华为DIGIX CTR赛题冠军分享

kaggle竞赛宝典 今天

以下文章来源于Coggle数据科学 ，作者lllllllyu

Coggle数据科学

Coggle全称Communication For Kaggle，专注数据科学领域竞赛相关资讯分享。

↑↑↑关注后"星标"kaggle竞赛宝典

每周干货，不错过

kaggle竞赛宝典

作者：lllllllyu, 转自Coggle公众号

赛题名称：华为DIGIX算法大赛A赛题（广告CTR预估 & 搜索相关性预测）

赛题链接：
https://developer.huawei.com/consumer/en/activity/devStarAI/algo/competition.html#/preliminary

赛题类型：机器学习、CTR

分享内容：**冠军思路+代码**、比赛数据

分享原文：https://blog.csdn.net/weixin_40174982/article/details/109802534

写在前面

华为赛（链接）终于是结束了。今年由于疫情原因，线下决赛搬到了线上进行，答辩的时候才去南京。决赛打榜在我们熟悉的环境进行，也给了足足5天时间，所以对我们还是有点好处的。

最后队友比较给力，稳住了初赛的成绩，南京答辩也一切顺利，拿到了最后机器学习赛道的冠军。这里偷偷插一句，现场答辩的时候我还是很亢奋的，结果到最后颁奖的时候就紧张了，担心会被逆袭，还好最后的结果还是好的，哈哈哈。

比赛结束后自然是要分享和开源方案。此前初赛结束的时候就分享过搜索相关性题目的方案和开源代码。这里就来讲一下我们ctr题目的方案，整体顺序将依托于我们答辩ppt进行。

比赛成绩

本次华为赛的机器学习赛道在初赛的时候包含两个题目： CTR预估 和 搜索相关性预测 。

我们在两个题目的A/B榜上都保持了长久的第一，其中CTR预估题目的第一名从8月24日持续到9月30日，搜索相关性预测题目的第一名从8月25日持续到9月30日。

决赛的时候只做CTR题，在A榜阶段我们掉到了第二，B榜的时候才回到第一。其实当时A榜的时候感觉都要凉了，还好最后B榜给机会了(￣▽￣)"

