

【关键模块】让数据驱动落地，你需要一个实验平台

2018-05-09 刑无刀

推荐系统三十六式

[进入课程 >](#)



讲述：黄洲君

时长 12:50 大小 5.88M



数据驱动这个口号喊了很多年了，这个口号也几乎成为了行业共识，但是数据驱动又像鬼一样，人人都在说，但几乎没人见过它长什么样子。

数据驱动和实验平台

要做到数据驱动，就要做到两点：第一点是数据，第二点是驱动。这听上去似乎像是废话，实际上不是。

这第一点的意思是，要采集数据，全方位，数据像是石油一样，没有它就谈不上驱动；第二点的意思是要让大家看数据，光采集了没有用，还需要让所有人盯着数据看。

而要做到驱动，需要一个 AB 实验平台。数据驱动的重点是做对比实验，通过对比，让模型、策略、设计等不同创意和智慧结晶新陈代谢，不断迭代更新。

对比实验也常常被大家叫做 ABTest，这个意思就是一个 A 实验，一个 B 实验，这样说可能有些模糊，所以我需要先和你说说什么叫做对比实验，然后再说说一个对比实验平台应该长什么样子。

你都可以把任何一家个性化推荐产品想象成一个函数，这个函数有很多参数影响它工作，函数的输出就是推荐物品列表。这些函数参数可以有各种组合，通过其中一种参数组合去面对一小股用户的考验，这就是一个实验。

要做实验，要做很多实验，要很快做很多实验，要很多人同时很快做很多实验，就需要实验平台。

要讨论实验平台，先要认识实验本身。互联网实验，需要三个要素。

1. 流量：流量就是用户的访问，也是实验的样本来源。
2. 参数：参数就是各种组合，也是用户访问后，从触发互联网产品这个大函数，到最后返回结果给用户，中间所走的路径。
3. 结果：实验的全过程都有日志记录，通过这些日志才能分析出实验结果，是否成功，是否显著。

把互联网产品想象一个有向无环图，每个节点是一个参数，不同的分支是参数的不同取值，直到走到终点，这一条路径上所有经过的参数取值，构成了服务的调用路径。

具体在推荐系统中，可能这些参数就是不同的模型与策略名称。每当一个用户经过这一系列的调用路径后，就为每一个分支产生了一条实验样本。

于是问题来了，每一个用户到来时，如何为他们决定要走哪条路径呢？这就要先经过实验对照来看。

实验要观察的结果就是一个随机变量，这个变量有一个期望值，要积累很多样本才能说观察到的实验结果比较接近期望值了，或者要观察一定时期才能说对照实验之间有区别或者没区别。

因为只有明显有区别并且区别项好，才能被进一步推上全线。

在设计一个实验之初，实验设计人员总是需要考虑下面这些问题。

1. 实验的起止时间。这涉及到样本的数量，关系到统计效果的显著性，也涉及能否取出时间因素的影响。
2. 实验的流量大小。这也涉及了样本的数量，关系到统计效果的显著性。
3. 流量的分配方式。每一个流量在为其选择参数分支时，希望是不带任何偏见的，也就是均匀采样，通常按照 UUID 或者 Cookie 随机取样。
4. 流量的分配条件。还有一些实验需要针对某个流量的子集，例如只对重庆地区的用户测试，推荐时要不要把火锅做额外的提升加权。
5. 流量如何无偏置。这是流量分配最大的问题，也是最难的问题。同时只做一个实验时，这个问题不明显，但是要同时做多个实验，那么如何避免前面的实验给后面的实验带来影响，这个影响就是流量偏置，意思是在前面实验的流量分配中，有一种潜在的因素在影响流量分配，这个潜在的因素不易被人察觉，潜在的因素如果会影响实验结果，那么处在这个实验后面获得流量的实验，就很难得到客观的结论。这个无偏置要求，也叫“正交”。

这些问题也是实验平台在设计之初要考虑的。试想一下，推荐系统中，算法工程师总是在尝试很多模型，或者在线下给出很多的模型调参，线下评测时，各种指标都是一片锣鼓喧天、红旗招展，恨不得立即上线去检验真实效果。

