技术人如何深入人工智能

时间 2017-08-18 07:13:54 公众账号

原文 http://mp.weixin.qq.com/s/ag5e_0bXK6lS2RKe15uiiQ (http://mp.weixin.qq.com/s/ag5e_0bXK6lS2RKe15uiiQ?utm_source=tuicool&utm_medium=referral)

主题 深度学习 (/topics/11020125) 人工智能 (/topics/11020194)

前不久趣直播举办了一场技术人成长交流会，邀请了《TensorFlow 技术解析与实战》作者李嘉璇来分享。以下是视频：

以下是文字版：

谢谢智维的介绍，介绍的太棒了，我其实没有他说的那么棒，让我压力好大，尤其巧哥又讲的那么好

我目前主要是做人工智能方向的，比较熟悉的是 TensorFlow，我有写过一本书，《TensorFlow 技术解析与实战》，我这里面再多说两句，因为有非常多的人会问我，你是怎么写出一本书的？或者说你写出一本书遇到什么样的困难？ 我现在面试的时候就会有人问我这样的问题，会经常让我去说这个事，我跟大家分享一下。

我在写这本书的时候，我脑子里没有第二件事了，我说的第二件事就包括吃饭和睡觉上厕所，脑子里没有这些，就是在我的意识里，这些东西都不存在的。

我眼里只有这一件事 ，我活着就是为了写书这一件事，所以我白天和黑夜就一直在做这件事，直到把它做完。

比较骄傲的是，我跟我朋友说，我想写一本书，我朋友说，她也想写一本书，但是她是某个创业公司的CEO，我们俩基本上同时开始，直到我的书出版了，她还没有写完，基本上还停留在一二章的样子，所以我觉得这个是我积累的一个心得。

我那时候也不发朋友圈，从来不发朋友圈，什么都不干，就只干这一件事，才把它完成。

目前我也希望再回归到这样一种状态，专心的写代码，不去干第二件事。

下面介绍一下我自己，最近这几个月基本上所有的人工智能大会我都有参加，也都有去做一些分享。

相关站点

公众账号 (/sites/ilzq6rE)

+ 订阅

热门文章

- 1. 如何更好地阅读包含很多数学相关内容的机器学习论文？ (/articles/beiaqyr)
- 2. [译] 用 Google 新开源的 deeplearnJS 预测互补颜色 (/articles/BJVVBbI)
- 3. 大规模知识图谱的构建、推理及应用 (/articles/uqYJbyu)
- 4. 数据科学家必须知道的 10 个深度学习架构 (/articles/7RfyAvQ)
- 5. Alex Smola论文详解：准确稀疏可解释，三大优点兼具的序列数据 (/articles/2Q73MvB)

之前在百度做深度学习方向的开发，目前是在一家人工智能创业公司做的 TensorFlow 的优化以及 TensorFlow 在 FPGA 上的编译，应用在一些物联网领域，实际上是做 AI 芯片的开发，就是如何让一个算法在硬件上能够更高效的运行？一方面是准确率更高，另一方面是速度更快，精度更高。 这是我做的一方面的事情。

目前在做的第二件事情就是跟招商银行那边去合作一些关于金融的智能聊天机器人方面的东西，根据用户的一些收支情况，用户的消费行为，给他推荐一些智能投顾的基金和产品。

剩下的是一些其他的杂事，比如说会有一些 CSDN 的约稿之类的去写一写。

今天我主要分享的就是这样几个方面，首先是从人工智能的应用开始介绍，尤其是机器学习和深度学习在我们产业界的应用，这是第一个方面。

第二个方面是人工智能非常有诱惑力。 诱惑力在哪里？我之前做过一段时间的网站开发，为什么后来我就特别迷算法？特别迷深度学习，特别着迷算法如何在硬件上编译快。

刚刚巧哥说的，我之前做的东西就是把别人的业务逻辑翻译成代码，我觉得我做的是一件翻译的工作，我做的时间太长了太久了，我翻译的越来越好了，我架构搭的越来越精美了，但是我觉得成就感反而越来越低了。

有一个笑话经常讲说，写 PHP 的可能就不带脑子，我觉得我带的脑子比较少，我希望今后能够更深入的思考，然后能够有一点点提升而兴奋的那种工作。

而不是像我以前那种，总是有现成的参考案例，或者总会有问题，能够问到别人，而做这个方向让我觉得，我基本上经常是问不到别人的。

我们开会也没有人能够给出一个解决方案，大家都只是猜测哪个方向会更好？我朝哪个方向可能会有一些优化的点？但是有没有都需要去做实验去尝试，可是这是跟以前的开发模式完全不一样的一个领域，而这个点是很吸引人的。

关于深度学习或者积极学习，如果要入门的话，总结了七个步骤。

关于 TensorFlow 方面的，TensorFlow 的话是一个非常容易简单上手的深度学习框架，然后简单介绍一下它的变成模型，以及如果你想深入到 TensorFlow，了解一个自然语言，处理这些领域或者图象这个领域，去讲解一下一个例子，就是 TensorFlow 在自然语言处理中的应用。

最后的话我会讲一下，关于深度学习的一些云的基础服务，对于国内深度学习研究方向，我总结大部分公司做的大概有三个方向。

第一个方向是关于云深度学习平台的建设，这个基本上是国内的公司，比如说第四范式的先知平台，百度的BML，就是机器学习平台，他们正在做的一些事情，这种东西是将来希望作为一个云深度学习平台，对外出售的，就像阿里云那种，就是卖服务的这种东西，这是AI平台。

第二个方面是我目前正在做的，AI 芯片，包括有地平线，有寒武纪，有深建科技，这些公司，主要是把我的算法跟硬件相结合，这个硬件就包括 GPO 和 FPG 这些硬件相结合，如何让这些算法在这个硬件上能够更高效的执行？并且速度更快，精度更高，功耗更小，成本更低，这些都是更，所以你就没有一个最终的标准，但是我们探讨如何去做这个事情?