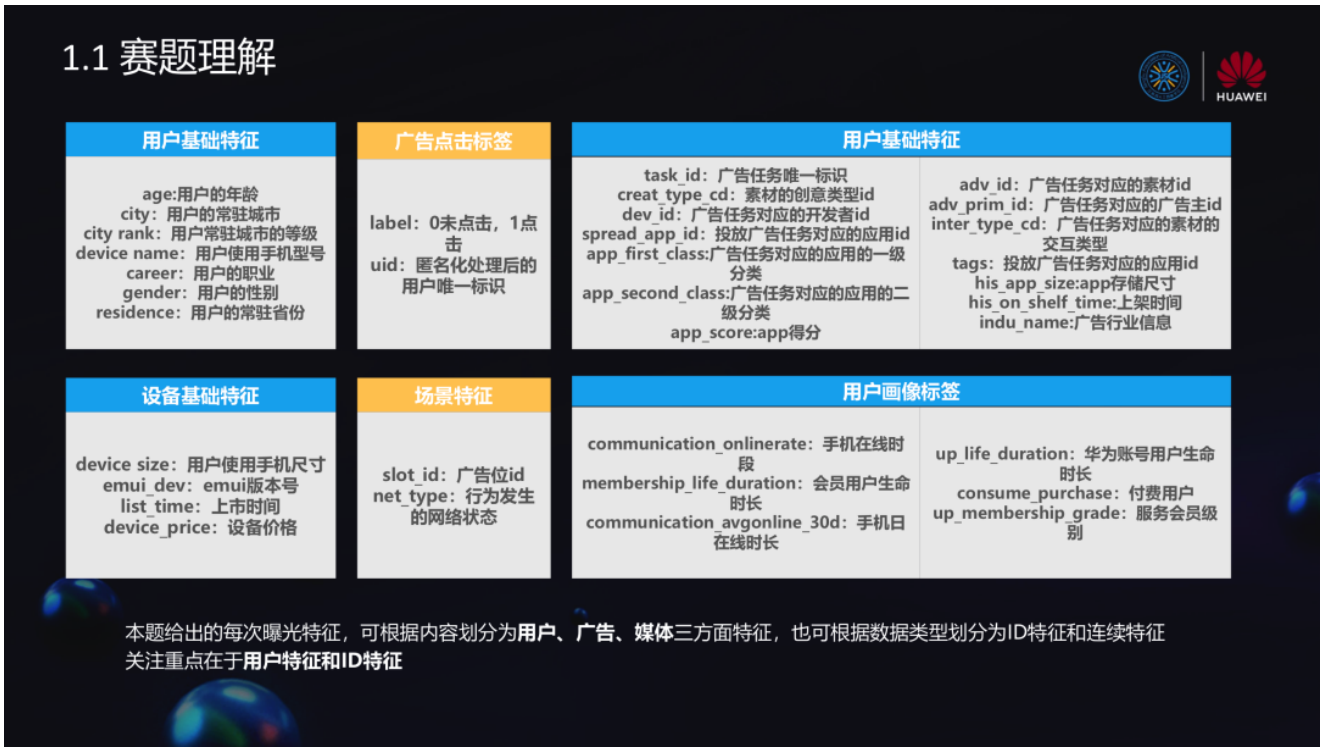
基础方案

赛题理解

本次 CTR预估题目 非常传统，给定前7天内每条曝光的点击行为，预测将来某一天内曝光的点击率，评价指标AUC。数据集划分如下：

- 初赛阶段：训练集1-7天，A榜测试集第8天，B榜测试集第9天。
- 决赛阶段：训练集1-7天，A榜测试集第8天，B榜测试集第10天。

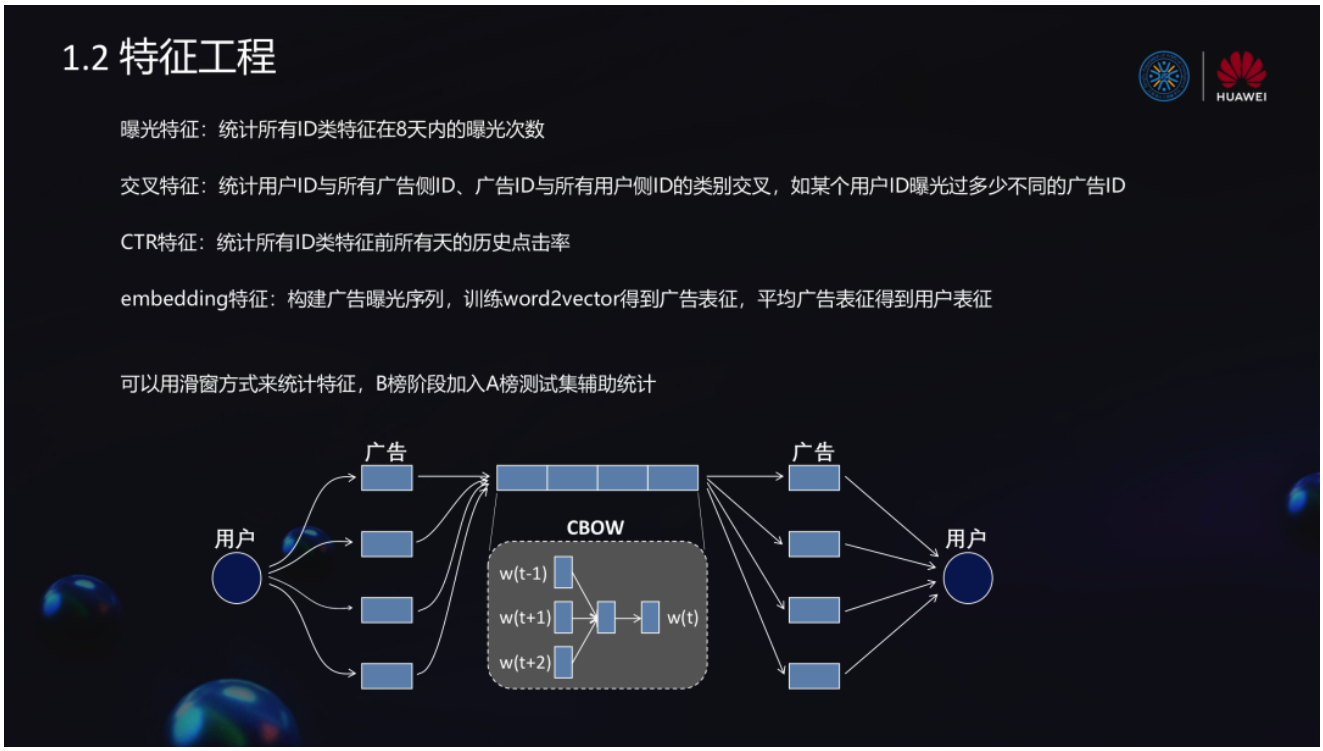
具体的特征可以看一下官网的描述，我们这里稍微统计了一下，将特征根据内容划分为用户、广告、媒体三方面特征，也可根据数据类型划分为ID特征和连续特征。



其中，关注的重点在于用户特征和ID特征。

特征工程

我们使用的特征工程非常的常规，以至于大家看完可能都会说一句：就这？包含四个特征：**曝光特征、交叉特征、CTR特征、embedding特征**：



- 曝光特征 ：统计所有ID类特征在8天内的曝光次数（即count特征）
- 交叉特征 ：统计用户ID与所有广告侧ID、广告ID与所有用户侧ID的类别交叉，如某个用户ID曝光过多少不同的广告ID（即nunique特征）
- CTR特征 ：统计所有ID类特征前所有天的历史点击率
- embedding特征 ：构建广告曝光序列，训练word2vector得到广告表征，平均广告表征得到用户表征

对于这些特征，还可以采取用滑窗的方式来统计，效果不一定更好，不过会有一定差异性。

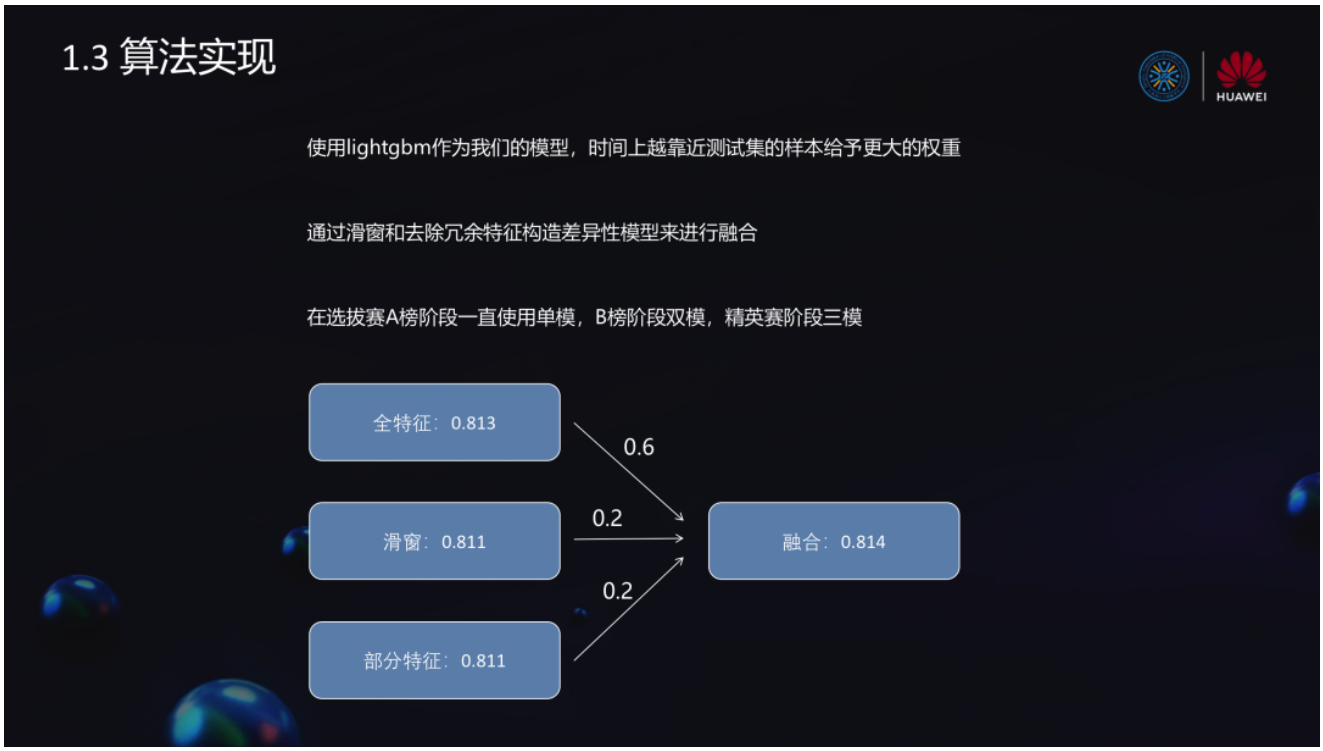
在B榜阶段，由于测试集是第9天或者第10天，与前7天的训练集隔了一到两天，会带来很明显的gap。这里我们将A榜的数据也放进去一起统计特征，就可以显著提升效果。

