
Facebook 面试全攻略 2019



本资料由九章算法独家倾情整理

www.jiuzhang.com

目录

一、电面(Phone Screen).....	2
二、现场面试(Onsite).....	4
三、行为面试 Behavioral interview.....	8
四、 Facebook 面经汇总.....	10
电面面经.....	10
Onsite 面经.....	20

拿脸书 offer 必备课程

《[简历修改与面试技巧点拨](#)》《[九章算法班](#)》《[OOD面向对象专题](#)》

《[系统设计班](#)》《[人工智能集训营](#)》《[硅谷求职算法集训营](#)》

Facebook 的面试过程分为电面(Phone Screen)和现场面试(Onsite)两个阶段，下面将从面试流程、面试内容、面试技巧等方面来教你如何准备 Facebook 软件工程师面试。

一、电面(Phone Screen)

电面(Phone Screen)将会有一位 Facebook 工程师参与，主要面试内容为 coding 面试。

面试流程:

- **自我介绍:** 首先面试官会要求你进行自我介绍，并说明想应聘的职位。
- **职业期望:** 下面的 5-10 分钟，面试官会对你的经历和职业期望进行提问。
- **coding:** 接下来的 30-35 分钟就要 coding 了。
 1. coding 面试会在一个你和面试官共享的在线编辑器上完成(如果是现场面试就写在白板上)。
 2. 你会得到一个或几个 coding question 问题都是长短合适的，可以在几分钟内解释好，10-30 分钟解答完。
 3. 面试官会在这一环节了解你解决问题的方式。
 4. 面试官一般不会问技巧或估算问题，他们并不关心海里能塞下多少个乒乓球。
 5. 你可以自己选择解题方式，但面试官可能会添加进一步的限制或要求。
- **向面试官提问:** 最后 5 分钟是提问时间。这是一个从内部角度了解 Facebook 工程师的好机会。

如何准备:

- **投入时间:** 温习面试技巧、编码技术和算法，对任何工程师来说都是很重要

的，即使资深工程师也是如此。面试和你的日常工作是完全不同的。尤其是第一次技术面试，一定要多做准备，对接下来的面试也是有益的。

- **刷题**: 不使用编译器，用一种最有效率 bug free 的方式练习解答算法问题，你可以在 Lintcode 上刷题，同时你可以在 uincode 标准答案查询上，到对应的最简洁、优质的答案。

- **用简单的编辑器写代码**: 面试中你会用没有语法加亮和自动填充的工具写代码 (比如 CoderPad) 。

- **练习手写编码**: coding 面试将在白板 上完成，准备时可以用白板或纸和笔练习解题。

- **计时练习**: coding 面试的解题时间有限，按时完成至关重要。可以和朋友一起做一次模拟面试。

- **复习数据结构、算法和复杂度**: 要对时间和空间的复杂度进行分析。别忘了复习数据结构，比如列表、阵列、哈希表(lists, arrays, hashtables, hash maps, stacks, queues, graphs, trees, heaps)等等，还有排序、宽度优先搜索、深度优先搜索、和遍历(sorts, BFS, DFS, and traversals)。还要记得复习递归和迭代方法(recursion and iterative)。

- **想想你 2-5 年内的职业期望**: 面试官会问及你的兴趣和作为工程师的优势，准备 1-2 个问面试官的问题: 面试最后五分钟是向面试官提问, 所以你要准备好 1-2 个能帮你增加面试官好感度的问题。

coding 面试技巧:

- **说出想法**: 面试官更多关注的是你解题的方法，这和答案的正确一样重要。把

想法说出来可以让面试官了解你的思考过程,跟随你的解题思路。另外还可以让他们在必要的时候指点一二。

- **找一个好的面试地点:** 选择一个安静的地方,确保你的网络连接和手机信号良好。使用耳机,确保可以双手写代码。
- **说话清晰:** 确保你说话清晰,如果你听不清面试官说话,立刻告诉对方,这样他们可以调整。谁也不想把时间浪费在努力听清对话上。
- **使用最擅长的编程语言:** 按时、准确地答题是很重要的,请使用你最熟悉的语言解题。
- **高效管理时间:** 花时间找出问题最理想的解答方案,不要脑海里第一个蹦出什么想法就急着写什么。如果在一定时间内你还没能找到更好的方案,先动笔写一个可行方案,在解题过程中再迭代改进。有些面试者最终没有给出答案就结束面试了,就因为他们找不到理想方案。有一个可行但不是最佳的方案比一个随意而不加思考的方案强很多。有可行方案之后,你就可以在此基础上再进行优化。
- **分享你的推理:** 确保被问到时你能解释你的方案。软件工程就是权衡的过程,你要时刻做好讨论的准备。
- **自己找到并修复 bug:** 不要等着面试官帮你找 bug。
- **留意面试官给出的提示:** 面试官通常都对问题有充分的了解,如果你卡住了,他们给出的提示可以帮助你。

二、现场面试(Onsite)

作为 Facebook 软件工程师职位的应聘者,你将会一天内经历 4 到 5 轮的面试,

这些面试会有 3 种类型:

1.代码面试 Coding interview -你需要解决一些基本的代码问题。

2.设计面试 Design interview -你会被要求展示设计能力。设计问题将根据你的背景决定集中在系统或产品上。

3.行为面试 Behavioral interview -你会被问到你的工作经验和求职动机, 还有其他一些行为问题。除了很早或很晚去参加面试的人, 其他人都会由招聘或工程部的人带着吃午饭。这是向面试官以外的职员提问的好机会。

1. 编程面试 Coding interview

面试流程:这次 coding 面试会比电面要难:面试官的提问会更难, 评估标准也会更严格。面试时间约为 45 分钟, 以下是比较经典的流程: .