第三个方向是图象领域的一些应用，当然也包括一些自然语言处理，图象领域就是一些 Yijia 公司 或者马龙科技，他们做的事情就是说，这些公司都是做服饰的，或者时尚的，他会把你人穿这件衣服，衣服的属性，人的面部表情，人的情绪给你提取出来。

会有一个巨大的图象和提取出来的这些标注的库，将来以及图象和视频方面，比如说你今后对视频做搜索的话，你可以直接通过文字就跳到视频的某一帧，类似于这种，因为每一帧的这个视频信息都给他结构化成文字的信息提取出来了，这是图象领域。

自然语言处理领域就是迅飞和搜狗可能做的，迅飞听见，迅飞语音识别，搜狗也是做类似的事。但是我觉得云深度学习平台也是一个非常基础的，深度学习服务的领域。

最后是一个小的彩蛋。

第一个方面是机器学习和深度学习的应用，我相信大家平时去看公众号，资讯，各种微博，都会知道，应该知道关于人工智能方面我们采用的方法是机器学习和深度学习。

那么他们的应用主要在这三个领域。

一个是图象，图象可以做图象的分类，图象分类我们经常说的图象识别，还有是目标检测。

目标检测要达到的效果有两点。 第一点是说我在一幅图象中把所有的物体给框出来，这是第一点。 第二点是物体框出来，把物体的位置定位起来了，但是还要实现第二个功能，就是把这个物体的类别也识别出来。

在图象分类或者图象识别上更难的一点，不仅要知道这个图象是什么类别？还要把这个图象的位置精确的框出来，就像这幅图里面显示的猫，狗和鸭子，不仅知道它的类别，并且框出来，这叫做目标检测。

图象分割的话指的是在一幅图象中，把图象根据象素的要求，把这个图象的轮廓标注出来。

这幅图中，最右边的一个例子，把这个猫和狗根据它的象素，把它的轮廓给标记出来，其实知道这点蛮重要的。

我之前带过一个深度学习的团队，当时我还算是他们的负责人，我的主管是做语音方面的，竟然对图象分割和目标检测是什么？竟然完全不知道，可能没有接触过这一方面。

我觉得在 AI 领域，如果你是做图象的，我认为你应该也要去对文本的处理方法，文本最新的研究要有一定敏锐度，因为你总做图象，其实你是缺乏灵感的，其实你经常需要从别的领域去吸收灵感，我认为是这样。

还有就是语音，这一块就是目前，迅飞正在做的一些事情，语音识别，包括语音听写，语音转写，还有命令词识别。

命令词识别大家就知道，比如 Google Now 这样的产品，你要唤起他会有一个命令词，就是你说一句话。 出门那种导航产品，你会说，你在车里面喊问问，然后它就会待命的听你说话，这是命令词识别。

语音合成，大家应该知道百度地图，给你导入林志玲或者郭德纲帮你导航，就是语音合成。

声纹识别，跟人脸识别差不多，就是支付宝可以刷脸支付，声纹识别跟刷脸支付的含义是一样的，就是通过声纹鉴别出你这个人来，这是语音方面。

文本是有些文本情感分析，主要用在百度糯米的一些产品，然后他会对商家有一些评论，然后你希望知道这些餐厅他的褒贬情况，然后文本情感分析，根据它的评论知道它是褒义还是贬义？就是它的分数应该归到哪一档?

文本分类就是假设你在知乎上，你去提一个问题，你把这个问题标题和描述写出来以后，希望它能自动的给你归到什么类别？比如说职业发展类，还是情感问题类，还是技术问题类等等？然后还有一些挖掘等等。

这是目前人工智能的应用，当然这些东西你都可以用于传统的学习方法来做，但是目前大家都流行的是，在做这些应用的时候采用深度学习方法，也就是采用深度神经网络的方法去做。

如何实现我上一个 PPT 做的功能呢？传统的方法是用机器学习方式去做，干什么的呢？比如从数据中，数据可能是语音的数据，文本的数据，以及图象的数据，一张图片就是数据，语音的话一段语音，语音就是数据，以及文本，文本就是你发的帖子或者你的评论都是评论数据。

从这些数据中抽取规律，提取有用信息，有用信息就是刚才我说的那些数据中的特征，然后我们用这些特征来代表数据，解释数据，从而达到预测未来的目的。

继续学习算法有这些，我就不去讲了，大家本科的时候如果学过计算机大家知道。

机器学习过程有哪些呢?

数据预处理的过程，你需要对原始的数据做一些去噪，或者归一化的处理，应该大家知道一般用的方法，比如说你把数据减去平均值再除以方差，类似于这种归一化处理，所以说你把数据最大值减去最小值再除以平均值，去做一些处理。

接下来就是我们最最重要的环节，拿着数据训练一个模型，这个模型在机器学习或者深度学习当中，它就指的是一堆参数，就是一个文件，这里面包含了一堆的参数。

训练出这个模型以后，如果再来新的数据，输入到这堆参数当中，这堆参数相当于，你认为他是做了一个函数的数学变换，然后再有一个新的数据经过这对参数的处理得到了你想要的结果。

