方向三作品展示页面内容

**<队员信息>**

队伍：日昌晶队

队长：余咏晴 华润小径湾贝赛思国际学校（11年级）

队员：黄裕涵 成都市七中育才学校（水井坊校区）（8年级）

队员：黄钰云 深圳国际交流学院（11年级）

小程序码



**<项目介绍>**

<项目名称> 吃不胖o

<产品介绍> “吃不胖”将通过识别照片中食物的类型，来估算食物所含的卡路里，从而给用户提供最健康的食物搭配建议，让用户在家也能用现有的材料做出健康低卡的食物，帮助大家健康高效地实现减肥计划！

* 主要功能：估算食物中含有的热卡。利用用户拍照、通过图像识别技术识别出食物的种类，调取数据库中相关食物的热量数据，再利用用户输入的各食物种类的大致质量，相乘估算食物中的热量卡路里。
* 次要功能：餐点建议。通过识别用户所有的食材，配合用户喜好，为用户提供健康美味的餐点建议，保证餐式多样、健康低卡。
* 次要功能：热量统计。用户可以记录自己的饮食情况、生成成长图以统计一段时间里自己的热量摄入，起到记录、分析、促进的效果。
* 次要功能：分享饮食数据。用户可以分享自己做的低脂餐以及自己的减肥进度，起到相互激励的作用。
* 次要功能：分享原创菜式。用户可以上传自己的原创菜式，通过审核后将加入“吃不胖”官方推荐菜单，分享给其它用户。

**<技术开发>（图文形式展现）**

**项目中使用的主要技术及实现流程：**

一、采用的模型

神经网络模型先后尝试了多层神经网络、卷积神经网络、VGG、ResNet、EfficientNet、MobileNet等结构。其中，EfficientNet没有训练成功，因为看不到日志无法确定原因；同时MobileNet已取得进展，模型识别率及大小可接受，所以后面就把精力全放在MobileNet的优化上。

二、训练使用的机器

训练的机器，主要是在腾讯智能钛服务器进行，还利用了自己家电脑和kaggle来做代码调试和简单模型训练。

最初的多层神经网络和卷积神经网络是在自己家的没有GPU的电脑上进行的，印象中训练五六十个epochs就要整整跑一晚上。从VGG开始，调好脚本后就放到腾讯智能钛服务器上跑。

三、训练数据集

训练用的数据集是自己从Kaggle搜索的food11-image-dataset。它将食物细分为11个类别：

"bread面包", "dairy\_product乳制品", "dessert甜点", "egg蛋", "fried\_food油炸食品", "meat肉类", "noodles/pasta面条", "rice米饭", "seafood海鲜", "soup汤", "vegetable/fruit蔬菜水果"

整个数据集压缩包大小1.1G，下分training/validation/ evaluation三个一级子目录（文件数9866/3430/3347），每个子目录下数量不等的11个食物种类二级目录，jpg格式。每个食物种类的图像数量各不相同，以training为例，最多的Dessert/Soup二级目录下各有1500张图片，最少的Rice二级目录下有280张图片。validation/evaluation下都类似，各食物图片数量要少一些，比例大致相同。

  

知乎上看到一段话：“如果训练数据存在极度不均衡情况，那么需要做图像预处理，欠采样或过采样或补充，目的是不让模型结果偏向于数据大的类别”。正好有过一段测试数据对照（本来不是为了验证上面这段话的），从结果看是不大符合。测试模型采用的是ResNet18的100轮训练结果，从11类食品各取15张已知正确分类的图片进行预测验证，总共165张图片的测试结果：数量最多的汤，众望所归地15张全对，准确率100%；同样数量的甜品却只有7张预测正确，跟油炸食品一样准确率垫底，不到50%；米饭图片这么少，结果应该不大好看吧？它预测对了9张（60%），比总体的准确率112/165＝67.88%稍低一点，反正不是最低的。关于这个问题我思考过，首先网上看到的知识仅供参考，其次它的结论往往需要有一定前提（例如我们的数据是否该归入“训练数据极度不均衡”），而且还有模型非最新或训练不足、测试样本量不足等问题。总之，这是学习和实践过程中的一个存疑吧。

人工粗略地浏览检查过各目录，大部分（百分之九十多）图片是正方形的（长宽像素数相同），少部分（3－5%吧）是长宽不等的长方形，少量最夸张的见过图形长宽比接近2：1比例的（例如512\*288）。看到的最窄的（长或宽）像素数都有280+像素。这部分数据特征其实一开始并没有（其实是来不及）刻意关注，这是后期思考如何提高模型准确率时才注意的。因为提高训练图像质量也是提高准确率的一个重要手段。复习了下直播课老师说过的内容，mobilenet模型经过各种技能加持之后，可达90%+，可我们只有70%+，差得很远啊？（这个问题后面还会讨论到。）那么其中一个担心就是训练图像了。例如像逐张图像检查修正这类高技术的体力活是来不及做了，首先把怀疑的眼光投向了resize……我们训练数据时指定了长\*宽参数，机器会作resize，那么这个resize上究竟是剪切还是按比例缩放？网搜了一下应该是后者，这也符合我们的思考判断——如果是剪切，需要机器有一定的智能来判断对图像的取舍，但这恰恰是我们希望通过AI模型训练，让机器“学会”的技能吧，它不大可能“提前”就掌握这个技能吧。好，既然是缩放，那就存在一个形变的问题，所以需要检查一下图像，目测九成以上都是（或接近）正方形，那就差不多了吧，这个因素应该影响不大。另一个因素是resize时最好是原图基础上缩小——而不是放大，目测——也只能先靠目测吧——最小的长/宽像素值也有接近300，所以resize成224\*224，应该问题不大。

