2017 **达观杯**个性化推荐算法挑战赛

龙樱 2017.08

目录页

01
OPTIO
N

成员介绍

- 张 杰
- 邸海波
- 徐铉
- 范林鹏
- 严欣雨

01
OPTIO
N

成员介绍

成员：张杰
简介：南京大学计算机系研二学生
邮箱：zhangj@lamda.nju.edu.cn
主页：http://lamda.nju.edu.cn/zhangj/

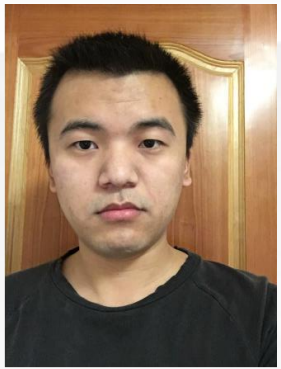
01
OPTIO
N

成员介绍

成员：邸海波
简介：天津大学机器学习研究生，微博机器学习&AILab算法工程师
邮箱：dihaibo@tju.edu.cn
主页:http://blog.csdn.net/a1b2c3d4123456

01
OPTIO
N

成员介绍



成员：徐铉
简介：南京维数 算法工程师
邮箱：xuxuan1003@163.com
主页：http://blog.csdn.net/corpsepiges

01 OPTION

成员介绍

成员：范林鹏
简介：南京大学计算机系研三学生
邮箱：fanlp@lamda.nju.edu.cn
主页：http://lamda.nju.edu.cn/fanlp/

01
OPTIO
N

成员介绍

成员：严欣雨
简介：南京大学计算机系研一学生
邮箱：yanxy@lamda.nju.edu.cn
主页：http://lamda.nju.edu.cn/yanxy/

02
OPTIO
N

赛题介绍

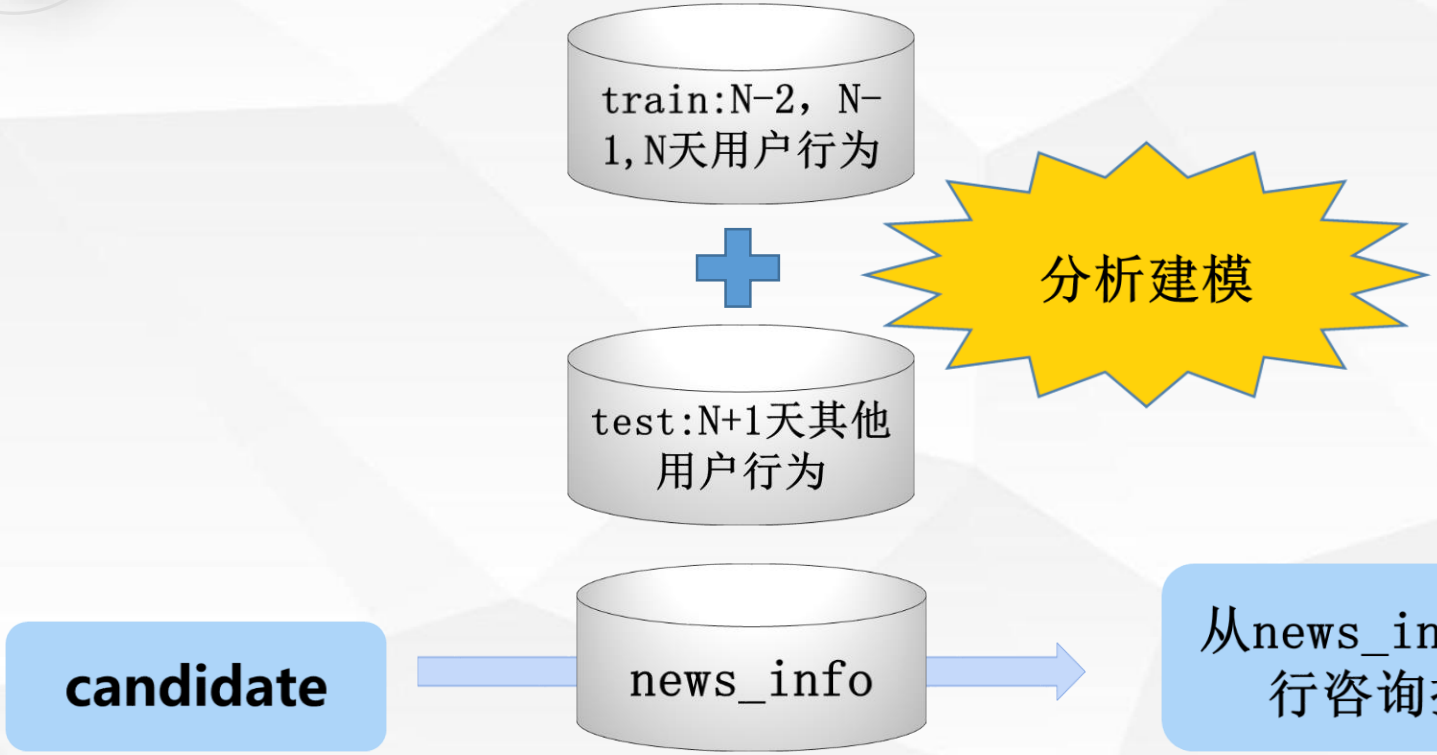
赛题简介

通过用户在**某3（记为第N-2天、第N-1天和第N天）天内对资讯内容的多种行为数据（train.csv）**，包括点击、完整阅读、评论、收藏、分享等，结合**第N+1天另外一些用户的浏览行为(test.txt)作为训练数据**，预测目标用户(candidate.txt）在第N+1天中对于候选资讯内容数据（news_info.csv）的浏览情况。

