湾沙滩青年流浪者队

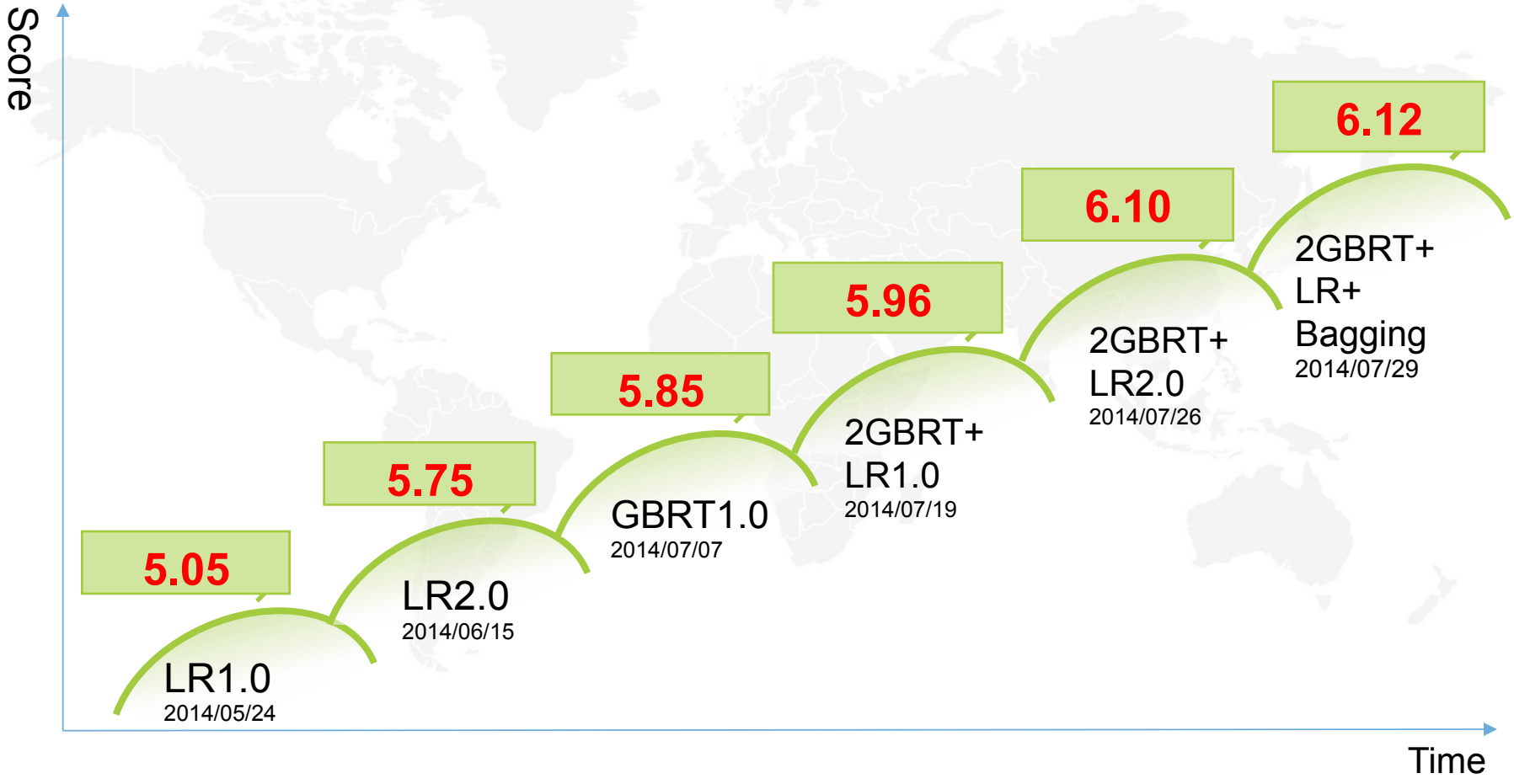
提纲

- 队伍及队员介绍
- 里程碑回顾
- 算法介绍
- 算法改进过程与分析
- 经验总结与建议

队伍及队员介绍

- **队员**
  - Zakklars
- **研究方向**
  - 数据挖掘与机器学习
- **感兴趣方向**
  - 数据挖掘与机器学习在广告,推荐,搜索,LBS,互联网金融等互联网领域的应用