这里的A榜数据只用于辅助做特征，并不参与训练。我们也试过将之前A榜的预测结果二值化然后当成训练数据来用，结果就会带来非常严重的过拟合。

我们的方案经过优化之后，占用内存不到20G，目前看来应该是最轻量级的方案。

算法实现

我们使用lightgbm作为模型，在实际训练的时候给予时间上越靠近测试集的样本给予更大的权重。具体来说，第7天的样本权重是1，第2天样本的权重就是2/7。



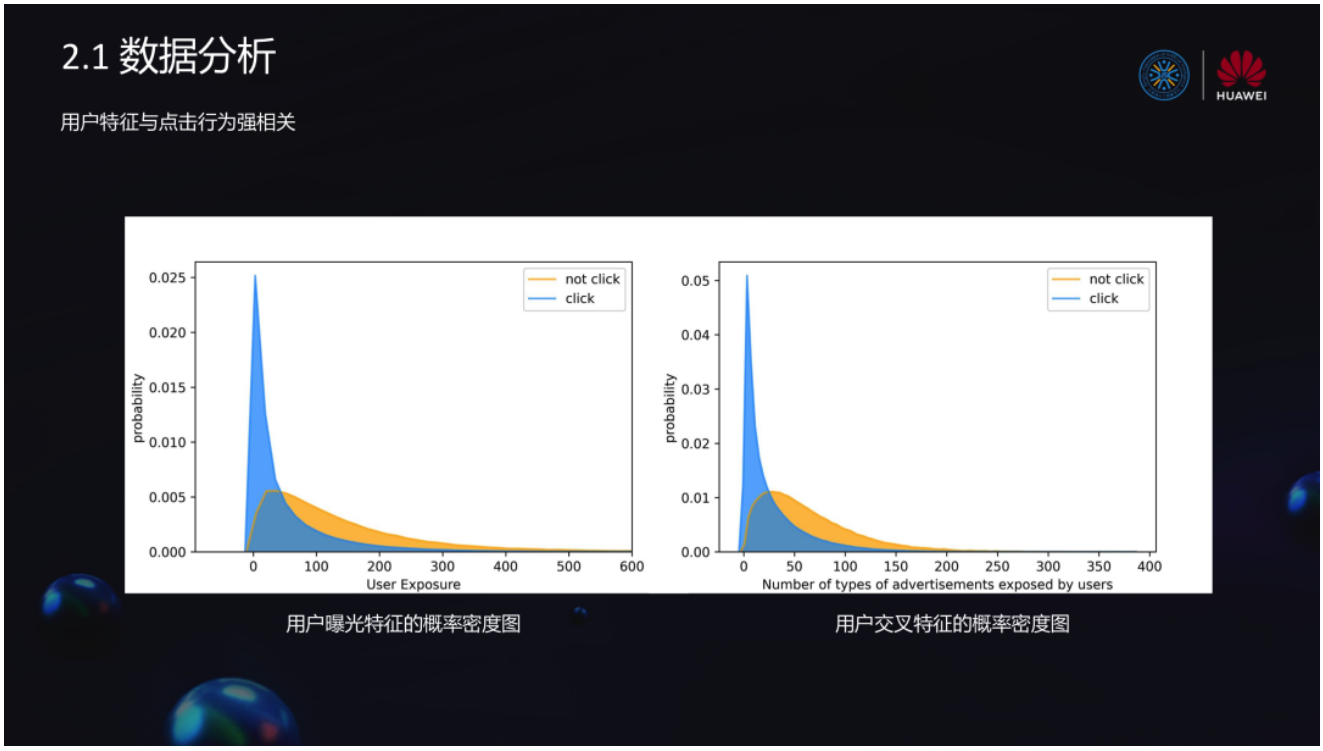
我们通过滑窗和去除冗余特征构造差异性模型来进行融合，在初赛A榜阶段一直使用单模，B榜阶段双模，决赛阶段三模。下图是我们决赛B榜的分数，其实我们单模就已经到了0.8137，刚好比第二的分数高了一个千。

