Real-Time Video Captioning with Yolo

小组成员：张迅 1600017704 倪临赟 1600017719 李思澄 1600017852

1 引言

在本次项目中，我们先根据原论文实现了YOLOv2，并且将其运用到image-captioning中，得到一个可以实时生成对于场景和物体描述字幕的模型RTVCY(Real-Time Video Captioning with Yolo)。我们在提交的文件中有相应的视频demo，在本文中，我们会先介绍有关YOLO实现的部分，接下来为对image-captioning的改进部分，最后有相应的图片测试数据。

2 You only look once

2.1 论文概述

YOLO是Joseph Redmon和Ali Farhadi等人于2015年提出的基于单个神经网络的目标检测系统。在2017年CVPR上，Joseph Redmon和Ali Farhadi发表的YOLOv2，进一步提高了检测的精度和速度。 YOLOv1中的创新点在于利用单一的神经网络实现物体的检测和边框的回归。

在YOLOv1中，会先将图像resize成448*448的大小，卷积并通过全连接层得到C*C*B*5的张量，其中 C*C为图片的网格数，B为单个网格中预测的框的个数，5为对应的坐标信息和置信度。对于这种特殊的输出，因而需要设计专门的loss函数，在YOLOv1中，设计有对坐标回归的损失函数，对物体分类的损失函数以及对于置信度判断的损失函数 ,在训练时的loss函数为上述三者按照不同比例的加和。由于YOLOv1的网络结构简单，可以实现在保证一定准确率的情况下进行实时的检测。之后提出的YOLOv2 为YOLOv1的改进版本，在v1版本中，将图片分成7*7个网格并且每个网格只预测2个框，在YOLOv2中，参照FasterRCNN，加入anchor机制，先将图片resize成418*418的大小，经过卷积和池化之后得到13*13*N*5的张量，其中N为anchor的个数。此时的13*13为网格数，并且在每个网格中预测N个大小比例不同的anchor，相比较之前的结构，这样会对较小的物体以及聚集在一起的物体有较高的识别率。 YOLOv2中也将网络改成全卷积的结构来适应不同的尺寸，并且在卷积层中加入批归一化，同时添加类似Resnet的短接层结构来获得较小物体的信息。

2.2 具体实现

由于网络结构较简单，YOLO系列的实现的难点也在于对loss函数的实现，在最初的实现中，没有加入批归一化，在载入初始化模型之后训练1天得到mAP52%的结果。但实时的利用摄像头的检测结果并没有太好表现，原以为是由于训练时间不足，但再经过两天的训练之后mAP没有提升并且实时的效果更差。我们认为其原因是预训练数据基础上训练时间过长导致了过拟合，或VOC2007训练集较小，物体类别较少，在实际场景表现难以令人满意。 之后更新网络结构变

