机器学习纳米学位

算式识别 金阳 Udacity

2018年12月16日

I. 问题的定义

简介

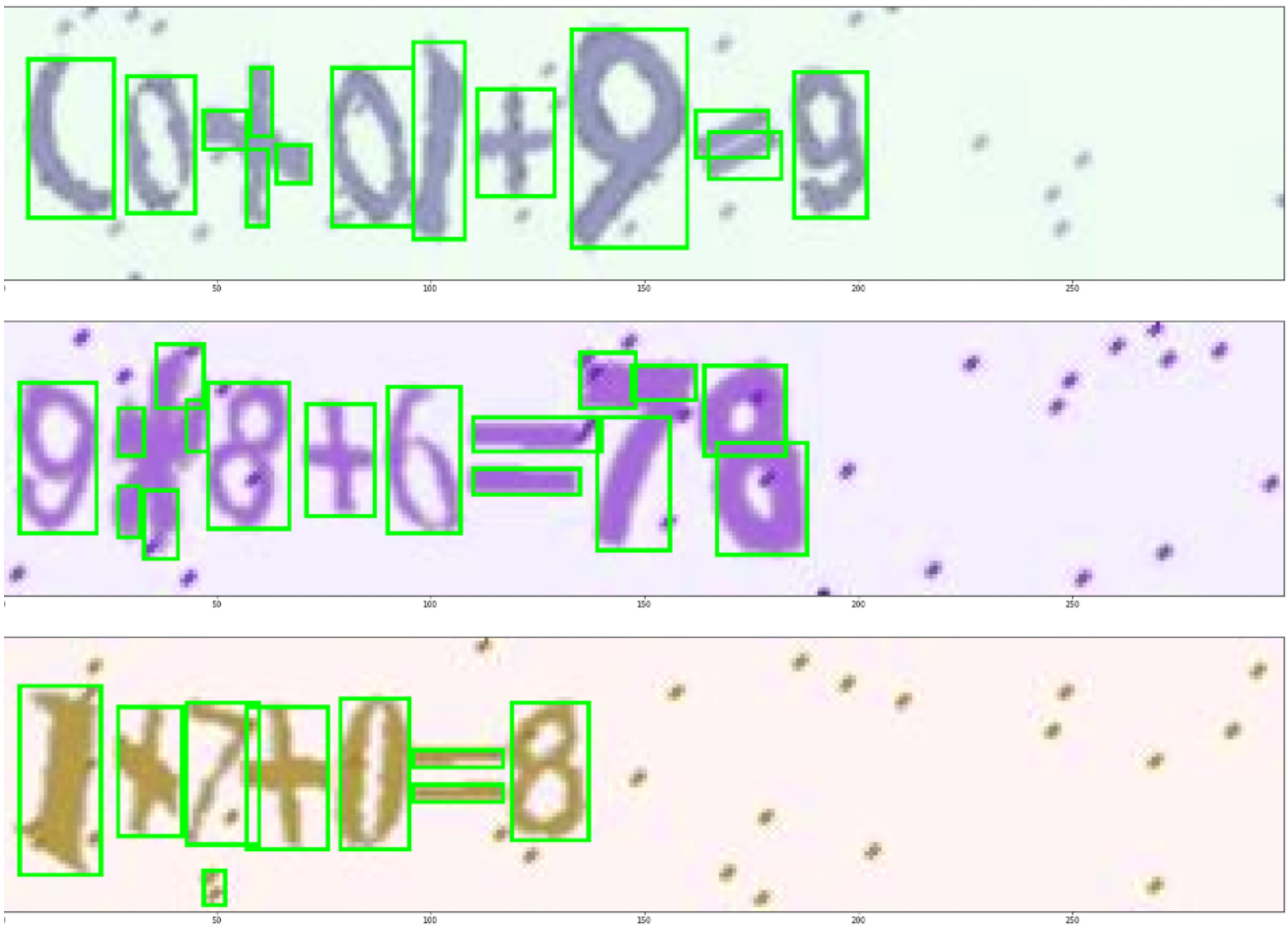
图像中的序列识别一直是计算机视觉里的一个热点。该项目是识别一张图片中的算式，就是图片序列识别的子集。该项目通过一个端到端的深度神经网络实现序列识别，在测试集上的正确率超过99%。

