

ML&DEV[17] | 算法工程师的技术深度

原创 机智的叉烧 CS的陋室 2020-07-19 21:30

China-A

徐梦圆 - CHINA (徐梦圆Original Mix)



【ML&DEV

】

这是大家没有看过的船新栏目！ML表示机器学习，DEV表示开发，本专栏旨在为大家分享作为算法工程师的工作，机器学习生态下的有关模型方法和技术，从数据生产到模型部署维护监控全流程，预备知识、理论、技术、经验等都会涉及，近期内容以入门线路为主，敬请期待！

往期回顾：

- ML&DEV[11] | 浅谈模型的局限性
- ML&DEV[12] | ML中的数学学习
- ML&DEV[13] | 快速从无到有建模完成思路
- ML&DEV[14] | 浅谈解决问题的能力
- ML&DEV[15] | pyspark杂记

最近在和大佬们讨论，算法工程师的发展路线会是什么样的，如何提升才是一个算法工程师合理的发展方向，做了一些思考和总结。

开发技能的深入

算法工程师，首先是一名工程师，然后才是算法，在有关工程的理解上本就应该多多益善。

首先，最基本的应该是，自己能简单的完成算法功能。例如一些简单的数据处理，文件操作、特征工程，复杂一些则是一些自动化打标的方法，最好的考查方式其实就是应届生考试里面的算法题。

紧接着应届生的算法题，其实考这个能考察的点很多，除了基本功能的实现（说白了就是考察这个人会不会写代码），另一点就是考察这个人对性能调优的能力怎么样，时间空间复杂度如何如何，很多时候所谓的模型压缩其实就是在干这个事情。

其次，就一些大数据技能，如何用分布式的方法实现对数据的离线处理，算法处理成千上万甚至更多的数据可以说是家常便饭了，使用大数据的方法能加快数据的处理，早期有mapreduce，在hadoop生态下现在的spark就是也非常成熟，会使用这些方法来进行处理，能大大提升处理的效率。

再者，就是工程上的理解，作为一名工程师，在整个系统中承担一环，但是由于需要处理很多模型，因此一不小心就会成为性能的阻塞点，因此，模型的选型必须考虑到各种性能。

对于技能点想点全一些的，其实可以往后端工程师的东西去看看，常用的服务可以尝试写写，如grpc之类的，毕竟模型最近好的上线方式就是单独构建服务上线，所以自己多少还是需要有所了解的。

算法能力的深入

算法工程师，算法肯定要很强，这个毫无疑问，否则就和低配版后端工程师没什么区别了。

首先就是对一些特定任务，要逐步有比较完整的认识，一方面是有哪些常用的方法，各个方法的优缺点要比较了解，以命名实体识别为例，要知道CRF、BILSTM、BERT-BILSTM-CRF之类的常用玩法，每个玩法的效果、性能都是什么样的，有哪些适用范围和改进空间，这些都十分有用。

在对一些稳定任务比较擅长后，开始了解更多的相关任务，开始提升，文本分类，文本相似度什么的，都开始慢慢了解。

这种深入应该是最体现算法工程师深度了，对于一个任务，自己应该要擅长的任务，应该清晰知道都有哪些方法，以便在进行特定场景下，一方面有方法可以选，另一方面是知道怎么选，知道所有方法的优缺点，能够因地制宜，最终实现更为精准的预测。

在一定基础后，该开始做的就是把这个坑挖大，例如从NER任务跳出到整个query理解任务，多去了解整个query理解任务是怎么做的，框架和架构是什么样的，有了这些思路，技术深度就加深了。

领域理论的深度

与工程岗位不同的是对算法理论需要有比较高的理解。

首先是数学，和很多人觉得数学难，生搬硬套模型完成任务，偷鸡是总有人可以偷得到，但是问题是偷鸡毕竟是少数，自己能偷的以后就不好说了，所以需要做的还是要扎实数学基础，这是对模型理解的一个上限，数学不懂是基本无法看懂复杂模型的，出来混总是要还的。

然后就是一些模型了，模型我们要学它的结构，但是从我的经验来看其实看看introduction之类的，对模型的理解会更深，前面提到的“优缺点”，其实每篇论文作者都会聊到，包括出现的场景是什么场景，出于什么原因而在前人的基础上做一些更新，所以一般确认了需要精读这篇论文，就不要想着捷径，不要想这简单，踏踏实实把事情做好吧。

另外，模型的深度除了在理论上，还需要落在代码上，多去看代码，尤其是一些比较出名的算法的代码，多看看多读读，有很多东西不落到代码层面是说不清楚的，例如albert的权重共享之类的，这些就非常重要吧。

其他应该提升的能力

首先，我的理解是解决问题的能力。要成长，从来都需要从点到面，要从慢慢做一个工作的人，成为一个能解决问题的人，举个例子，举个例子，以前更多是有个leader告诉我该做什么，怎么做，现在要升级为自己想出一个合适全局、效果最好的方法。

然后，是一个有长期意识的人。方案都需要考虑一个模块甚至一个系统的长期发展，短期内的方便，如果会导致未来改进的成本很高，那可能就要修改了，另一方面，还是要考虑有一些未来需要的基础工作，可以尽快从现在可以一起做，逐步构建起来。

第三，就是沉淀能力，对于校招的新人，工作其实本身就是成长，因此需要学会沉淀，这里的沉淀是指有一些自己的工作，可以抽象出来，日后是可以用的，这个能有效提升未来的工作效率所以很有用。

小结

近期因为一些事情，重新思考自己的提升方向，所以有了一些结论，欢迎大家讨论。



喜欢此内容的人还喜欢

心法利器[55] | 算法工程师读论文思路

CS的陋室