对每个候选集中的用户推荐5个最可能且不重复的资讯。最终成绩以**Mean Average Precision @ 5，MAP@5**作为评价指标进行判定.

02 OPTION

赛题介绍



03 OPTION

解题过程

01 两个难点一个假设

02 数据预处理

03 特征工程 + 单模型

04 模型融合

05 模型融合

3.1 赛题分析

■ 两大难点

✓ **数据仅给出用户三天的浏览记录，训练集和验证集的线下构建会受限**

- 用第16天的数据提取feature,第17天的用户观看记录构造label;再用第17的数据进行特征构造,用第18天的数据构建label，从而形成对应的训练集和验证集;无法用16-17天的数据构建feature,第18的数据构建label,这样的话会较难构建线下的验证集;

✓ **如何利用好第N+1天新增的test集合，**该集合是由N-2,N-1,N天中未出现用户N+1的浏览信息组成,如何较好利用这个数据集,这个数据集能带来什么也成为一个问题.

■ 猜想

✓ **新增的test集合可以认为是第N+1天的一个随机采样,**该集合能帮助我们得到当天热门产品的信息,

而如果热门的产品分布较广(假设有90%的用户都会看最热门的产品,那么毫无疑问我们就可以以此为突破口,进行候选集的构建与建模).

01

两个难点

一个猜想

3.1 OPTION

赛题分析

■ 假设验证

✓ 分别统计16,17,18日三天最热门的top100的咨询占当天所有用户的比例

- 第16日中top100的咨询有78.99%的用户进行了浏览
- 第17日中top100的咨询有80.07%的用户进行了浏览
- 第18日中top100的咨询有79.96%的用户进行了浏览

■ 结论

✓ 每日top100的咨询分布在80%的用户当中，如果我们能把这些热门咨询大部分都推荐正确的话,那么就可以获得非常高的分数;而之前的test集合恰好可以用来构建19日当天的热门咨询.

01

两个难点

一个猜想

数据预处理

■ 数据预处理

✓ 刷咨询用户检测与处理:N-1,N,N+1天中两天以上都有记录(特殊爱好);某一天多次操作(刷单),保留一次记录.

		hit_counts
user_id	item_id	
15216521	549829	62
1fd30a9290bf16952d888d5a90d7103f	164800	39
1732453	541978	34
12257557	544294	24
1019135	544271	23
6324815	553482	22
16733192	527519	20
9400641	524648	20

3.3 OPTIO N

特征工程+单模型

- 用户特征
  - ✓ 用户浏览所有咨询的总次数
  - ✓ 用户对每类cate_id的咨询的浏览总次数
  - ✓ 用户对每类cate_id的浏览次数占用户对所有浏览次数的比例
  - ✓ 用户每日对top10item的浏览情况(浏览就是1,未浏览就是0)
  - ✓ 用户浏览了最热门的top50,top100,top200,top500咨询的次数（例如top100个咨询看了其中的20个就是20）
  - ✓ 用户最后一次浏览的情况(最后一次浏览的cate_id的咨询次数以及总次数)
  - ✓ 用户最后一次浏览的时间距离下一天的时差

- 咨询特征
  - ✓ 咨询生成的时间
  - ✓ 咨询被view的次数占所属cate_id的比例
  - ✓ 咨询所属的cate_id进行one-hot编码
  - ✓ 咨询被浏览的总次数
  - ✓ 咨询是否是当日的top10的热门咨询(是就是1,不是就是0)(训练集1)
  - ✓ 咨询是否是当日的top10的热门咨询(是就用对应咨询被浏览的次数来计数)(训练集2)
  - ✓ 咨询是否为16,17,18,19当日的产品(是就是1,不是就是0)

03
特征工程
+单模型

3.3 OPTION

特征工程+单模型

■ 数据集1（咨询信息用top1-10进行one-hot编码）

✓ 训练集1：用16,17日的信息提取用户特征,18日的信息提取用户特征和产品特征,再利用18日的信息构建label;

✓ 测试集1：用17,18日的信息提取用户特征,19日的信息（test集合）提取用户特征和产品特征;

■ 数据集2（咨询特征部分用top1-10浏览次数计数）

✓ 训练集2：用16,17日的信息提取用户特征,18日的信息提取用户特征和产品特征,再利用18日的信息构建label;

✓ 测试集2：用17,18日的信息提取用户特征,19日的信息（test集合）提取用户特征和产品特征;