里程碑回顾


Score
5.05
LR1.0
2014/05/24
5.75
LR2.0
2014/06/15
5.85
GBRT1.0
2014/07/07
5.96
2GBRT+
LR1.0
2014/07/19
6.10
2GBRT+
LR2.0
2014/07/26
6.12
2GBRT+
LR+
Bagging
2014/07/29
Time

提纲

- 队伍及队员介绍
- 里程碑回顾
- 算法介绍
- 算法改进过程与分析
- 经验总结与建议

整体思路

- 任务
  - 监督型二分类→概率预测(或排序)→机器学习
- 数据
  - 样本如何得来? 什么数据作为特征? 什么数据作为标记?
  - 如何划分训练集,验证集和测试集?
- 特征
  - 从"特征数据"中如何抽取转换成某个机器学习模型适用的"特征信息”?
  - 业务理解和模型原理的双重结合
- 模型
  - 选择合适的模型
  - 选择合适的参数
  - 不同模型的结果融合

数据

- 样本定义
  - 每个<user_id,brand_id> pair标识一条样本,表示当前样本的用户,品牌.我们的任务是根据该用户以及该品牌过去的历史数据中抽取特征,预测该用户未来一个月是否会购买该品牌
- 数据集划分
  - 线下训练集,线下验证集,线上测试集

数据

- **数据抽样和过滤**
  - 未交互过滤
    - 在历史信息中没有任何交互的样本线上和线下均丢弃
  - 训练负样本下抽样
    - 负样本不放回抽样10%或20%,抽样之后样本数量在1000万到3000万之间
  - 训练正样本过抽样
    - 直接翻倍正样本使正负样本比例1:10左右
  - 抽样8%的样本作为验证集,不用做下抽样和过抽样

特征(业务理解)

特征类别	特征描述
当前用户总体特征	用户最近7/15/30天有多少天有交互/购买任意品牌
	用户最近一次/第一次交互/购买任意品牌的时间
	用户最近1/3/7/10/15/30天购买/点击所有品牌的总次数
	用户最近1/3/7/10/15/30天购买/点击过的不同品牌的个数
当前品牌总体特征	品牌最近第1/2/3个月以及最近7/15天被多少不同用户交互/购买
	品牌总共有多少人购买
	品牌总共有多少人在不同天重复购买
	品牌总共有多少人在不同周重复购买
	品牌第一次/最后一次被任意用户交互/购买的时间

特征(业务理解)

特征类别	特征描述
品牌竞争	当前用户与当前品牌的最后一个交互天的前一天,当天和后一天这3天中一共交互了多少个不同其他品牌
	用户最近1/2/3/4/5/6/7/8/9/10/12/14/16/18/20/25/30天有多少单天/三天/单周/十天/双周对当前品牌存在纯交互日
	用户最后一次交互当前品牌距离用户最后一次交互任意品牌相隔多少天
	当前用户与当前品牌的最后一个交互天的前一天,当天和后一天这3天中一共点击/购买/收藏了多少次其他品牌
当前用户当前品牌特征	用户交互当前品牌的最近/第2近/第3近/第4近/第5近/第6近/第7近/第8近的交互天的点击数/购买数/收藏数以及时间
	用户交互当前品牌的最远一个交互天的点击数/购买数/收藏数以及时间
	用户第一次交互/购买当前品牌和最后一次交互当/购买前品牌相隔多少天
	用户最近1/2/3/4/5/6/7/8/9/10/12/14/16/18/20/25/30天有多少单天/三天/单周/十天/双周有交互/购买当前品牌以及交互/购买/收藏的总次数
	用户30天/60天以前有多少单天/三天/单周/十天/双周有交互/购买当前品牌以及交互/购买/收藏的总次数

特征(模型适用)

- LR特殊处理
  - 所有特征都进行Dummy Coding

    基本思想: 将有多个取值的1个特征转换成多个取值为0,1的特征

    举例: 以一个用户对一个品牌有交互的天数nday(nday<8)为例,假设一条样本n_day=5,那么dummy之后变为8个特征,其含义和取值如下:

nday=1	nday=2	nday=3	nday=4	nday=5	nday=6	nday=7	nday=8
0	0	0	0	1	0	0	0

变种: 对数值型特征,基于大于等于或者小于等于的包含式dummy

nday>=1	nday>=2	nday>=3	nday>=4	nday>=5	nday>=6	nday>=7	nday>=8
0	0	0	0	1	1	1	1

特征(模型适用)