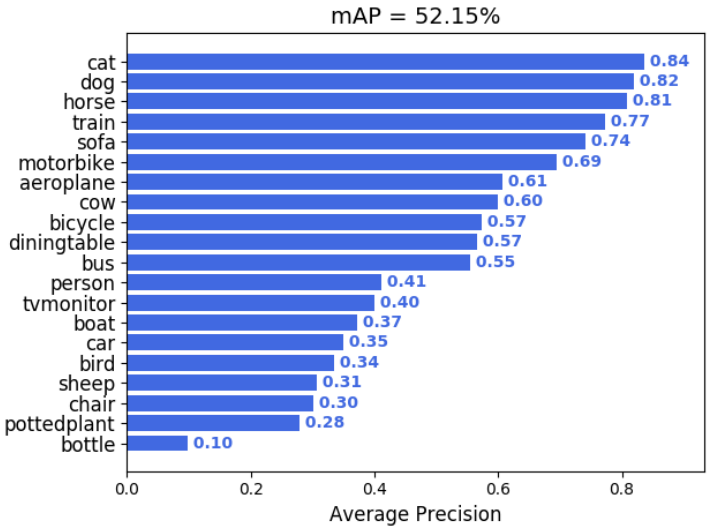

图 1: YOLOv1测试结果

为v2的结构，移除v1中的7*7的卷积核，全改为3*3或是1*1的卷积，在卷积层中去掉偏置并且加入批归一化，并且保存第17层的卷积结果，将其短接在26层的卷积结果之后。对于最后一层的卷积层加入偏置并且去除归一化，将之前的网格数改为13*13,并利用了5种不同比例的anchor，在loss函数方面，先将原本的输出进行处理，对于x,y和物体置信度进行sigmoid处理，对于物体类别的预测进行softmax,对于w和h，为了防止训练时进行指数运算导致inf,限制在-10到10之间(w和h会作为e的指数来得到对于anchor长宽的偏移量)。之后根据论文描述的，对物体负责的框计算坐标回归，分类等误差。以及非负责框的置信度误差。之后利用预训练模型再训练2天后，在VOC数据集上测试得到如下结果。可以看到YOLOv2的结构可以达到比YOLOv1更好的效果。 之后将需要判断的类别改为80类，使用COCOtrain2014作为训练集进行训练。虽然在原论文中也提出了YOLO9000,利用WordTree来进行训练，并且在对于全卷积网络的训练中作多尺度训练，但因为硬件条件和时间有限，在具体实现时只是在预训练数据基础上进行单尺度的训练。

2.3 代码结构

在提交的代码中，含有COCO和VOC的名称的文件为对COCO数据集和VOC数据集的处理，对于COCO数据集，由于官方提供了Api接口，可以调用相应的函数，得到x,y,w,h和相应的图片

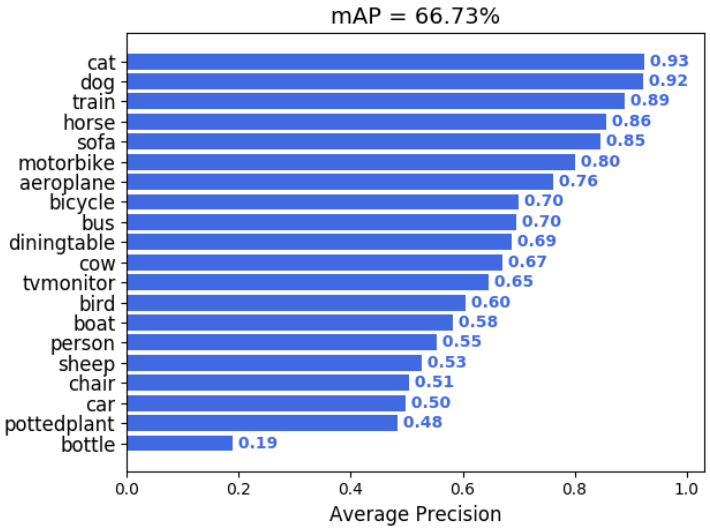

图 2: YOLO2测试结果

路径之后，转换成训练需要的坐标格式，并且将转化后的标签存入pkl文件中。对于VOC数据集，需要从官方提供的文本文件下读取相应的图片序号，并且根据相应的序号在annotation中得到对应的xml文件，从中获得坐标信息和物体分类。 test_picture和realtime为对YOLOv2网络的测试代码，分别为单一图像的测试以及实时的摄像头测试。 train为主要的训练程序类，在被调用时会生成相应的data类和net类，并且开始对net类进行训练。由于邮件上限的限制，所需要的COCOapi接口不在提交的代码当中。在yolo_net中定义了YOLOv2的网络结构以及损失函数。

3 Real-Time Video Captioning

3.1 概述

Image-Captioning任务目标为自动生成对于图片内容的文字描述。传统的Image-captioning使用Vgg-16或Resnet模型提取图像特征，但层数较多的vgg或resnet计算时间较长，不能实时生成字幕。因此我们便考虑使用可以达到一定准确率的并且能够实时给出结果的YOLO模型。我们利用前文提到的实时提取特征效果较好的YOLOv2模型，作为CNN结构提取图像特征并使其适应RNN-LSTM的输入，传入RNN-LSTM模型中，并运用Attention machanism实现字幕的生成。