- **自我介绍:** 开始的 5 分钟是介绍时间, 可能还会简单问一下你的背景。
- **coding:** 接下来的 30 分钟会有一个或多个编程问题(coding question) ,大部分是算法面试题, 可以参考 Lintcode 上的算法面试题。
- **向面试官提问:** 面试官会留最后 5 分钟作为你对面试官提问的时间。这是从内部工程师身上了解更多 Facebook 和你感兴趣的事的好机会。

如何准备: 同电面(phone screen)

2.设计面试 Design interview

面试流程:设计面试有 45 分钟。整个面试你只用说与写。正如所有面试一样,面试官会留最后五分钟让你提问。面试的目的是评估应聘者解决重要工程设计问题的能力,最后面试官会问你一个非常广泛的设计问题并评价你的方案。设计面试有两种类型:系统设计和产品设计。

如何准备:

(1) 系统设计

- 模拟系统设计面试。邀请一个工程师帮你模拟面试。让他提出一个系统设计问题,如果正好是他正在做的项目那就再好不过了。不要把它当成是一个面试,而是放轻松地去思考问题,并提出你能想到的最佳解决方案。
- 在实际的系统中去实践。你可以在既有的 OSS 中去练习,也可以与朋友合作搭建一个系统。对于课堂中的系统设计作业,不再把它仅仅当成一个学术训练,而是把它当成实际问题,思考系统设计过程中的架构和博弈。正如我们生活中遇到的大多事情一样,只有做了才知道其中会遇到什么问题,从而真正学到东西。深挖开源系统的运行特点。例如,你可以看看 levelDB。这是一个干净、小、且编写良好的系统。你可以读读执行命令,了解它是如何在硬盘中存储数据的,如何将数据压缩成不同的层?你也可以多多反思一下的博弈问题:哪种数据和大小是最优的?什么情况下会降低读写速度?(提示:比较一下随机写和顺序写)
- 多了解一下系统中数据库和操作系统是如何运行的。这些技术并不只是你口袋中的工具,它们往往会在你设计系统的时候给你带来启发。如果经常像 DB 或 OS 一样思考它们如何处理各自的问题,你也会把这些思考方式应用到其它的系统设计

计中去。

(2)产品设计

· 反思自己的项目:思考已完成的项目,哪些是容易的部分,哪些是困难的部分?

面试官可能会问:“你会如何设计一个电子邮件服务器?”你可以先考虑这些问题:

- 如何储存邮件?尤其是当系统扩大一台机器装不下时?
- 如何处理有大量收件人的邮件列表?
- 如何处理人们滥用垃圾邮件系统?
- 面对潜在的系统故障,如何保证系统是可靠的?

另一个面试官可能会问:“你会如何设计用来构建丰富文档编辑器的客户端服务器API?”你可以先考虑这些问题:

- 客户端如何从服务器请求文档数据?尤其是文档大到我们不想在单一请求中下载它的时候?
- 如何在API响应中显示文档中粗体和斜体这样的元素?
- 在不破坏原有客户端的情况下,想要给服务器添加新特性,应该如何设计系统?

设计面试技巧:

- **提炼你的方法:** 当你采取你的方法时,要先概述它,然后再考虑如何把它进行拆分。
- **找到关注点:** 你没有时间去讨论每个设计的细节,所以你需要找出有趣的关注点和难以解决的问题。
- **注重整体和细节:** 你既要重视整体的解决方案,也要注意其中的一些细节。一

一个好的解决方案需要在方方面面都能做到完美的

- **探索内在的权衡:** 在工程问题中,你需要对权衡做出明智的决定。你需要把你的方案与其他方案进行比较,探讨其中存在的问题,并且选择出最佳的解决方案
- **主导讨论:** 面试官希望看到的是你通过艰难的经历学会了构建大型系统。你的预测能力和解决典型问题能力都能让面试官为你亮绿灯。
- **积极对话:** 问清晰明确的问题,但确保话题要往好方案方向发展。

三、行为面试 Behavioral interview

面试流程:

行为面试实际上由一部分行为面试和一部分编码面试组成。行为部分考察的是你本人、你的经历、你的简历和你的动机。行为面试的目的是评估求职者是否能适应 Facebook 的工程师文化。面试中有时也会包括一道题作为前两个面试的补充。

如何准备:

- **了解自己:** 花时间检查回顾你的简历,面试官很可能会问你工作经历中的重要事件。
- **了解求职动机:** 你要清楚你的求职动机是什么?为什么想在 Facebook 工作?为什么要做软件工程师?
- **了解如何与同事相处:** 你会如何与同事合作?会如何化解矛盾?

- **有具体的例子或轶事:**

下面是一些常问的行为面试问题，每个问题的回答都要有具体的事例作支持:

- 你创建过的最好的项目有哪些?
- 什么让你引以为豪?
- 如何做到更好?
- 你与他人一同合作过的成功经历有哪些?
- 讲述一次你主张自己的想法，尽管有人反对仍坚持推进的经历?
- 你会怎样处理矛盾?
- 你喜欢如何给予和接收反馈?
- 什么样的技术使你最为兴奋?
- 为什么选择 Facebook?

面试技巧:

- 熟悉 Facebook 的 5 个核心价值观 (快速行动，勇往直前，专注于影响力，保持开放，创造社会价值)。Facebook 在核心价值观的影响下才能共同努力让世界更开放更一体化。面试官寻找的是把它们铭记在心并在实际中运用的人才。
- 做自己!保持开放并诚实面对自己的成功与失败。
- 保持谦逊，注重团队合作与领导能力。

四、 Facebook 面经汇总

电面面经

1.[面试经验] 脸家刚出炉的店面

两轮前后一起来

一面:

LeetCode 680:讨论各种各样的 corner case, 我大概给了五六个吧

LeetCode 938: 时间优化

二面:

LeetCode67: 时间复杂度

LeetCode139: 变种, 要返回列表。follow up:(1)时间复杂度

2.[面试经验] 脸书新鲜电面

刚面完, 简单说一下时间线: 九月中旬内推, 十月中旬来消息说给一次电面

一个星期之内就安排上了 45 分钟电面

面我的是一个印度小哥, 非常直接, 连自我介绍都省了, 直接做题

给两个 interval arrays, 找出所有 overlapping intervals。

解题思路: 两个 pointers, 分别指向两个 array。先找 overlapping interval, 找到了, 推入 return vector. 根据两个 intervals 的 end 来判断更新哪个 pointer.

之后讨论了 testcases, runtime, 和优化

3.[面试经验] FB 技术电面 面筋

其实是上上周五面试的了, 上周一收到 onsite invite SQL 部分是 spam 原题.

Table user_actions .

