客户画像比赛解决方案（算法说明）

CCF 国家电网 大连理工大学信息检索实验室

1.队伍介绍：

> - 队名：大连理工大学信息检索研究室
> - 指导老师：林鸿飞 教授
> - 队员：费鹏，大连理工大学，研三
> - 网址：ir.dlut.edu.cn

2.解决方案概述

2.1.赛题背景

参赛者需要以电力用户的95598工单数据、电量电费营销数据等为基础，综合分析电费敏感客户特征，建立客户电费敏感度模型，对电费敏感用户的敏感程度进行量化评判，帮助供电企业快速、准确的识别**电费敏感客户**，从而对应的提供有针对性的电费、电量提醒等精细化用电服务。

2.2.比赛数据

赛题给出2015年全年，浙江省1029,244个用户的数据，其中训练集658373个用户，测试集370871个用户。数据共包含12个表，数据来源广且复杂。

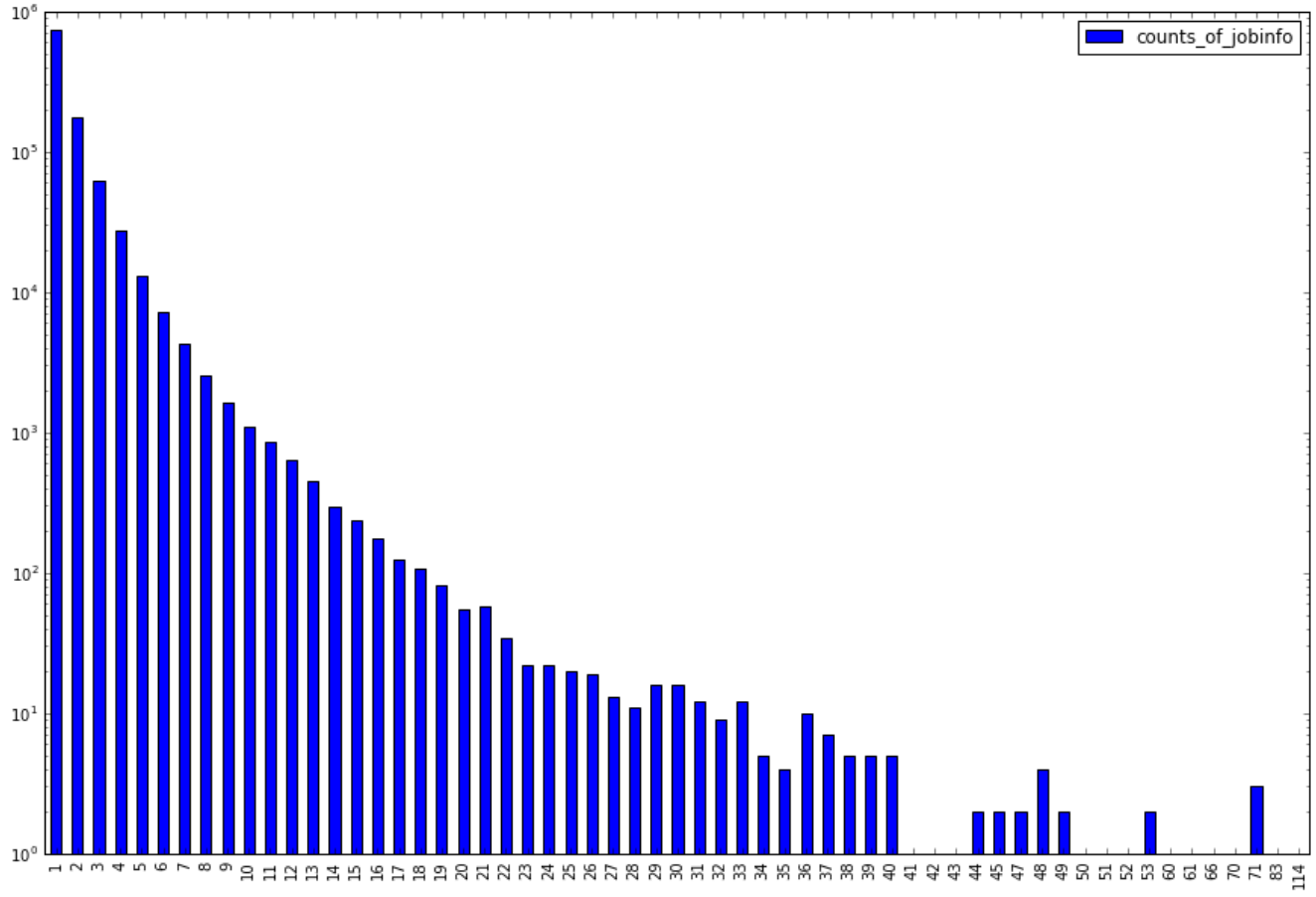
在综合分析考虑了各个数据表的完整性，通过对比实验衡量各类数据对于敏感用户识别的特征贡献程度后，最终我们只选用了表1、表2和表9三个数据表进行特征的构造。

数据表单名称	数据概况	数据缺失率	是否使用
01 95598工单信息	包含全部训练集和测试集，*核心数据*	0	**是**
02 客户通话信息记录	包含训练集用户656641个，测试集用户369560个	0.29%	**是**
03 催办督办信息	包含训练集用户2757个，测试集用户355个	99.69%	否
04 用电客户信息表	包含训练集用户656282个，测试集用户134325个	23.18%	否
05 用户电价信息表	包含训练集用户291857个，测试集0个	71.64%	否
06 低保户信息表	包含训练集用户2455个，测试集用户1505个	99.61%	否
07 费控用户信息表	包含训练集用户22192个，测试集用户10954个	96.77%	否
08 实收电费信息表	包含训练集用户282152个，测试集用户311222个	42.34%	否
09 应收电费信息表	包含训练集用户555748个，测试集用户201702个	26.40%	**是**
10 运行电能表示数	-	-	否
11 运行电能信息表	包含训练集用户641159个，测试集用户0个	37.70%	否
12 收费记录	包含训练集用户431833个，测试集用户222635个	36.41%	否