3.2 具体实现

在我们的代码中，base_model.py与model.py定义了网络结构与训练、读取、测试、保存网络等过程。 dataset.py定义了从目标文件夹读取数据的过程，config.py与config_yolo.py分别定义了CNN与RNN的相关配置参数。在将Vgg16或Resnet50替换成YOLOv2后，最初构想为利用YOLOv2的输出作为RNN-LSTM模型的输入来进行训练，但实际效果并不理想，YOLOv2最后一层卷积返

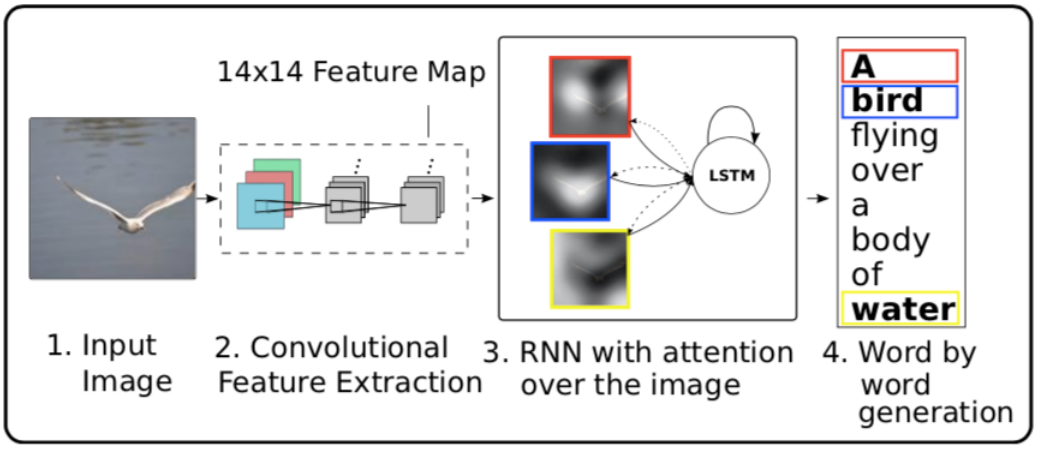

图 3: image-caption

回精确坐标，在Image-Captioning任务中不需要精确定位，因此我们去掉了YOLO结构中的最后一层卷积层。将特征图reshape成32*196*512的张量作为RNN-LSTM结构的输入。我们将训练好的YOLOv2模型的模型参数导入后进行RNN-LSTM部分的训练，训练集为COCOtrain2014。训练时直接运行main_train.py并将phase设为’train’即可。测试时，将需要测试的图片放到’./test/images’下，运行main_image.py即可在’./test/results’文件夹下看到生成的字幕。

使用YOLOv2作为CNN结构后支持实时检测与实时生成。在main_video.py中我们加入了摄像头接口与视频接口，可以与摄像头连接生成实时的对周围环境的描述，或者可以输入视频对视频进行实时生成解释文字。在main_video.py的cap.VideoCapture()中输入视频路径即可。若参数为0即为连接本机摄像头。

4 测试结果

4.1 图片测试结果

下图为我们模型的图像测试的结果，由于硬件设备、训练时间、数据集词汇量等的局限性，对于一些情况可能会得到较为荒唐的结果。但在场景不过于复杂的前提下可以看到该模型具有着不错的表现。

4.2 视频测试结果

具体视频测试结果在一并提交的文件当中，命名为demo的文件即为视频测试结果。

图 4: image-caption

参考文献

[1] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[2] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. *arXiv preprint*, 2017.

[3] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057, 2015.

[4] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[5] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[6] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[1] [2] [3] [4] [5] [6]