Date | user_id | post_id | action {view, like, reaction, comment, report, share} | extra {extra reasons for the action ex. Love, spam, nudity}

1. How many posts were reported yesterday for each report reason

2. What percent of daily content that users view on FB is spam?

Product 也是关于 Spam Filter 的题

问如何 test effectiveness of spam filter, 我说 AB testing

然后追问做了 AB testing 但是 user time spent 下降了为什么, 给出你的 hypothesis, 然后如何 validate

注意不要盲目套模板, 我一听问题说会不会是 seasonality 的问题, 面试官直接问我 AB testing 为什么要 worry about seasonality, 我一想对哦, 立马改变方向说我的 hypothesis, 跳过那些 context checking

4.[面试经验] fb 2020 实习电面

加州 office 的印度小哥哥，随便介绍了自己之后开始做题。

1. 确认一个 array 是不是 monotonic, 两个两个比就行。我真的太紧张了...这题还写出俩 bug。自己发现时候都乐了...很快改好 move to the next question

2. LeetCode 76 我花式卡住，那位哥在 comment 里面都给我讲了一遍了快..很希望我能 get 的亚子。无奈我太菜，最后他都不说话了，由着我瞎写，到了也没写完...

唉都是缘分 重在参与吧..

5.[面试经验] 肥死不可第一轮电面

第一题: $m \times n$ 的矩阵，每一行由 0000001111111111 组成，求含 1 的最小 column 值。先给了二分法的思路，面试官直问我能不能再想个更好点的，然后急中生智强行现场想出了 $O(m+n)$ 的解。

第二题:

给往返机票依据天数的价格:

去程:

[10 6 3 7]

回程

[3 8 9 10]

让你求最小的往返票组合，回程天数得大于或等于去程。

6.[面试经验] fb data scientist 电面面经

面试当天早上临时告诉我面试官换了，我反正也觉得无所谓。然后面试时候居然发现有两个面试官，原来的那个其实等于是监面，新的那个应该是实习面试官...不过也不重要。只是全程我觉得我和小哥没搭上线，他问我的问题我都答偏，或者是我没理解到他的意思之类的...我觉得看寂静的时候你很清楚问题是什么，因为写寂静的人已经把面试官的问题用他自己的语言叙述出来了，一下子就理解了，但是真正面试的时候面试官发问的方法可能不那么好懂。而且我这位也不会说 "make sense" 什么之类的...不会给 feedback，所以我都不知道我有没有答到他的点子上...比如他问我导致某某问题的原因可能有什么，我说 A,B,原因是什么什么，他就会面无表情说，“还有呢？”

先是 product, 是 unfriend 那道题，但是我感觉问的方向有点不同。先问我有什么 pros & cons, 然后我说 pros 是 friends 多的话 user 就会更愿意呆在 FB, 小哥问我怎么 measure engagement, 然后我丢了一些 metrics 过去; 然后问我 cons 是什么，我说有一些我不想看到的 posts 会出来，从而让我不想用 FB, 因为很多信息都没用

小哥: 那你说的 pros 是增加 engagement, cons 又是减少 engagement, 那怎么平衡?

我: 这...平衡? 那就 deranking 不想看的 posts 吧, 这样既不用减少 quantity, 又能提高 quality

小哥: (反正说了一堆 deranking 也怎么怎么不好以后) 那你怎么知道别人不想看一个 post?

我: (走偏了) average...time spent...on...the...person's...post? 比如这个数只有 0.5 秒, 那可能就真的对这个人发的 post 不感兴趣

小哥: 那对于这个 metric 你怎么确认一个 threshold?

我：哦…(又掉沟里了)不同的内容应该有不同的 threshold, video, photo, text, 把内容分类, 每个弄一个 distribution, 找比如 5th percentile 当作 threshold (我在说什么…)

小哥：(我放弃了…) ok. 现在我们来说就是有一些人你想 unfriend, 怎么判断?

我：根据他们以往的 interactions, 比如 comments, likes, gifts exchanged, messages, (又多嘴了一句) 我觉得我会 weigh more on my comments on his posts than his comments on my posts 因为我的账号我做主!

小哥：那你怎么判断这个 weight?

我：(崩坏) 6, 4 开吧也许?

小哥：为啥呢?

我：我随便说的

小哥：好吧。但有些人本身就和朋友的 interactions 很少, 很少留 comments, 那这样的话我们就会要让他 unfriend 90% of his friends?.

我：这样是不好的…(嗯, 除了说这个我还能说什么)

小哥：那你还能想出什么 signal 能帮助我们判断是否 unfriend?

我：(真的想不出来了)…毕业学校?

小哥：毕业学校为什么是一个好的 signal?

我：我…我也不觉得他是个好的 signal…只是一个 potential signal。同一个学校毕业可能相熟的几率更大吧, 比如很有可能有共同朋友, 参加共同活动之类的。

小哥：(放弃了) ok, 那你要有 what statistical method 来判断需不需要 unfriend?

我：statistical method? (我真的不知道在问什么) 没有 y 那就 clustering 吧

小哥：什么是 clustering?

我：(解释了一遍 clustering)

小哥：OK。那我们现在推出了这个 feature, 怎么确认 work 与否?

我：metric 确认了, a/b testing…

小哥：怎么判断结果到底有没有 difference?

我：确认那个 difference 是 statistically significant

小哥：怎么知道是 statistically different?

我：用 t-test

小哥：怎么用 t-test?

我：(我怎么说…) 用 t-test 你会算出一个 t-statistic/p-value 告诉你。

小哥：什么是 p-value?

我：(解释了 p-value)

小哥：如果 p-value 大了怎么办?

我：(我不是统计出身我什么都不会) 那我重新 design experiment, 也许 increase statistical power? (完全在乱说)

小哥：什么是 statistical power?

我：(崩溃地乱解释了 statistical power)

小哥：OK cool…

(中途其实还有很多被问到的细枝末节的问题我已经想不起来了, 我觉得他们的面试手册肯定有一条就是, dig deeper! Dive in!)

SQL 就是配套的那个题, 两个问题, 1. 前十多朋友的人, 2. 和每队朋友的 comments 数量。小哥跟我说 output table 一共两列, 一列是 friend_pair_id, 一列是 num_comments, 我想: what? 什么 friend_pair_id? 这什么东西? 他说你要 compute

这个 column 出来…我到现在都没想通他要我干什么，求指教？于是我确认了他只是想每一对朋友只有一条 entry，就还是 output 了三个 column， user1, user2, num_comments.