比如说做文本分类的话，我在拿了一百万的文本去做了分类，每一个文本都可以分成很多类，然后接下来如果拿一个新的文本，经过模型以后，就直接能够输出这个文本的类别。

还有一些训练模拟以后，需要一些评估模型方法，最后应用模型的话就是说，我们把这个模型上到线上，这样一个过程。

这里就简单提一下机器学习算法应用，其实大家看过周志华的书或者李航的书，机器学习或者统计方面的书，大概的话他们的应用都有这些。

回归有一些房价的预测或者有些用户消费行为的预测，然后分类，分类的话有文本的分类，图象的分类，图象分类就是图象的识别，垃圾邮件的分类等等。

推荐是一个目前很火的领域，也是薪资比较高的领域，比如说做一些电商的推荐，还有一些视频推荐，还有一些短视频的推荐。

排序领域，主要应用的是方法，比如文本的相关性检索，就是在谷歌或者百度的搜索结果页产生之后，去找出跟你的查询词最相关的网页做一个排序。

我讲的第二点就是深度学习入门的七个步骤，这是我大概总结了一下。

如果你完全没有接触过深度学习，我认为做深度学习之前，很多同学也问我，我现在想入门深度学习了，然后我需不需要先从机器学习开始看？我需不需要先从高数，线性代数然后那些知识开始学起？逐步把各个算法都弄明白，然后我再上深度学习？我觉得是不太需要的，但是这个过程是个相互的过程。

我下面就跟大家讲一下入门的 七个步骤 。

首先第一个步骤就是你去学习或者回忆一些数学知识，但是其实也不是说特别刻意的去学习，因为比如说我刚才说过，就是一个模型它是一堆的参数。

其实训练过程，我们在深度学习我们都用深度神经网络的方法，每一层的话需要一些，我们输入的数据也称为特征，我们会把这些特征做一些转换，通过一些线性或者非线性函数转换。

所以我这里说学习回忆一些数学知识，指的就是你要掌握一些，比如说概率统计，高等数学，线性代数的知识，就是如果你学过了那是最好的，大学学过最好，但是很多同学大学没学过，然后没学过也没有关系，你就知道大概原理是干什么的，每一个概念它的原理大概是干什么就可以了。

有兴趣的时候可以涉及到一些推导和证明，没有兴趣的话推导和证明其实我认为在这个阶段也不需要的，你就只知道这个名词大概是什么意思就可以了，甚至说你把这个名词输入到百度百科当中，大概知道这个是干什么的就可以了，因为这是第一个步骤。

你可能学到第七个步骤了，你忽然觉得学习这个，你需要真正知道这个乱发的含义，你需要知道这个算法中每一个参数到底是什么含义？你需要知道这个算法的推导是怎么样的时候？你可以再回来去学习，这个没有关系。

第二个步骤就是你掌握一些机器学习理论和基本算法，其实我在做深度学习的时候，我绕过了机器学习，没有去看机器学习，所以做了很长时间以后，我才回过头来，我觉得我机器学习需要补很多东西，然后才开始把这些都给补上的。

其实大家现在看起来就是，其实蛮难的，比如说 SVM 各种推导，然后 SVM 各种理论，引入了合函数，就是每一步为什么要这样做？为什么那样做?

当你做一个新的实践项目的时候，你光看理论其实很难理解的，但是这时候可以先做一些积累的储备，比如说你学一些 SVM，逻辑回归，决策树，贝叶斯分类器，随机森林。

我现在看起来就觉得这个非常简单，但是当时的话看每一个名词都觉得，都像一本书一样困难，然后关联性分析，人工神经网络和传播算法，一些 PCA 降维的方法，以及什么叫过拟合？什么叫正则化等等这些。

然后就是掌握一种编程语言了，我估计大家现在可能都是 iOS 开发，很多同学都是 iOS 开发，所以其实，可能推荐大家去掌握一下 Python 语言吧，当然如果不懂 Python 语言的话，我知道 iOS 新出来的 iOS11，就是前两个月看到，目前也有对 iOS11 当中也有工具包，所以也就是说你应该用 iOS 的语言，也可以做一些相关的东西。

会有一些后面直接的 API 去调用，而这个 API 还特别丰富，涉及到我刚才第一页幻灯片说的关于图象，文本，还有语音各种API都非常的全，然后我也有同事在公司内部也分享过新的 iOS 在继续学习上的应用，大家也可以看一看，那时候也可以说不去掌握这些语言。

但是实际上掌握 Python 的话，原因是说你做数据处理和数据分析是非常方便的，同时有 Numpy，Pandas，这些第三方的库，然后能够让你做数据分析跟查询 MySQL ，就是跟处理数据库一样方便，你像处理数据库，做一些具体函数的处理，那些都没有问题，用这种 Python，Pandas 库都可以实现。

然后你也可以自己多试试一些机器学习的库，比如说 sklearn，就实现了我上一页 PPT 说的各种逻辑回归，知识向量，PCA，朴素贝叶斯各种等等算法，用一行代码就可以搞定的，你可以看一看它，大概更深一层的理解了，前面说的各种机器学习算法的一些原理。

还有就是你可以去试一下 MATLAB 或 R，因为我们公司现在目前就是特别多清华大学，很多很多清华大学的实习生，我发现他们都用的 MATLAB ，我也去看一看，其实它的实现思想跟 Python 也是比较相似，可能 MATLAB 在学术界用的比较多，Python 在工业界用的比较多。

我最开始是做目标检测的，然后那时候目标检测一些脚本，其实也是用 MATLAB 写的 ，但是也非常容易，你坐在那很认真，然后一天就能把它看懂。

第四个步骤就是去读一些经典的论文，关注一些动态和研究成果。

你要去训练一个手写数字识别的模型，这个模型是深度学习最最简单的一个任务，就是说对手写数字做一些图象的识别，最古老的方法就是 LeNet，1998 年提出来的一个神经网络方法。

比如说做目标检测的话，非常非常简单的网络，用 MSCNN 的方法去做，这时候你要做之前，实际上非常重要的是去先读一下关于 LeNet 或者 MSCNN 的论文 ，他们论文都非常早。

LeNet 是 1998 年提出的，1998 年的论文，也是大牛写的，论文非常朴素，不是说看不懂，还是比较容易看懂的，那具体论文怎么搜索呢?