2.3.项目总体思路

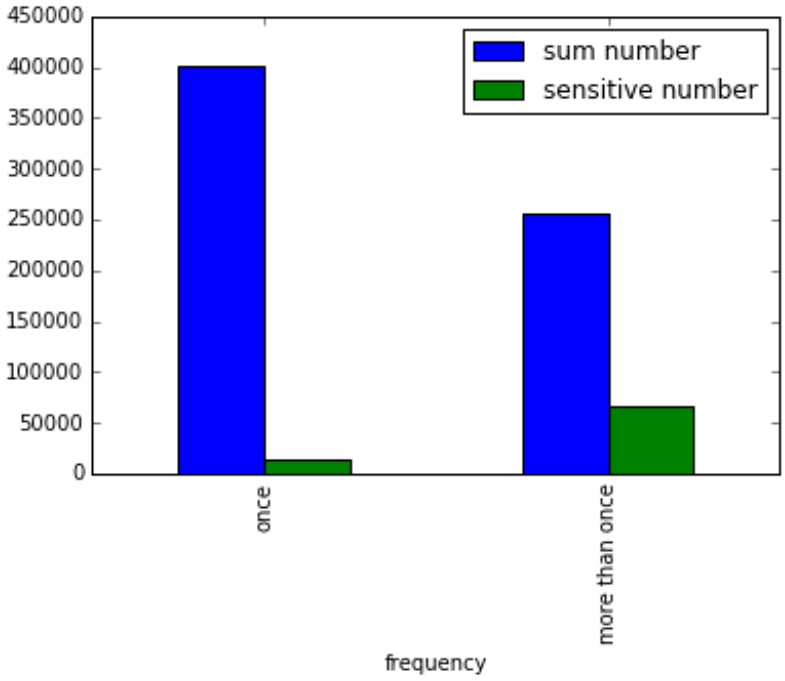
首先，我们分析核心数据95598工单信息发现，每个用户拥有的工单记录数量在[1,114]之间，并且随着工单记录数量的增加，对应的用户数量越来越少。

`x轴代表工单记录次数，y轴代表对应的用户数量（取log）`



显然，对于不同工单记录数量的用户，用来衡量他们是否是敏感用户的标准是不一样的：只有1条记录的用户，我们关心的是他们仅拨打一次95598的通话时间、通话内容以及拨打的时刻等角度；而对于多条记录的用户，我们更关心的是他们拨打95598的频率、每次通话的内容以及各通话记录之间的联系。因此，考虑到两类用户特征不同，我们将所有用户按照95598工单记录的数量不同分为**低敏感度用户**（只有1条95598记录）和**高敏感度用户**（有多条95598记录）两类，分别进行特征的构建和模型的训练。

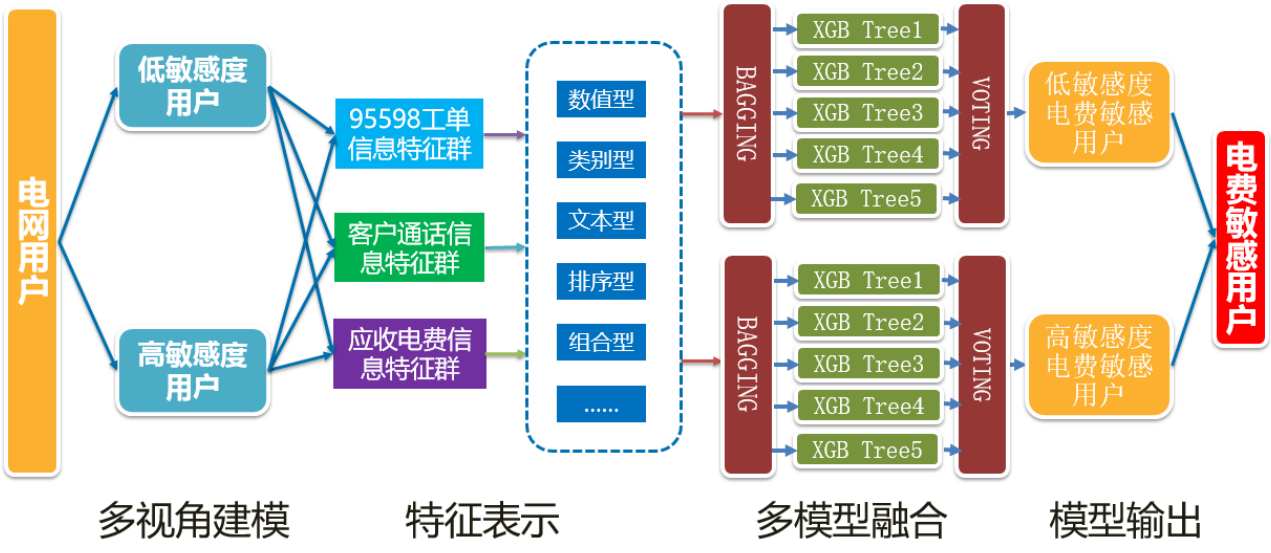
x轴代表两类用户，蓝色代表用户总数，绿色代表敏感用户即正样本数量



而由上图也可以看出两类用户的分布特点：

- 低敏感度用户：总体数量多，电费敏感用户占比小
- 高敏感度用户：总体数量少，电费敏感用户占比大

于是我们设计出的整体模型框架如下图所示：



接下来，本文将从数据预处理、特征工程、特征选择、模型设计，模型融合等方面介绍我们的算法。

3.数据预处理

3.1.通话信息部分

表2通话信息记录了每一条95598工单的通话开始时间REQ_BEGIN_DATE和通话结束时间REQ_FINISH_DATE，而通话时间的长短对于通话的内容是一个很好的度量，是判断用户是否是敏感用户的一个非常有效的特征。