可能我从没面过大厂的原因，所以我觉得他们好像并不 care 你这个人…和我面一些 medium/small size 的公司感觉不一样。后者让你感觉蛮有人情味（不管是真的还是装的），所以初次和 fb 电面让我感觉不是很好。所以我觉得真正值得 fb 的 offer 的人一定是第一对这个公司的产品有热情，喜欢这个公司的产品，公司文化，并且很坚定的要去拿到这个 offer 的。所以找工作就像相亲，一定是要 match 才行的。

7.[面试经验] FB phone interview

1. find intersection of two sorted arrays

A = {1, 2, 3, 3, 4}

B = {2, 4, 5, 6, 7}

Two pointers and write the code

Follow up: A is huge and B is small, binary search and do not need to write the code

2. Remove invalid parenthesis

删除字符串里所有不合法的括号

"(ab)()(d)aa" => (ab)()(d)aa

使用栈来解决，(push, letter or digit push,) pop until (or stack is empty

第二题写的有 bug

面试的是一个印度小姐姐，真心把我说懵逼了，全程几乎听不懂她在说什么，好像说第二题有更好的解决方案，并且给了 hint，但是完全听不懂。

8.[面试经验] 脸熟 系统垫眠

PE 刚问，上周一 面试，系统

经典问题"what if query for database become extreme slow? "

45 分钟全程快问快答，基本碰到一个点他想问，他就会脱离原来轨道来问你那些内容...

面试前跟自己说语速千万别太快，就慢慢扯。结果回答的时候还是想要快速的把问题回答完，感觉拉扯了好几十个问题。个人感觉还是说慢一点好，能够增加思考时间，减少问题数量。最后结束的时候，小哥感叹已经问了那么多问题:)

可能问的问题例子：

提到可能是分布式数据库，所以系统挂掉可能在分布式系统内部 如 query node - data node，他就问网络问题，如何确认分布式机器间网络没问题

提到资源，他就问 VM 为什么存在，PAGE FAULT 为什么要处理，INTERRUPT 机制

提到资源耗尽，就问进程 OOM，OS 会如何处理

提到 Soft irq(interrupt request), 他就问举例 syscall vs softirq 的区别

lz 这道题其实没仔细想过, 最后小哥其实想问单机 db 如何加速的问题, 我回答的是充分利用总线, RAID 并行载入, 小哥说有点接近, 后来反应出来是想把 db 全载入内存在硬件物理访问速度上做优化

如果能一开始就答到痛点估计会很快乐

9.[面试经验] FB 电面

FB 电面一面

一个国人大叔, 第一道题 validate BST, 第二道题 kth largest element in array。

题都没啥难的, 不过会非常仔细的过 test case。

第二道题写了 heap 然后口头过了 quick select 实现了 partition。

10.fb 实习 电面

昨天中午电面了下, 今天 hr 发邮件说开始 teammatch。

直接上:

印度哥:

1. LeetCode 304
2. LeetCode 215 以及一些堆的用法的考察

大英帝国哥:

1. 给一串字符串, 找到最左边重复的字符, 比如 reuuioie, 输出 e
2. LeetCode 560

11.fb 热气腾腾 (非典型) 电面

第一题: 判断 valid binary search tree, lz 给了一个时间空间都是 n 的, 问一下优化把空间优化成 1 (我简直不敢相信自己会遇到这个题); 第二题: LeetCode 523, 但不是求 target 的 n 倍数而是直接求 target, 虽然这个题 lz 之前做过, 但只看了 n 平方的解法, 小哥想要线性的, 最后还是没给出来, 解在 lc 上都有, 这是 lz 觉得最凉的地方...

面试官应该是白人小哥, 人还挺好的, 会在我过于紧张大脑一片空白的时候安慰我说 it's okay, 提出一些问题的时候也会给比如 awesome 或者别的一些反馈, 体验还不错

12.[面试经验] FB 新鲜电面面经

上来双方简单自我介绍, 她目前是 ins 广告组的, 之前是 ins 开拓 marketing 的部门。

没问我的工作经验, 可能是因为我自我介绍了稍微久了点。

先是 product 题

1. About issues with too many friends. what are the pros/cons. 只要我回答 cons, 根据我的回答 follow up 继续深挖。

朋友越多导致看到的内容越多，但你真正 care 或者 interested 的内容（如 close friend, family 的 post）会减少被看到的机会）/ 朋友过多之后，根据的 connection 许多本来 irrelevant 的 advertisement 会成为你的 spam 出现在你面前。

2. 根据第一个，how to evaluate the effect? 给出 metric

我大概下意识给了个 # of interaction (e.g. Likes, Comments, Repost) / # of DAU，和她大概分析了这个 ratio 会因为朋友增多而怎么变化，有点磕磕绊绊

继续问，how to compare this ratio?

我在这里和她纠结了许久，我听到 compare 给出了一个 control group (users with less friends/connection size)，但被她揪着问你的 test 如何设置，既然有 control，那 experiment 给了什么 treatment，

我实在没法绕出来，只能说这不是一个 experiment，没有 hypothesis，只有两个不同的组而已。根据这两个组的不同来看我的 ratio 如何变化。感觉她是想揪出你的定义上的毛病，所以还是不懂就容易被抓着问。

最后，我提出改善，说 numerator 上其实可以更 reasonable 的改成 (# of interaction / # of post viewed)，稍微解释比如 20 条 post 中，点赞了 2 条，取 2/20 这个 ratio 作为 effect 的评判。感觉也就稍微好一点点，面试官说 make sense (恶魔的低语 lol)

3. 第二个话题，关于 defriend 功能，你如何判断一个 friend 需要被 recommend 为 defriend 的对象

我给了三条 metrics，临时想的也没多思考，只能赶紧解释一通。

of interactive between user and this friend / # of private message (like Messenger chatting) between user and this friend / activity of this friend

第三个 metrics 实在是瞎说了，大致按照顺序解释一通，首先两者之间不常互动的先选出来，其次不互动也不互发私信的，最后再根据 activity，我的本意是判断一下是这个 friend 到底是不用这个 product 了还是经常潜水。

但我可能也没解释清楚，总是我说的是 depend on different situations, whether users want to keep inactive friends or only keep active friends. 面试官没啥表示，make sense 带过。

然后是 SQL 题

Table1 User1| user2
 123 456
 456 123

1. count top 10 users with most friends

直接用 count() group by user1 就好

2. 加了个表

Table2 sender_id|recipient_id|action|content_type|date

Q: output the interaction for each friend pairs in table1 in 2018

要求: user1 | user2 | interactions

123 | 456 | 25

类似这样的 output

Action 是 create 或者 delete.