比如说你想实现这个任务，你直接在 GitHub，然后你输入 LeNet 这些关键字，然后它会有 LeNet 在 GitHub 上会有一些实现。

他都会给你标注出来他采用了哪篇论文?

这些论文的作者是谁?

这个论文的链接是什么?

特别规范，所有参考的东西，所有引用的东西都非常规范，所以你直接去看这篇论文，基本上知道它的网络结构是什么样子了?

还有就是你要关注最新的动态和研究成果，这一点在工作中，我觉得灵感是非常重要的，因为这个工作跟传统的以前开发工作特别大的区别就在于，以前开发工作就是巧哥说的，很大程度是翻译工作。

我们现在这种工作就是，经常是没有方向的工作，就是大家坐在一起，忽然有一个问题，就是死活提高不上去了，比如说精度到 98%了，但是我们希望它的功耗更低，比如说功耗现在是 10 瓦，我们现在它达到 5 瓦以下，5 瓦以下就可以更省电。

我现在待机，一个智能的产品，比如说我们这个芯片，物联网嘛，放在智能的空调家电等等，不同的智能设备当中，我们希望它在断电的情况下，假设它还能撑一段很长的时间，或者说它能功耗更低更省电。

所以这时候，可能我们算法上觉得没有优化的地方了，所以每天关心一些最重要的论文或者资讯是很重要的。

尤其是在你面试过程中，这个面试就有一个套路，面试官会经常问你，你最近读了哪些论文? 2017 年CVPR大会上的哪些论文，你对它理解比较深? 可能大家不会问你具体去写的代码了，一般都会问你对这个论文的理解，这个论文当中有什么关键点，哪些比较吸引你? 所以多看一些论文笔记以及公众号，还有微博的资讯。

第五步的话就是自己动手训练神经网络，首先就要选择一个深度学习框架，这里面我自己一路一直是在做 TensorFlow 方向的开发，所以我也推荐使用 TensorFlow。

原因是什么? 首先它的 QQ 群和微信群活跃度非常高，然后各种杂志公众号，还有微博关注的人很多，你只要写 TensorFlow 相关，非常多的编辑会愿意帮你约稿，或者他们平时老会追着你让你去写一些东西。

谷歌 GDG 也在中国非常愿意去做这些东西，而且影响力又很大，并且说全球关注很多，比如在写书的过程中，我会经常遇到一些问题，因为那时候研究的非常前沿，我觉得我身边我问不着人了。

但是我在网上了总是在每一个角落翻了十几页以后，或者在 Facebook 上，某一个人说了一句话，我突然觉得原来是这个样子? 没有说解决不了的问题，因为很多人都在关注它。

还有行业交流和技术峰会的讨论非常多，以及国内外研究信息同步，这个是指，实际上在人工智能领域有个 arxiv.org 那个网站，这个网站每天全世界各个的科学家都会往上面去发论文。

这些论文是预备的论文，只要有人发论文，立刻就有其他科学家会把这个论文用代码实现，提交到 GitHub 上，可能时间非常短，可能一到两天就有了，而且他们是用 TensorFlow 实现比较多，基本上你现在见到的论文模型在 TensorFlow 上面都有实现。

你如果看到一篇论文的话，你可能都自己不用手写代码，然后自己直接从 GitHub 上面找一找就能找到一个现成代码去做这件事了。

而且 TensorFlow 它的第三方框架，上层的应用库也比较丰富，就相当于我们以前做的开发的翻译工作，你脑子里想一个神经网络是一个什么样子？你就写出来了，你想什么样就写出来，就是跟我们实现业务逻辑差不多，这就是 TensorFlow 提供的非常优秀的第三方库。

也得到了 TensorFlow 官方的支持，然后目前 TensorFlow 今天的数据，他的 star 数大概是六万，已经有六万多个数，并且官方每天都会回答各种解决它的艺术问题，所以我说有 bug，他们修复 bug 特别及时，一个月出三个版本，然后每个月出一个大版本，然后进步非常快，然后性能也提高的很好。

然后就是你深入你感兴趣的感性领域或者你目前工作相关的领域，比如说你计算机视觉，因为业界大家按这个分的，你是做视觉的还是做 NRP 的，结合这两个一起发挥价值的，业界一般都是按这个分的。

所以你要如果今后想做视觉领域，计算机图象处理，图形学处理方面的书，也就一两本，我觉得基本上把整个系统知识有一个很好的理解，然后你结合你目前的工作深入到某一个领域，完全读这个领域去写实现这个领域的模型。

这里我也举了一些例子，比如说这是单纯的我们说它的应用领域，但是如果你要结合某个行业，比如说在医学行业当中有一些医学影像；比如穿衣的话，衣服搭配，衣服款型的识别；比如说保险业，通讯业的话，可能会有智能问答机器人这些；然后到智能家居领域，可以做一些人机的自然交互，这些都是依赖于你上面你所从事的这个领域。

然后就是说，训练好模型以后，可能会出现一些问题，比如准确率的问题，识别速度，最重要的就是识别速度，识别速度是非常关键的，准确率的话，其实大家采用的算法差不多，模型差不多，准确率基本上差的不是很高，比如说我 97%，你 98%，其实差的不是很高，但是你要说误差率 97% 和 98%，是差 30% 的，对吧?