每一个算法工程师都这么想，但是线上流量有限，因此需要重叠实验，废水循环，最好能够做到洗脸的水冲马桶，这样灵活的实验平台长什么样子。

Google 公司的实验平台已经成为行业争相学习的对象，所以今天我会以 Google 的实验平台为主要对象，深入浅出地介绍一个重叠实验平台的方方面面。

重叠实验架构

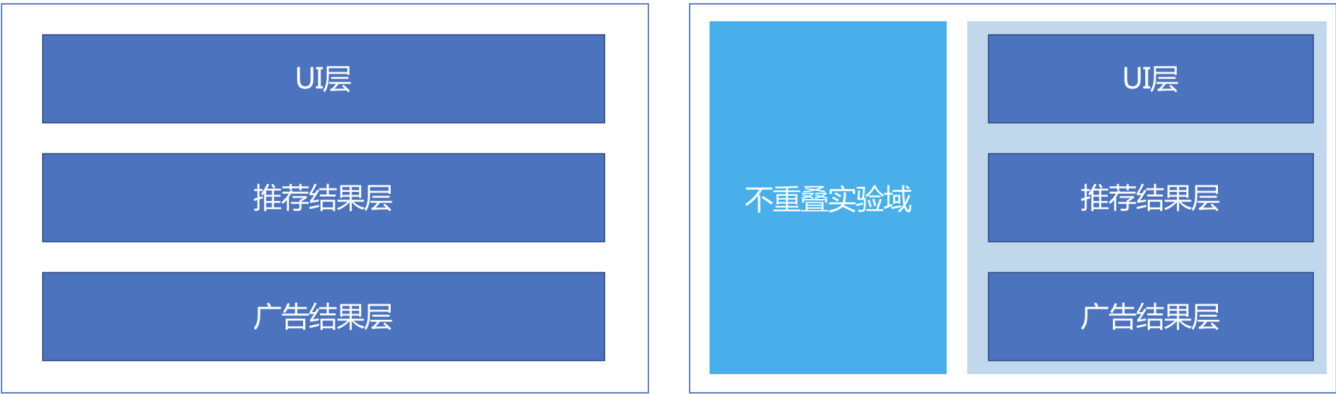
所谓重叠实验，就是一个流量从进入产品服务，到最后返回结果呈现给用户，中间设置了好几个检查站，每个检查站都在测试某些东西，这样同时做多组实验就是重叠实验。

前面说了，重叠实验最大的问题是怎么避免流量偏置。为此，需要引入三个概念。

1. 域：是流量的一个大的划分，最上层的流量进来时首先是划分域。

- 2. 层：是系统参数的一个子集，一层实验是对一个参数子集的测试。
- 3. 桶：实验组和对照组就在这些桶中。

层和域可以互相嵌套。意思是对流量划分，例如划分出 50%，这 50% 的流量是一个域，这个域里面有多个实验层，每一个实验层里面还可以继续嵌套域，也就是可以进步划分这 50% 的流量。下面这两个图示意了有域划分和没有域划分的两种情况。



