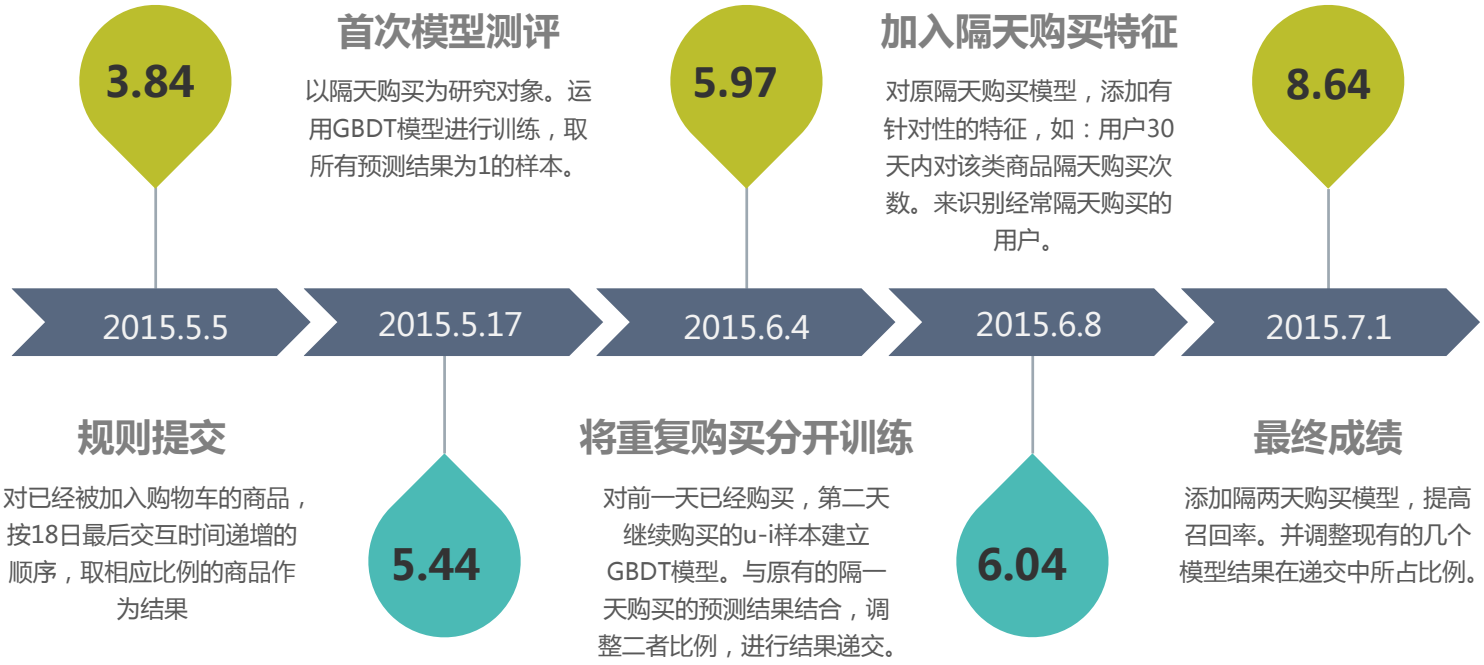
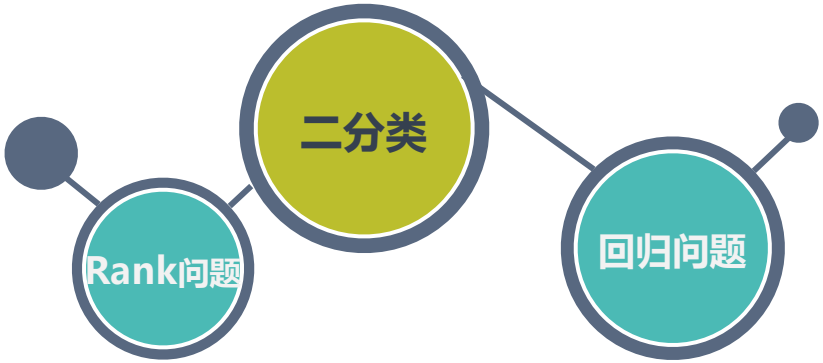

阿里移动推荐算法大赛答辩
2015天池大数据竞赛
TIANCHI天池
NEU-Smart团队

回顾


3.84
首次模型测评
以隔天购买为研究对象。运用GBDT模型进行训练，取所有预测结果为1的样本。
5.97
加入隔天购买特征
对原隔天购买模型，添加有针对性的特征，如：用户30天内对该类商品隔天购买次数。来识别经常隔天购买的用户。
8.64
2015.5.5
2015.5.17
2015.6.4
2015.6.8
2015.7.1
规则提交
对已经被加入购物车的商品，按18日最后交互时间递增的顺序，取相应比例的商品作为结果
5.44
将重复购买分开训练
对前一天已经购买，第二天继续购买的u-i样本建立GBDT模型。与原有的隔一天购买的预测结果结合，调整二者比例，进行结果递交。
6.04
最终成绩
添加隔两天购买模型，提高召回率。并调整现有的几个模型结果在递交中所占比例。