■ 数据集3

✓ 训练集3：用16日的信息提取用户特征,17日的信息提取用户特征和产品特征,再利用17日的信息构建label;

✓ 验证集3：用17日的信息提取用户特征,18日的信息提取用户特征和产品特征,再利用18日的信息构建label;

✓ 测试集3：用18日的信息提取用户特征,19日的信息（test集合）提取用户特征和产品特征;

03 特征工程+单模型

3.3 OPTIO N

特征工程+单模型

■ 单模型XGB + 调参

✓ 因为数据集的构建方式不同,对于*数据集1和数据集2*我们采用线上调参;这种方式效率较低,但是线上的效果却相对不错,以数据集1单模型XGBoost可以达到0.0699的线上分数,数据集2达到的效果类似可以达到0.0701的分数.

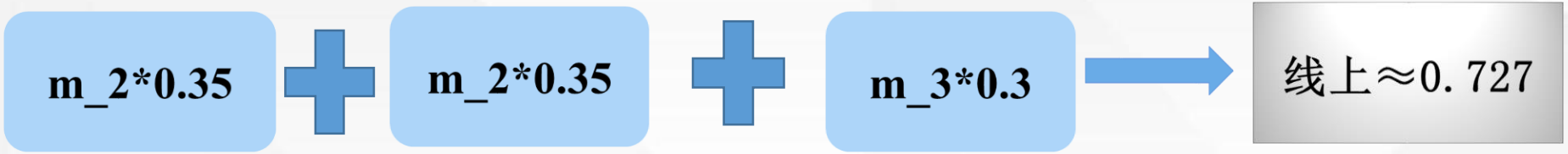
✓ 对于*数据集合3*我们采用验证集线下调参的方式进行;最终单模型的线上的结果能达到0.0694.

✓ 因为我们的模型只进行简单的调优,所以还有一定上升的空间,我们在调参的时候也发现参数的不同可以使得我们模型从0.0682 - 0.0701之间波动,此处我仅给出训练集2的最优参数供大家参考.

```
paras = {'booster':'gbtree', 'eta':0.15, 'max_depth':6, 'subsample':0.9,
         'colsample':1, objective':'binary:logistic', 'lambda':3, 'alpha':8,
         'num_boost_round':300, 'eval_metric':'auc'}
```

3.4 OPTION

模型融合



04

模型融合

3.5 OPTIO N

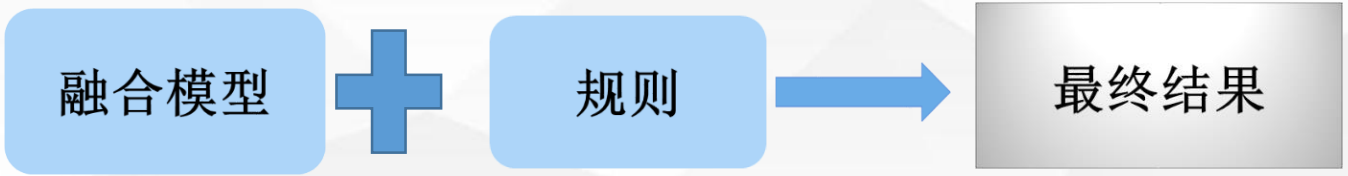
模型+规则

■ 两个规则

✓ 用户癖好探索:用户三天同时都看了哪些咨询,我们默认这些咨询用户在第四天还会看.

✓ 深夜用户挖掘:统计熬夜用户(在凌晨的时候还在看咨询),我们认为这些用户在前一天还没有看完的产品会在接下来的一天继续观看,即前一天view的产品很大程度上在后一天会变为deep_view.直接推荐即可.

• 例子：我们发现在17天深夜还在view的咨询有86.9%概率顺延到了下一天，于是我们将第18天最后的2分钟最后阶段还在view的数据继续首推给对应的用户.



05 模型融合

04 总结

赛题难点

1.本次赛题的最大难点在于如何构建候选集合,当候选集合构建完成之后,后面的建模过程即可回到原始的问题当中;
2.本次赛题中出现的test集合打乱了我们传统的建模思路,这也导致我们不得不将我们的思路转变到上述的方式中.尝试中也发现传统的基于协同过滤的方法的效果较为一般(线下测试的结果).

后续工作

1.本次赛题可以扩充的地方还有很多,比如特征工程构建模块,我们受限于机器的原因,只能提取少量的特征,不然内存会溢出,所以特征工程这一块还可以做很多的补充
2.我们建模的过程仅仅使用了XGBoost和 Random Forest,其他的类似于FFM的流行模型尚未使用,所以可以考虑使用这一类的包.
3.调参也有很大的改进,线上的效果对比我们发现,参数调整可以带来较大的差异,从0.0607 - 0.0701,个人感觉应该还有较大的上升空间.
4.候选集构建过程中我们仅仅使用了top100的咨询,并未做其他调整,所以可以调节topN中N的大小.

感谢您的聆听！

欢迎各位专家批评指正！