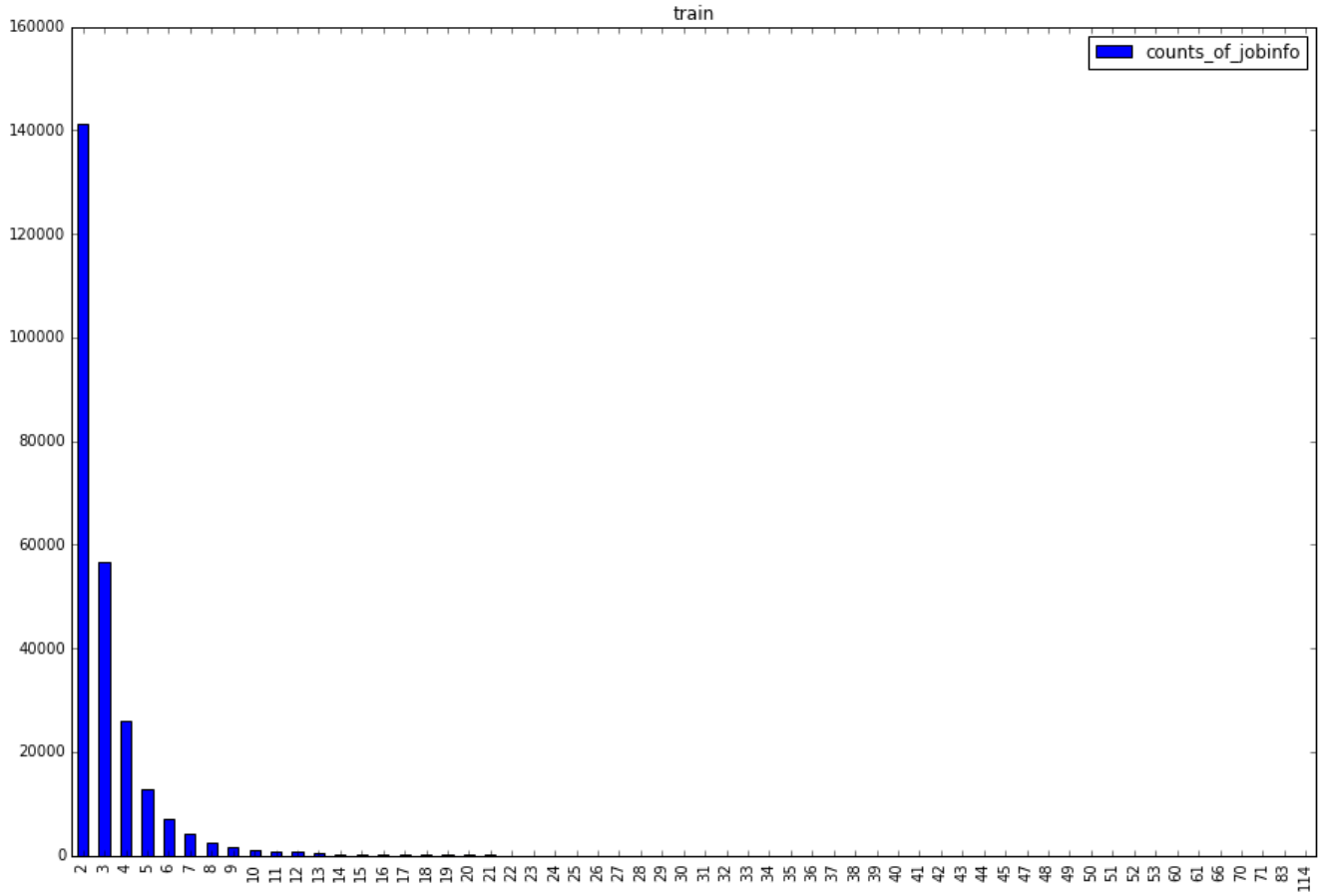
	APP_NO	HANDLE_ID	COMM_NO	REQ_BEGIN_DATE	REQ_FINISH_DATE	ORG_NO	BUSI_TYPE_CODE	WKST_BUSI_TYPE_CODE
0	1000000619515828	NaN	1818693	2015/12/24 12:37:29	2015/12/24 12:39:39	13406	999.0	3
1	1000000267085485	NaN	1344134	2015/2/4 14:10:21	2015/2/4 14:11:49	15421	999.0	3
2	1000000241905876	NaN	1475375	2015/1/11 11:18:29	2015/1/11 11:23:20	1542316	999.0	15
3	1000000243214087	NaN	1634225	2015/1/12 16:50:25	2015/1/12 16:51:33	15404	3.0	3
4	1000000289386345	NaN	1316851	2015/3/3 21:51:13	2015/3/3 21:52:12	15404	999.0	3