但是实际上在工业界应用的话，如果不是在自动驾驶等等那种安全领域应用，比如说在安防领域，97% 和 98% 其实如果谁先工业化落地谁先抢占了市场，不是差很多，但是最重要的是识别速度问题，然后有一些 badcase 都可能是你遇到的瓶颈，然后这些训练好的模型需要不断优化。

如何去优化呢？就是说我们工作中如果遇到了问题，然后去看新的论文中涉及到数学知识再去学那一方面的数学知识，涉及到经济学的算法再去了解那些方面，逐步去优化这个模型。

这个就是刚才总结的七个步骤，从数学知识开始，数学知识了解以后会学一些算法，学习算法以后，编程工具这个大家可以同时去掌握，比如说 Python，其实认认真真坐在那两天左右就基本上也会的差不多，然后如果你会 C++ 的话更好了，基本上不愁找不着工作，各个大公司都会抢着要会 C++，并且懂深度学习开发的人员，然后还有就是研读公众号和博客，那个也比较重要。

然后你自己动手训练一些案例，最后结合工作中相关领域，你选择事业方向和语音识别方向去深入到，结合你工作和行业领域去做一些东西。

给大家稍微去讲解一下 TensorFlow 编程模型，TensorFlow 是一个什么东西呢？它是一个机器学习框架，之所以说它是机器学习框架，原因是说，它不光对深度神经网络做了实现，对整个机器学习界的所有算法，都有实现。

所以你只要去调 TensorFlow 的 API 来说，机器学习的算法都实现了，不光是集中在神经网络这个领域，所以大家也有一种说法认为 TensorFlow 算是一种新的语言了，虽然它是基于 Python 的，但是 TensorFlow 本身的文档非常全面，你甚至可以说，大概非常浅薄的知道一点点 Python 的语法，你就深入到 TensorFlow 整个框架或者语言当中去学习，也是可以的。

认为它是一个机器学习或深度学习框架，并且它实现了异构设备的分布式计算，什么叫异构设备？指的是我们底层芯片可能用的是 CPU 或 GPU 以及 FPGA 这种不同的芯片。

但是对于我们开发者来说，我们对底层异构设备是透明的，所以我在开发程序的时候根本不用关心底层我用的是哪一种芯片在运算，当然图象处理 GPU 当然是更快的，但是 TensorFlow 已经帮我们封装好了，它对每一个算子每一个操作，它在 CPU 和 GPU 上都去做了实现。

还有就是 TensorFlow 对上层有一些编程接口，尤其是 C++ 语言或者 Python 语言，以及目前正在完善的 Java 和 Go 语言，都有一些编程接口，所以如果会上层语言也可以做一些上层开发。

用神经网络训练的过程就是这一些，这一些过程跟我们前面讲的机器学习训练过程也是类似的。

这就是一个 TensorFlow 整个的编程模型，如果大家去学过 TensorFlow 的话，对这个也是比较了解，具体来说，我们如何来训练一个神经网络？或者说如何来训练一个数据？得到一个模型。

实际上是说我们把一个数据X输入进来，塑性可能做一些数据预处理的操作，然后我们就进入到，神经网络是由各个层来组成的。

所以我们进入第一个层，我们叫做隐藏层，隐藏层的话，是在神经网络中最主要的是矩阵乘法运算，稍微知道一句线性数学的知识就知道了，就是 W×X+B，其中W是一个矩阵，然后 B 是一个列的矩阵。

我们需要学习的就是刚才说，模型中一堆的参数，最终那个模型中就是 W 和 B 一些参数组成的模型文件，然后我们把这个数据输入进去，经过矩阵乘法运算，然后经过 RELU 这个函数的非线性变换，这时候第一个隐藏就处理完了。

然后再接下来，这一个非常简单的神经网络只有三层，接下来输出层的话也是 W×X+B，输出层以后，就是计算我这个输入。

比如说我做一个图片分类一个操作，我计算这个输入，它对应的输出，在各种类别下的概率，比如我输入一个猫，它可能输出猫 90%，狗 5%，兔子 5%，输出的就是这样一个概率分布。

然后我们会对这个概率分布和原始类标记做一个交叉，原始的类标记就是我输入一个猫，然后这个猫的概率肯定是 100%，这个 100%和我刚才得到的做个交叉熵，原数据的分布和我们目标得到的，预测数据分布做一个交叉 熵 ，整个就完成了。

再计算梯度，就是反向传播的过程，就是为了使误差更小，反复进行神经网络训练，直到去训练一个误差比较小的 W 和 B 值，然后把 W 和 B 值存储下来以后就是一个模型，这个模型将来我们拿着他，这个模型就是一个文件，你下一次输入一张图片，我立刻就告诉你这是一个猫还是一个狗，立刻告诉你。就跟百度在最强大脑上类似，就是一个模型文件输入双胞胎，我就立刻给他检测出来，就是那种。

这是 TensorFlow 编程模型，然后 TensorFlow 本身的架构是什么样的?

TensorFlow 本身是由 TensorFlow 核心 API，分为上下两个部分，其实我们工程师大概都用上面那一部分，上面部分叫做前端部分，前端部分就是用来搭建编程模型，就是我搭建我上一个 PPT 当中显示的整个神经网络是什么样子?