输入影像方面，可改进方案而没来得及实施的，至少可在数量较多食物种类（如甜品和汤）下，把一些长宽比偏大的图像删除吧。

四、模型训练过程

1、MLP多层感知神经网络、CNN卷积神经网络、VGG神经网络

该阶段处于环境搭建与熟悉代码的磨合期。其中MLP与CNN尚可在自家PC上训练，训练60个epochs花十个小时的样子。VGG训练3轮也需要花8个多小时，只能在腾讯智能钛服务器上训练，但初期可能是使用方式选择不当，看不到日志。日志问题是在ResNet后期、MobileNet之后解决的。

中间碰到的问题是，CNN训练的后30多个epochs，training accuracy逐渐上升至95%+，而val\_accuracy却在0.3x后徘徊不升反有微降。网查资料并结合思考，这应该是一种过拟合现象。模型中加入Dropout()层后现象缓解，即val\_accuracy不再呈下降趋势，但升到一定数值（0.5）后小范围波动也不再上升。

相应的VGG模型在智能钛上能够正常跑出结果，因看不到日志反馈，无法得知训练过程中training accuracy和val\_accuracy数据。估计它的表现应该比CNN强一些。

上述3种模型，自行调用evaluation中图像来测试时结果都不如意。h5模型大小都在200M以上，都偏大不适宜手机小程序使用。必须往前研究。

2、ResNet

先试通了ResNet18，并跟随老师的授课笔记和代码，初步试用了图像增强技术，包括水平翻转、上下平移或左右平移等。初步学习使用动态learning rate。

学习率采用初始值0.001（或者也会尝试0.01/0.0001等），每30个epoch右移一个小数点的方式。

h5模型大小降至135M。仍然偏大。测试识别率达68%，初步可用。

根据思考，将老师示例代码中的图像增强技术，由val\_data\_gen移至train\_data\_gen。因为图像增强应该是用在training训练环节，相当于增加训练图像数的效果；而应用于validation验证环节，应该相对没那么重要。

回忆老师讲的通过减少降采样次数（即某些环节卷积时，原模型stripe=2老师改为stripe=1），提升模型识别率。考虑到老师讲课用的训练数据是采用32\*32的数字，故需减少降采样以避免图像太早降至1\*1，而我们的模型为224\*224，是ImageNet标准输入，可能没必要这么做。因此根据老师的课件和网上的标准图，找出老师的两次减降采样代码，恢复了标准ResNet18模型。

尝试过自行学习ResNet50结构并在老师提供的ResNet18代码基础上尝试改写。未成功。

同期还尝试修改试跑了EfficientNet的代码，未能成功生成模型。

随后因MobileNet调试成功，集中精力研发MobileNet

3、MobileNet

此模型调通后，陆续以10epochs为单位续训，根据日志判断，其最优val\_accuracy出现在epoch=55，为72.52%，但当时为每10epochs续训、保存一次，故未保存下该模型。后续有机会保存下来的模型为epoch=218的72.05%。

MobileNet模型大小约为ResNet模型的1/9左右。h5模型为15M，转换后6.8M，在手机小程序端加载起来相对快一点。

因为对mobilenet其他参数的研究，一开始是在持续地分段提交，陆续训练了300个epochs，希望找到一个较好的val\_accuracy值。对其他参数的探究，是这几天才开始的。但是，智能钛服务器的响应情况似乎有了变化，这几天发现val\_accuracy普遍下降了，包括重跑原300次训练的脚本，连重上65%都难。说明一下这个重跑，严格意义上来说并不是把原来的脚本一字不变地重新跑一遍：原来是调通后，每10次epochs提交一个脚本，一个半小时后再搜集日志并提交下一脚本的，在睡觉时间或长期上课不便上网时就一次性提交个训练50/60次的脚本；现在是改为一次性提交300个epochs的训练脚本，中间自动保存。因为原来的分次脚本很有规律，除epoch外其他参数都没变动过，所以现在提交的一次性300个epochs，自我理解是等价的python脚本。但是自己作为初学者，很可能有忽略的地方，所以首先不能排除的是自己的原因（虽然自己没能找到这个原因）。其次，在这段时间的学习实践中，认识到了一个大的系统、像腾讯智能钛服务器云端这样大系统是有很多人在辛苦地协同工作、会有很多因素在影响其运行的，不排除某几个因素综合作用结果，恰好影响到了这个AI训练的环境，例如随机数产生序列恰好产生了某种变化（想得是不是有点多了，嘿嘿）。总之归结为一句话：最近几天发现，提交的mobilenet训练脚本，其val\_accuracy（这是我们找到的主要评价指标）普遍下降，原因不明，导致某些新参数的试用效果无法评价。