Cotent_type 可以是 comment, reaction, reshare

Content_id 就只是 id of content, 如果是 comment 的话就是 comment_id

Q: output the interaction for each friend pairs in table1 in 2018

似乎没有别的那些那些算 fraction 的复杂, 只要 $\text{sum}(\text{case when action} = \text{'create'} \text{ then } 1 \text{ else } 0 \text{ end})$ as interaction 作为 output 就好, 表也只要 left join 起来就好,

但第一次 draft 写完自己也查不出什么问题, 但我也知道这不是最终答案, 索性就直接问他让她帮我查, 他说 group by 的不对, 我解释了一遍思路, 发现需要再 group by 一下 user2, 改完问她 ok 吗, 他说 ok.

最后我问了她不少问题, 她似乎不是很热情, 大概我没有给她展现出我很聪明的亚子把, 也可能她的性格就不是很热情。。

坐等两天后的拒信, 可能要回答的很出彩才能去 onsite 啦, anyway 这是我第一次店面大公司的 product 题, 之后再接再厉吧!

13.[面试经验] 脸家电面

刚面完脸家, 两道 LeetCode 原题, 一道是 isBadVersion, 一道是 304. 感觉自己思路讲的不是很好, 啊啊啊啊求过啊

14.[面试经验] FB DS hr+电面

下午刚面完电面, 本来因为 sql 太久太久不写了 比较担心 sql, 但是感觉面下来 product 说的不好。。

sql 的 table 是

date | user_id | post_id | action | extra

第二个 table 是

date | reviewer_id | post_id

第二个 table 是如果 reviewer 看到被 report 的 post, 并决定删除, 就会出现在这个表

第一个题是找出每一个 action 的 post 有多少个

第二个题是, 昨天被 report 为 spam 的 post, 有多少被 reviewer 删除

product 是一个 ml solution 可以把 spam downrank

(1) 什么 metric 可以衡量这个产品好不好

(2) 怎么确定 metric 的高是因为我们的产品 以及一些 abtest 问题.

(3) lunch 这个 product 后 time spent 变少了, 怎么解释

(4) 用什么 data 来 back up 刚刚的解释

之前 hr 面，就是几个最简单的 sql 快问快答。left join 什么的 order 的 default order

anyway，今年 fb 家政策也很迷，面面就当 practice 了。

15.[面试经验] facebook 电面

facebook 电面就面了一道题。一开始面试官就说了是一道题。

题目是 Read N Characters Given Read4 可以重复 call 的那个版本。

很快写了 code 后，面试官说很好，代码没有问题。我们来讨论一下这个方法在现实中可能会有什么问题，然后再来优化一下，问什么时候可以不用 extra buffer，直接用已给的 read4 api 读到目标 buffer 里。

讨论了输入输出后，说可以加一个地址指针，写完了代码。中间因为理解问题修改了一次代码。

第二天收到 hr 通知说过了，可以约 onsite 了。

ps: 我的 hr 说 fb 面试有一题的，也有两题的，一般会在一开始的时候就告诉你。2 题一般是没有 follow up，做完直接下一题。

16.[面试经验] 脸书电面面经

贡献一个 脸书 intern 电面面经 及其 timeline

8 月中旬收到 hello from FB

两轮电面 09/25:

第一轮：声音好听的美国妹纸： LeetCode 211 跑 test case 算时空间复杂度

第二轮：中国大叔： LeetCode 33 跑 test case 算时空间复杂度

感觉第一题写 code 写太慢了，答得不是很好！当时以为挂了，结果第二天 给了加面！

加面： 不知道某国的大叔，口音有点重，迟到了 10 分钟： 两道题

LeetCode 138

LeetCode 297

跑 test case 做复杂度分析，大叔还是比较认真，我每写一部分，他都问我为什么这么写，可以更简介优化，对我用的每一个数据结构都问了下大致具体实现，可以用别的数据结构实现吗，不停的 follow up 一度以为我之前做的是错的。语速太快加上有语音，导致我很紧张。

感觉这一轮时间很紧张！ 但好在打得比较快还是做完了！

上周末 HR 通知过了 technical 准备 team match!!!!

总结教训：做题还是要用心做，至少遇到各种 follow up 不慌！！感觉自己还是太浮于表面，导致 follow up 有时候会懵！！ 有时候听不懂面试官的问题也是尴尬，可能还是缺乏经验，

得加强锻炼了

17.[面试经验] Facebook 实习二轮 video 面试

现在 Facebook 的二面好像基本都是 video，要下一个 bluejeans 的软件，然后在 codingpad 上写题目。

一上来先是自我介绍，讲一个最近自己做的 project。面试官是一个英国老哥，一上来还有点不习惯这个口音。

然后两道题目，一道是刷题网 523，一道是刷题网 215。我当时以为二面题目会更难就准备了一堆 hard，应该也是自己运气比较好，面试官没有为难我。然后写完以后可能写太快了，follow up 问了 5-6 个，比如说 k 大的时候要什么 algorithm, k 小的时候改怎么办，然后就是时间空间复杂度。全部问完了以后还剩了 10 分钟，就愉快的聊天，问了问工作的环境，然后探讨如何在保证 freedom of speech 的同时去创造一个良好的讨论环境，以及如何保护用户的 privacy（因为楼主这一年做的 research 都跟 security/privacy 有关，加上这个大哥也是负责类似的部分，所以聊得很开心。

总的来说是非常顺利的，有一两个小 bug 但是面试官提醒一下也马上改过来。每一题都是在沟通中不断的改进 time/space. HR 说 10 天内给回复，求好运!!

18.[面试经验] FB DS 店面过经

timeline:

9/5 内推

9/6 HR reach out

9/17 确定可以店面.

10/10 店面

10/11 店面 pass

自我介绍+product+sql

Product 是经典的 Too many friends 那道题。大家看面经的时候不要太拘泥于面经里问的问题，因为问题是会变的。主要要根据面经题目发散开来去思考。我这题虽然是同一个大背景，但是具体的问题是根据当时的回答来的：

1. 朋友太多好不好

这里大部分面经都直接说不好会 crowded，但是我还是讨论了一下，如果是 new user 的话，其实朋友多是好的，可以让他们感受到 fb 带来的和朋友的 interaction，提高他的 stickness，对 user retention 是好的。如果是 existing user，但是所有朋友都是 highly interactive 的话，其实也是好的，证明这个 user 就是有很多 meaningful 的 interaction。最后如果 existing user，又有很多非 close 的 friends 的话才会有 crowding issue。