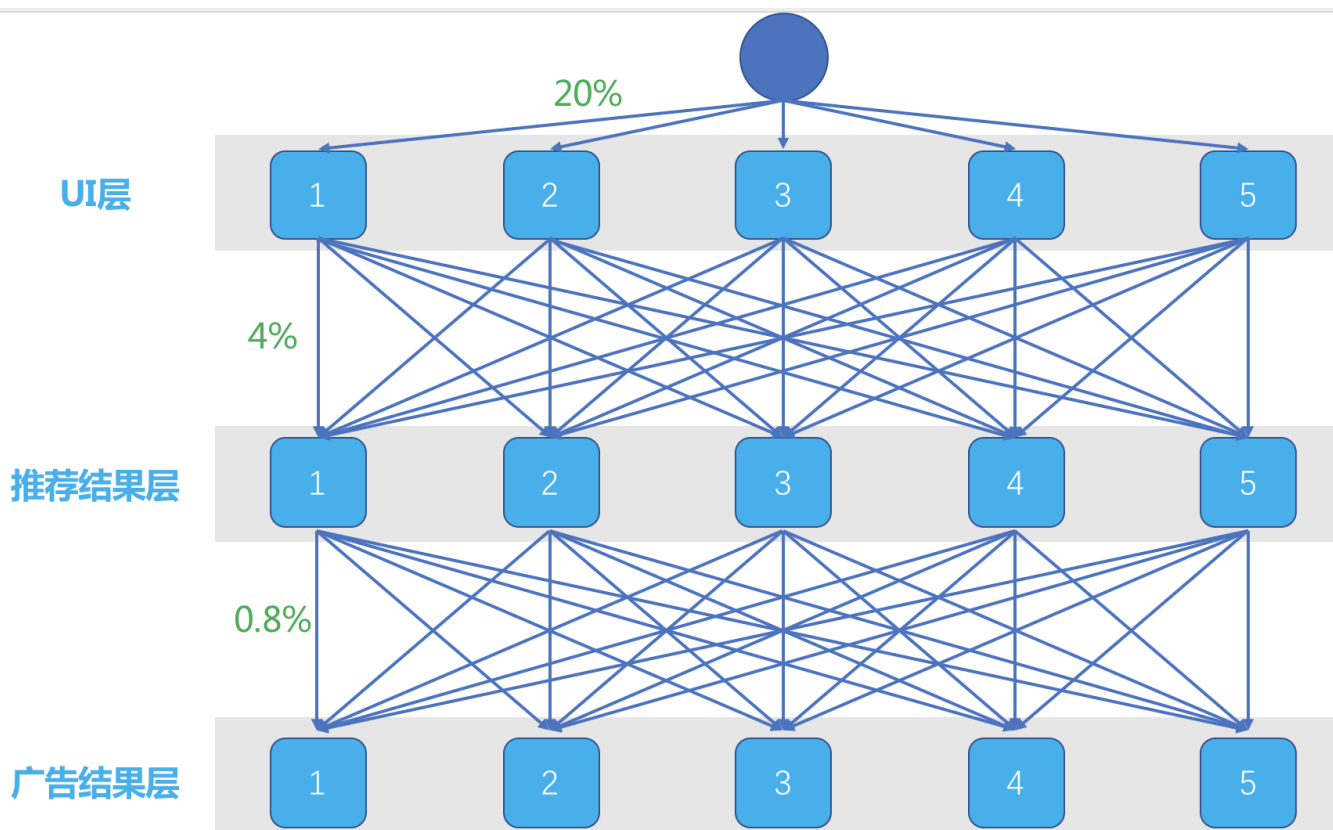
图中左边是一个三层实验，但是并没有没有划分域。第一层实验要测试 UI 相关，第二层要测试推荐结果，第三层要测试在推荐结果插入广告的结果。

三层互不影响。图中的右边则添加了域划分，也就是不再是全部流量都参与实验中，而是被分走了一部分到左边域中。剩下的流量和左边的实验一样。

这里要理解一点，为什么多层实验能做到重叠而不带来流量偏置呢？

这就需要说桶的概念。还是上面示意图中的左图，假如这个实验平台每一层都是均匀随机分成 5 个桶，在实际的实验平台上，可能是上千个桶，这里只是为了举例。

示意图如下：



这是一个划分域的三层实验。每一层分成 5 个桶，一个流量来了，在第一层，有统一的随机分流算法，将 Cookie 或者 UUID 加上第一层 ID，均匀散列成一个整数，再把这个整数对 5 取模，于是一个流量就随机地进入了 5 个桶之一。

每一个桶均匀得到 20% 的流量。每一个桶里面已经决定好了为你展示什么样的 UI，流量继续往下走。每一个桶的流量接着依然面对随机进入下一层实验的 5 个桶之一，原来每个桶的 20% 流量都被均分成 5 份，每个桶都有 4% 的流量进入到第二层的每个桶。

这样一来，第二层每个桶实际上得到的依然是总流量的 20%，而且上一层实验带来的影响被均匀地分散在了这一层的每一个桶中，也就是可以认为上一层实验对这一层没有影响。同样的，第三层实验也是这样。