问题分析

问题归纳

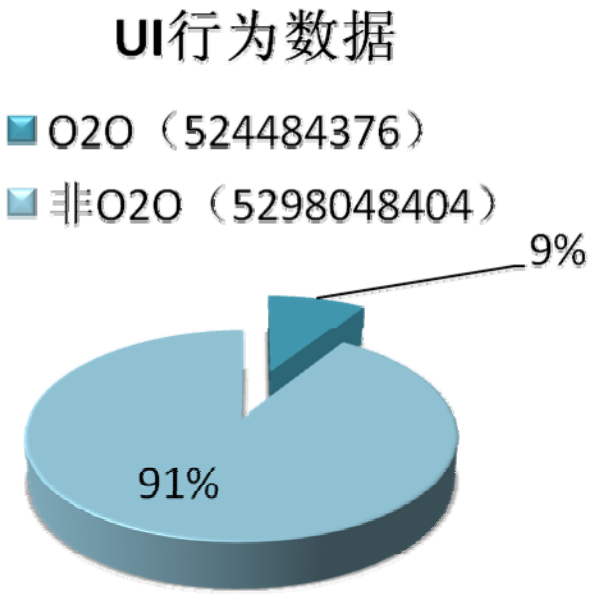


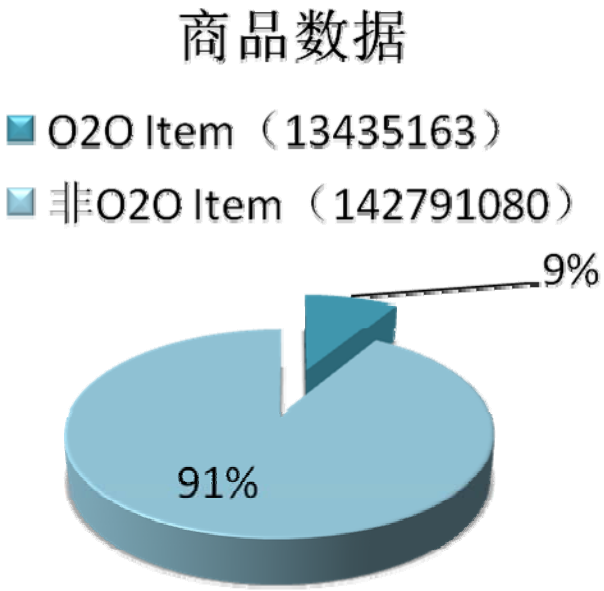
已知数据信息：

字段	字段说明	提取说明
user_id	用户标识	抽样&字段脱敏
item_id	商品标识	字段脱敏
behavior_type	**用户对商品的行为类型**	**浏览、收藏、加购物车、购买**
user_geohash	用户位置的空间标识，可以为空	由经纬度通过保密的算法生成
item_category	商品分类标识	字段脱敏
time	行为时间	精确到小时级别