负责构造计算图，因为 TensorFlow 是基于静态图模型的，实际上就是图论，我们计算机中图论的模型，所以前端系统主要是用来提供编程模型，并且来负责构造计算图，那么后端系统的话主要是来运营构造我们的计算图，并且提供运行时的环境，负责执行构造的计算图。

后端系统大家看到其实 TensorFlow 在底层实现了一些 GRPC 以及远程过程调用协议，还有远程的内存直接访问这些协议，可以不关心底层。

在数据操作层主要实现了各个算子，比如说常数的算子，矩阵操作算子，卷积的算子，非线性函数的算子，队列的算子等等，数据操作层。

然后在图计算层当中，TensorFlow 刚才说它是异构设备的分布式计算框架，分布式当中，你就理解大家做过大数据，能够实现分布式的计算，但是它也可以在本地进行计算，因为有好几种部署方式。

一方面实现了在本地可以计算图，另一方面它可以分布式计算图，分布式计算图依赖的就是远程过程调用，以及远程的内存直接访问，因为各个节点上需要直接的通讯以及数据传输。

但是这些我们其实大家都不需要关心，这是 TensorFlow 开发者需要关心的事情，然后我们关心就是 TensorFlow 核心 API 上面这些如何用 Python 这些语言去构建一个神经网络？可能在优化过程中，可能会修改一些数据操作层的东西，比如说把一些算子做融合，然后使得整个计算图的运营效率更高。

这就是 TensorFlow 技术栈，我给大家稍微说一下，其实跟上面差不多，但是如果你要是真正从事人工智能方面开发的话，这些技术栈必须得有的，比如说你需要了解硬件中 GPO 的编程，然后你需要了解 GPO，gRPC 或者是RDMA 协议， 这种网络协议在 TensorFlow 当中怎么实现的?

数值的计算层，就是这几个数值计算的库，还有就是高位的计算层，就是Eigen， 这个 用 C++ 写的数组计算的第三方库，但是这些都是开源的，还有就是比较重要的 TensorFlow Core，这里面实现了计算图的优化，以及我们神经网络最重要的前向传播和反向传播的过程都在 TensorFlow Core 当中实现的。

我的意思就是说，如果你真从事人工智能方向的工作，只要把 TensorFlow 原码下下来，TensorFlow 用两种语言写，一种 Python 语言，一种 C++ 语言 ，把它看懂，我觉得你对 TensorFlow 的架构就有一个很清楚的了解了，然后这是做本身底层优化。

但是如果你不做底层优化方向，你只是做上层应用，比如说训练一个模型，把它应用到工业界的话，你可能会关心，比如直接使用 Keras 或者 TF Slim，这些东西直接它给你现成帮你实现了业界经典的网络，以及帮你现成训练好了模型，都放在 GitHub，你可以直接把他训练的模型下载下来，当作你训练下一个模型的预训练模型去做，这个就比较简单。

还有 TensorFlow 可视化工具，TensorBoard，最重要的有两个功能，一个功能是说，它可以观测你在每一步训练迭代过程中你的准确率和损失值的一些变化，然后你知道这个模型到底是好是坏？ 然后你知道你训练过程中可能出现什么样的问题？这是第一个优点。

第二个优点，因为我们学出来的一堆参数，参数就是 W 和 B，他可以知道 W 和 B 的分布，这个分布是否平滑？平滑是好的，如果不平滑的话，就需要找一些原因，这就是可视化工具。

TensorFlow 在自然语言处理模型当中应用，首先这个就是非常简单的自然语言处理模型，RNN，叫做循环神经网络，其实这只是把循环神经网络做了一个展开，实际上单独的循环神经网络就是在 A 中做循环，一直循环下去，它是跟普通的神经网络没有什么太大的区别。

这个公式就不给大家看了，公式大概知道就行，因为我最开始看的时候，我跟大家也是，应该现在感觉一样的，就是完全看不懂，然后你多看几遍，然后再回过头来再去看就比较容易。

如果说你要入门，比如说我现在做 TensorFlow 方向，然后我面对的行业领域是做自然语言处理相关，自然语言处理当然也分为文本分类，文本挖掘等等，但是你要选择模型的时候，你可能会选择，我是用普通的 RNN，你就必须知道各个模型的优缺点在哪里?

这里不做算法，只是做工程方向，你就需要知道我在应用这个项目的时候，应该采用哪一种模型？所以就需要知道，比如说 RNN 当中模型的优点缺点，每个模型存在的问题，存在问题应该如何去解决？这是我们做工程方向应该知道的，这都完全不涉及算法和优化的层面。

比如说 RNN 当中，目前来说它存在两个问题，第一个问题就是我说的，这个梯度爆炸和梯度消失的问题。

什么叫梯度爆炸梯度消失？我们知道反向传播的过程是从后面往前传播的，所以梯度消失你就可以理解为 0.9，0.9 的 100 次方，就是非常小的一个值了，然后梯度爆炸你就可以理解为 1.1 的 100 次方，就是一个非常大的值了，所以就会导致模型不收敛。

我们当然希望它是 1 的 100 次方，最终的损失值就是一个恒定的，相对来说训练到一个局部最小值，所以梯度爆炸梯度消失我说的那两种情况，RNN 会存在这两种情况，为什么？具体的话大家可能需要知道原因就是需要看公式，但是你不想知道原因，你知道它存在的问题就可以了。

第二个问题会存在的问题是 conflict 问题，也就是我的数据输入进来以后，没有一个东西帮忙挡着，因为 RNN 的话是一个数据序列的输入进来的，我可能需要控制我上一个序列对下一个序列的影响。

而传统的 RNN 来说是控制不了的，是完全把上一个序列对它的影响完全放行或者完全屏蔽掉，它没有说是控制，放行 50%，放行 30% 等等，没有这个控制，而这个控制是我们将来可以优化的点，这个优化的点就在于，我们可以把这个控制当成一个参数去学习。