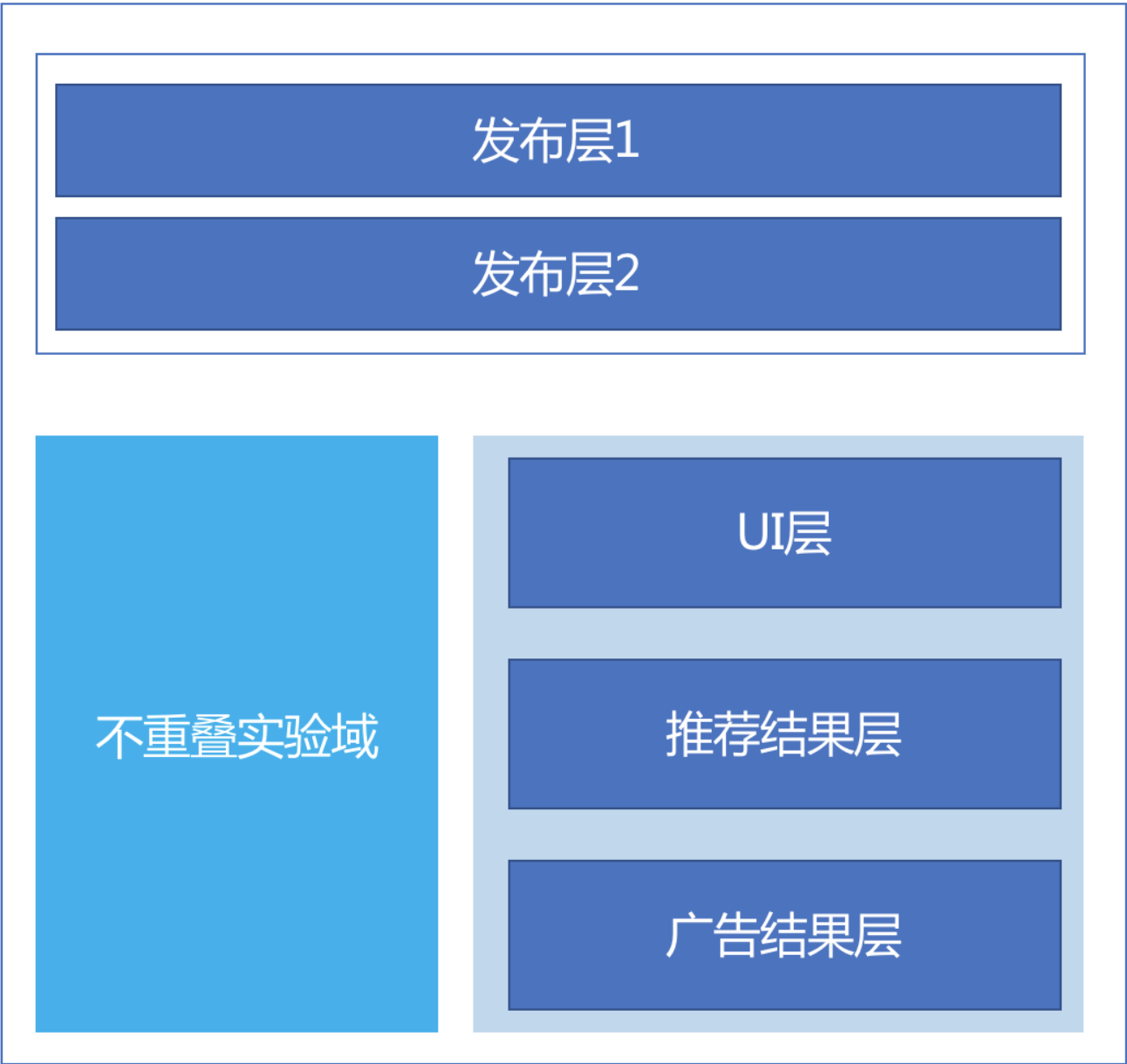
这就是分层实验最最基本的原理。在这个基础上，增加了域的概念，只是为了更加灵活地配置更多实验。

关于分层实验，有几点需要注意：

1. 每一层分桶时，不是只对 Cookie 或者 UUID 散列取模，而是加上了层 ID，是为了让层和层之间分桶相互独立；
2. Cookie 或者 UUID 散列成整数时，考虑用均匀的散列算法，如 MD5。

3. 取模要一致，为了用户体验，虽然是分桶实验，但是同一个用户在同一个位置每次感受不一致，会有损用户体验。

Google 的重叠实验架构还有一个特殊的实验层，叫做发布层，优先于所有其他的实验层，它拥有全部流量。这个层中的实验，通常是已经通过了 ABtest 准备全量发布了。示意图如下：