任务：根据11月18日~12月18日的用户行为记录 **预测** 12月19日用户对O2O商品购买行为

数据分布

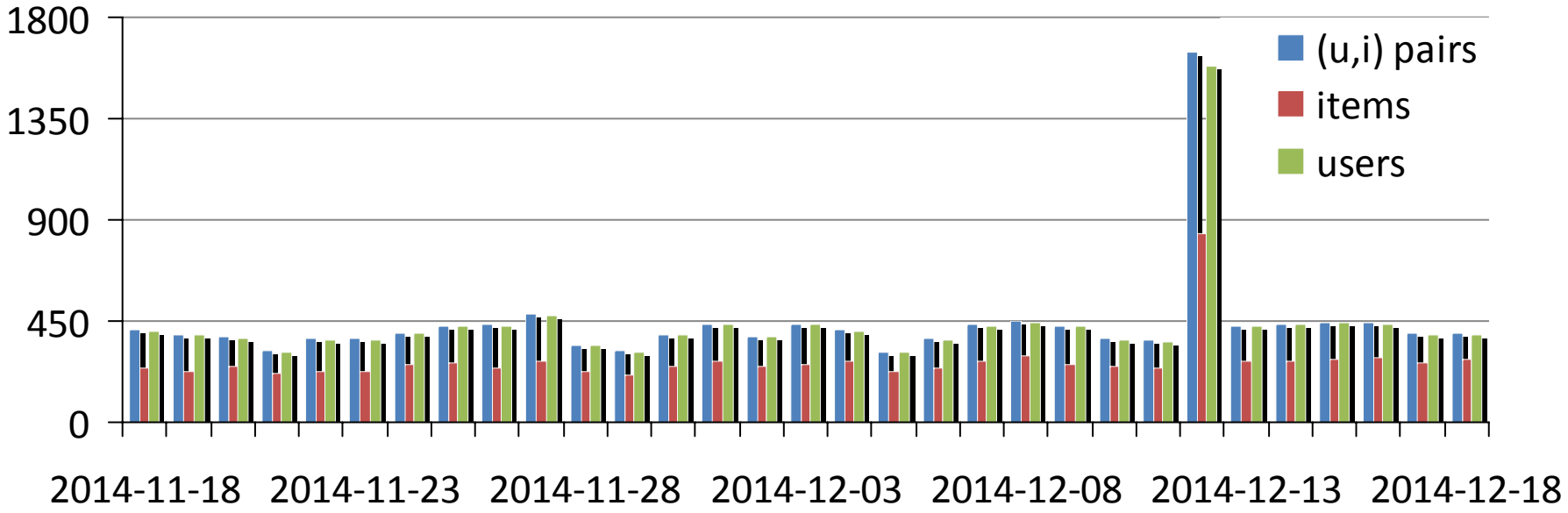




行为数据与商品 **数据分布一致**，O2O数据基本上占总体数据的9%
利用纯O2O行为数据的同时，充分考虑如何利用非O2O的行为数据

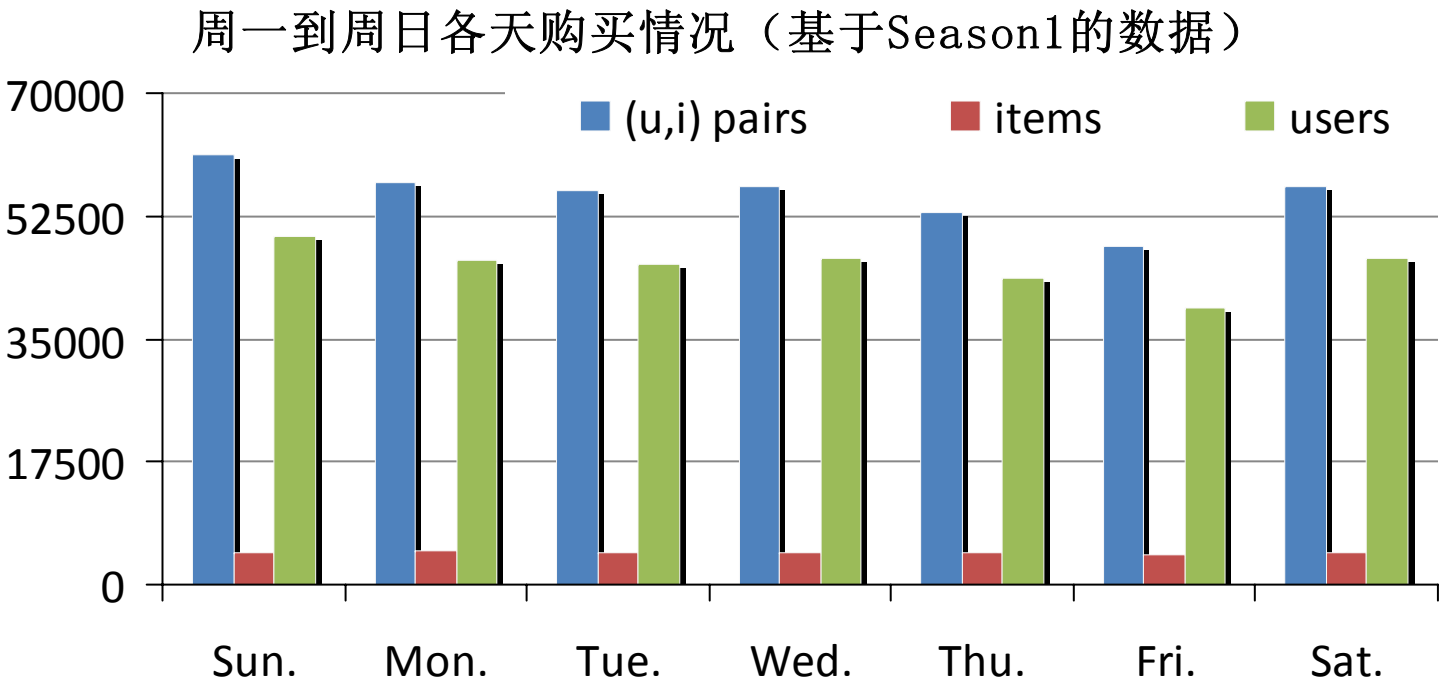
数据分析

一个月时间内各天的购买情况（基于Season1的数据）



1212销售量明显高于其他时候，**需要特殊处理！**

数据分析



相对比较平稳，总体来说预测日（周五）的**购买需求不强**

业务分析

预测的商品子集主要是O2O的服务行业。

- 服务地点存在一定规律
- 服务商品存在一定的时间规律
- 和非O2O商品的关联关系
- 周期性购买服务
- 区域间的竞争
- ....

线上消费、线下服务

特征工程


User Feature
用户活跃度
用户购买力
用户转化率
用户行为模式
01
Item Feature
商品活跃度
商品销售量
商品转化率
商品竞争优势
商品隔天/重复购买模式
02
UC Feature
用户对类别的喜好程度
用户对类别的转户率
用户对类别的购买力
用户对类别的隔天购买模式
用户对类别的重复购买模式
06
Category Feature
商品类别活跃度
商品类别销售量
商品类别转化率
商品类别竞争优势
商品类别隔天/重复购买模式
03
UI Feature
用户对商品的喜好程度用户
购买商品的行为模式
行为占比
05
04
Rank Based Feature
基于U分组排序
基于UC分组排序
基于GEO分组排序
GEO
地区购买力
地区转化率
地区间竞争优势

特征工程——User特征

- 最近n天的各个行为和
- 最近1,3天的购买行为总和（所有行为数据包括非O2O数据）
- 用户最近n天活跃度
- 最早一次交互/购买距离预测日前一天的时间间隔
- 最后一次交互/购买距离预测日前一天的时间间隔
- 最近N天隔天购买比率
- 最近N天隔天购买次数
- 新用户标注

特征工程——User特征

考虑week因素

十一月 2014

周日	周一	周二	周三	周四	周五	周六
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

十二月 2014

周日	周一	周二	周三	周四	周五	周六
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

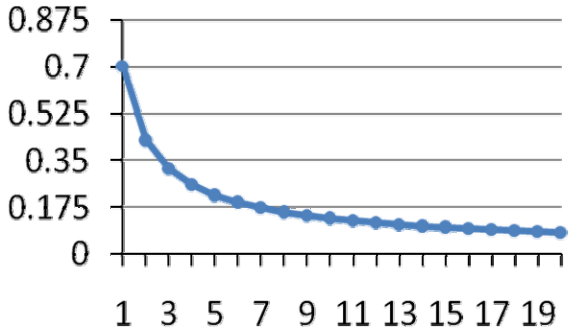
用户在各个周五(周四/周三)的活跃度、购买力、购买可能性

考虑时间衰变

用户最近n天的时间衰变行为和

$$decay(x) = \lambda^{\log d^2+1}$$

$$W_u = \sum_{d \in Dates} \sum_{i \in I} behavior\_times_d(u,i) \times decay(d)$$



特征工程——Item特征

注：同时考虑week影响，提取商品在各个周五(周四/周三)的活跃度、销售量、销售可能性

特征工程——Category特征

Category特征是Item层面的聚集,与Item特征类似

隔天购买的两种情况

第一天交互的item与第二天购买的item是同一个item

看什么item
买什么item

第一天交互的item与第二天购买的item在同一个Category

看什么category
买什么category

特征工程——UI,UC特征

最近N天的行为和

最近n天的权重行为和

最近n天的权重行为平方和

最近N天各行为占比

最后一次交互/购买距离预测日前一天的时间间隔

最早一次交互/购买距离预测日前一天的时间间隔

用户最近n天的时间衰变行为和

UC行为在U中占比

注：UC特征是在UI层面的聚集

特征工程——U-GEO特征


区域购买力
从购买数量推测该地区对指定category的购买力
用户的活动范围
识别用户类型，如果用户最近的活动范围突然扩大，很可能是有出差等外出活动
用户与商品距离
计算用户最近几天，与商品的最小距离
商品分布范围
移动商品一个不可忽视的特性就是变化的地理位置，如果商品的分布范围广、购买方便很可能影响到商品是否被购买

特征工程——转化率

Browse to Buy　　Collect to Buy　　Cart to Buy

方法一：
用户周期内购买总次数/用户周期内浏览总次数

方法二：
以用户收藏转化率为例，不能单纯的用购买总次数/收藏总次数，因为有些购买不来源于收藏行为，如用户对item1的行为。

方法三：
以用户浏览转化率为例，只统计最终被用户购买item的浏览次数，计算该用户购买商品前需要多少次浏览。

Lily	Browse	Collect	Cart	Buy
Item1	4	0	1	1
Item2	2	1	0	0
Item3	3	1	1	1
Item4	1	0	0	0