2. 现在就假设你说的最后一种情况，你觉得这种情况有什么不好
看不见真正 close friend 的 post，会间接降低 user engagement

3. 你还能想到什么其他 user story 吗

这里说实话我没懂要问什么。。clarify 了以后感觉还是没懂，就说了个 time spent 降低。。

4. 怎么判断有没有 crowding issue

我先问有没有一个 flag, 说这个 user 是 close friend or no, 面试官说有 (这里我记得其他面经明明说是没有。。三脸懵逼)

那我说最简单的就是 # close friends / # total friends, 然后我还说了一个可以把两个 group 的 post 做比较因为 ultimately 我们担心 user 看不到想看的 post, 所以这个 metrics 应该更直接。感觉面试官觉得这个 metric 很有意思, 说很少听到有人说。所以大家不要拘泥于常规的 metrics

5. 我们现在要推出 unfriend recommendation, 怎么 recommend? 这里开始 assume 之前的 flag 没有了

这里如果没有 flag 的话就不能直接 supervised learning。但我还是说了如果有 flag 的话, 可以用 supervised learning 算一个 score, 然后说了我认为应该放到 model 里的 factor, # of posts 啊 共同的 event 啊之类的。然后我说但是考虑到没有 flag 的话, 可以做 survey 得到 target variable, 或者可以直接做 cluster analysis。面试官问我还有没有什么办法, 我实在没想出来。。三脸懵逼 again

6. 如果现在我是个 teenager, 我妈妈加入了 fb, 我们没有 interaction, 你的 model 是不是会让我 unfriend 我妈?

吓了一跳考起父母加入 fb 了。。。。我就说可以加一个 logic, 根据 last name, 地理位置, 是否共同在家庭群里面判断是不是一个家庭的。

7. 如果我好朋友主要用 twitter, 我们在 fb 上没啥 interaction, 怎么办?

可以用共同好友, 看共同好友占总好友的比例 (这里其实我觉得自己的答案不是很 make sense, 因为如果用户 A 的好友基数很大, 那他和好友 B 的共同好友也许在他的好友圈里面比例并不高, 我把这个 concern 自己提出来了, 面试官说没事, good enough。鼓励大家想到就要说, 不要憋在心里)

8. 假设我们把这个功能投放给一个小镇的用户, 一半一半。experiment group 很多人都用了这个功能, 但是 #post made by close friend / # total post 并没有区别, why?

可能 unfriend 的人本来就不怎么发 post, 所以没啥影响。或者因为 sample size 不够所以 statistical power 不够高。

第二部分 sql 就是很简单的了, interaction table 找出最 popular 的 user, 没有陷阱。

19.[面试经验] FB SDE 电面

是非常 nice 软萌的国人小哥哥

找到 BST 中第 n 小的值, 应该是经典题 (但是我全然没记忆了 现场想的 orz 自己拿 testcase 来 verification 代码的有效性。

两个 followup

如果 n 是已知的一个不变的数字, 能否略过某些步骤?

如果树是满的, 能否略过某些步骤?

以上所有的时间和空间。

20.[面试经验] fb 大概率挂电面

国人小哥面，上来聊了一下暑期实习然后开始做题

第一题 easy 题，insert node to sorted linkedlist

第二题 medium，LeetCode 399，套了个马甲，变成汇率转换，大致思路有了，反正就是 map 一层层嵌套查找，用 dfs 做，但是没有写完，这道题也不太熟。估计是挂了
好不容易有个面试，move on 吧

Onsite 面经

1.[面试经验] 脸书安卓昂赛

第一轮：国人小哥：

给一个字符串，只有数字和“+”和“*”，要求计算出值

Ex:

输入：“1 + 2*3*4 + 5*6”，

输出：55

第二轮：

behavior，研究简历，问问过去的经历，最有挑战性的 project 是什么；

第三轮：

两个排好序的数组，找到公共元素：

Ex:

输入：

[1, 3, 4, 4, 5]

[2, 3, 4, 5, 6]

输出： [4, 5]

有很多 followup，例如有很多重复元素，数组很大，如何时间或者空间最优等

第四轮：

安卓系统设计，加入现在的手机都没有推送通知系统，自己设计一个，该考虑哪些方面等

2.[面试经验] FB system engineer(PE) onsite 挂经

本来以为有希望还是挂了

说是我 networking 方面太弱 还有 coding 速度太慢

1. vmstat 解释 2. server 很慢 troubleshoot 3. 各种 signal 解释

Coding: 1. 用 bash 写一个 script 把 server 上所有 txt 中的邮箱地址改掉 2. 实现 fortune

Behavior: standard 问题集

Networking: 主要围绕 TCP/IP 封包从最底层讲到最上层 NAT IPv6 尤其重要 我记不熟到后面懵了 越说越心虚 索性后来直接承认不懂 放弃治疗

system design: 这个比较奇怪 他问一个大的安装档如何在很短的时间传输到所有伺服器在有限的频宽上 比较像是数学问题

3.[面试经验] facebook 挂经

面的岗位是 research scientist, onsite 一共四轮。1. 亚裔小哥, coding, 类似于 LeetCode 415 以及若干 follow up。

2. 印度小哥, system design, 类似 web crawler。

3. 印度小哥, coding, 第一道题 merge two sorted array 变种, 第二道题是一道需要用 heap 去做 bfs 的网格题, 从来没见过类似思路的 bfs 题, 没有做完。

4. 白人小哥, 主要谈 research, 聊的不错。

这是楼主第一次 onsite, 缺乏实战经验, 运气不佳, 遇到两位印度小哥, 都没有给好评, 感觉考 design 的印度小哥好像有意刁难, 一开始嫌我的设计细节不够, 最后 hr 给的反馈却是我设计的东西细节太多, 忽略了许多宏观设计。

经验教训:

1. 对于一般人来说, 面试真的也很看运气, 不可控因素太多, 所以失败了也不要太难过, 就当是一种历练, 楼主总结了一下这次 fail 的教训, 最终还是拿到了别的 offer。

2. 任何地方都有可能出问题, 这次面试, facebook 的 interview 房间巨坑无比, code 要写在不平整的墙上的, 而且我的那个房间的白板笔没有一个好用的, eraser 的效果还不如用手, 面试结束我的手全黑了, 严重影响发挥。我接下来的几个面试都自己带用顺手的笔跟 eraser 了, 发现很有用。

3. 我发现一边讲话一边写 code 很容易降低自己的做题智商, 建议一口气把一个函数或者功能写完之后再交流, 抓重点讲就可以, 能写完题目最重要。

4.[面试经验] 脸熟现场

1. anagram in string, string 里啥都有 space, punct, letters in upper & lowercase, 不过 upper 和 lower case consider as different character, $O(n)$ 用 hashmap 做。不知道为什么这道题做慢了, run 了几个 test case, 然后时间就到了, 一个 session 只做了一道简单题

