

基于深度学习的物体检测 - 作业 6

peng00bo00

September 6, 2020

1. 本次作业是对 ATSS 代码进行调试, 现将代码核心内容总结如下:

ATSS 结合了 FCOS 框架主要由 2 部分组成, 分别为 Backbone 以及 ATSSModule:

- (a) Backbone 由 ResNet 和 FPN 组成, 用来生成多尺度特征金字塔。
- (b) ATSSModule 为 ATSS 算法的核心组成部分, 其主要包括 ATSSHead、anchor_generator、box_selector 以及 loss_evaluator:
 - i. ATSSHead 是 ATSSModule 的主要预测模块, 用来预测锚框的类别、回归值和中心性。得到特征图后会分别进入 ATSSHead 的 2 个分支 cls_tower 以及 box_tower 进行预测。与 RetinaNet 类似 ATSSHead 在不同尺度的特征图上是共享的。
 - ii. anchor_generator 的作用与 Faster RCNN 中相同, 用来在不同尺度的特征图上产生锚框, 不过这里在同一个位置只需产生 1 个锚框而不是像 Faster RCNN 或 RetinaNet 那样产生多个不同高宽比的锚框。
 - iii. box_selector 在模型训练阶段不起作用, 但在模型推断阶段能够根据 ATSSHead 预测的锚框类别、回归值以及中心性来生成候选框。
 - iv. 得到预测值与 anchor 后由 loss_evaluator 计算损失, 具体流程如下:
 - A. 对于每个标注框在每个尺度下根据锚框中心到标注框中心的距离选择最近的 k 个锚框。
 - B. 计算 k 个锚框和标注框的 IoU 并统计所有尺度下 IoU 的均值 (mean) 和方差 (std)。
 - C. 以 mean+std 作为阈值将选取锚框中大于阈值且中心在标注框内的作为正样例, 其他作为负样例。
 - D. 如果一个锚框可以匹配到多个标注框则将该锚框匹配到具有最大 IoU 的标注框上。
 - E. 将锚框对应的类别以及正样例锚框对应的回归值作为锚框的分类和回归目标。
 - F. 根据正样例锚框的回归目标计算相应锚框的中心性。
 - G. 计算锚框的分类回归以及中心性损失, 需要注意的是计算回归以及中心性损失时只考虑正样例而分类损失则考虑全部锚框, 且回归损失需要按照中心性目标进行加权。

(c) 得到损失后将损失传给优化器对网络参数进行优化

2. 基于锚框的单阶段法 RetinaNet 和无需锚框的中心域法 FCOS 的区别如下:

- (a) anchor 的使用方法不同: RetinaNet 把 anchor 作为锚框中心, 而 FCOS 则把 anchor 作为目标内的一个点
- (b) anchor 的数量不同: RetinaNet 在每个 anchor 的位置要设置多个锚框以适应不同的高宽比, 而 FCOS 每个 anchor 位置仅有 1 个
- (c) anchor 的回归目标不同: RetinaNet 对 anchor 回归得到锚框中心和长宽的偏移量, 而 FCOS 对 anchor 回归得到 anchor 到目标框 4 个边界的距离
- (d) anchor 的匹配方式不同: RetinaNet 是把每个 anchor 匹配到具有最大 IoU 的目标上, 容易产生正负样本不均衡的问题; 而 FCOS 则是将落在目标框内的所有 anchor 都匹配上, 所以正负样本的比例要好很多
- (e) 损失函数不同: RetinaNet 对每个 anchor 要预测类别和偏移量, 而 FCOS 除此之外还要预测一个中心性 (centerness)
- (f) 多尺度机制不同: RetinaNet 在不同尺度的特征图上对不同大小的目标进行匹配, 而 FCOS 则是按 anchor 到目标中心的距离进行多尺度匹配