项目概述

使用深度学习识别一张图片中的算式。

由于卷积神经网络（CNN）的快速发展，卷积神经网络在处理图像分类问题上效果非常好。但是该项目是识别图像中的序列，图像中有不定数目的对象需要识别，所以不能直接套用卷积神经网络模型。

我开始决定先识别出字符的位置，切分出单个字符然后使用卷积神经网路模型处理。由于数据集是没有字符位置标注的，所以不能使用深度模型来做识别，我选择了使用传统机器学习中的轮廓检测来做，但是效果不理想，不能很准确的识别出字符位置。所以这个方案不成立。



然后我想到，识别序列是递归神经网络（RNN）擅长处理的问题，所以我决定结合CNN和RNN，使用一个端到端的模型来完成项目。

这是一个图片识别问题，所以需要用到卷积神经网络（CNN），并且需要对图片数据做一些预处理 算式图片中出现的长度是不定长的，需要用到递归神经网络（RNN）得到计算结果。 我决定使用卷积神经网络提取出特征之后，输入到递归神经网络中，识别出其中的算式。

评价指标

算式图片正确率=识别正确的算式数量/算式的总数

当算式图片识别的每一个字符都正确时，该算式为识别正确。

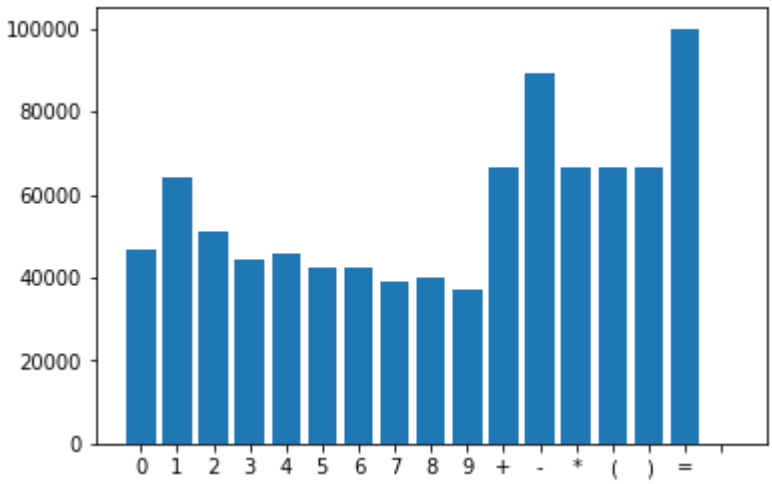
II. 分析

数据的探索

- 此数据集包含10万张图片，每张图里面都有一个算式。
  - 每个算式可能包含 `+-*` 三种运算符， 可能包含一对括号，可能包含0-9中的几个数字，以及每个算式包含一个等号。所以一共出现的字符总数是16种。
  - 每个字符都可能旋转。
  - 图片大小统一是300*64。
  - 图片字体是各种颜色的，背景也是各种颜色的，但是背景都是浅色（接近白色）
  - 图片中有一些噪点。

探索性可视化

统计标签中各个符号的数量，画柱形图，发现分布均匀。



算法和技术

项目中主要使用keras框架。使用keras框架能快速搭建模型，利用GPU计算优势，快速迭代模型。

基准模型

一个类似的项目，音符图片的序列识别论文中，An End-to-End Trainable Neural Network for Image-based Sequence Recognition and Its Application to Scene Text Recognition，模型在测试集上的正确率有84%，考虑到该项目识别的内容更加简单，目标设定在正确率90%