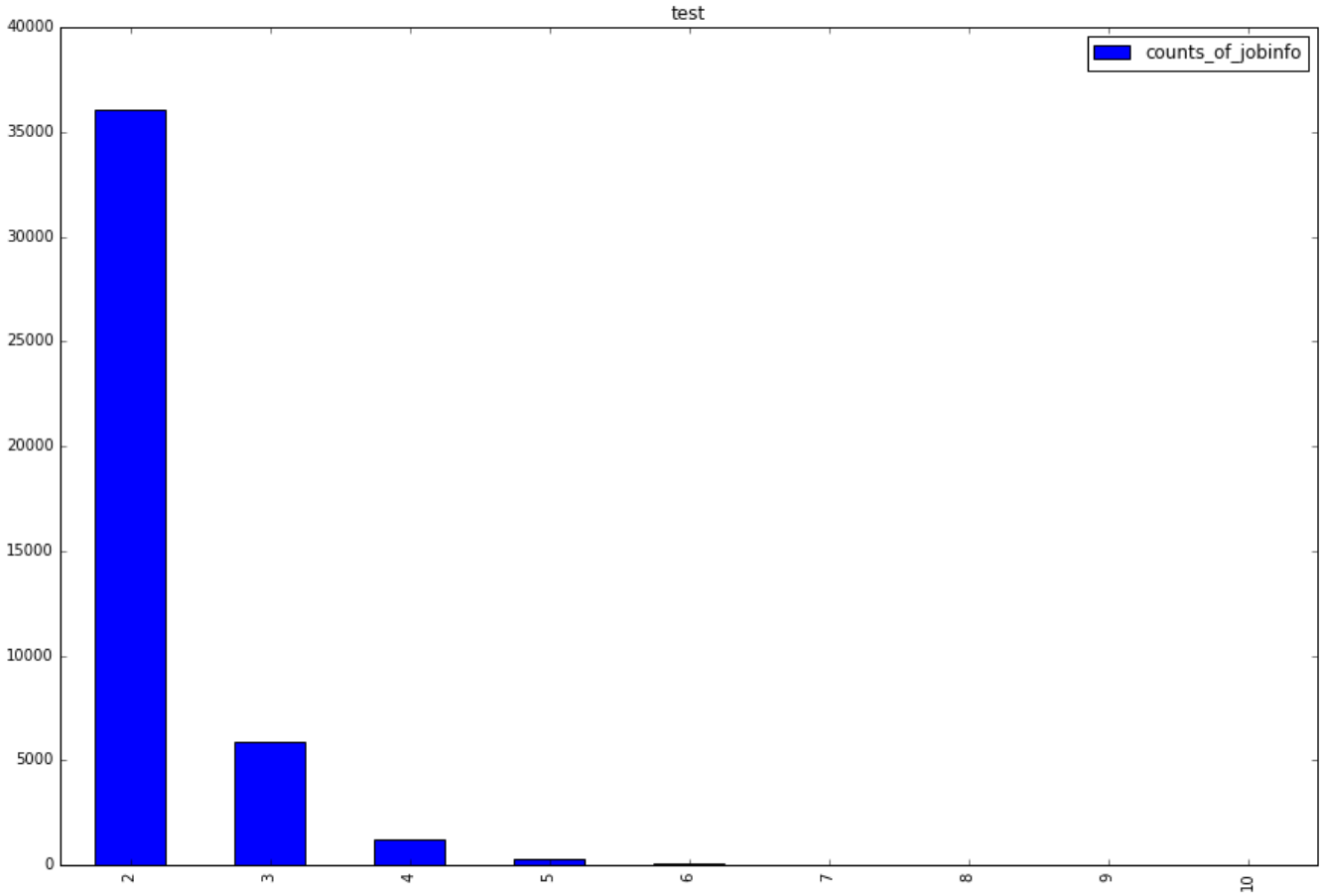
经统计：

1. 在低敏感度用户中，有2815个用户是没有表2信息的，其中训练集1548个（全部是非敏感用户），测试集1267个，这部分用户属于数据缺失，将其剔除；另外有6个样本的REQ_BEGIN_DATE > REQ_FINISH_DATE，属于数据异常，也将其剔除掉。
2. 在高敏感度用户中，有224个用户没有表2信息，其中训练集181个（全部是非敏感用户），测试集43个；加上另外6个REQ_BEGIN_DATE > REQ_FINISH_DATE的异常样本，一并将其剔除。

3.2.高敏感度用户部分

在高敏感度用户群体中，训练集与测试集的样本分布情况并不一致：


train
counts_of_jobinfo
160000
140000
120000
100000
80000
60000
40000
20000
0
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 32 33 34 35 36 37 38 39 40 41 42 43 44 45 47 48 49 50 51 52 53 60 61 66 70 71 83 114


test
counts_of_jobinfo
40000
35000
30000
25000
20000
15000
10000
5000
0
2 3 4 5 6 7 8 9 10

训练集中，用户工单记录数量的值域为[2,114]，而测试集仅为[2,10]。考察工单记录非常多的这部分用户，大部分的工单内容都是“测试工单”，内容形如“本工单为分中心测试工单。请以咨询办结处理，并回拨或腾讯通通知测试人员，告知测试人员咨询办结工单号”、“【回访多媒体工单】本工单为分中心测试工单。请以咨询办结处理，并回拨或腾讯通通知测试人员，告知...”等，该部分数据并未在测试集中出现。如果将全部训练集都参与模型训练，势必会影响模型训练精度。经实验验证，我们将训练集中工单数量大于10的样本视为离群点，将其剔除。