注：User、Item、Category、User-Category的转化率类似

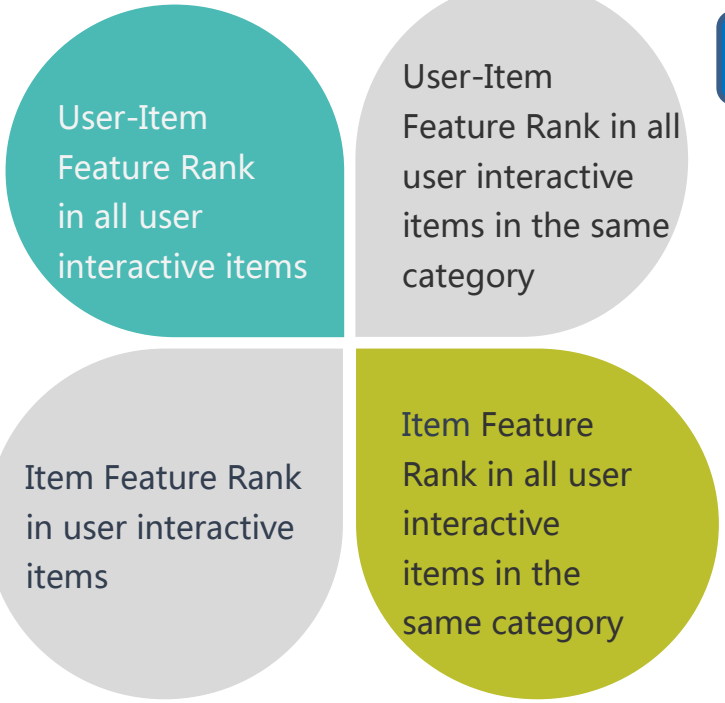
特征工程——分组排序特征

关键点

分组排序特征在我们最终的模型中起到了关键性作用，将特征进行排序作为一个新的特征。

显著提高命中样本数100+！

四种rank特征类型



两种rank方式

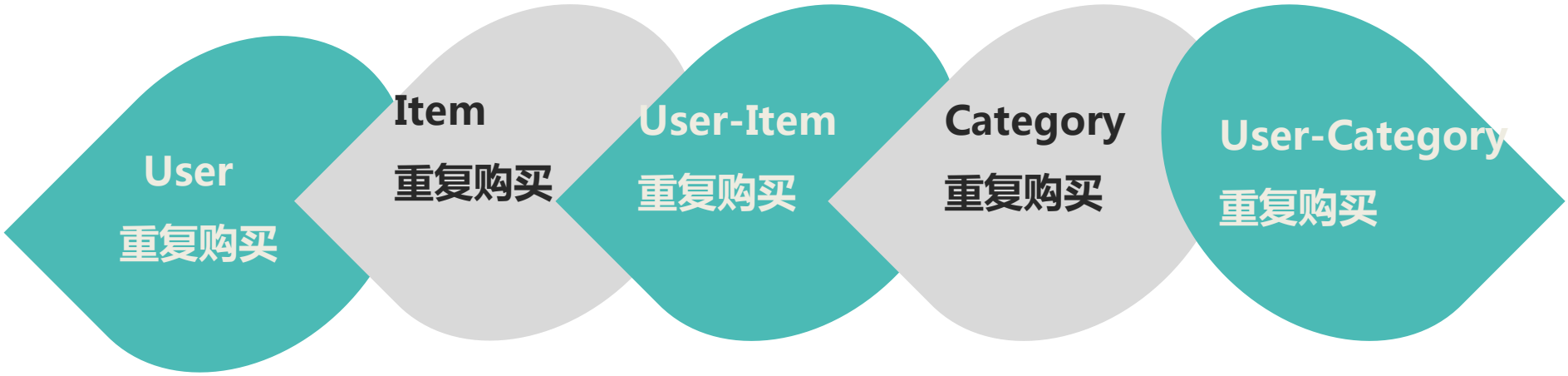
- Dense Rank
- Percent Rank

搜索问题

举例： 对用户交互过的商品通对浏览次数逆序排序，可以体现用户对于商品的喜好程度，一般情况下，浏览的越多购买的可能性越多，通过rank之后，可以形成一个基于用户空间的对比。

特征工程——重复购买

重复购买行为特征，主要是为了训练重复购买模型



特征工程——其他

各个类型的行为得分

$$behaviorScore = behaviorTimes \times conversionRate$$

去中心化

$$f(a,b) = \frac{a-b}{b}$$

有些人/商品浏览次数明显高，对其进行去中心化处理，可以更突显其喜好

- Superiority about user-item over user-category-item average:

$$SP_{ui\_uc} = \frac{count(u,i) - mean\_count(i \text{ in u,c})}{mean\_count(i \text{ in u,c})}$$

- Superiority about user-item over category-item average:

$$SP_{ui\_uc} = \frac{count(u,i) - mean\_count(i \text{ in c})}{mean\_count(iinc)}$$

- Superiority about user-item over user average:

$$SP_{ui} = \frac{count(u,i) - mean\_count(u)}{mean\_count(u)}$$

- Superiority about user-item over other item average:

$$SP_{ui\_i} = \frac{count(u,i) - mean\_count(i)}{mean\_count(i)}$$

- Superiority about user-category over category average:

$$SP_{uc\_c} = \frac{count(u,c) - mean\_count(c)}{mean\_count(c)}$$

- Superiority about user-category over user's category average:

$$SP_{uc} = \frac{count(u,c) - mean\_count(cinu)}{mean\_count(cinu)}$$

特征分析

Positive Data VS Negative Data

Field	Positive Max	Negative Max	Positive Min	Negative Min	Positive Mean	Negative Mean	Positive SD	Negative SD
ui_last_browse_score_dense_rank	163	556	1	1	6.510745064	11.8156915	9.573269746	15.69164258
ui_last_browse_score_percent_rank	1	1	0	0	0.264368563	0.409883815	0.291397499	0.303935502
ui_last_all_score_dense_rank	166	556	1	1	5.546016508	11.89054029	8.926351216	15.81741401
prd4w_ui_beh_wgt	36.81622594	59.78890597	1	1	6.543109788	2.579570701	3.874982195	2.07600573
ui_uc_beh_ctr	7.084745763	11.65185185	-0.939271255	-0.966666667	0.602182057	-0.027891395	0.908338691	0.442425973
pro2_chr4_cnt	5	10	-1	-1	-0.019126577	-0.053824731	1.10473262	1.092368423

Training Data VS Testing Data

Field	2014/12/18 Max	2014/12/17 Max	2014/12/18 Min	2014/12/17 Min	2014/12/18 Mean	2014/12/17 Mean	2014/12/18 SD	2014/12/17 SD
ui_last_browse_score_dense_rank	163	160	1	1	6.510745064	6.675534508	9.573269746	10.10631393
ui_last_browse_score_percent_rank	1	1	0	0	0.264368563	0.264995866	0.291397499	0.291118607
ui_last_all_score_dense_rank	166	162	1	1	5.546016508	5.708684449	8.926351216	9.452401317
prd4w_ui_beh_wgt	36.81622594	33.40021109	1	1	6.543109788	6.521531862	3.874982195	3.856140511
ui_uc_beh_ctr	7.084745763	13.28571429	-0.939271255	-0.857142857	0.602182057	0.590909019	0.908338691	0.908131327
pro2_chr4_cnt	5	5	-1	-1	-0.019126577	-0.034194582	1.10473262	1.094221576

失败的特征

销售增长率

- 商品需求淡旺季
- 商品促销
- 新商品上市
- ....