2. return 两个 array of intervals 的 intersection. Follow up - 能不能 optimize, 我就说如果有一个特别长, 用 binary search, 然后写完了 code

3. bq: most impactful project, handling conflicts; coding 是一个 array 里有 0, 把所有非零的数挪到左边, return 非零 array size

4. 第一道 easy 题, 问给一个 string 能否重新 arrange 变成 palindrome; 第二题, LeetCode 785, bipartite graph, follow up, 一开始没懂, 但最后就是想我回答, 如果一个 circle 里面 node 的数量是 even number 就是 bipartite, odd number 就不可能 bipartite

5.[面试经验] 脸家 Onsite 挂

最近收到的第一个拒信, 还有后面陆续的几个巨头拒信, 欢迎大家一起讨论

FaceBook 挂 (DS role)楼主是被内推的, 电话面了两轮, 考 coding SQL, Python/R, stats (基础) 和 business case。

onsite 一共六轮, 两轮 case, 两轮 cross-functional, 两轮 tech

Case 一般就是考 modeling, analytical thinking, 没怎么像其他面筋里面说的考 ABTest, 考了关于如何做 customer classification 问题

Cross function 就是问了一些 BQ 问题, 遇到问题如何解决这种, 我胡乱侃了一通

Tech 考了一些 stats 基础知识(还让我现场写了公式)和 coding, coding 要求一定要用 SQL, 以

及 R 或 Python 选一种. SQL 大概考了 4-5 题, R/Python 考了两个,这样看来他们真的很重视 SQL... 最后楼主是挂在了 SQL 上, recruiter 给的反馈是 modeling 部分 HM 挺满意的, SQL 方面没有达到他们的 bar. 所以我就这样悲剧了. 总结就是脸家的 DS 真的对基础要求 bar 比较高。

6.[面试经验] FB onsite 面经

coding 题总的来说都是刷题网上的 Facebook 公司的高频原题,或者一些变形

1. 一道 bq 题, LeetCode 273, find predecessor in binary search tree(指找比 target 小的最大的数,只讲思路, 没写 code)
2. LeetCode 958, follow up 是用 recursive 的办法写 (只讲思路), LeetCode 158
3. dot product of sparse vector, LeetCode 253, follow up 是找出任意一个重叠最大的区间, LeetCode 477
4. bq 轮,问了一道简单的题, LeetCode 523

7.[面试经验] 脸家现场表演

10/07 的 UDay, 10/14 收了口头 offer.

第一轮: 1) 一串儿数字, 要求找出出现次数为奇数的那个数 (仅有一个或不存在)。 2) LeetCode 161 要求想各种 corner case

第二轮: 一个在 Ads 组做算法的英国 (?) 大叔。

1) Given a data stream, implement a GetFirstUniqueCharacter() in amortized O(1) ex. abcbega.... => return 'c'

2) LeetCode 74 变形。Given a sorted 0-1 matrix, (where 0s appear to the left of 1s), find the leftmost column that contains a '1'.

[0 0 1 1]

[0 0 0 1]

[0 0 0 0]

[0 1 1 1]

=> return 1

(就是这一轮让我很虚。我两道题都没有立刻想到最优解, 都是面试官各种 hint。)

第三轮: BQ. 你最不喜欢和什么样的同事工作, 你领导给过你的建设性的反馈, etc. 反手出了一道 LeetCode 621 变种 要求 preserve task order

面完还以为挂了, 无心参加后续游园活动, 于是吃完午饭就溜了。

8.[面试经验] 脸书 PE 现场吃灰贴

Recruiter 联系的, 因为对 PE 这个职位还是蛮感兴趣没准备就上了 妥妥的炮灰了

1. system design: 如何把一个文件传送到一个 cluster, 如何把这个 cluster 里面的文件 update 而保证 service 正常
2. coding: 判断 palindrom ("a ?cb, cA*" -> true) 比较简单, 但是 follow up 试如果一个超

级大的文件怎么处理 (map reduce)

3. system: 这个问的太多了 大概一些重点: shell, fork, exec, system call, file system(inode struct) 感觉需要重新学一遍操作系统

4. soft skill: 基本就那些

5. network: tcp 3 次握手 dns 具体咋 work 数据如何传输, 竟然还问到了物理层数据咋传输的, nat, ipv6

9.[面试经验] 脸家 onsite

面了 5 轮

1. a. 会议安排最小要求会议室, 具体题号忘记了, LeetCode 原题。b. 合并 k 个有序 list, 也是 LeetCode 原题。
2. 设计不同语言翻译和推特。
3. a. 二进制的字符加法。b. 两个字符串, DFA 那个变种题目。忘记具体问题了, 就是多字符串匹配, 但是多字符串是来源于一个大字符串的子序列。
4. BQ+简单的 coding。coding 忘记了。。不好意思。
5. word ladder 那个 LeetCode 原题。要求输出字符串 1 到字符串 2 的路径, 路径在字典里面。b. 打乱字典序列的字符串排序, 就是字典序列不是 abcdefg 这种, 有可能是 gdca 这种。

10.Facebook 2020new grad 电面跪经

个人的肥死不可二进宫, 又稳稳得跪了, 没办法, 准备不充分, 小伙伴们一定多刷题, 多看面经, 做好万全准备。

题目大概就是给定一个任务执行序列, 有不同类型的任务, 每种任务执行完毕只需要 1 个时间单位, 而相同任务的执行需要至少相隔 cooldown 个时间单位, 求按照给定执行序列执行完所有任务需要花费多长时间。

LC 上只找到 621 稍微相近一点, 但是 LC 那题是可以随意排序 task 的, 求最少执行时间, 而这里是给出了 task 的执行顺序, 求一个确定的总的执行时间。