4.特征工程

4.1.95598工单信息

BUSI_TYPE_CODE、URBAN_RURAL_FLAG、CITY_ORG_NO

对于这三种类别特征，特征构造方式相同：

用户类型	**特征构造方式**
低敏感度用户	one-hot独热编码
高敏感度用户	*bag-of-category*（类似于bag-of-words方式，权值取对应类别的数量和比例）

ORG_NO

供电单位编码，经过统计，该编码包含4种形式：

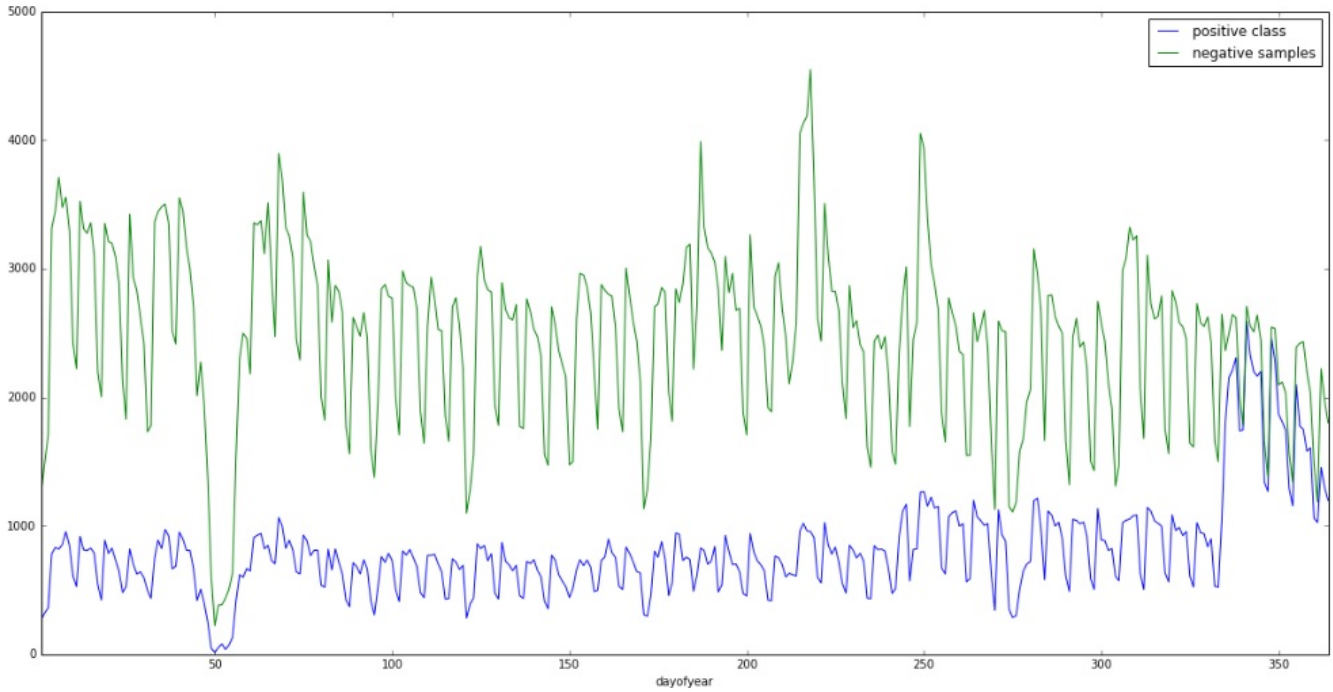
- 12个一级编码，由5位数字组成：33401，33402，.... ,33410， 33411， 33420；
- 75个二级编码，由7位数字组成：前缀（33401，33402，.... ,33410， 33411）；
- 96个三级编码，由9位数字组成：前缀（33401，33402，.... ,33410， 33411）；
- 1个四级编码，由11位数字：33406400142。

ORG_NO的长度反映了用户的供电单位级别，因此对ORG_NO和ORG_NO的长度分别构造特征：

用户类型	特征构造方式
低敏感度用户	对于ORG_NO各类别使用hash trick转换成该类别下正样本的比例；对于ORG_NO的长度使用one-hot表示
高敏感度用户	分别对ORG_NO和ORG_NO的长度使用bag-of-category表示方式

HANDLE_TIME

按日期统计训练集中的用户数量，正负样本分别统计，得到如下的曲线图，横坐标是日期（20150101至20151231），纵坐标是对应每天的用户数量。蓝色曲线是正样本的数量，绿色曲线是负样本的数量。



可以发现敏感用户在全年的分布是十分有规律的，比如二月份样本总量很小，正样本的数量也非常少；12月份的正样本明显增多；另外样本数量按照10天左右为一个周期进行变化。