没有提升

可能原因

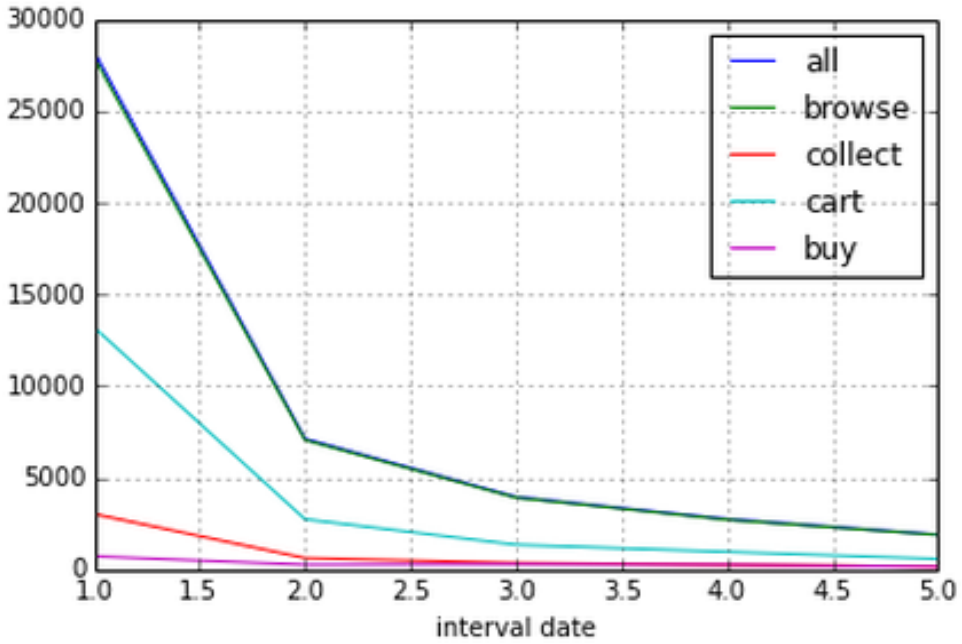
- 总体周期短
- 促销持续时间短
- 隔天增长不一致
- ...

星期一	星期二	星期三	星期四	星期五	星期六	星期日
	11/18	11/19	11/20	11/21	11/22	11/23
22/24	11/25	11/26	11/27	11/28	11/29	11/30
12/01	12/02	12/03	12/04	12/05	12/06	12/07
12/08	12/09	12/10	12/11	**12/12**	12/13	12/14
12/15	12/16	12/17	12/18	**12/19**		

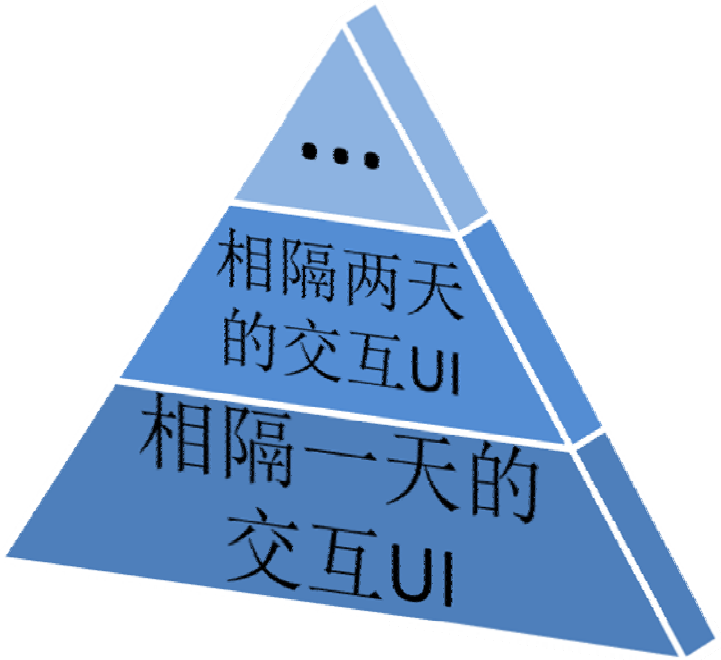
buy_trend_1216
 * d1: 15 -> 16
 * d2: 12/7+12/8 -> 12/14+12/15,
 * w3d1: 11/18+11/25+12/02
 -> 11/19+11/26+12/03
 * p3w1: 11/18+11/24 -> 12/02+12/08

各个行为影响衰减分析

如图，最近1天贡献最大，后面逐渐衰减，时间间隔3天以后衰减的更加严重



抓住主要问题，首先从相隔一天的交互UI入手，逐个突破



隔一天预测模型

样本数据

Label Date	正样本	负样本	正负比
2014/12/17	29481	7463340	1:253
2014/12/18	28215	7283858	1:258

为了减少模型的混乱程度，将当天已购买的UI与未购买的UI分开处理

看了后隔1天购买样本			
Label Date	正样本	负样本	正负比
12/17	28718	7283594	1:254
12/18	27501	7109520	1:259