III. 方法

数据预处理

字符识别中颜色影响不大，首先把图像从RGB图转化成灰度图，可以减少图像层数为一层，大幅度减低计算量。图像的形状都是统一的，不需要处理。然后对图片进行归一化，使之均值为0，方差为一，方便后续计算。

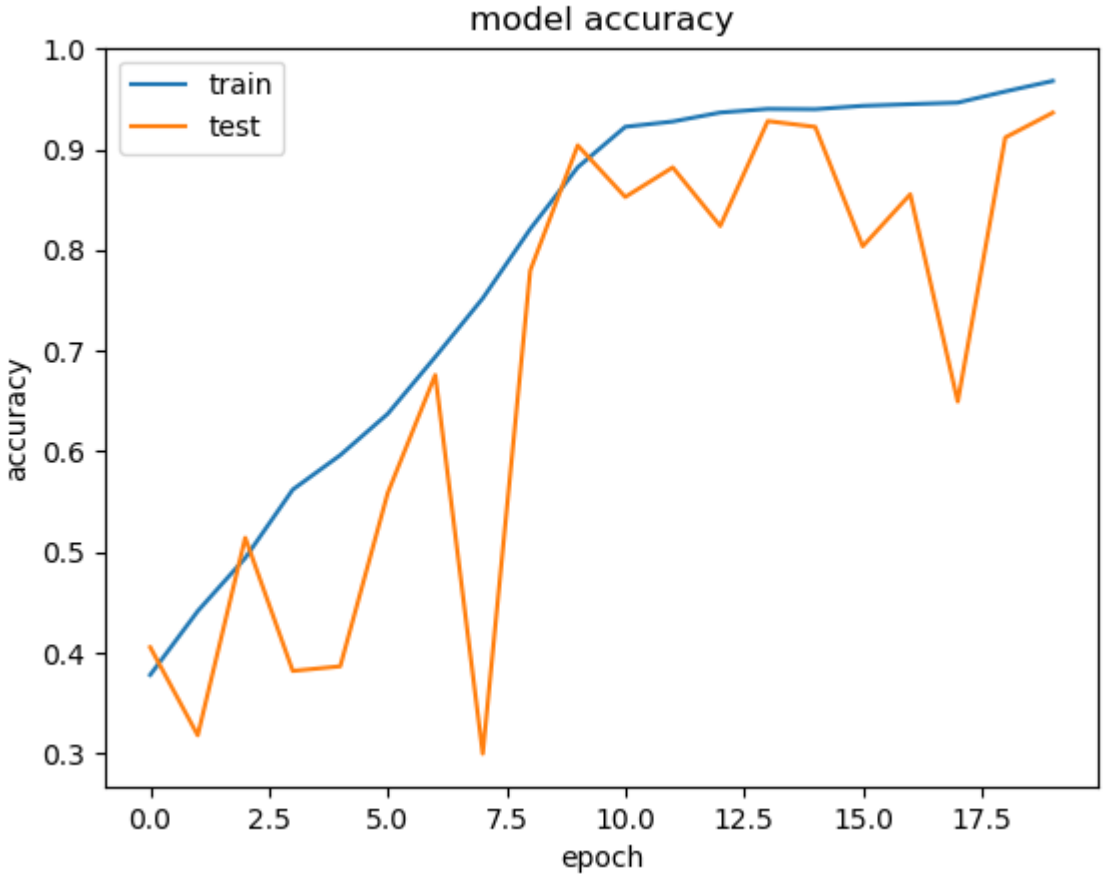
由于数据中最大长度为11，可能出现的字符为17种：'0','1','2','3','4','5','6','7','8','9','+','-','','(',')','=',''。*所以把标记（label）内容处理成11*17的one-hot形式，方便后续计算。