大家看到这里是否觉得我们的方案平平无奇？不用急，下面才是我们工作的重心😉

冷启动探索

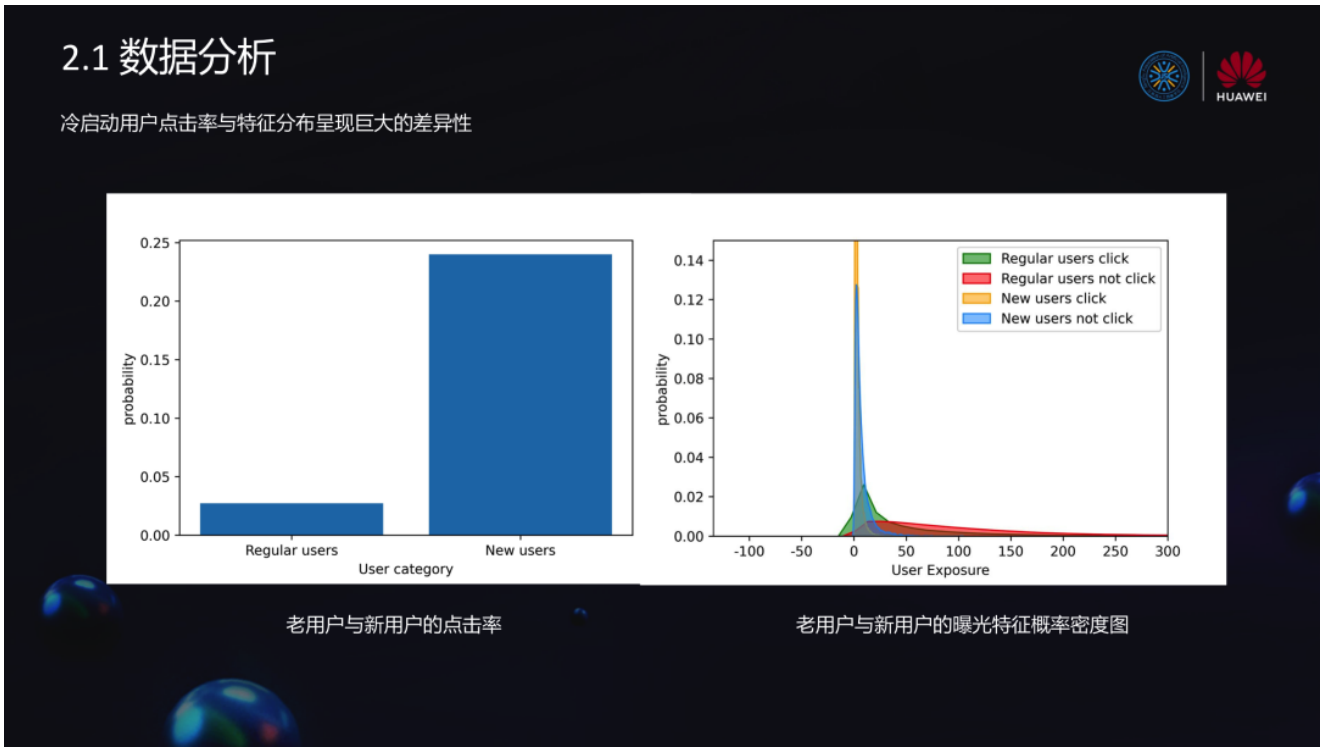
数据分析

我们首先把用户的曝光特征以及交叉特征的概率密度图画出来。



可以很明显的看到，点击与不点击样本的用户特征分布差异非常明显，这说明了本题中用户特征与点击行为强相关，这也验证了我们一开始的以用户特征为重点的想法。

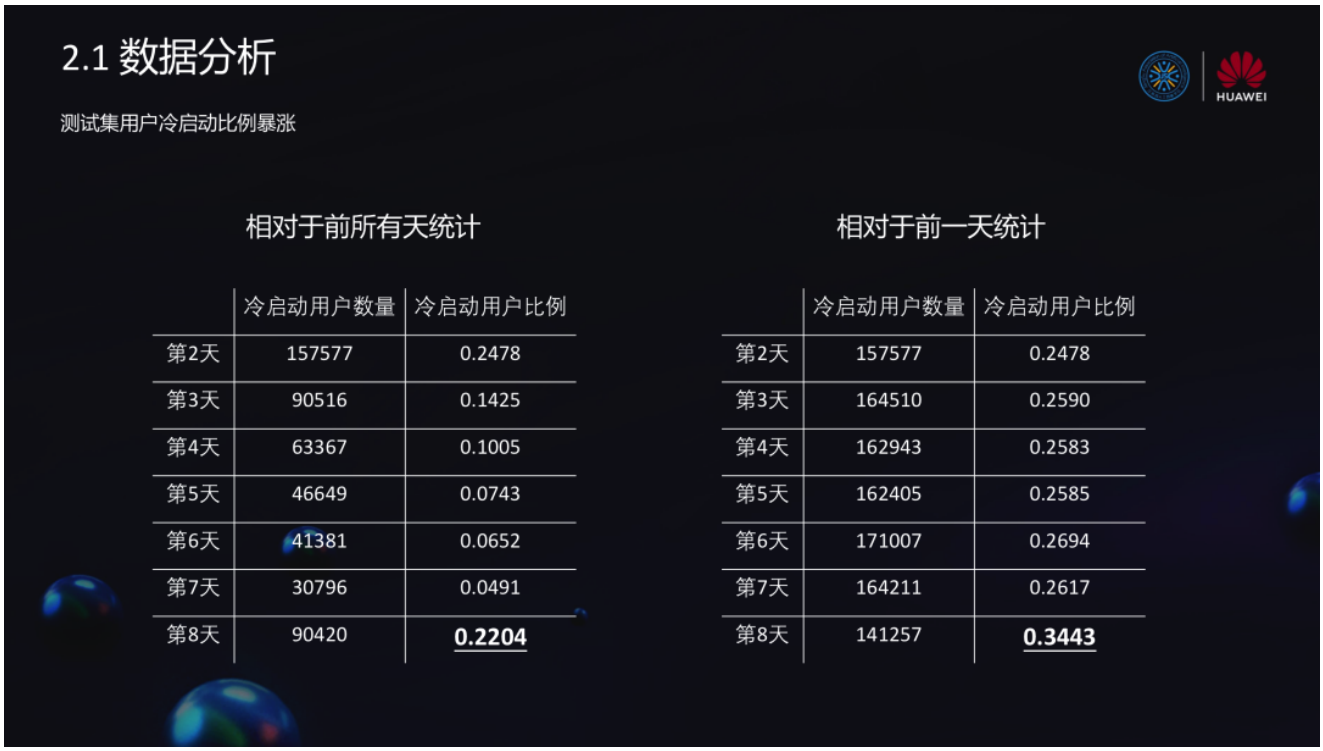
接着来看一下用户冷启动的情况。这里统计了第7天训练集中冷启动用户的情况。左图是冷启动用户与老用户的点击率差别，冷启动用户的点击率非常高，是老用户的10倍。虽然我觉得这种现象非常奇怪，但是在本题的数据中，只能认为这个广告场景中冷启动用户的特性特别明显，以至于广告系统能够很好把握。



右图是两种用户群体的特征分布，两条尖的曲线就是冷启动的分布。两个图的结合能够说明冷启动用户点击率与特征分布呈现巨大的差异性，也就是说，只要模型能够正确区分出冷启动用

户，就会有很高的分数。

接下来统计测试集的用户冷启动情况。这里以第八天的测试集为例统计。



	冷启动用户数量	冷启动用户比例
第2天	157577	0.2478
第3天	90516	0.1425
第4天	63367	0.1005
第5天	46649	0.0743
第6天	41381	0.0652
第7天	30796	0.0491
第8天	90420	**0.2204**

	冷启动用户数量	冷启动用户比例
第2天	157577	0.2478
第3天	164510	0.2590
第4天	162943	0.2583
第5天	162405	0.2585
第6天	171007	0.2694
第7天	164211	0.2617
第8天	141257	**0.3443**

左图为相对于前所有天的统计结果，随着时间的推移，前七天冷启动比例的统计结果逐渐稳定到0.05，但是第八天出现了暴涨，直飙0.22。考虑到相对前所有天统计是不公平的，我们继续统计了相对前一天的冷启动，发现前七天都处于比较稳定的状态，而第八天依然暴涨。至此，我们确定了，第八天测试集中的冷启动用户确实比例非常的不正常。