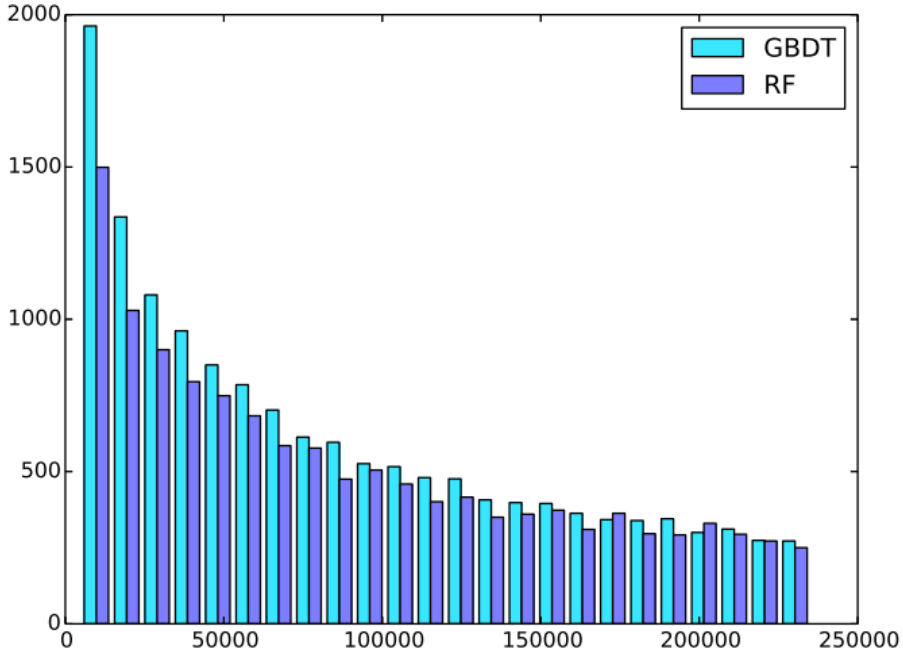
VS

重复购买样本			
Label Date	正样本	负样本	正负比
12/17	763	179746	1:236
12/18	714	174338	1:244

从对比表可以看出买了又买模式的转化率比看了购买模式要高，
当初直接踢掉当天购买的UI是不可取的

GBDT VS RF

Training Data:17 label date 样本
正负比例1:6，
抽取100w
Testing Data: 18 label date



预测数量与hit数量分布图

参数设置

GBDT

- Metrics: Normalized Discounted CumulativeGain
- Trees: 1,000
- Learning rate: 0.04
- Maximal number of leaves: 32
- Minimal leaf size: 500
- Ratio of instances for training: 0.6
- Ratio of features for training: 0.6
- Maximal splits: 500

RF

- Decision Tree Algorithm: Mixed of ID3, C4.5 and CART
- Trees: 1,000
- Minimal leaf size: 100
- Number of featurs: log *N*
- Maximal number of instances per tree: 600,000

GBDT明显好于RF

GBDT单个模型

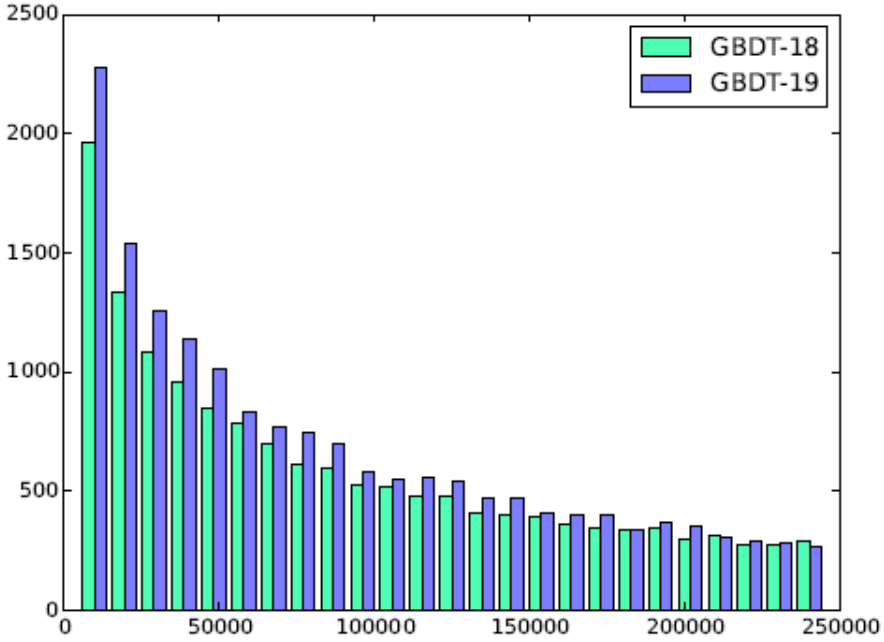
训练数据:正样本-----17_Pos*2+18_Pos*4

负样本-----Random Sample 100w(17_Neg*1+18_Neg*2)