- LR特殊处理
  - 使用当前样本要预测的品牌id作为特征
  - 使用conjunction特征(两个或多个binary特征的乘积作为一个新的binary特征），主要以品牌id和其他数值型特征conjunction为主
  - 使当前品牌id和其他品牌id进行conjunction

    举例:

    样本当前要预测的品牌id和当前用户有过点击/购买的品牌id的conjunction

    样本当前要预测的品牌id和当前用户7天内有过购买的品牌id的conjunction
  - 将出现次数极少的特征直接删掉

特征(模型适用)

- LR为什么要用dummy coding?

  实现单特征的非线性权重，以及为了使用非数值类特征（比如品牌id）

- LR为什么要用包含式的dummy coding?

  解决特征值长尾部分数据稀疏的问题

- LR为什么要用品牌id特征?

  品牌完全个性化，但又不是不同品牌完全独立训练,**效果提升非常明显**

- LR为什么要用dummy conjunction?

  依旧是为了拟合二次或高次非线性，实现类似树模型的特征与特征之间的interaction,**效果提升非常明显**

- LR为什么要用品牌id和品牌id的conjunction?

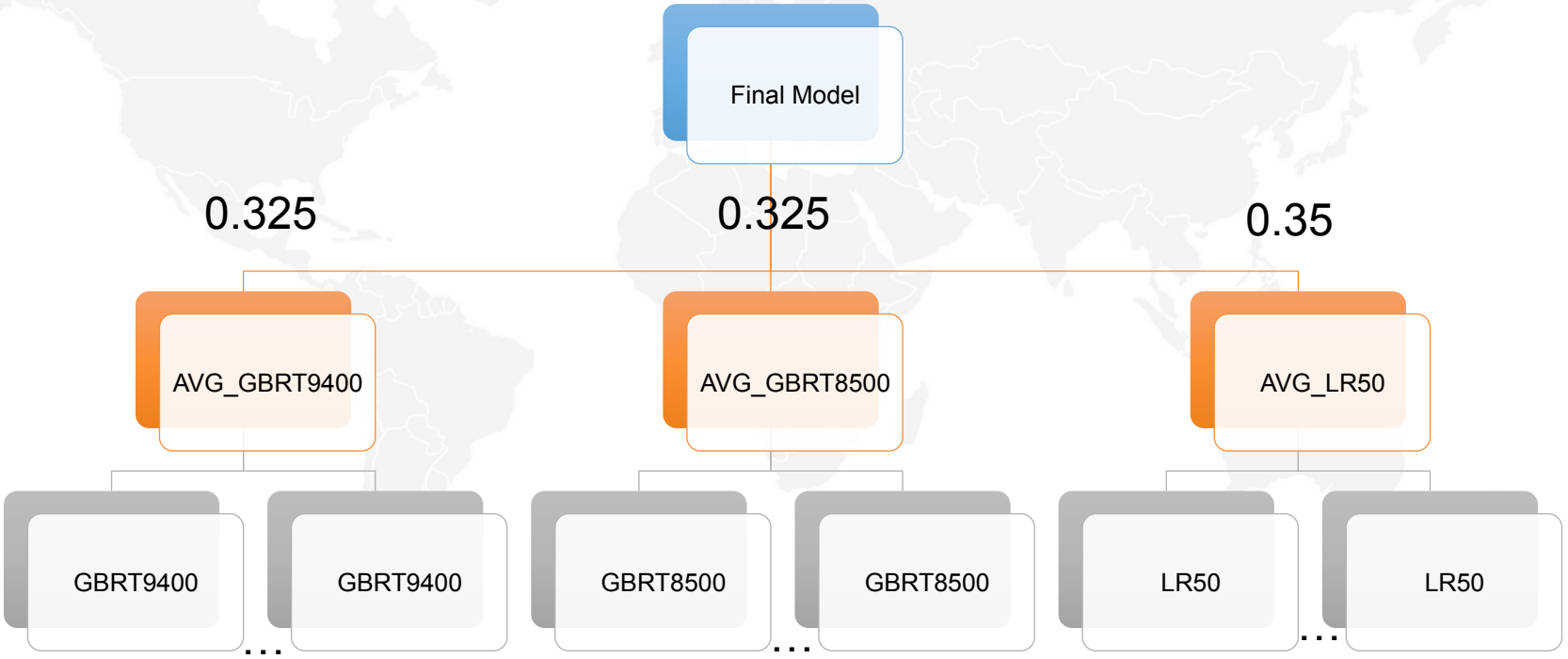
  实现关联规则，但和纯关联规则算法不同，这里依然融合了其他重要特征

模型

- 单模型
  - GBRT9400
    - 1500万样本,正负比1:10,300个特征,深度9,学习率0.1,迭代400次
  - GBRT8500
    - 1500万样本,正负比1:10,300个特征,深度8,学习率0.1,迭代500次
  - LR50
    - 3000万样本,正负比1:10,130万特征,L1正则,正则项系数50,迭代1000次

模型

- 模型融合


Final Model
0.325
0.325
0.35
AVG_GBRT9400
AVG_GBRT8500
AVG_LR50
GBRT9400
GBRT9400
GBRT8500
GBRT8500
LR50
LR50

模型

- 为什么选用如此简单的模型和如此简单的模型融合算法?
  - 线上可用性高，特别是LR，训练和预测均较快，并且能实现增量式的online learning
  - 线性模型和树模型预测结果差异大，单凭线性融合效果提升就已经非常显著
  - 线性融合效果较稳定，调参方便，不容易过拟合，通用性强

算法特色与优势

- 简单是美

  简单→ 高效,通用,稳定

换数据之前从未进入过前10

换数据之后从未跌出过前10

算法特色与优势