然而，冷启动只有比例不正常吗？我们继续对这个群体探索。

新用户异常

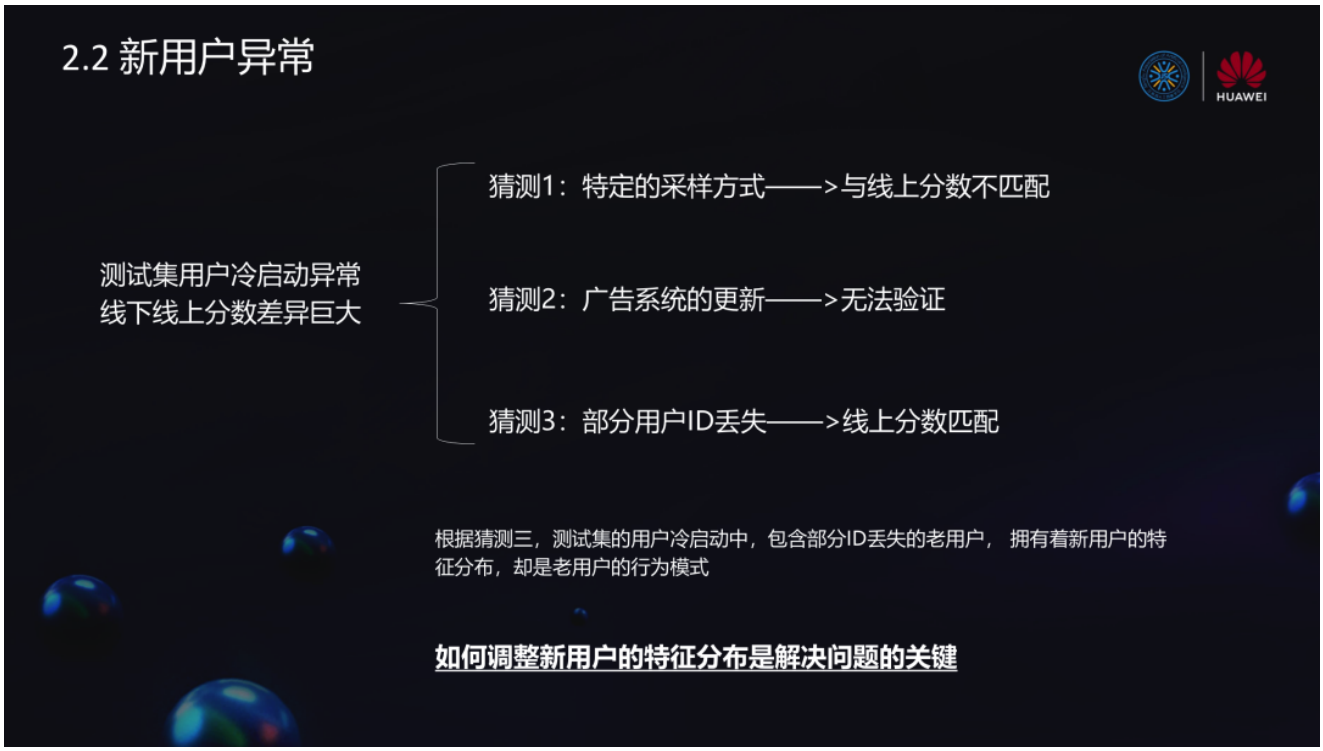
刚刚有提到，用户的特征非常重要，其中最重要的一个特征是曝光特征。但是之前的曝光特征是八天一起统计的，老用户和冷启动用户自然gap非常大。这里，我们将每一天随机采样到一定的总曝光量，然后统计不同用户群体的日曝光，以观察它们特征层面的差别。

	老用户日曝光	冷启动用户日曝光
第2天	3.1581	1.7704
第3天	3.0466	1.6316
第4天	2.9012	1.5452
第5天	2.8454	1.5070
第6天	2.8517	1.5274
第7天	2.8578	1.4983
第8天	2.5307	**2.1773**

	老用户日曝光	冷启动用户日曝光
第2天	3.1581	1.7704
第3天	3.3022	1.8363
第4天	3.2297	1.8062
第5天	3.2405	1.8146
第6天	3.2824	1.8598
第7天	3.3251	1.8471
第8天	2.9030	**2.0186**

从日曝光上可以看到，老用户的日曝光一般都接近冷启动用户日曝光的两倍。但是第八天中，冷启动日曝光再次暴涨。这点就让人非常奇怪，似乎有种感觉，冷启动用户里面混入了部分老用户，因而拉高了它们的日曝光。

有跑过这个数据的同学都知道，线上线下分数的差别非常巨大。目前看来，这么大的gap原因是测试集中冷启动用户在比例和日曝光上和训练集的差异。我们对这个差异出现的原因进行了分析，提出了几个猜想：



- 猜测一 ：测试集经过了特定的采样。我们将第七天当作验证集，然后尝试去采样成第八天的比例和日曝光，但是我们发现，经过采样之后的验证集分数其实下降不大，完全达不到

第八天线上分数的效果，说明这个猜想是错的。

- 猜测二 ：在第七天到第八天中广告系统进行了更新。这个猜想就无法验证了，当然我也觉得出题方不至于这样搞事情。

- 猜测三 ：在第七天到第八天的过程中出现了用户ID丢失的情况，部分老用户在第八天变成了新用户。我不知道在实际场景中是否会出现这种情况，但是只要出题人把老用户的ID改成新的ID就能出现这种效果。我们在第七天尝试了一下，发现是完全可以得到线上分数的效果的。

（这里补充说一下，我一开始十分坚定地认为是出题人改ID导致的，但是后来现场和第二名的老哥沟通后，发现其实删掉前七天的部分老用户，也可以达到一样的效果，即可能出题方是对训练集而不是测试集进行了针对的采样。所以，后面就统一描述为用户ID丢失的情况。）