所以，后面提到的几点，都是一个思路和可能手段。要评价其是否有用，单看一轮两轮跑脚本的结果，应该是存在很大的偶然性的。可能需要更长的时间、每次固定其他因素，专门研究对比某一项要素变化的影响，结论才比较可靠。

为了取得历史上曾达到过的val\_accuracy峰值（epoch=55时72.52%），曾尝试从保存的epoch=50模型起步，重新跑51－60，每一步都保存模型。因为训练参数的图像增强本身是具有随机性的，所以实际运行的过程中各参数肯定不同。曾尝试过两个脚本，一个是日志方式完全不改（参见后面简化日志的说明），另一个是考虑到日志写入理论上对模型训练无关而我们只关心val\_accuracy，故日志作了简化。结果是日志不改的版本的高者达到过71.61%、71.55%、71.41%等值，日志简化版高者有过两次71.14%。而此前（指分多次运行的1－300#epoch）多次达到过71.8%以上的，这两轮新的51-60#epoch都没有达到。这个测试是今天才想起，临时做的，可能对于上面一段说到的“小小背景情况”有一定的辅证作用。

这几天，还尝试过将val\_data\_gen也加上图像增强参数，结果现象是val\_accuracy下降（峰值60多）。不确定是最近这几天val\_accuracy普遍下降的原因还是更严格地验证导致的下降。据思考，validation环节与training是独立使用不同图片，故这一改变应该对提升训练效果本身无影响，反而影响评价的一致标准。所以在其他尝试中，没再于val\_data\_gen上应用图像增强参数。

在train环节的图像增强增加 rotation\_range=20 参数，意在使图像随机旋转正负20度角。效果上同样是val\_accuracy下降，峰值60多。

与前类似，直播课老师还是使用了减少降采样的代码（共计3处）。我们尝试了额外增加1处减降、恢复该3处减降以变回标准模型，这两个方向的尝试，想作对比。结果还是val\_accuracy在下降，峰值还是60多点。

还尝试通简化日志，通过设置环境变量TF\_CPP\_MIN\_LOG\_LEVEL=3以及设定fit\_generator()参数verbose=2，只显示我们关心的val\_accuracy，效果不错，日志简洁多了。

最后，fit\_generator()中的参数workers=n，允许最大线程数，适当设置可提升训练速度。但逐步尝试着提高该参数过程中，有一次workers=8导致无日志（应该是服务器上其他因素综合导致的，稍后重新运行该脚本未改workers参数又能看到日志了，此处仅如实记录测试现象）。

五、当前技术实现存在哪些问题：

有些问题在讲数据集和模型训练的环节中已有述及，不再赘述。提一个前文没怎么说到的问题：准确率。

直播课老师说到mobilenet的准确率时，有这样一张图：



按照我听课印象（不知有没有记错），标准的mobilenet已可实现76.69%的识别准确率，而加上“减少降采样”和“数据增广”两项技术后，可实现惊人的92.10%准确率！我们这里训练出来的food11的mobilenet模型，数据增广是已包含进去了，减少降采样方面因为是在老师代码基础上修改，也被动带入了3处减降措施。所以我们的mobilenet模型是应该对标那个92%的，但实际只有72%。我想，直播课老师用的是cifar10的数据集，识别的是手写数字，相对于现在识别食物类别，难度应该是不同的，所以识别准确率下降应该也在情理之中。

**<成长收获>**

在整个未来营的的学习实践过程中，有哪些心得体会，做经验总结。

一、关于用户数量的影响

在后期的模型训练中，很多摸索和参数调整，效果只是提高几个百分点或零点几个百分点的准确率。在本次演示中，提高这百分之几的准确率可能影响不大。然而，像腾讯这类用户基数巨大（可能数以亿计）的很多项目中，这千分之几、万分之几的差距都会有巨大影响。所以说，有些改进手段，是否值得作这个努力，需要看它是放在什么环境下，用户基数和对准确率要求是如何。

二、关于探索未知

这次未来营的学习，收获很大。老师已经为我们比较系统地整理了数学、人工智能方面的知识体系，为我们这些原来的AI零基础的中学生入了门。虽然学习过程中困惑很多，但收获更多，它为我们进一步的学习打下了基础、了解了一个体系。我们以后有机会系统地学习相关知识的时候，就更知道学习的方向。

虽然学习过程中，有很多未解之谜。是的，有些困惑会让人迷茫，但有些困惑却能让人懂得在接触新事物时应专注于哪些方面。我想腾讯星火营的学习和实践中的困惑，它属于后者。思考越多问题往往也会越多，关键是搭好体系，就能把碎片化的东西整合成有效的知识。我们思考过程中的困惑，就好比一个神经元伸出的更多树突，它会增强神经元的感知能力。当学习过程中发现某个新知恰好解答了某个旧惑时，我应该能比很多人更深地理解这个知识点，也会更明白和珍惜这些知识对我的价值。