

# NJU\_NLP\_SummerCamp\_2019\_report\_week4

2019.07.22-2019.07.28

@author Eric ZHU

## 本周学习内容

1. 论文复现: *Show and Tell: A Neural Image Caption Generator*, 论文[链接](#) (自己实现了部分 + 最终还是主要用了助教提供的代码, 自己写的接口和助教提供的有点差别, 导致数据读取存在一定问题)

## 部分知识点

1. 集束搜索: Beam Search

In computer science, **beam search** is a heuristic search algorithm that explores a graph by expanding the most promising node in a limited set.

集束搜索使用广度优先策略建立搜索树。在树的每一层按照代价对节点进行排序, 然后留下预先确定个数 (Beam Width-集束宽度) 的节点, 在下一层仅拓展这些节点, 剪去其他的节点。

在Beam Width为1时, 集束搜索退化为贪婪算法

在Beam Width为正无穷时, 集束搜索退化为宽度优先搜索

2. Python库 tqdm: Tqdm 是 Python 进度条库, 可以在 Python 长循环中添加一个进度提示信息用法:

```
tqdm(iterator)
```

使用方法: 参考[此文章](#)

3. 关于模型搭建与训练的技巧细节: 见个人代码库内代码注释部分。因本项目脚本数较多, 为避免赘余, 暂不在此处贴出。

## 存在问题

因本周不在学校, 暂时未能完成模型的训练与测试过程, 记录测试方案如下:

1. Windows环境下直接运行: Pycocotools没有windows支持, 安装较为麻烦, 经尝试后放弃。
2. Kaggle: 对多文件脚本的支持不佳, 且难以长时间持续训练, 加载保存点, 经尝试后放弃。
3. Google Colab: 数据集太大, 难以导入 (Google Drive), 经尝试后放弃。
4. 因身边没有易用的Linux机器, 采用Win10的Ubuntu子系统: 成功, 脚本可以正常运行, 但受制于笔记本性能, 训练速度过慢 (一个Epoch需要数天), 只能暂缓训练, 下周一回校后开始。

## 附件

1. 7.8-7.14\_Show&tell 实验记录 (Draft) (见下页)

# Show and Tell : Image Caption based on NIC

时间：2019-07-22

本项目是对 [Show and Tell: A Neural Image Caption Generator](#) 论文的复现

以下代码参考了 [此代码库](#) 与 [此代码库](#)

## 文件结构

名称	描述
temp\	自己实现的部分功能脚本 // TODO 优化
log\	训练日志与模型的保存文件夹
utils\	部分辅助脚本的保存文件夹
pic\	文档所需的部分图片的保存文件夹
process.py	COCO数据集的预处理脚本
data_loader.py	COCO数据集的Dataset定义脚本
model.py	模型的定义脚本
train_nic.py	模型的训练脚本

## 数据集

本项目采用的数据集为 [Common Objects in Context](#), 具体信息如下：

- Training: 2014 Contest Train images [83K images/13GB]
- Validation: 2014 Contest Val images [41K images/6GB]
- Test: 2014 Contest Test images [41K images/6GB]

## 运行环境

使用的运行环境如下：

PASS // TODO

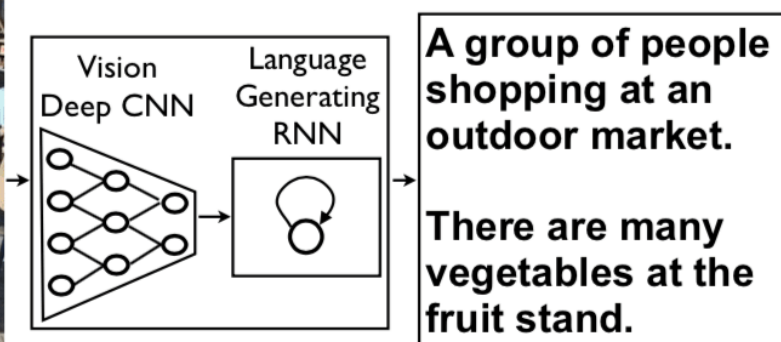
训练机配置如下：

PASS // TODO

## 模型结构

NIC结构由两个模型构成：Encoder与Decoder.

- Encoder: 卷积神经网络, 采用经过在 ImageNet 上预训练的 ResNet152 模型 (`torchvision.models.resnet152`) 并进行微调 (*fine tuning*), 目的是提取图片特征, 创建对图片特征进行语义描述的定长向量 (Feature Vector)
- Decoder: 循环神经网络, 采用LSTM/GRU作为结构单元, 目的是在 Feature Vector 的基础上生成对图片的自然语言描述文本 (Caption)



以上图片来源于[此处](#)

## 模型训练与结果

// TODO