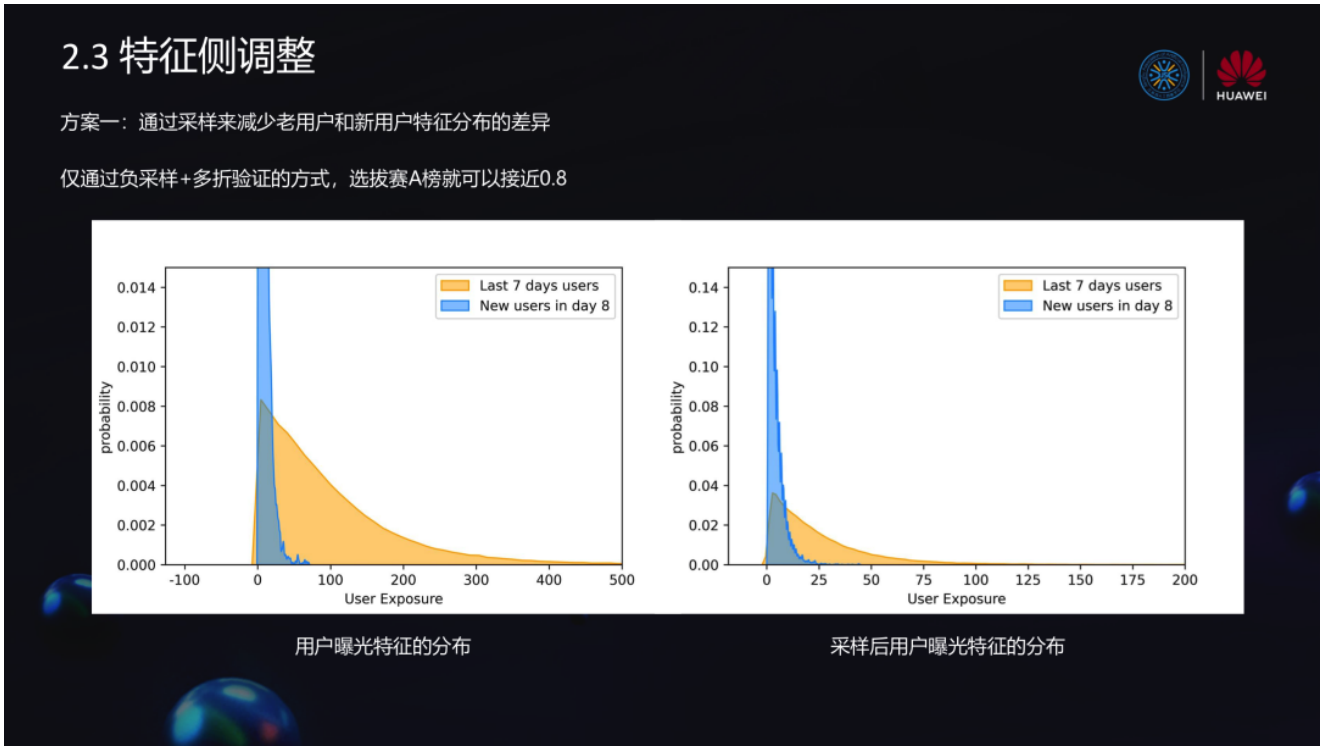
根据猜测三，测试集的用户冷启动中，包含部分ID丢失的老用户， 它们拥有着新用户的特征分布，却是老用户的行为模式。模型遇到这部分用户，会给予一个比较高的点击率，因为训练集中的新用户就是点击率偏高的，然而这部分用户本身应该是较低的点击率，所以就会导致严重的性能下降。因此，如何调整新用户的特征分布是解决问题的关键。