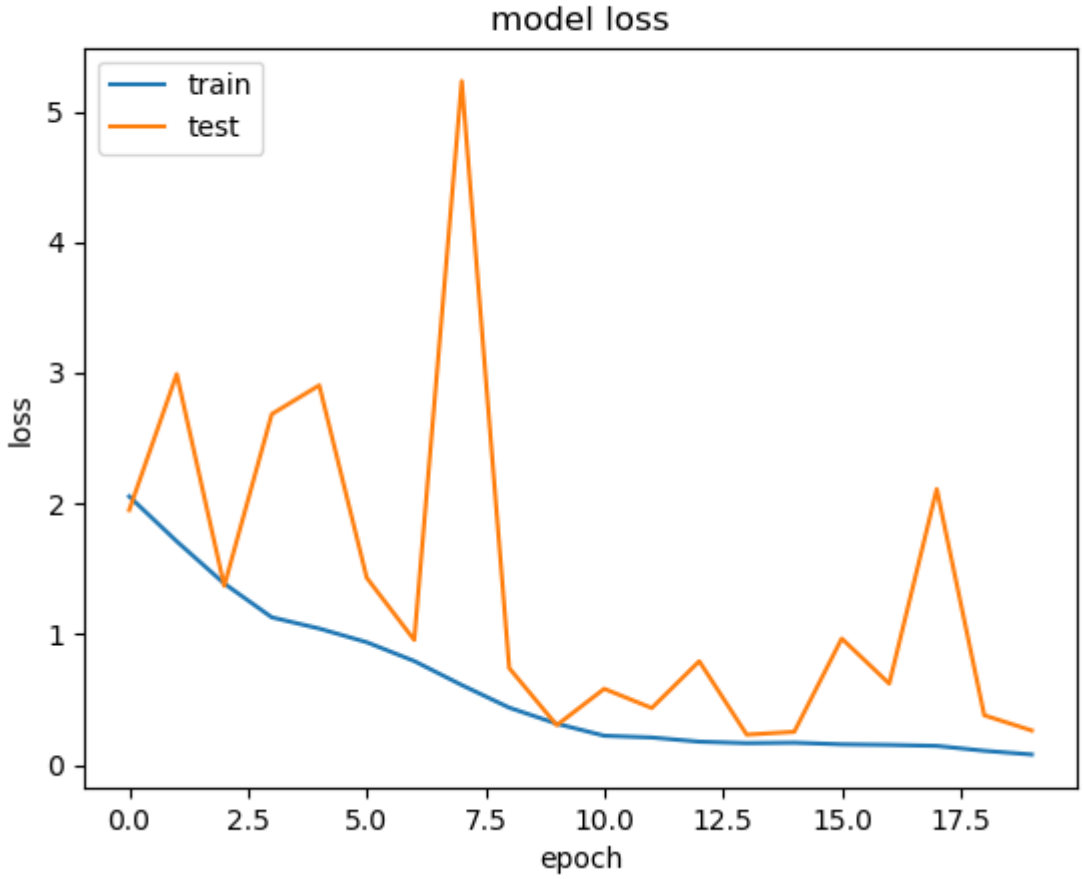
然后把图像和标记内容划分为训练集，验证集和测试集。

执行过程

