

work@m7-model-gpu01 CTC使用速成手册

韩景涛
22/02/18

概述

CTC模型训练和预测的主文件在根目录下CTC_lower和CTC_upper. 两个文件里内容几乎完全相同, 除了parameter.py文件的内容以外, 设两个文件夹是因为服务器有两张显卡, 故可以同时训练两个不同的模型。

生成数据放在根目录下 ~/data/toy_data_original/ 里面有两个文件夹, lower代表小写金额数据, upper代表大写金额数据, 数据对应的label文件也在这里

总之, 基本操作的流程是先修改parameter.py里的参数值, 再运行 python main_ctc.py 来执行任务

编程环境

用 source activate py35 激活配置好的python环境, 我安装了 tensorflow 1.4.1 和 keras 2.0.9

如需关闭py35环境, 用 source deactivate py35

建议使用tmux开新窗口后, 在该窗口内运行程序, 这样即便退出服务器后, 程序仍会运行。

模型训练的步骤

1. 图像预处理

修改parameter.py, 将mode改为 mode='preprocessing'

将original_folder_path 设为原始图片所在的文件夹地址, 如此处小写金额应设为

original_folder_path = '/home/work/data/toy_data_original/lower/'

将training_folder_path 设为预处理过后的图片保存的地址, 如此处经过预处理的小写金额图片可以放在 training_folder_path = '/home/work/data/toy_data_padding/lower/'

将height和width设成希望图片预处理过后能达到的分辨率, 根据之前的经验, 建议对小写金额设成 height=400, width=48, 对大写金额设成height=500, width=70

其他参数随便设值, 不影响预处理环节。

保存parameter.py后, 运行 python main_ctc.py

然后程序会在original_folder_path里读取原始图片, 进行预处理, 把图片放缩至 (height, width) 的分辨率, 再将新的图片保存到training_folder_path里, 后面的模型训练预测, 都是针对 training_folder_path内的预处理过的图片进行的。

2. 模型训练

修改parameter.py

mode = 'train'

model_type = 'model00', 其他几个model适用于其他场景, 对金额场景建议使用model00, 如果想自己搭模型, 则选择'model_custem', 再在model.py文件里搭自定义模型

optimizer_type = 'adadelta' 或 'adam'

is_pretrain_flag = False, 这里false代表不使用与训练模型, true代表使用, 现在暂时没有特别好的预处理模型, 建议一直设成false

chars_file_path = './chars_lower.csv', 字符集文件所在位置

label_file_path = '/home/work/data/toy_data_original/label_lower.csv', 数据对应的label文件位置

gpu_id = '0', 我们两张卡, 一个实验占用一张卡, 所以gpu_id在'0'和'1'中选一个

其他几个参数改的收益不大, 不建议一开始就改动

保存后，运行 `python main_ctc.py`，会进入模型训练的画面，每个epoch结束后都会对validation set进行一次预测评估，得到val acc和val loss，程序会自动保存历史上val acc和val loss较好的模型参数，模型会保存在logs里面。

如果很快val acc就非常高，可以按ctrl+c强制结束程序，节省时间

3. 模型泛化性测试

`mode='infer_with_labels'`

`model_for_inference = 'CTC_model.h5'` 用来在testing集里测试模型的能力，这里用来测试的模型的名字可以选取自logs/里的任何一个模型

其他参数保持和第2步中的参数一致

保存后，执行 `python main_ctc.py`

会计算出模型在testing set上预测的acc，这里的test acc应该接近val acc

模型预测的步骤

到这里我们已经有了一个不错的训练好的模型，比如叫做CTC_model.h5。经常行方会再给我们提供一个不带label的最终验证用的数据集。我们希望用CTC_model.h5在这个数据集上进行预测，并将结果保存在prediction.csv这个文件中。依旧打开parameter.py

`mode = 'infer_without_labels'`

`model_for_inference = 'CTC_model.h5'`

`testing_folder_path = 'the_path_to_the_given_database/'`

其他参数不用变

保存后，运行 `python main_ctc.py`

会预测并生成一个predicton.csv, 这个文件和main_ctc.py在同一个文件夹内

试玩版任务

在work里，我提供了两个数据集：

大写金额数据集： `/home/work/data/toy_data_original/upper/`

小写金额数据集： `/home/work/data/toy_data_original/lower/`

参考 **模型训练的步骤** 章节里的3个步骤，对上述两个数据集，分别训练出CTC识别模型。

大写金额识别模型的val acc and test acc baseline: 0.95

小写金额识别模型的val acc and test acc baseline: 0.92

出了任何毛病，请随时骚扰韩老师: hanfeng@4paradigm.com