当时我想这题很快有了思路, 不过实现过程坎坎坷坷, 因为小细节有点多, “还是我的脑子开始有点乱~”, 而且结尾面试官问我能不能优化一下时间复杂度, 我一下想不出来怎么优化了, 然后就草草收尾了。还是把自己的实现贴在下面, 供大家参考:

```
// Given a list of task execution order: [1, 1, 2, 1], same number indicates same type of task;
// Executing any type of task takes only one time slot, while there should be at least a
cooldown time cd between executing same type of task
// Write a function processTasks(int[] order_list, int cooldown) that returns the interval of
executing all tasks
```

```
// For example: order_list = [1, 1, 2, 1], cd = 2, then executing situation would be:
// [1 _ _ 1 2 _ 1], the interval is 7, so processTasks() should return 7
```

```
// Complexity: assume M types of tasks, and N tasks to execute;
// Time: O(NM), Space: O(M)
```

```

public static int processTasks(int[] orderList, int cooldown){

    int interval = 0;

    int execute_time = 1;

    // key: type, value: cooldown time signal
    Map<Integer, Integer> signalMap = new HashMap<>();
    for(int i=0;i<orderList.length;i++)
        signalMap.putIfAbsent(orderList[i], 0);

    for(int i=0; i<orderList.length;i++){
        int type = orderList[i];
        if(signalMap.get(type) <= 0){ // executable
            // execute it, plus execute_time
            interval += execute_time;
            // reset corresponding signal
            signalMap.put(type, cooldown);
            for(Integer decreaseType : signalMap.keySet())
                if(decreaseType!=type)
                    signalMap.put(decreaseType, signalMap.get(decreaseType) - 1);
        }else{
            // wait until cooldown = 0, then execute it
            int wait_time = signalMap.get(type);
            interval += wait_time + execute_time;
            signalMap.put(type, wait_time);

            for(Integer decreaseType : signalMap.keySet())
                if(decreaseType!=type)
                    signalMap.put(decreaseType, signalMap.get(decreaseType) - wait_time
- execute_time);
        }
    }
    return interval;
}

```

[hide]

11.[面试经验] 非已不可 欧赛特 BQ 比较坑

上周五的四轮面试，基本挂了

第一轮：

算法：一个 BST 和一个 Target，找出 closest node to the target，要求 logN 复杂度

算出岛的个数， LeetCode 200

然后是算面积和周长，没做完时间就到了，算法基础较弱。。。

第二轮: BQ

感觉回答的不好, 很多题都没准备到, 回答的有些犹豫和慌张, 比如

和同事的最大 conflict, 和 leader 的最大 conflict, 和其他组合作的最大 conflict, 别人一直拖着活不完成导致你没发开始改怎么处理,

用户紧急的需求和 你不做其他 team member 就会被 block 的情况, 改怎么处理 (假设都只有你能完成)

吃饭, 周五的伙食还不错, 有酸奶, 中餐做的不错, 还有两种点心, 和面试官聊了聊项目组业务和项目的发展方向。

第三轮:

UX + 前端的项目设计: 很新颖的方式, 感觉面试官很有水平, 只画了图, 没有要求写代码, 很考验知识点宽度和深度。

提出了一些需求, 根据要求画出大致的 UX wireframe

然后该怎么设计这个项目, 问了怎么配置 LESS, 怎么配置 Redux, 和后端怎么配合, 怎么做前后端分离。。。

前端怎么优化项目, 针对 React 有哪些优化方式

中间穿插了很多的前端基础面试题, 和 React 面试题。还聊了 React16 的源码优化, 异步渲染, fiber 原理。源码我只看了 fiber 部分, 其他主要是听面试官说, 时不时插上两句自己的想法。

这一轮面完感觉以后面试可以从很多方面去扩展, 收获很多。

第四轮:

前端面试, 写 autoComplete 的时候主动加上了 debounce 和 infinite scroll, 把 cache 给忘了。。。

然后就聊天

面了好几家公司, 不得不说, FB 的面试官水平明显要高一些, 问的问题无论深度还是广度都更有难度。

大厂还是难进, 感觉应该没戏了, 继续面吧。

12.[面试经验] 脸熟昂赛

昂赛三轮:

1. Add operator
2. Alien Dictionary
3. Find the first 1 in a matrix for each row

13.[面试经验] 脸家 现场

1 behavior

美女经理, 常规 bq, 但是所有问题都需要个 concrete example 就和其它的不太一样了, 因为例子的准备不充分, 有几问感觉没答好

2 coding

小印, #1, LCA, 在只有 parent pointer 的情况下

#2 meeting room II, follow up return 每个 meeting room 包含具体的 meeting, eg Room A (1, 2), (2, 4), Room B (1.5, 3)

3 coding

用 trie 的带 wild card 的 string matching, *除了可以表示任何 char, 还可以是 blank

4 design

wiki crawler

14.[面试经验] 非死不可现场面试经历

据猎头说, 上次昂赛一个面试官和同事意见不和。为慎重起见, 决定加面一次。我还以为全程 behavior 呢。可是全程技术面试! 第一题: 给出 YOY DAU 增长率 15%, YOY posting user 增长率 8%。分析数据背后的故事。

这题很刁钻。我就死马当活马医, 心一咬, 牙一横, 开始从人口分布说起, 又说到人口老化问题, 再说到产品 feature 发布, 最后在面试官提示下, 说到用户使用终端从 desktop 到 mobile, mobile 安卓和苹果之争的问题, 等等。

其实这道题问的就是数据的 dimension 问题。

第二题: 给出一个数据库 table, 有四个列:

userid int

timestamp datetime

action enum {"login", "send", "update"}

client enum {"android", "iOS", "desktop"}

问如何得到每天数据

DAU

send count

iOS send count

新用户

date

我根据以往的经验, 就说根据上述数据, 建立一个 table, 每天分区。这两个 table full outer join 后这样的数据很容易得到。

第三题: 给出一群 metric table, share 共同的列“date”, 其他的列 follow 同样的结构, 就是

- 有一个 dimension 列, 每个 table 名字不一样
- 有一个或多个 metric 列, 每个 table 名字不一样

如何写 script 产生一个 query, 把这群 tables 数据变换成如下格式并插入到 warehouse 数据库里?

date

dimension

dimension_value

metric

metric_value

面试官很善良地给 hint。我各种配合。

最后的解是：用 Python 写脚本程序，用 JSON 传递数据库 schema，用 txt 文件写 template。

```
def query_gen(json, template):  
    return ' union '.join([table_query_gen(table, template) for table in json.tables])  
def table_query_gen(table, template):  
    return ' union '.join([col_query_gen(table.name, col, table.dim, template) for col in  
table.metrics])  
def col_query_gen(table, col, dim, template):  
    return template.format(table=table, col=col, dim=dim, metric=col)
```

结论：数据工程师职位不好拿呀。需要 product sense。面试时候套路挺多，挺独特的。

www.jiuzhang.com