# 讯飞AI营销算法大赛说明书

本次大赛我们基于特征工程和lgb模型+stacking模型获得第12名的成绩。在这里简单介绍一下我们团队地思路，主要包括数据分析、特征构造和模型算法地使用。

## 数据分析：

1. 初赛和复赛数据加起来有300多万条，一共35个字段，其中包含32个类别变量、2个数值型变量和标签变量‘click’;
2. 目的：根据广告地信息、用户的信息、材料信息预测用户是否会点击广告；
3. 对每个字段进行分析，我们发现：
4. 很多字段中某一类别出现的次数为1，比如adid中，有较多类别出现的次数仅有一次，即广告得被展示次数是不均匀的，需要构造广告的其他属性判断其类似的广告；
5. advert\_id和advert\_name基本上是一一对应的关系，可删掉一个字段；os和os\_name同理；
6. 字段creative\_is\_voicead的值均为False,即均不为语音广告，该字段信息熵为0，可删除；creative\_is\_js同理;
7. 所给信息中只给了用户标签和设备信息，未给用户id。不能构造用户历史行为信息；因此从设备信息中，推测用户uid，并进行用户历史行为特征构造；
8. 在对设备信息如osv,make,model进行观察，发现数据结构不一致，比如大小写不同，猜测数据来源不同，质量不同。

## 二．特征工程：

1. 基础特征：对所有类别型特征进行独热编码（onehot），因为维数较大，使用sparse稀疏矩阵形式保存；
2. 属性特征：对所有类别型特征进行计数，计算每个类别的出现次数，可以衡量该广告的热度等信息，体现品牌的占有量；
3. 交叉特征：计算两两字段之间的nunique值。例如每个adid被投放在多少个不同的城市，用于衡量广告的覆盖率；每个adid被投放在多少种不同的app上。基于这些信息，可以判断广告是被定点投放还是全面撒网；
4. 设备信息特征：对osv,model,make等字段进行清洗，对大小写进行统一，对不合规值进行填充；对osv提取第一位数值；识别各品牌的高低端机。例如小米手机是否是红米，苹果三星是否是Plus等；
5. 用户属性特征：操作一：对所有属性进行CountVectourizer；操作二：对所有属性进行分类：21，22，ag,gd,mz等型，计算用户在每种类型中数量；
6. 类别特征与数值特征的交叉：如计算每种adid的面积、长宽的平均值、标准差；
7. 趋势特征：计算类别特征每天出现的次数并计算其标准差；
8. 竞争性特征：计算每种类别特征的竞争性特征。比如每种广告在当天出现的次数占当天总次数的比例；
9. 构造用户uid并构造用户历史行为特征：以city、os、make、carrier、model五个字段判定是否样本是否属于同一个用户，并构造用户历史行为特征；

## 算法模型

本赛题我们使用的是LIghtGBM算法

1. LIghtGBM算法简介：

1.1 优势和改进

训练过程中，我们使用的是Lightgbm算法。相比Xgboost、GBDT等，Lightgbm具有以下优势。

更快的训练效率

低内存使用

更高的准确率

支持并行化学习

可处理大规模数据

LightGBM在传统GBDT的基础上做了改进，主要包括：

基于Histogram的决策树算法

带深度限制的Leaf-wise的叶子生长策略

直方图做差加速

直接支持类别特征(Categorical Feature)