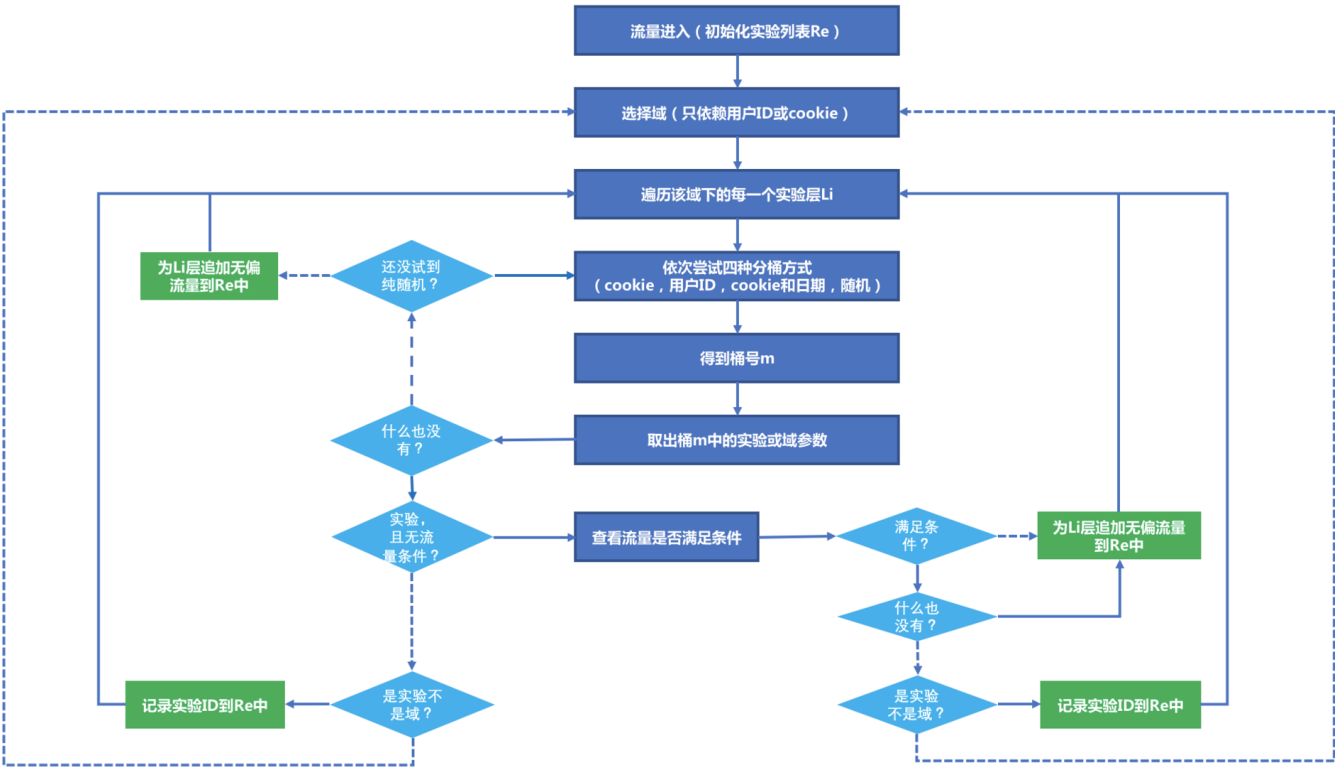
前面举例所说的对用户身份 ID 做散列的流量分配方式，只是其中一种，还有三种流量分配方式，一共四种：

- 1. Cookie+ 层 ID 取模；
- 2. 完全随机；

- 3. 用户 ID+ 层 ID 取模;
- 4. Cookie+ 日期取模。

在实验中，得到流量后还可以增加流量条件，比如按照流量地域，决定要不要对其实验，如果不符合条件，则这个流量不会再参与后面的实验，这样避免引入偏置，那么这个流量会被回收，也就是使用默认参数返回结果。

在 Google 的架构中，由于层和域还可以嵌套，所以在进入某个层时，可能会遇到一个嵌套域，这时候需要按照域划分方式继续下沉，直到遇到实验或者被作为回收流量返回。整个实验平台，工作的示意图如下所示：



说明如下：

- 1. 图中涉及了判断的地方，虚线表示判断为假，实线表示判断为真。
- 2. 从最顶端开始，不断遍历域、层、桶，最终输出一个队列 Re，其中记录对每一个系统参数子集如何处理，取实验配置的参数还是使用默认参数，其中无偏流量表示使用默认参数，也就是在那一层不参与实验，流量被回收。
- 3. 拿到 Re 就得到了全部的实验，在去调用对应的服务。

统计效果

除了分层实验平台之外，还存在另一个问题，每一个实验需要累计获得多少流量才能得到实验结论呢？

这涉及了一点统计学知识。实验得到的流量不够，可以说实验的结论没有统计意义，也就浪费了些流量，而实验在已经具有统计意义之后，如果还占用流量做测试，则也是在浪费流量。