考虑到敏感用户跟时间线有关，我们对HANDLE_TIME构造了多种表现优异的特征：

- 低敏感度用户

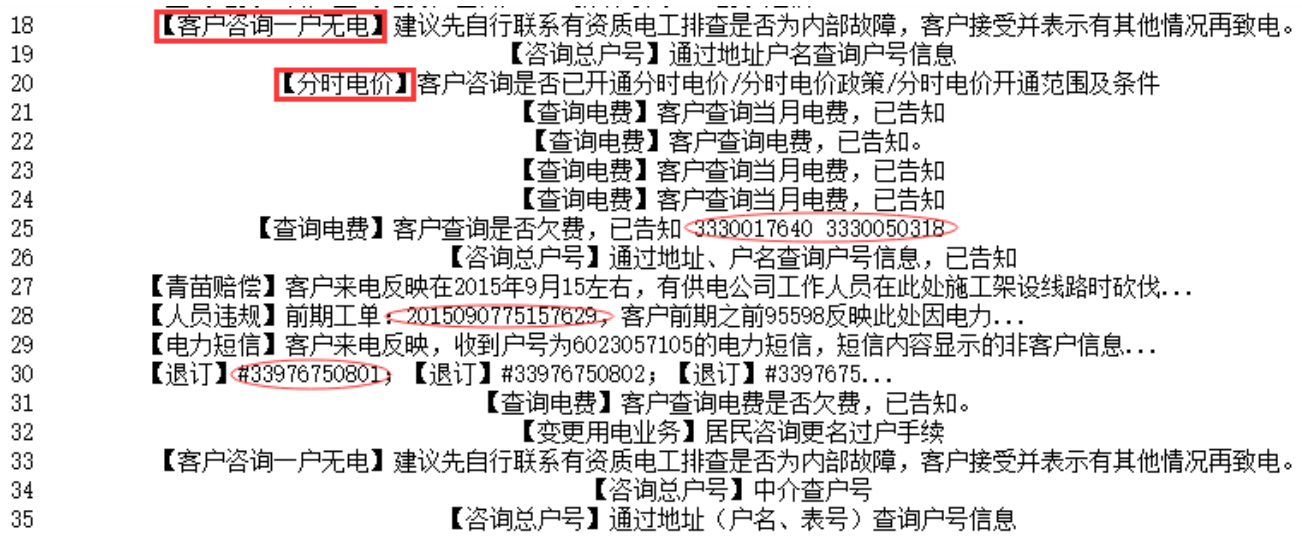
```
3个类别特征：月份、一个月的第几天、小时
3个二值特征：是否是上旬、是否是中旬、是否是下旬
```

- 高敏感度用户

```
月、天、小时三种粒度的bag-of-category
统计用户有几个不同的日期
第一条记录和最后一条记录间隔几天
平均几天会有一条记录
平均一天有几条记录
各记录之间的最大、最小、平均间隔
对各间隔取标准差和中位数
最多一个月有几条记录
记录日期dayofyear的标准差
```

ACCEPT_CONTENT

该属性记录了每一条95598工单客户所反映的内容，对于判断用户意图，识别电费敏感用户起到了十分重要的作用。下图展示了部分工单的文本信息。



对于此部分信息，我们做了如下处理：

> - 使用jieba分词工具对文本进行分词，并且去掉停用词
> - 添加自定义词典以改进分词效果，添加词示例：
>
>   `'户号', '分时', '抄表示数', '工单号', '空气开关', '脉冲灯', '计量表', '来电'`
> - 注意到文本中有很多的数字，例如“2015090775157629”、“33976750801”等等，这些数字各不相同，显然对于特征表示没有任何帮助，因此我们总结了文本出现的数字类型，将它们分别按照对应格式映射成“手机number”、“户号

> number”、“退订number”、“工单number”、“停电number”5类，这样也大大降低了文本特征的维度
> - 将每条文本开头的“【topic】”提取出来作为该文本的类别标识，使用hash trick得到了每一类【topic】对应的正样本比例，将其定义为该【topic】下的敏感分数

- 低敏感度用户

```
tfidf特征, ngram=(1,2)
文本长度
文本中用到词的种类数
```

- 高敏感度用户

```
tfidf特征, ngram=(1,2)
平均、最小、最大的文本长度（因为对应着多条工单记录）
文本长度的标准差、中位数
用词种类数
将多条文本【topic】取和、平均值、最大值、最小值、中位数、标准差得到6维数值型特征
```

ELEC_TYPE

用电类别编码规则为3位数字，从100到900，可以看出首位数字相同的属于同一大类，例如100，101，102都是大的工业类用电，所以对此数据需要从编码本身和编码首位数字两方面分别处理、构造特征：

- 低敏感度用户

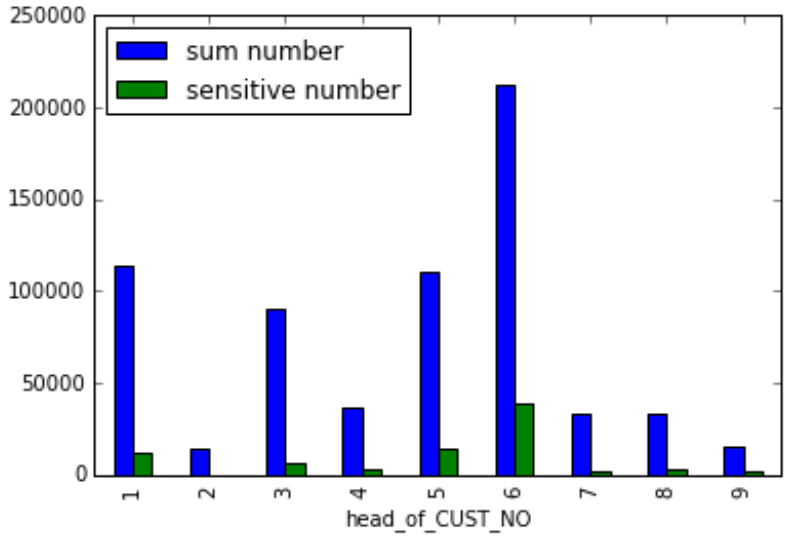
```
对于ELEC_TYPE各类别使用hash trick转换成该类别下正样本的比例
对于ELEC_TYPE的首位数字类别使用one-hot表示
```

- 高敏感度用户

```
ELEC_TYPE ---> bag-of-category
ELEC_TYPE按首位数字聚类 ---> bag-of-category
```