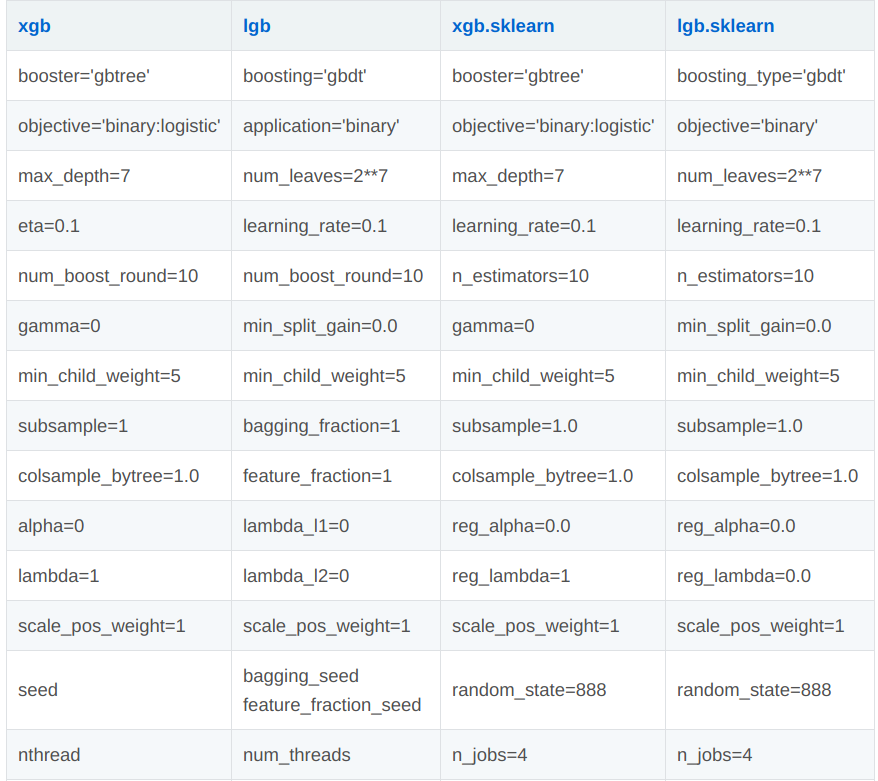
Cache命中率优化

基于直方图的稀疏特征优化

多线程优化

了解LightGBM的原理以及特点将更有助于我们使用该模型在比赛中获得更好的分数。

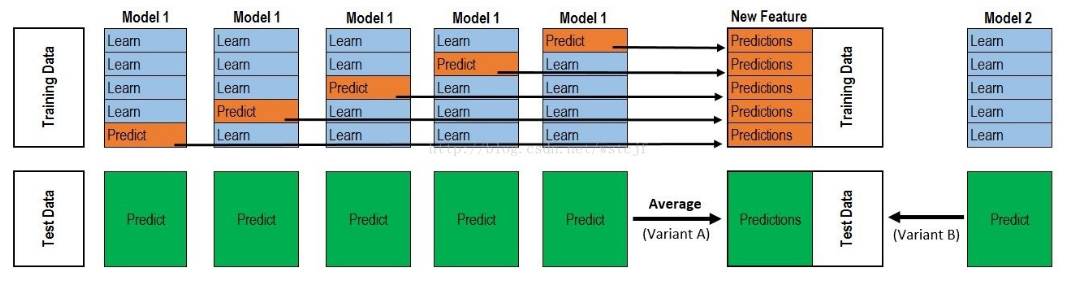
1.2 Lightgbm和Xgboost的参数对比：



1.3 调参建议：



**2**.Stacking算法简介：

  
算法流程：

* 1.将Training Data进行划分，分别训练基模型，此过程和交叉验证类似；
* 2.对每一次的划分的Predict，使用基模型进行预测，将多折预测结果拼起来，得到一个完整的Traning Data;
* 3.同时，对Test Data使用基模型进行预测，然后将得到结果进行平均，得到新的Test Data;
* 4.使用第二层的模型，对新生成的数据集（Training Data、Test Data）训练做预测。

### 我们团队的模型应用

第一步，我们基于LightGBM模型，使用基础特征、属性特征、交叉特征、设备信息特征、类别数值交叉特征，进行8折交叉验证，选取表现最好的前5折，得到sub1.csv文件.复赛A榜得分0.422958；

注：在生成交叉特征时，由于内存有限，因此对每个字段的20多个交叉特征只取特征重要性的前3个；初赛中我们使用的是pca降维，复赛改成筛选效果更好。

第二步，在第一步的基础上，增加平均长、宽、长宽标准差、以及趋势性特征后，使用8折交叉验证，得导sub2.csv文件，并于sub1.csv进行5-5融合，线上得分0.422894；

第三部：构造uid特征。由于此时特征数量过多，我的队友提出使用stackIng进行处理，8折得到sub3.csv文件，线上得分0.42292；与sub1.csv,sub2.csv进行平均融合。最终线上的得分0.42282；

由于我们的多特征多模型，控制了过拟合，最终B榜得分0.421180，位列第12名。