- Logistic Regression专项击破

为数不多的使用纯dummy特征的队伍之一，LR分数可以达到将近5.80，是TOP10参赛队中最高的

LR全部使用dummy特征

- 解决数值特征的非线性问题以及数值特征无法与非数值特征进行conjunction的问题

LR对数值特征采用包含式的dummy

- 配合L1正则化,实现了hierarchical smoothing,解决特征值长尾稀疏的问题

LR使用品牌id与品牌id之间的conjunction dummy特征

- 实现了二项关联规则与LR的融合

提纲

- 队伍及队员介绍
- 里程碑回顾
- 算法介绍
- 算法改进过程与分析
- 经验总结与建议

算法改进过程与分析

- **尝试过觉得有用的**
  - 在LR中使用brand_id特征以及brand_id特征和其他特征进行conjunction dummy
  - 线上测试集的特征使用4个月的数据而不是3个月的数据
  - 想到可能有用的稍微有业务含义的特征就添加,哪怕不太确定,或者觉得用处不大,或者觉得和已有特征关联较大

算法改进过程与分析

- **尝试过但没有用的**
  - 在LR中使用user_id特征
  - 在GBRT中尝试对未购买但有点击的样本赋予一个介于0～1之间的label
  - 在LR中对于未购买但有点击的样本降低样本权重
  - 在LR中尝试使用人工数据分片分开训练
  - 在LR中根据每月销量判断一个品牌的稳定性,对于不稳定的品牌不使用其brand_id特征

算法改进过程与分析

- **未尝试感觉有用的**
  - 用pairwise的loss来训练LR模型(相当于训练排序模型）
  - 将带brand_id conjunction brand_id的LR模型用于未交互的样本
  - 进一步细化特征到更细的时间粒度,用类似random forest的思路每次随机选取一些特征出来跑模型,最后平均融合不同模型
  - 在LR中融入factorization
  - 在LR中实现自动conjunction选择以加入更多的conjunction特征
  - 在LR中融入gradient boosting

提纲

- 队伍及队员介绍
- 里程碑回顾
- 算法介绍
- 算法改进过程与分析
- 经验总结与建议

个人经验总结

- 线上调参
  - 不要过早陷入根据线上结果调整算法,要多依赖线下结果
- 特征选择
  - 从追求分数的角度来说,不要花太多时间反复尝试人工删除特征并验证效果
- 人工规则
  - 不要过早陷入人工规则的调节
- 数据分布
  - 多关注线下和线上数据的不一致性
- 不断尝试
  - 不能光从原理上分析就拍板结论,也不能不懂原理盲目不断尝试