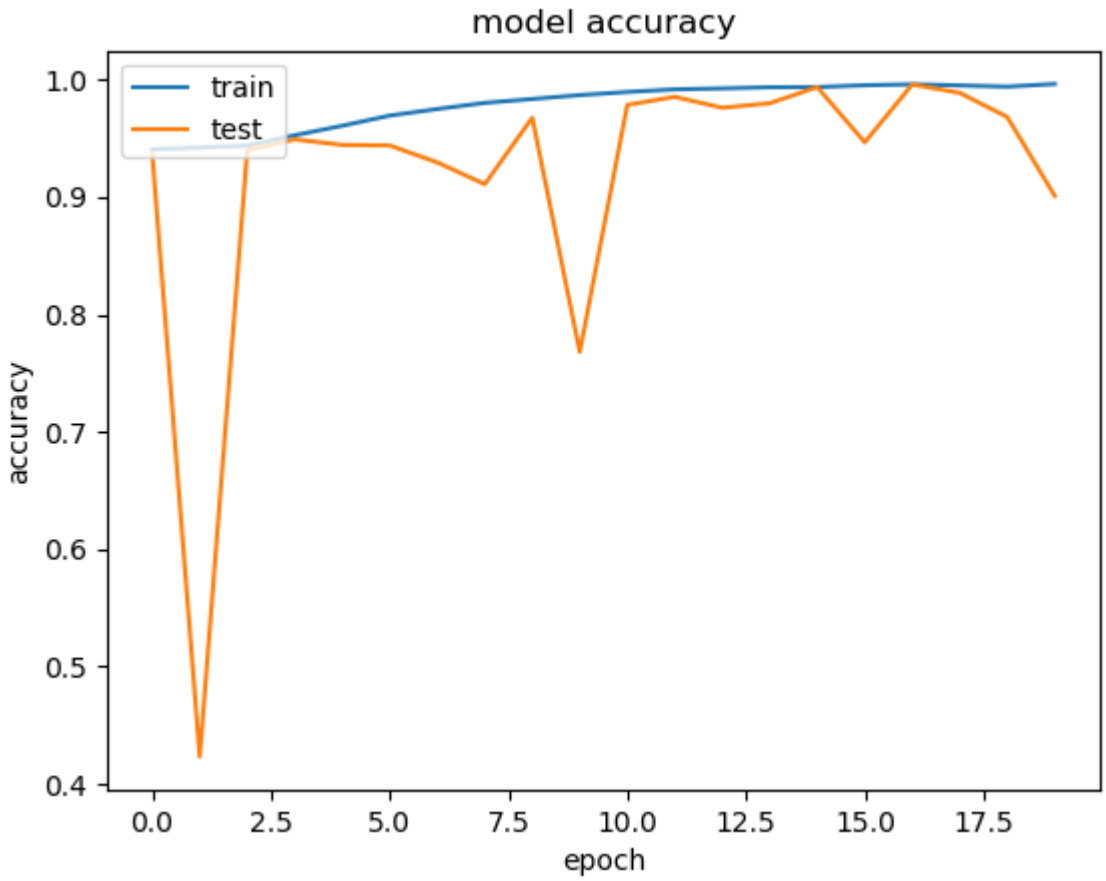
执行过程，把数据输入模型，执行20个批次，选择保存在验证集在误差（loss）上最小的模型，观察随批次误差和正确率的情况。