分布调整方案

针对上面提出的问题，我们提出了一系列的解决方案，这些方案可以比较好地解决线上线下不一致的情况，也是我们上分的关键点。

采样

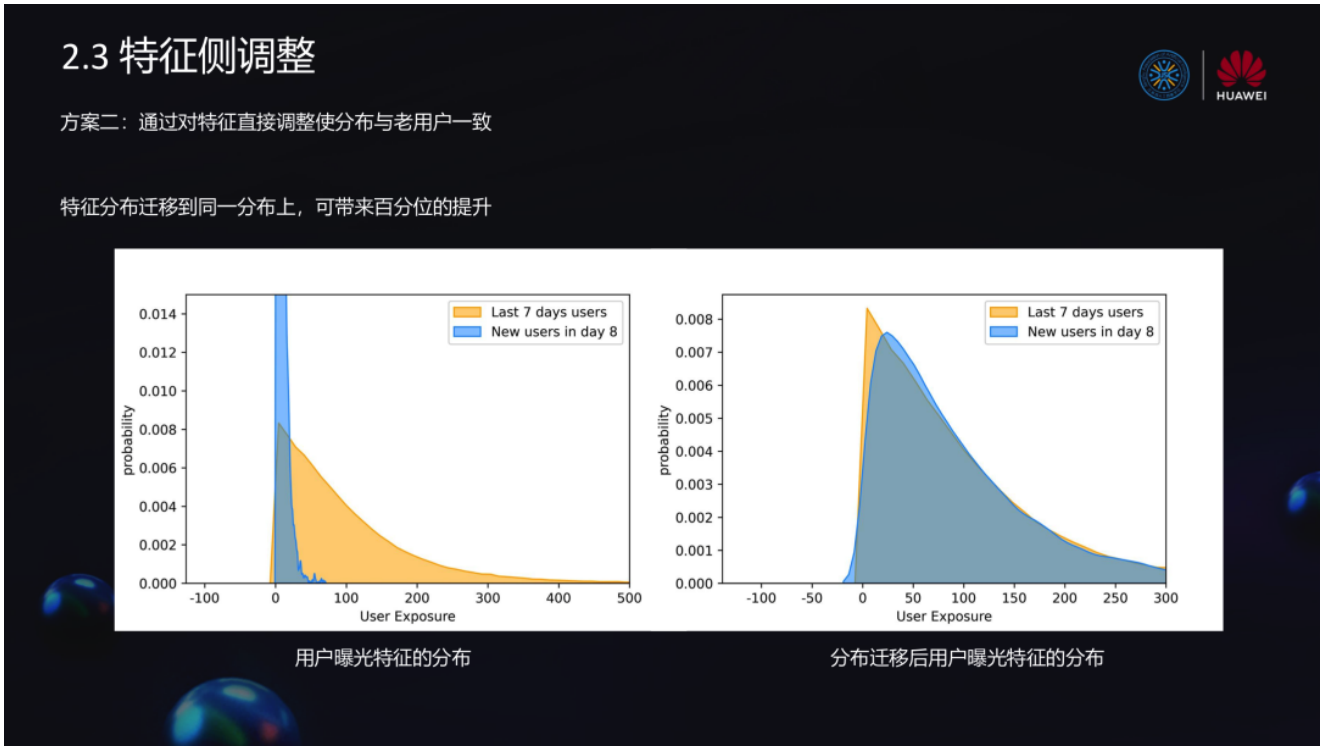
新用户与老用户之间特征分布的差异性，可以很简单地用采样的方式来缓解。

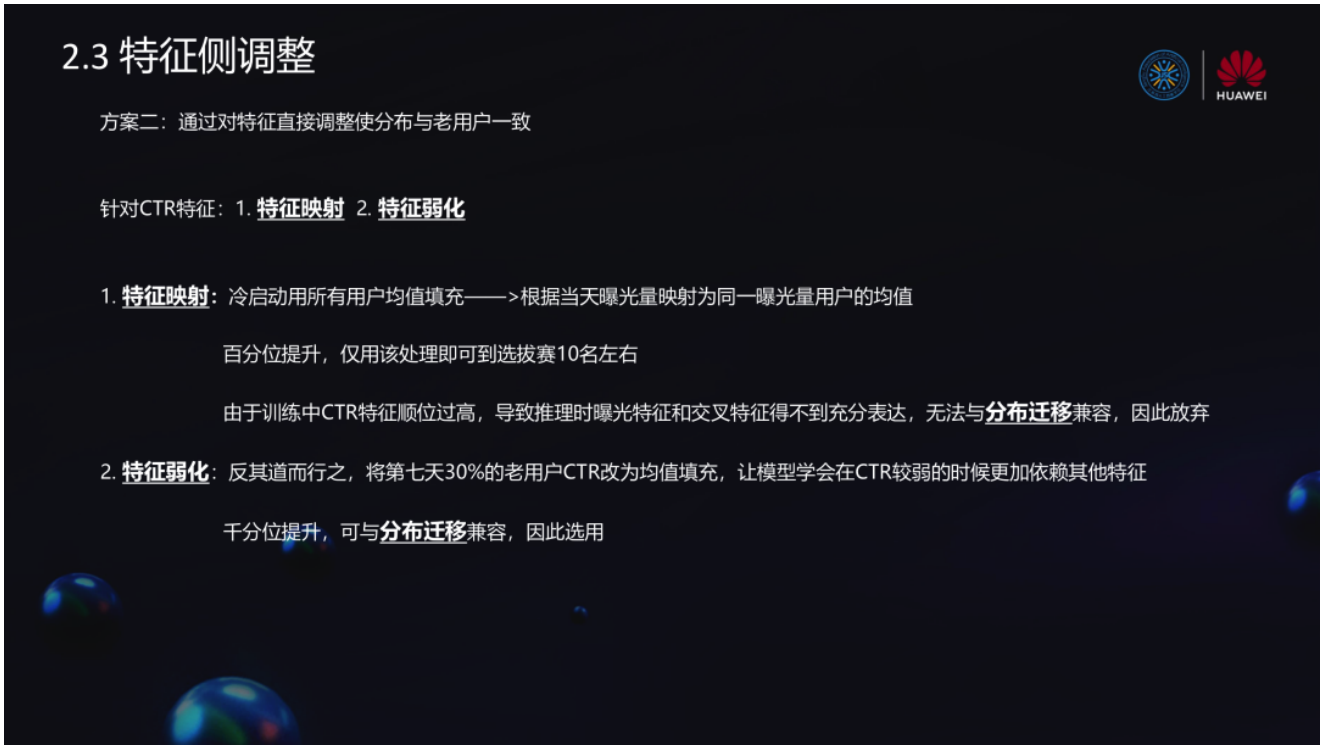


右图是随机采样之后的特征分布情况，新老用户之间的差异被明显降低了。当然，随机采样会导致严重的信息丢失，因此我们最后采用的是负采样。仅仅使用负采样+多折验证的方式，就可以在初赛A榜到达接近0.8的分数。需要注意的是，负采样在这里并不是为了解决类别不平衡问题，而是为了缓解分布不一致问题。

特征调整

我们还对所使用的四个特征分别进行特定的调整，来减少每一个特征带来的gap。





- 分布迁移

对于曝光特征和交叉特征，我们提出一种非常简单的分布迁移方案。这两个特征的概率分布均为长尾分布，无法用特定表达式的参数来拟合，因此我们简单地认为可以用均值和标准差来大致表示分布情况。

然后样本均值为概率分布均值的无偏估计，修正样本标准差为概率分布标准差的无偏估计（直接用无修正的样本标准差也行，毕竟样本标准差是渐进无偏估计），我们对样本计算这两个统计量，然后简单地归一化。注意，我们只对测试集的新用户进行该操作。

可以看到，经过分布迁移之后，新用户的特征分布已经基本拟合前七天了。这个操作背后的想法是测试集的新用户本来应该是老用户，而我们要恢复这部分用户原本的特征。这一个简单的操作，可以给我们带来不止一个百的提升。

- 特征映射&特征弱化

针对CTR特征，我们首先提出特征映射的方案。之前对于冷启动用户的历史点击率，我们使用均值点击率来填充，但是这种操作会带来不少的信息丢失。

更为严重的是，模型看到用户历史点击率为均值的样本，会认为是新用户并且给出比较高的点击率预测，然而实际上应该是低点击率的老用户。

因此我们这里将测试集冷启动用户的点击率根据当天曝光量映射为前七天同一日曝光量用户的点击率的均值。这样子可以得到一个相对准确的CTR特征。这一操作也会有百分位的提升，并且仅用这个操作就可以在初赛A榜达到0.812的分数，预测在B榜应该是10名左右的成绩。

但是由于CTR特征是很强的特征，在训练的时候顺位很高，也就是说会在树模型靠前的地方划分分支。我虽然没有打开看过，但是划分出来的效果应该是一条分支是均值，然后另外很多条分支是基于点击率细致的划分。在推理的时候，如果样本的CTR特征进入了这些细致划分的支路，那么后面出来的预测结果基本跟这个点击率差不多，不会再根据其他特征有较大的调整。这就是一种强特掩盖其他强特的现象，本质上是因为树模型划分节点时分而治之的思想。因此，经过特征映射之后的CTR特征会极大影响最后的输出结果，但是这个特征映射并不能得到绝对准确，不能和前面提出的分布迁移方案兼容，因此我们最后放弃了这个方案。

但是我们反其道而行之，提出一种特征弱化的方案。具体来说，就是将第七天30%的老用户CTR改为均值填充，让模型学会在CTR较弱的时候更加依赖其他特征。换一种说法，就是构造出了一部分点击率是均值的老用户样本，和测试集中的新用户样本对应，让模型学习如何处理这一类样本。这个特征弱化的方案有千分位的提升，并且可以与分布迁移兼容，是最后采用的方案。