如何确定实验规模呢？Google 给出了如下公式：

$$N \geq 10.5 \left(\frac{s}{\theta} \right)^2$$

公式中：

1. s 是实验指标的标准差。
2. θ 是希望检测的敏感度，比如想检测到 2% 的 CTR 变化。

上面这个公式计算出来的实验规模，表示以 90% 的概率相信结果的显著性，也就是有 90% 的统计功效。

对比实验的弊端

AB 测试实验平台，是产品要做到数据驱动必不可少的东西，但是这种流量划分的实验方式也有自己的弊端，列举如下：

1. 落入实验组的流量，在实验期间，可能要冒着一定的风险得到不好的用户体验，在实验结束之前，这部分流量以 100% 的概率面对这不确定性；
2. 要得到较高统计功效的话，就需要较长时间的测试，如果急于看到结果全面上线来说有点不能接收；
3. 下线的实验组如果不被人想起，就不再有机会得到测试。

诸如此类弊端，也可以考虑在实验平台中用 Bandit 算法替代流量划分的方式，通过 Bandit 算法选择不同的参数组合、策略，动态实时地根据用户表现给出选择策略，一定程度上可以避免上面列举的弊端。

总结

实验平台是推荐系统要做到数据驱动必不可少的东西，但是如何做到科学高效快速地做实验呢？

常见的做实验，只是简单地选择一个尾号的用户 ID 作为实验组，再选择另一个尾号作为对照组，甚至选择剩下所有的用户 ID 作为对照组。

这样做出来的实验，显然是有问题，因为并不知道通过用户尾号来分组是不是能做到无偏？另一个问题是，这样就只能在一个时期只能做一个实验，非常低效。

本文以 Google 开放的实验平台架构作为原型，对其核心技术做了详细介绍。这个实验平台做到了同时无偏地做多组对照实验。因为它巧妙地引入了三个概念的嵌套结合：

1. 域；
2. 层；
3. 桶。

三个概念层层相扣，流量划分得到了一个可行的方案。这个实验平台方案已经应用在很多公司中，你不妨在自己的公司尝试做一下。

最后留给你一个问题，关于分层实验的原理，你是否已经理解了为什么层和层之间可以做到毫不影响，欢迎给我留言讨论。

推荐系统 36 式

解决你推荐系统 起步阶段 80% 的问题

刑无刀

资深算法专家



新版升级：点击「 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 **【关键模块】** 让你的推荐系统反应更快：实时推荐

下一篇 **【关键模块】** 推荐系统服务化、存储选型及API设计

精选留言 (10)

 写留言



行行行

2018-05-10

 8

<https://storage.googleapis.com/pub-tools-public-publication-data/pdf/36500.pdf>
原文链接



聂文峰-峰...

2018-11-22

 2

因为每一层的没个桶的数据都均匀随机的受到上层实验的影响，所以在该层实验之前有差异就是同一层各桶实验引起的



shangqiu86

2019-05-08



目前我接触的我们的平台，并没有域层桶的概念，就是以pin划分，随机划分10%，或者5%的流量（pin）走入需要对比效果的算法实验里，一旦划分这个pin的都是走这个算法实验

展开 ▾



shangqiu86

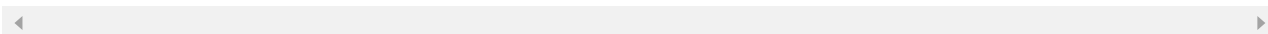
2019-05-08



老师，这里有多少个桶是不是我们就可以上多少个实验同时进行效果观察

展开 ▾

作者回复: 对。



shangqiu86

2019-05-08



下一层得到了上一层每个桶内的流量，这样若上层有个桶内有偏置，下层的每个桶内均匀分得了这部分偏置数据，也就不偏置了



帅帅

2018-09-25



小团队的推荐系统，玩这个太大了；
初期我还是用userId%2来区分好了



林彦

2018-05-13



每一个流量标识只会属于一个桶，每一个桶只会属于一个层，它只会属于一个特定层的实验，直到这个实验结束。



曹林华

2018-05-10



层与层之间有 id 区分
hash 取模的时候加入层 id 来避免不同层层影响

我们做实验采取的是每层通过加salt 的方式，效果其实和层 id 类似
展开 ▾



仲杰

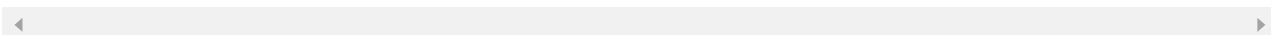
2018-05-09



请问老师，资料的原文链接有吗

展开 ▾

作者回复: 有，我最后会单独整理一篇整个专栏的参考文献列表，请注意查收。



极客星星

2018-05-09



老师你好 请问google的平台有没有对应的论文介绍 如果有的话 能否提供下链接 谢谢

作者回复: 有，最后会整理一篇我所有的参考文献列表，请注意按需阅读。