CUST_NO

用户的索引由10位数字组成，首位数字对应的用户群体正样本的比例不同，如下图所示，横坐标表示CUST_NO的首位数字，蓝色表示样本总数，绿色表示正样本数量。



显然用户索引中蕴含着某种信息，我们将用户CUST_NO从小到大排序，得到1维的排序特征。

4.2.客户通话信息

通话时间

使用REQ_FINISH_DATE减去REQ_BEGIN_DATE可以得到每条工单记录的通话时间，即holding_time，通话时间的长短反映了通话内容的不同，该特征在最终的模型重要性输出上得到了很高的分数。

- 低敏感度用户

```
holding_time 归一化
```

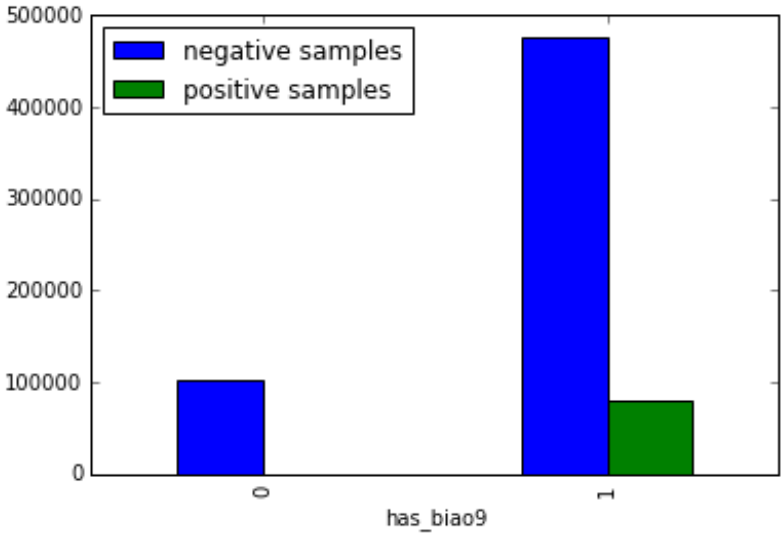
- 高敏感度用户

```
通话记录数量
min_holding_time_seconds    最小值
max_holding_time_seconds    最大值
sum_holding_time_seconds    总数
std_holding_time_seconds    标准差
```

```
median_holding_time_seconds 中位数
mean_holding_time_seconds   平均数
```

4.3.应收电费信息

虽然该部分的数据缺失率有26.40%，但是经过统计发现，在训练集中有102625用户没有应收电费信息记录，而他们全都是负样本，如下图所示，横轴0代表没有应收电费信息记录，1代表有。



此外表9中的电费、违约金等信息对于判断用户是否电费敏感也十分重要。无论是低敏感度用户还是高敏感度用户，对应的应收电费信息都包含多条记录，所以对这两类用户采用了相同的特征构造方式：

```
（二值特征）是否有应收电费信息记录
应收电费信息记录的数量
应收金额（最大、最小、总数、平均、标准差）
实收金额
应收金额减去实收金额
总电量（最大、最小、总数、平均、标准差）
电费金额（最大、最小、总数、平均、标准差）
违约金（最大、最小、总数、平均、标准差）
应收违约金和实收违约金做差
每个用户有几个月的记录
平均每个月几条
```

5.特征选择

在特征工程部分，我们对低敏感度用户和高敏感度用户分别构建了文本的unigram+bigram特征，分别达到了2W维和4W维，这么多维特征一方面可能会导致维数灾难，另一方面很容易导致过拟合，需要做降维处理，常见的降维方法有PCA和SVD，这类方法的计算复杂度比较高，并且在文本类的高维稀疏特征上的效果并不是很好。

除了降维还可以采用特征选择的方法来降低特征的维度，这里我采用了一种比较高效的方法，基于xgboost模型来做特征选择，xgboost模型训练完成后可以输出特征的重要性，据此我们分别保留了200和800维的文本特征。下图展示了我们保留的重要程度高的文本特征：

6.模型设计

在各种的机器学习算法中：

- 逻辑回归（LR）因为其训练高效，训练结果的可解释性强等特点而在互联网领域应用最为广泛
- 基于决策树（DT）的树模型，因为模型结构而不需要对数据进行预处理，数据的准备相对简单

- 随机森林（RF）和梯度提升决策树（GBDT）分别采用bagging和boosting的模型融合方式获得不错的表现
- xgboost是boosted tree的一种实现，效率和精度都很高，在kaggle等各类数据挖掘竞赛中成为大杀器

✎ 对于低敏感度用户，训练集40W，测试集32W，为了模拟线上提交，我们按照测试集的比例从训练集划分18W作为验证集，剩下22W作为训练集，供模型调参使用。
✎ 对于高敏感度用户，训练集25W，测试集4W，直接从训练集抽取4W作为验证集，剩余21W作为训练集。

各模型的结果f1值如下：

模型	**低敏感度用户**	**高敏感度用户**
LR	0.716	0.780
DT	0.752	0.780
RF	0.810	0.784
GBDT	0.857	0.812
xgboost	**0.901**	**0.834**