- GNN传递

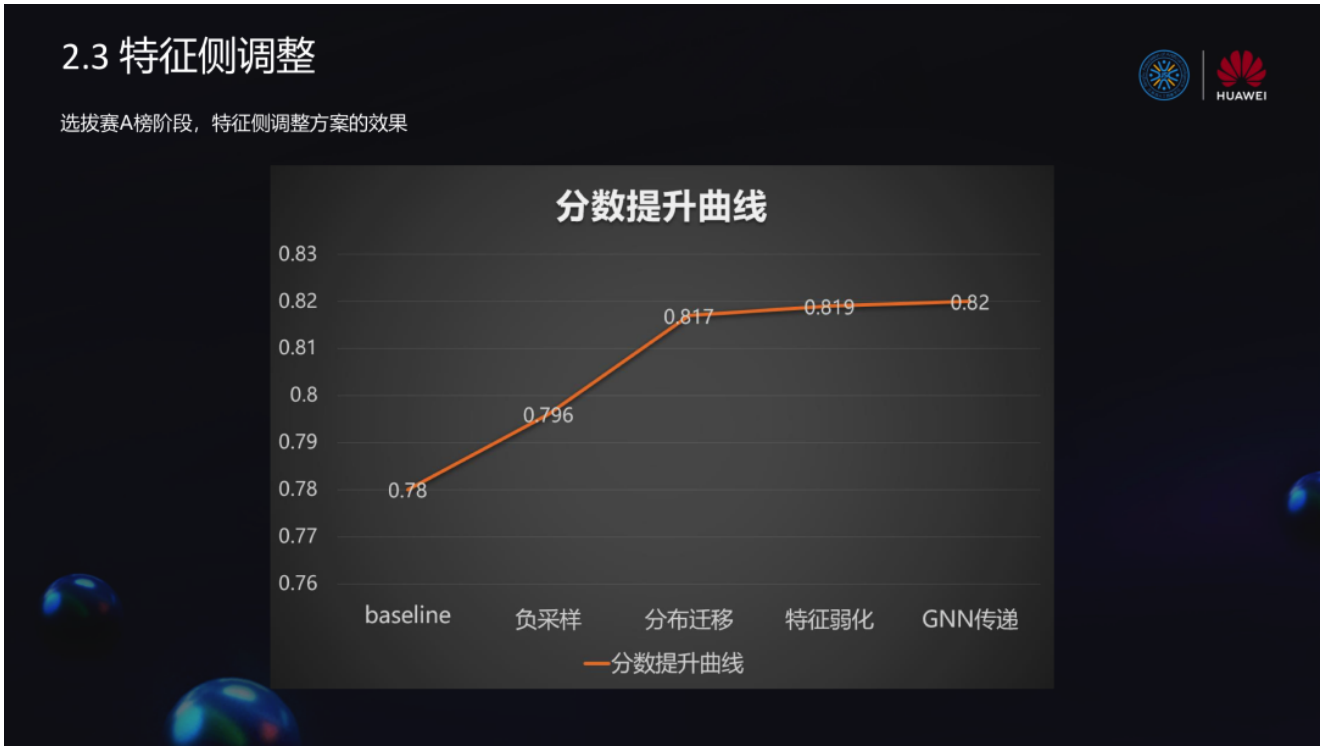
针对embedding特征，我们发现它其实也带来了线上线下的gap。这个主要是因为用户的embedding是由广告的embedding平均而来的，但是第八天的新用户只有第八天的广告曝光信息，而第八天的广告分布和前七天的有差别，所以得到的用户embedding也会和老用户有差别。

为了解决这一个问题，我们参考GNN中GraphSAGE的消息传递思路，将特征在用户与广告之间再传播一轮，从而联系上前7天的其他广告特征。

操作其实非常简单，就是用户的embedding再次平均到广告，然后广告再次平均到用户。当然，大家都知道GNN有非常严重的特征平滑问题，而且我们这里与普通的GNN比只有聚合没有特征映射，平滑更严重了。所以，最后这个方案是只有万分位提升，幅度并不大。

特征调整效果

上面我们针对如何调整新用户分布的问题提出了一些方案，包括负采样、分布迁移、特征弱化和GNN传递。它们在初赛A榜的上分曲线如下所示。



可以看到我们针对性提出的方案都是有提升的，而其中提升最多的是负采样和分布迁移。

总结&不足

总结

我们的方案仅包括四大基础特征：曝光特征、交叉特征、CTR特征和embedding特征，亮点在于针对新用户的分布问题提出了两大方案：负采样和特征调整，其中特征调整又分为三个方案：分布迁移、特征弱化和GNN传递。

不足

其实我们工作比较明显的不足，一个是没有用深度模型（没有做出好效果），一个是模型差异度不够，融合上分很少。

然后是一些没有做出来的东西，首先是我们一直以来就有的想法，那就是既然新老用户的特征分布差异巨大，那么是否分别用不同的模型来处理两批用户。

但是由于最后的指标AUC是一个排序指标，两个模型可能可以使得两批用户的内部排序更优，却很难解决相互排序的问题。决赛有其他队伍采用了两个模型的方案，通过折线映射来使得两个模型的输出平衡，但是我觉得这种方案应该不太稳定。一个可能的方案是还需要第三个整体效果比较好的模型，用它来指导两个模型的融合。

上面也提到了GNN的平滑性问题，我们在后面有想过用self-attention代替均值聚合，似乎可以缓解一下平滑情况，但是后面没有时间和机会去尝试了。除此之外，还有尝试增量训练，点击率贝叶斯平滑、graph embedding等，都没有做出来。答辩的时候跟其他队伍交流，点击率的贝叶斯平滑在除了用户以外的特征上做就会有提升，可能这也是我们没有做出来的原因。

写在最后

历时好多个月的华为赛终于结束了，这次的时间线拖得真是长，当然最后得收获也是满满的。我觉得这次比赛能有这个成绩，主要是因为一个思维的转变。

刚开始遇到线上线下不一致的时候，我们先尝试的是不管怎么样把线下弄高再说，结果收效甚微；然后回过头来研究减少线上线下的Gap，发现这才是上分的重点。

当然也可能是我们参赛经验比较少，所以会有一些比较奇怪的想法哈哈哈。当然最重要的还是好队友，多进行思维的碰撞才会产生新的想法□。

数据竞赛C3 | Datawhale

该二维码7天内(12月4日前)有效，重新进入将更新

阅读原文

喜欢此内容的人还喜欢

Datawhale在浙大分享总结！

Datawhale

我们与Datawhale的故事！

Datawhale

二分类、多分类、回归任务，一个项目get竞赛必备模型

机器之心