



我的人缘 0

2 主题 | 3 帖子 | 291 积分

发消



分享帖子到朋友圈

erica119120 3 天前 | 只看该作者

1 楼 电梯直达

本楼: 100% (9) 0% (0)
全局: 100% (53) 0% (0)

2019(10-12月) MachineLearningEng 博士 全职@Amazon - 内推 - 技术电面 Onsite 视频面试 | Other | 在职跳槽

本帖最后由 大队管理员 于 2019-11-26 12:02 编辑

地里的面经很多, 但是很少有人写总结, 尤其是Machine learning这一块。我根据自己面试的经验, 总结了一下ML design的面试中, 解题的思路, 抛砖引玉, 分享给大家。

求加米, 给我米不会减少你自己的米, 举手之劳, 互惠互利。

1. 结构化表达:

通过结构化的表达来让答案显得很有逻辑。结构化表达, 针对一个问题, 一般讲三点答案, 逻辑上依次有递进的关系, 这样就会显得你逻辑清楚。超过三点, 听的人容易lost, 所以也没必要。

比如被问了一个design的问题, 答案比较复杂, 有很多部分。回答的时候, 第一句话先说, 这个问题可以从2(数字)方面来考虑: 用户和系统。用户这方面有2(数字)个solution, 系统这方面有3(数字)个solution。因为面试官不知道你想讲什么, 在讲细节之前, 先把框架说出来, 这样有助于面试官follow 你的逻辑, 即使中间在细节上仔细询问你, 他也知道你后面还有内容要讲, 会配合让你讲完。

2. 讲自己做过的项目, 怎么讲?

讲自己的项目时, 主题最好和这个组的方向相关, 技术越新越厉害, 越好。出于NDA, 最好避开自己在当前公司的工作。雷区: 这样可能会导致只能讲几年前的工作和技术。怎么解决呢? 在报告的结尾, 给一个rethink, 讲两部分。

第一, 反思如果现在做同样的问题, 哪些地方可以提高。[重点显示你的见识提高了, 不是反思你的缺点]
分三个部分: 新技术[比如以前用 logistic regression, 现在可以用 deep learning/LSTM], data size scale up[比如以前只有10k的数据, 现在是100M, 可以使用更复杂的模型], system[single pc -> distributed system/GPU, training could be paralleled or in the cloud. 更advanced approach to collect labels, 比如大公司都会有自己的crowd sourcing judgment platform]。

第二, 概述一下, 你在目前的职位做过的工作, 使用了哪些最新的技术, 这样显得你的skillset在与时俱进, 没有过时。

3. - It is your opportunity to shine! 你会被要求讲自己的经历, 可能很多次, 面试官根据你的经历来问问题。在讲经历的时候, 突出哪些关键字和技术? 一是和对方相关, 二是新技术, 三是显得你水平高的内容。当然一定要是你熟悉的, 如果你被问到什么, 答不上来, 会特别减分。

4. - 讲到使用哪个model的时候, 列举2-3个相关的model, 答案的重点是讲一下各自的优缺点, 以及当前问题的特质, 基于这些比较, 从而决定选哪一个。

以下内容需要积分高于 100 您已经可以浏览

5. - Lead conversation, 主动推进谈话。比如讲完一段话或者回答了一个问题, 主动询问: Does it make sense to you? 比如讲一个复杂的系统之前, 先告知对方: I'm going to talk about a few components. If you feel interested in some particular ones, you can point out and we'll discuss more over them.

6. - 会沟通。有些问题问的非常模糊, 可能是面试官故意的, 也可能是面试官的表达能力差。这种时候, 通过rephrase问题 + make concrete example, 来主动probe面试官, 去除面试官自己的误差, clarify需求。

7. - 会接话, 能听出面试官的concern, 不要等面试官停下来就接过话题, address面试官的concern。有的面试官在交谈中, 不是每个问题都会explicitly的问, 然后停下来, 然后等你答。你要能在交谈中, 听出弦外之音的问题, 立刻想答案, 然后看准时机, 当机立断的接过面试官的话题, 给出让人满意的答案。

8. - Stress test. 碰到面试官很不友善的时候, 没法develop一个很好的答案, 要坚持the principle of problem solving: Break down problem to solvable subtasks. 比如被问到这个问题 "你deliver了一个ML的产品/系统, 用户使用以后, 汇报系统的accuracy 远低于你自己test的accuracy, 哪些方面可能出问题了? 要求不能看log。" 从problem solving的角度来看, 整个系统有两个element: 你的ML系统, 用户的application。每个element都可能出问题。那么把大问题break down成两个element自己的问题+element之间衔接的问题, 就是一个很好的答案。两个element自己的问题包括: 产品的问题(overfitting, training data coverage, etc), 用户的application的问题(使用产品的domain和develop 产品的domain不一致, 使用方数据的distribution和training data不一致, etc), 用户的问题(没有按照设计的方式来使用系统, measure的方法不对, 使用了和开发方不一样的metrics, etc)。