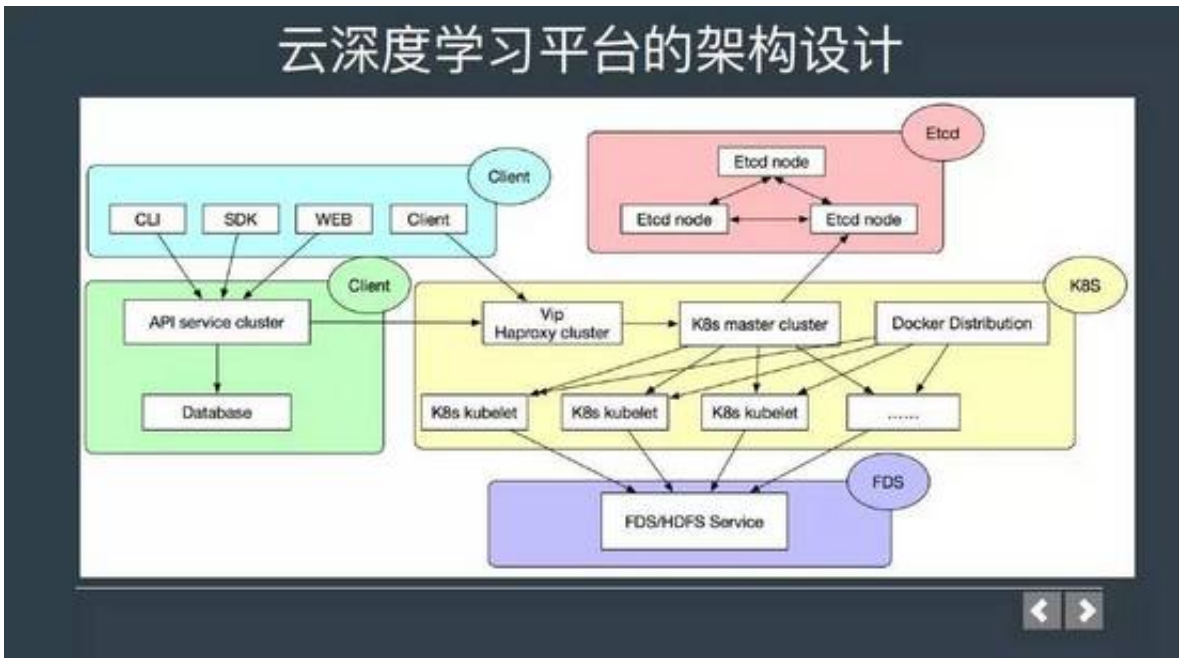
所以这是传统 RNN 存在的问题，就是说无论在输入领域，上一个层向下一个层的输入，以及下一个层向再下一个层的输出，都存在是否完全放行，或者是否完全屏蔽？然后这里面把这个冲突叫做 conflict，无法去调整，只能完全放行或屏蔽。

如果我们不理解那个理论其实也没有关系，我们直接看 RNN 在T ensorFlow 中如何实现的？那么 TensorFlow 中采用的是编程的方式，直接说每一个 RNN，RNN 它都需要实现四个函数。

如果 RNN 具备了这四个函数，那么它就相当于你就实现了 RNN 的模型了，所以每一个 RNN 都需要这几个属性，那么这几个属性，然后它需要的参数输入本时刻的输入，以及上一时刻的状态，经过 output 得到下一时刻状态。

在 TensorFlow 当中直接用一句 RNN Cells 就可以搞定了，输入的话就是这种，基本上输入的参数和输出在 TensorFlow 当中都是固定的。

针对刚才不是有两个问题吗？一个是梯度爆炸和梯度消失的问题，另一个问题是对是否完全放行和是否完全屏蔽的问题就提出了，在 1997 年提出了这个模型。



通过什么方式解决刚才那两个问题？第一个问题引入了 TEC 的架构，TEC 的架构实际上就是我们在这里面，给大家看一下，其实做了两个点，一个是在这里引入了 CEC 的架构，然后这个斜杠代表线性函数，并且它的权重为 1，然后并且还引入了两个门，这个门就是用来调节刚才说的完全放行或者完全屏蔽的问题，就是说我可以达到 30% 的放行 50% 的放行，而不是说完全 0 和 1 的这种行，这是 1997 年提出的一个架构。

又优化了一个版本，叫做标准的 LSTM，实际上引入了 forget gate，原因是说，比如说说两句话的时候，在第二句话，因为 RNN 是序列输入的，在第二句话，希望把第一句话完全忘记掉，因为 RNN 本来是一个有记忆的网络。

这时候就引入一个 forget gate，就用来控制是否记住或者是否忘记上一时刻的状态？这就是现在我们常用标准的 LSTM，左边的话就是它的架构，右边就是如何把这个负载的架构用数学来表达。

这个我研究了很久很久才知道原来这个架构是可以用数学去这样表达出来的，其实是非常简单。

大概知道理论之后，你就可以直接调用 TensorFlow 函数了，它直接就帮你实现了，同时它还有非常丰富的解释。

我总结一下 TensorFlow 下面的关于自然语言处理的一些实践小结。

最后我给大家介绍一下，我刚才说过，现在做 AI 公司有几个领域，一个就是云基础服务，就是出去可以卖服务的，还有 AI 芯片领域，就做安防或者物联网上面的 AI 芯片，第三个领域就是关于图象或语音的领域，图象你跟行业结合。