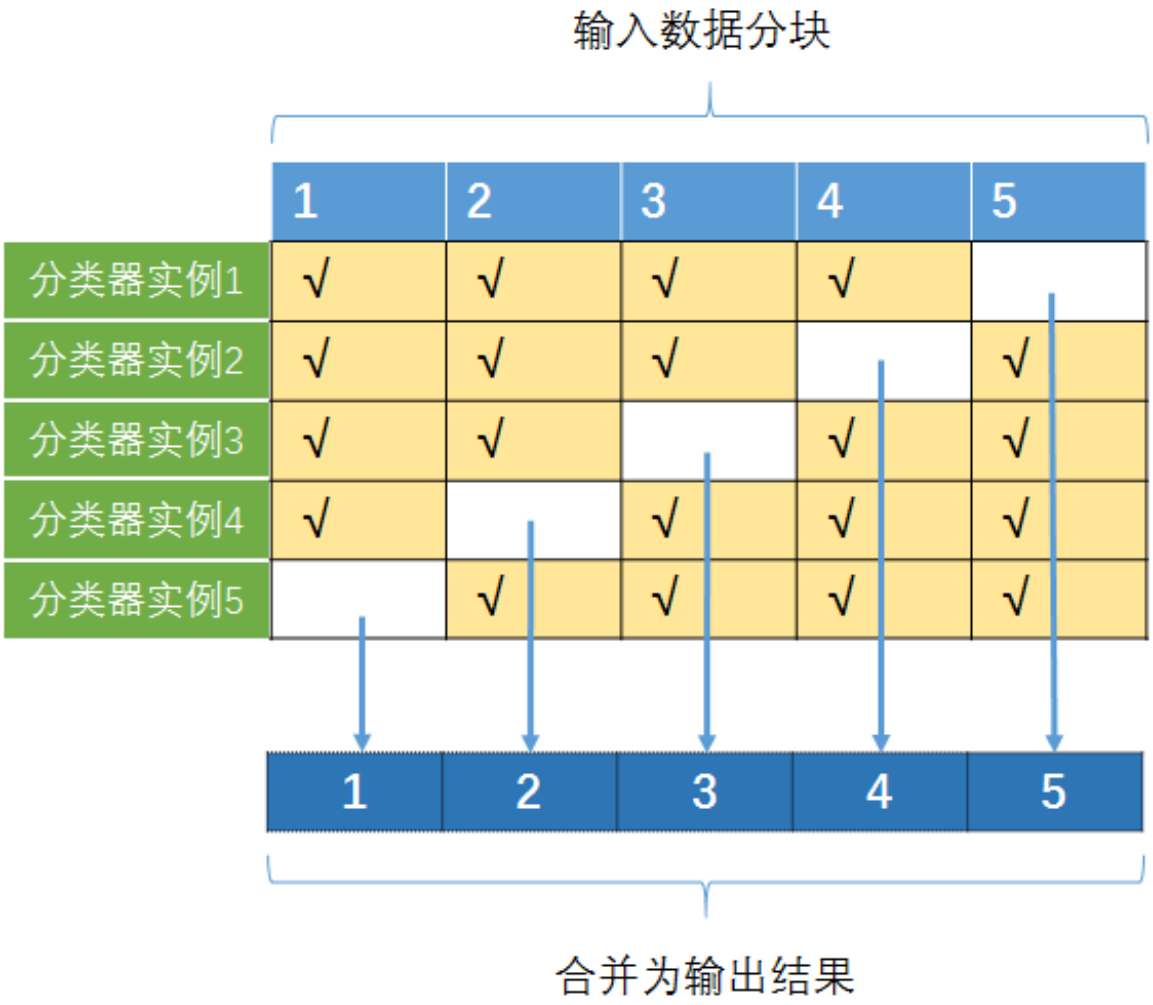
从结果可以看出单模型方面xgboost的表现最佳，因此我们一直使用它作为基础模型更新迭代不同的特征版本。

7.模型融合

模型融合通过将多个学习器进行结合，常可以获得比单一学习器显著优势的泛化性能。模型融合可以带来两个方面的好处：

1. 从统计的方面来看，由于学习任务的假设空间往往很大，可能有多个假设在训练集上达到同等性能，此时若使用单学习器可能因为误选而导致泛化性能不佳，结合多个学习器则会减小这一风险；
2. 从计算的方面来看，学习算法往往会陷入局部极小，有的局部极小点所对应的泛化性能可能很糟糕，而通过多次运行之后进行结合，可降低陷入糟糕局部极小点的风险。

在kaggle等高水平的数据挖掘大赛中，诸多冠军队伍都采用了模型融合技术，常见的方法有bagging和stacking。bagging指多个模型进行投票；而stacking方式如下：



如图所示，将训练数据集分为5块并每一块保证不产生id重叠，对于同一个模型，产生5个分类器实例，每个实例使用不同的1个数据块作为验证集，对应的其余4个数据块作为训练集。每个实例针对自己的验证集都可输出一个结果，并且这5个结果最终又可合并为整个数据集的大小。从而实现了所有数据从输入特征到输出特征的特征变换，而且因为每个分类器预测的数据块均未参与到该分类器的训练，因此可以有效防止过拟合的发生。

我们对单模型、bagging和stacking三种方式进行了对比，其中单模型采用xgboost；bagging采用5个xgboost（设置不同的随机种子）进行投票；stacking设置LR、RF、ExtraTreesClassifier、xgboost作为基分类器，LR作为二级分类器，实验结果如下所示：

融合方式	低敏感度用户	高敏感度用户	线上得分	训练速度
单模型	0.901	0.834	0.89538	快

融合方式	低敏感度用户	高敏感度用户	线上得分	训练速度
bagging	0.913	0.845	**0.90379**	中
stacking	0.913	0.840	0.89913	慢

综合考虑模型预测的准确性和训练速度，最终我们采用bagging方法作为最终的模型融合方式。