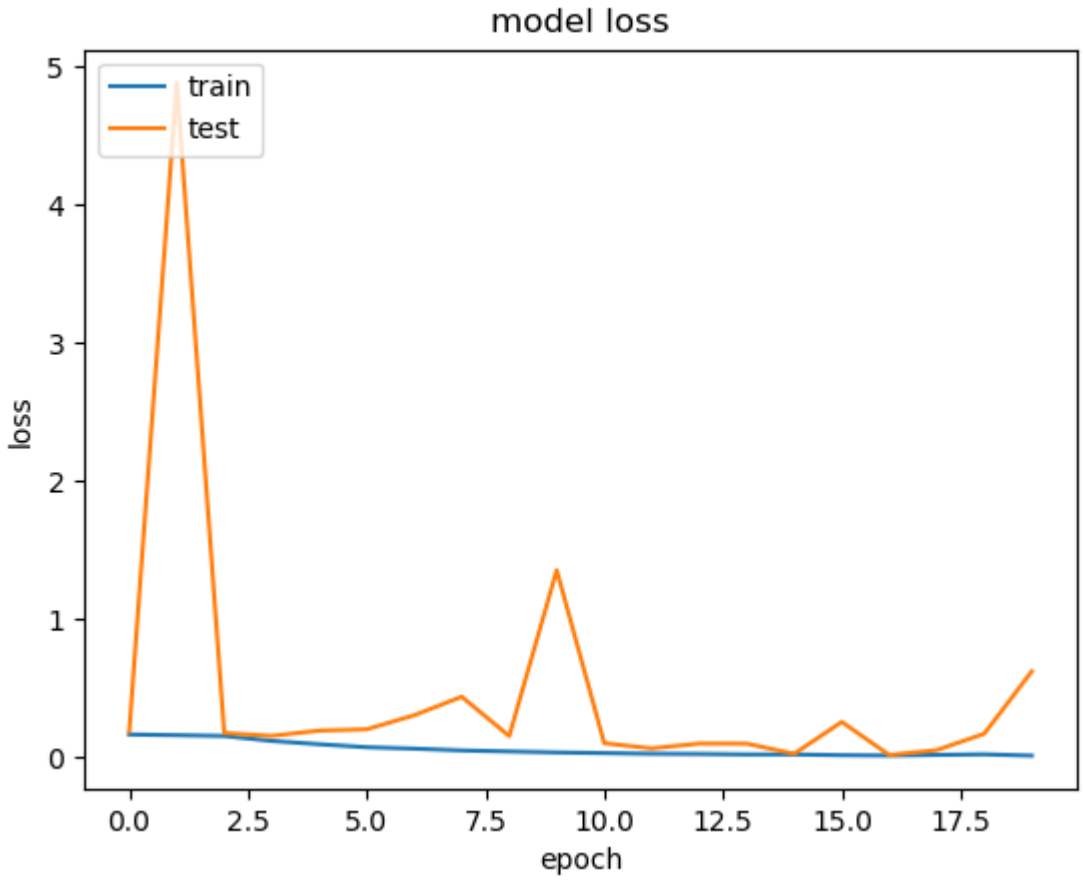
观察到误差基本稳定下降，正确率上升知道平缓。验证集上的表现与测试集表现相对贴合。


model accuracy
train
test
accuracy
epoch
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5


model loss
train
test
loss
epoch
5
4
3
2
1
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

由于误差还有下降空间，决定在训练20次，观察到正确率和误差曲线逐渐平缓，验证集上正确率（字符）最高时达到99.6%。





正确率和误差曲线逐渐平缓，将学习率改成0.001，再训练了40个批次，在验证集上正确率（字符）达到99.89%，达到预期目标，训练结束。

执行过程中遇到的问题是：

- 内存溢出，决定不适用全部数据，使用60000张图像做训练集和验证集。
- 同样由于内存溢出，切分训练集验证集的时候，不使用train_test_split，直接把前面55000张图片做训练集，后5000张做验证集。

完善

因为一个算式里只要有一个字符没识别对，这个算式就算错误，所以字符识别错误率会比算式识别率低接近10倍。

开始使用比例为0.2的Dropout，训练80个批次，但是验证集误差下不来。

后来去除了Dropout，发现误差下来了。

Dropout:	**0.2**	**无**
训练集误差:	0.0003	0.00002
验证集误差:	1.7	0.00632
训练集正确率（字符）:	0.9999	0.9999
验证集正确率（字符）:	0.9981	0.9989
验证集正确率（算式）:	0.9868	0.9912

发现dropout是需要考虑到具体问题的，需要的时候才加。

IV. 结果

模型的评价与验证

模型由CNN和RNN两个部分组成:

CNN部分:

有5个模块组成，前四个模块的结构类似，以第一个为例，组成为：两个卷积核大小为3*3的卷积层，一个标准正则化层，一个relu激活层。然后每个模块卷积层的卷积核的数量。

最后一个模块在两个卷积层后，通过全局平均池化层，把输出变成一维（1600），然后再把输出的形状调整成（11*100）后，为输入RNN部分做准备。

RNN部分:

每个模块为一个LSTM层，如此4个模块后，加上一个全连接层，一个*Softmax激活层，最终输出的形状是（11 * 17）

我先后尝试两次训练，最终在测试集上都到超过99.5的正确率，我觉得模型的鲁棒性不错，比较稳定。

合理性分析

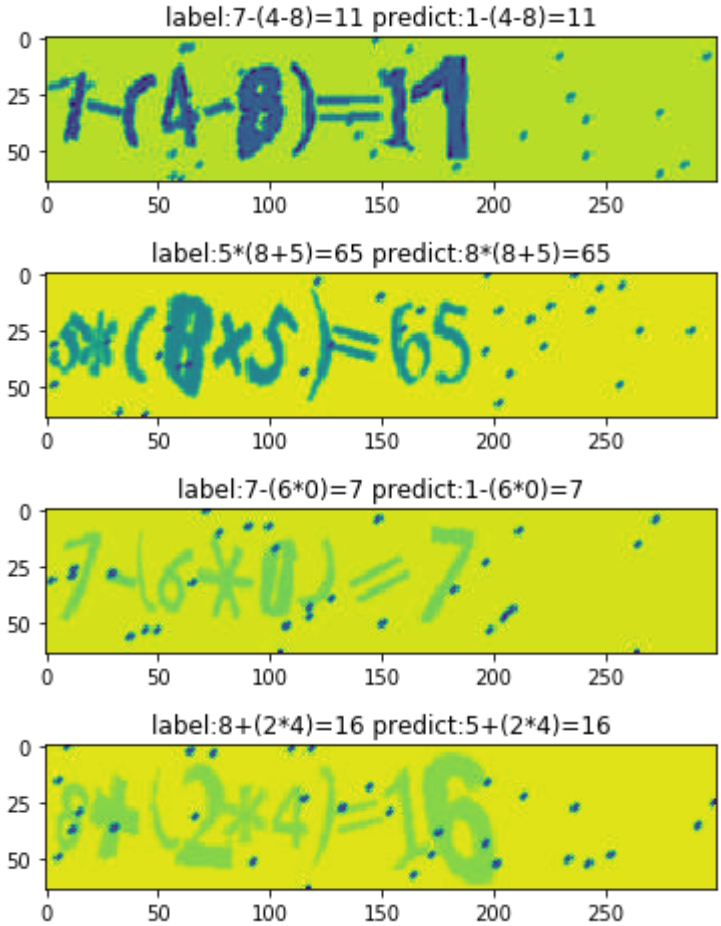
验证集正确率（算式）= 0.9912

模型在测试集上的正确率达到了设定的目标。

V. 项目结论

结果可视化

查看显示错误的图片，发现这些字符还是比较容易被人正确识别的，所以该算法还有提升的空间。



对项目的思考

结合CNN和RNN的端到端模型效果很好，数据图片中的噪点以及符号一定程度的旋转，不需要手动的处理，也不会影响到结果，模型能从这些干扰中找到正确的结果，所以深度学习还是很强大的，在计算机视觉项目中，可以减少很多人工对图片的预处理。

比较困难的地方是收到机器内存的限制，无法使用全部图片，所以机器学习内存很重要。

我觉得这个模型在正确率和易用性上符合我的期待。对于其他通用场景的问题，我觉得在对处理车牌识别，或者幼儿算术题识别，街道文字识别等领域有一定的交界相通的地方。

需要作出的改进

开始设计的模型效果就挺好，我觉得可以尝试的是，使用更简单，层数更少的模型，这样能在保持正确率的情况下，加快计算的速度，节约计算资源。

使用更大内存的机器训练，就可以使用更多数据了，这样子泛化性能会更好。也可以用上图片生成器了。

引用

Baoguang Shi, Xiang Bai，Cong Yao（2015）An End-to-End Trainable Neural Network for Image-based Sequence Recognition and Its Application to Scene Text Recognition