比如衣服的一些标签，以及视频当中你输入文字就能搜索到视频的某一帧，然后语音的话就是刚才说的，比如迅飞和搜狗正在做的事。

然后今天我给大家介绍，我前些日子有在做的云深度学习平台的一些架构设计。

什么叫云深度训练平台？就是希望你的数据输入进来，我们希望提供一个深度学习的基础服务，给各个业务方提供深度学习基础服务。

比如说你的业务部门有文本分类服务，只要每天告诉你，给你输入日志，然后我做个云深度学习平台的部门，我给他训练模型就行了，鉴黄部门，他把图片给我，我给他训练模型，鉴暴恐的部门，他把图片给我，我给他训练模型就行了，这个云深度学习就是做这个的，大家可以去看一下。

这是我们目前的一个架构设计，然后这个设计大家要会参考，就是谷歌的机器学习引擎去设计，然后也就是说前端我们通过 web 的方式，我们直接以拖拽的形式去组合一个神经网络，然后我在 TensorFlow 的集群容器当中就可以直接去训练这个神经网络，并且实现自动调参的功能。

比如自动调参功能是怎么实现的呢？我在输入这些数据的时候，设置一个参数范围，我在云深度学习平台当中就可以直接生成 20 套的分布式神经网络的环境，然后每一套环境都去训练这个数据，最终选择出一到两套，模型效果最优的，把它拿出来。

因为这个是集训练，参数调优，测试一体化的一个平台，所以直接训练好模型以后，就不用人工干预，自动上线，自动提供给业务方，所以业务方每天做的事就是每天把他日志扔给我，然后第二天他就拿到我的训练数据接口，就直接去使用了。

我非常重要跟大家说的，就是在深度学习界非常头疼的问题就是调参的问题，大家老说是，可能是需要很多经验才可以的，所以目前深度学习平台存在很大异议就在于自动化的调参。

而自动化调参是怎么实现的？就是说设置一个参数的最小值以及最大值，因为有多个超参数，然后把它们组合一下，生成大概 20 套或者 30 套不同的参数，然后每一套参数都给他生成一个分布式的深度学习训练环境，对同一批数据同时做训练，这时候选出一到两个模型效果最好的，把这一到两个对应的参数作为我这次训练的参数。

最后就是彩蛋，彩蛋就是调参经验，很多同学入门的话会有的问题，就是常常网络或者模型不 work 的时候，老是觉得自己调参经验少，但实际上往往其实也不是，往往可能那篇本身就不怎么 work，调参经验多的人也做不了，但是掌握一些调参经验还是很重要的，至少能排除一些在网络结构设计上的一些问题。

然后调参经验的话一般来说非常重要的就是我们其实机器学习当中也有这种概念，集成模型，其实调参经验也有这种思想在里面，就是说你可以采用不同的参数处理化模式，采用不同的数据预处理方式，然后以及有一些训练技巧，你就千万不要害怕爆显存，直接把模型怎么往大了放？怎么能够利用更多的显存就把它全部占满。

还有一些可视化的技巧，TensorFlow 本身就提供了，这个也是 TensorFlow 相比其他机器学习框架一个很大的优势，就在它可视化上面做的非常完备，然后 Ensemble 就是把前面这五种方式，把它分别结合起来。

比如说我用同样的参数，我用不同的方式，或者采用一些交叉验证的方式，选出参数最好的几组等等，最后会有一些经验的参考，专门去介绍调参经验的。

然后就是我的书了，这个就是广告，谢谢大家。

更多阅读:

李艳鹏：技术人如何修炼内功 (http://mp.weixin.qq.com/s?__biz=MzIyMTU2OTY4Nw==&mid=2247484793&idx=2&sn=3bf3af08d4cd57364d60e6f9cb6d5e5d&scene=21#wechat_redirect)

唐巧：技术人如何成长为管理者 (http://mp.weixin.qq.com/s?__biz=MzIyMTU2OTY4Nw==&mid=2247484793&idx=1&sn=103dc756da92916507b91b3eb5e75989&scene=21#wechat_redirect)

活动回顾 | 2017技术人成长交流会 (http://mp.weixin.qq.com/s?__biz=MzIyMTU2OTY4Nw==&mid=2247484743&idx=1&sn=66eb97984c3733ef4225f31eaba03539&scene=21#wechat_redirect)

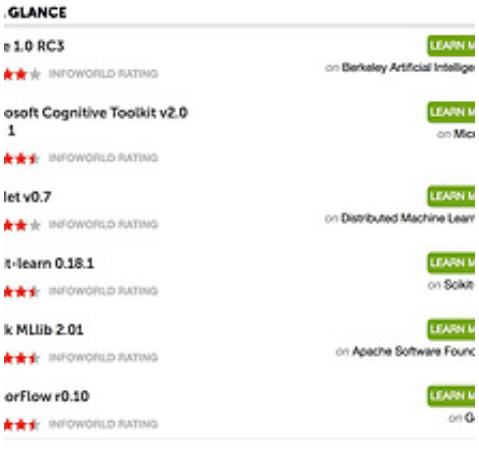
分享

☆ 收藏 ⚠ 纠错

推荐文章

- 1. 如何更好地阅读包含很多数学相关内容的机器学习论文？ (/articles/beiaqyr)
- 2. [译] 用 Google 新开源的 deeplearnJS 预测互补颜色 (/articles/BJVVBbI)
- 3. 迈向通用人工智能：星际争霸2人工智能研究环境SC2LE完全入门指南 (/articles/veQNneV)
- 4. 大规模知识图谱的构建、推理及应用 (/articles/uqYJbyu)
- 5. Andrew NG 深度学习课程笔记：梯度下降与向量化操作 (/articles/ui2QVfY)
- 6. 数据科学家必须知道的 10 个深度学习架构 (/articles/7RfyAvQ)

相关推刊

- by firecream 

(/kans/3192075353) 《tensorflow》

  (/kans/3192075353) *224*

我来评几句

请输入评论内容...

发表评论

已发表评论数(0)