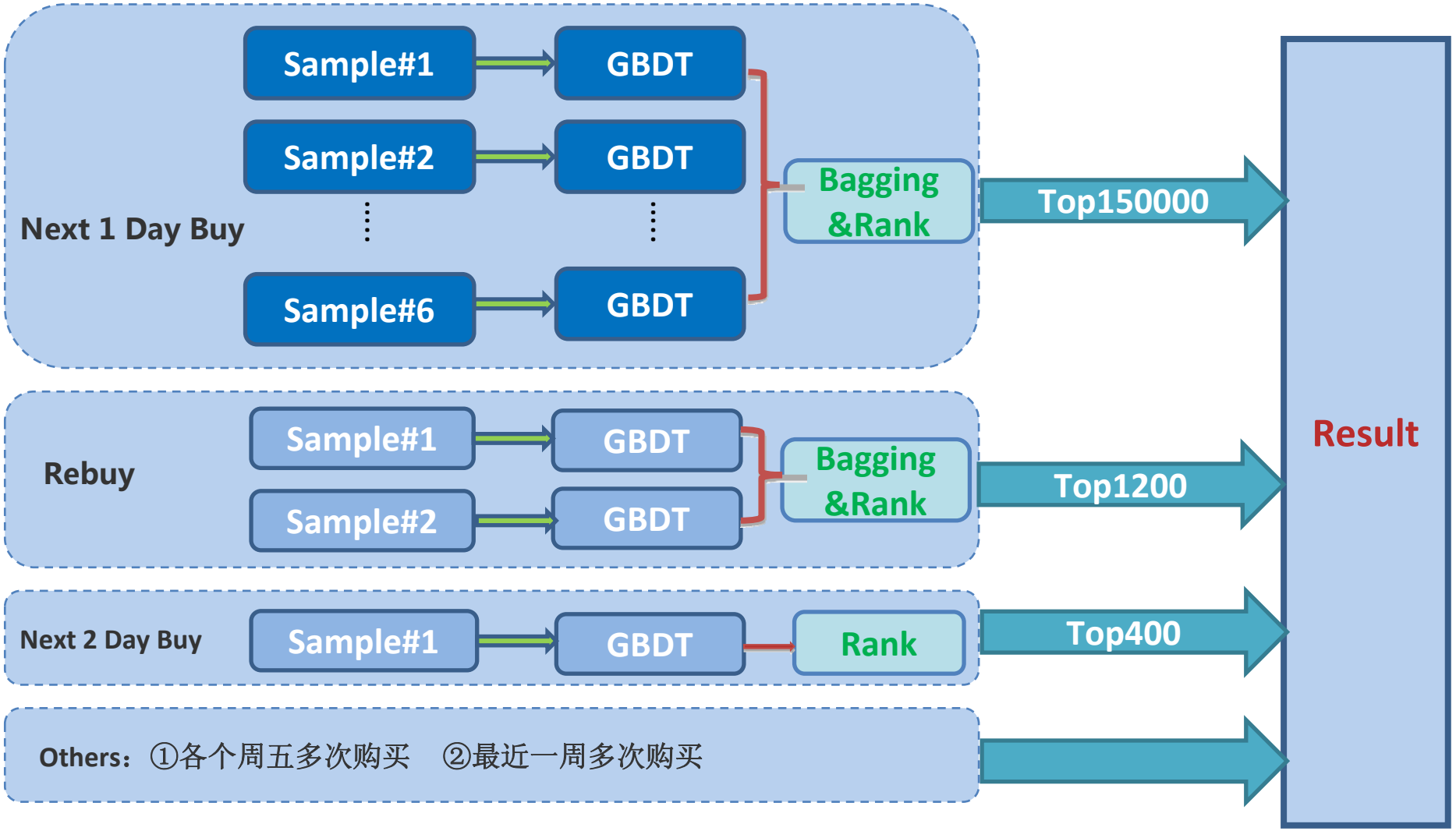
负样本**先融合后抽样** 比 **先独立抽样后融合**效果好

参数设置

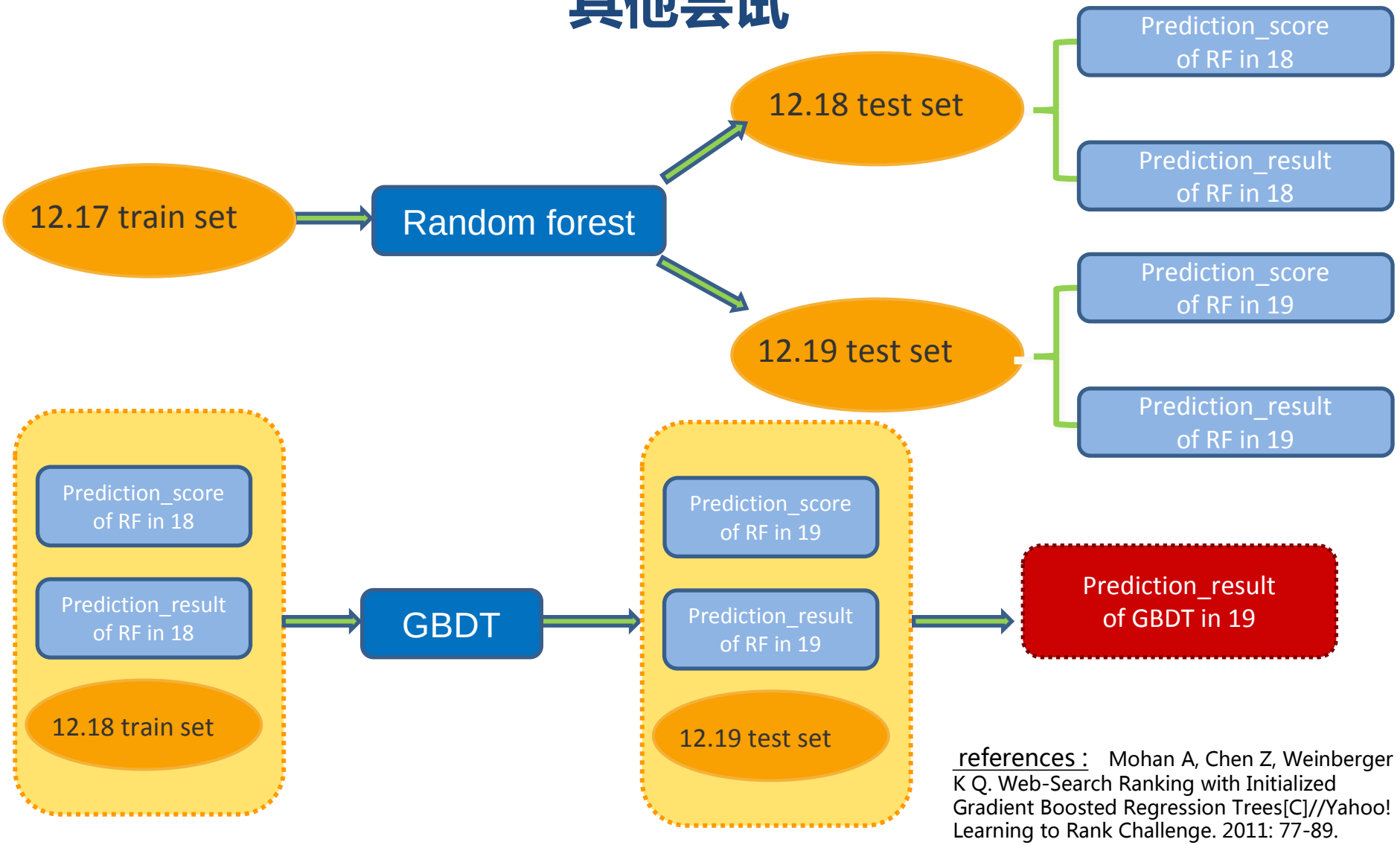
- Metrics:NDCG
- Trees: 10,000
- Learning rate: 0.04
- Maximal number of leaves: 70
- Minimal leaf size: 500
- Ratio of instances for training: 0.8
- Ratio of features for training: 0.6
- Maximal splits: 500



Model Ensemble

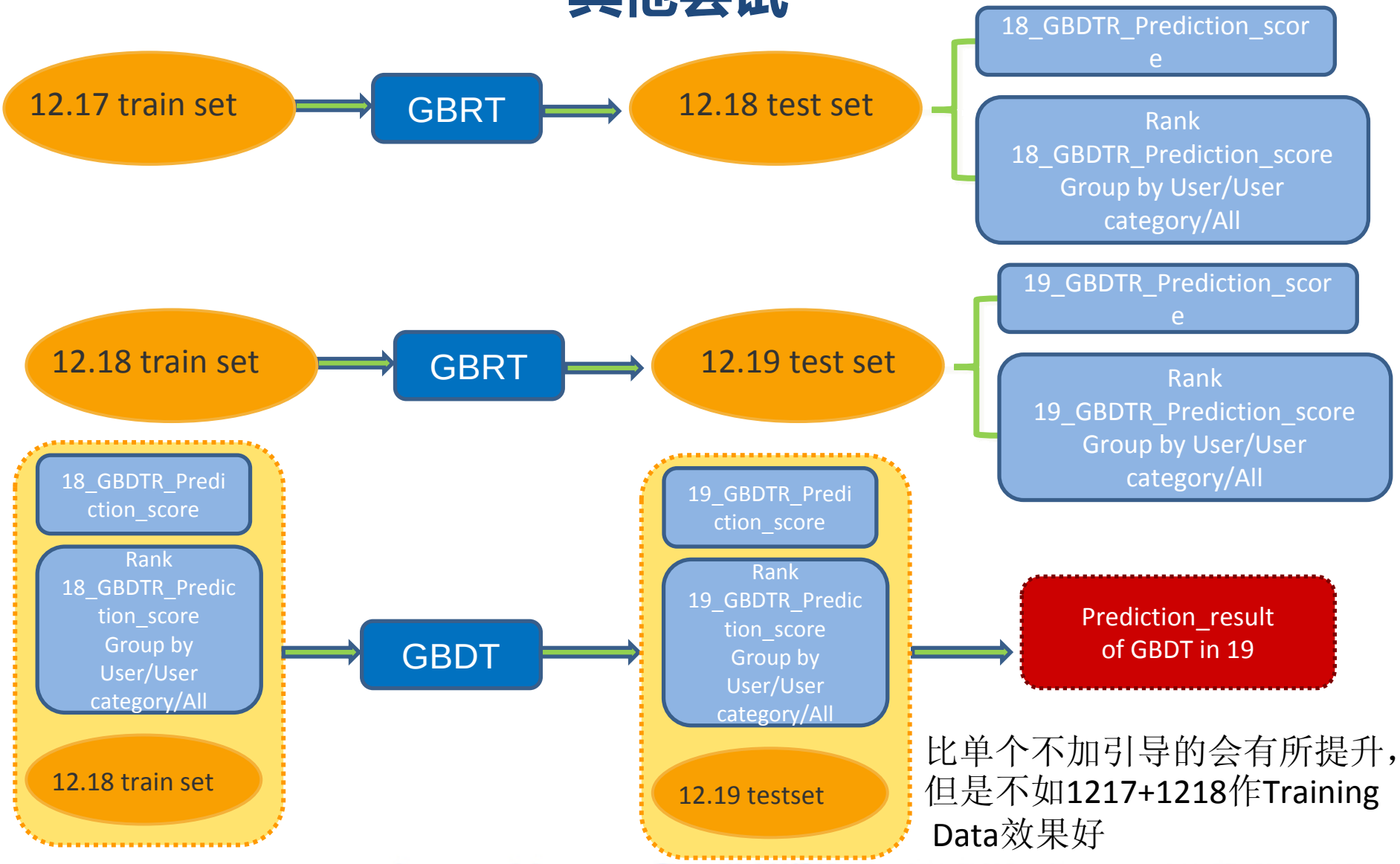


其他尝试



references： Mohan A, Chen Z, Weinberger K Q. Web-Search Ranking with Initialized Gradient Boosted Regression Trees[C]//Yahoo! Learning to Rank Challenge. 2011: 77-89.

其他尝试



比单个不加引导的会有所提升，但是不如1217+1218作Training Data效果好

总结

- 特征工程决定上界，算法决定接近上界的程度
- 重视数据分布

  Training Data VS Testing Data

  Positive Data VS Negative Data
- 团队合作（工作协调，时间管理）
- 理论与实践相结合
- 注意不要穿越，不要偷看未来预测数据
- 训练多个模型，取舍比例控制
- 我们尝试的模型远比我们呈现的要多

建议

天池平台总体感觉还不错

建议天池平台加入任务定时执行以及等待执行功能

考虑到实际应用场景，衡量标准借鉴一下**rank**的衡量标准是否会更好些呢，比如**NDCG**，需不需要限制每个客户的最多推荐个数等等

团队成员邮箱：lambertzhao@foxmail.com 赵光明

yfw100@163.com 王植

mawenjianeu@gmail.com 马文佳

参考文献

[1] Alibaba, “Ali Mobile Recommendation Algorithm.” [Online].Available: http://tianchi.aliyun.com/competition/introduction.htm?spm= 5176.100066.333.5.tYfFbq&raceId=1& lang=en US

[2] ——, “Yushanfang Big Data Platform.” [Online]. Available: http: //www.yushanfang.com

[3] B. Leo, “Random Forests,” *Machine Learning*, vol. 45, no. 6, pp. 422–432, 2001.

[4] J. H. Friedman, “Stochastic gradient boosting,” *Computational Statistics& Data Analysis*, vol. 38, no. 4, pp. 367–378, Feb. 2002.

[5] Z. Zheng, K. Chen, G. Sun, and H. Zha, “A regression framework for learning ranking functions using relative relevance judgments,” in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, ser. SIGIR ’07. New York, NY, USA: ACM, 2007, pp. 287–294.

[6] C. J. C. Burges, “From RankNet to LambdaRank to LambdaMART: An Overview,” Microsoft Research, Tech. Rep., 2010.

[7] T. G. McKenzie *et al.*, “Novel models and ensemble techniques to discriminate favorite items from unrated ones for personalized music recommendation.” *JMLR W&CP*, 2012.

[8] P.-L. Chen *et al.*, “A linear ensemble of individual and blended models for music rating prediction,” *JMLR W&CP*, vol. 18, 2011.


谢谢聆听
2015天池大数据竞赛
TIANCHI天池
NEU-Smart团队