目标检测 | 清晰易懂的SSD算法原理综述

原创 石头 机器学习算法那些事 1月3日

SSD（Single Shot Detection）是一个流行且强大的目标检测网络，网络结构包含了基础网络（Base Network），辅助卷积层（Auxiliary Convolutions）和预测卷积层（Predicton Convolutions）。

本文包含了以下几个部分：

（1）理解SSD网络算法所需要理解的几个重要概念

（2）SSD网络框架图

（3）SSD网络中几个重要概念的详细解释

（4）SSD网络如何定位目标

（5）SSD网络的算法流程图

（5）小结

1.理解SSD网络所需要理解的几个重要概念

Single Shot Detection：早期的目标检测系统包含了两个不同阶段：目标定位和目标检测，这类系统计算量非常耗时，不适用实际应用。Single Shot Detection模型在网络的前向运算中封装了定位和检测，从而显著提高了运算速度。

多尺度特征映射图（Multiscale Feature Maps）：小编认为这是SSD算法的核心之一，原始图像经过卷积层转换后的数据称为特征映射图（Feature Map），特征映射图包含了原始图像的信息。SSD网络包含了多个卷积层，用多个卷积层后的特征映射图来定位和检测原始图像的物体。

先验框（Priors）：在特征映射图的每个位置预先定义不同大小的矩形框，这些矩形框包含了不同的宽高比，它们用来匹配真实物体的矩形框。

预测矩形框：每个特征映射图的位置包含了不同大小的先验框，然后用预测卷积层对特征映射进行转换，输出每个位置的预测矩形框，预测矩形框包含了框的位置和物体的检测分数。比较预测矩形框和真实物体的矩形框，输出最佳的预测矩形框。

损失函数：我们知道了预测的矩形框和真实物体的矩形框，如何计算两者的损失函数？
损失函数包含了位置损失函数和分类损失函数，由于大部分矩形框只包含了背景，背景的位置不需要定位，因此计算两者的位置损失函数用L1函数即可。我们把背景称为负类，包含了物体的矩形框称为正类，不难理解图像中大部分的矩形框只包含了负类，若用全部的负类和正类来计算损失函数，那么训练出来的模型偏

向于给出负类的结果。解决办法是在计算分类损失函数时，我们只选择最难检测的几个负类和全部正类来计算。

非极大值抑制（Non-maximum Suppression）:若两个矩形框都包含了相同的物体，且两个矩形框的重叠度较高，则选择分数较高的矩形框，删除分数较低的矩形框。

2.SSD网络框架定义及其应用

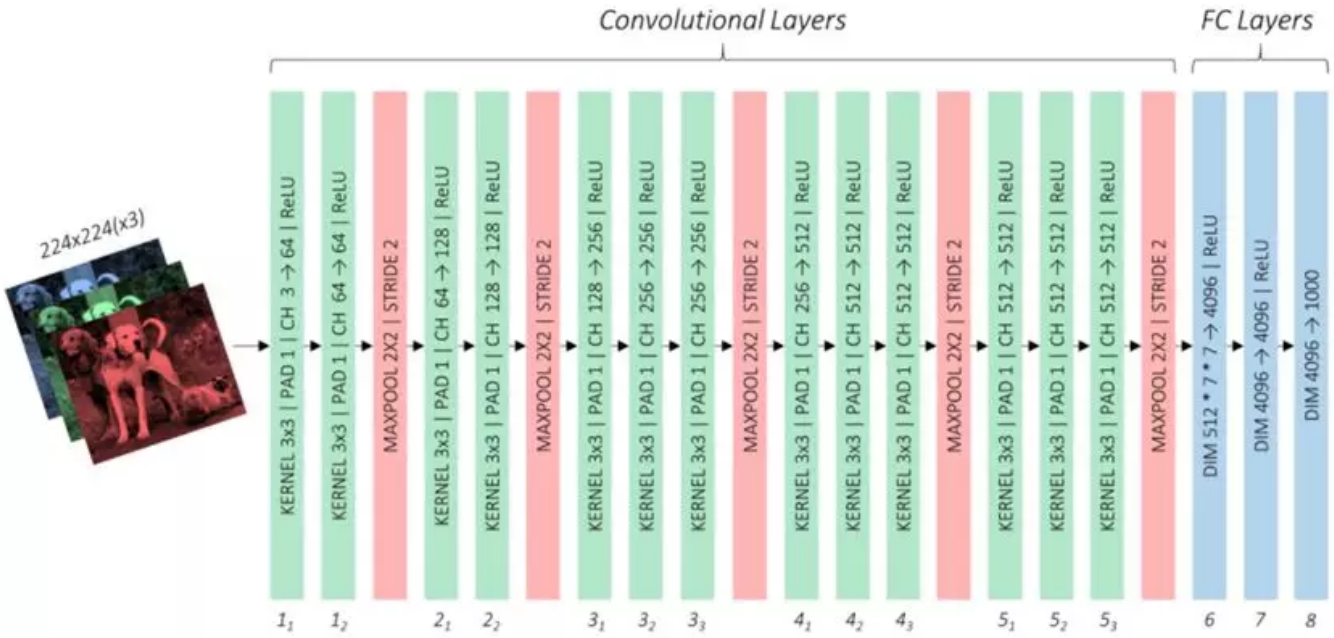
SSD网络包含了基础网络，辅助卷积层和预测卷积层：

- 基础网络：提取低尺度的特征映射图
- 辅助卷积层：提取高尺度的特征映射图
- 预测卷积层：输出特征映射图的位置信息和分类信息

下面介绍SSD网络的这三个部分

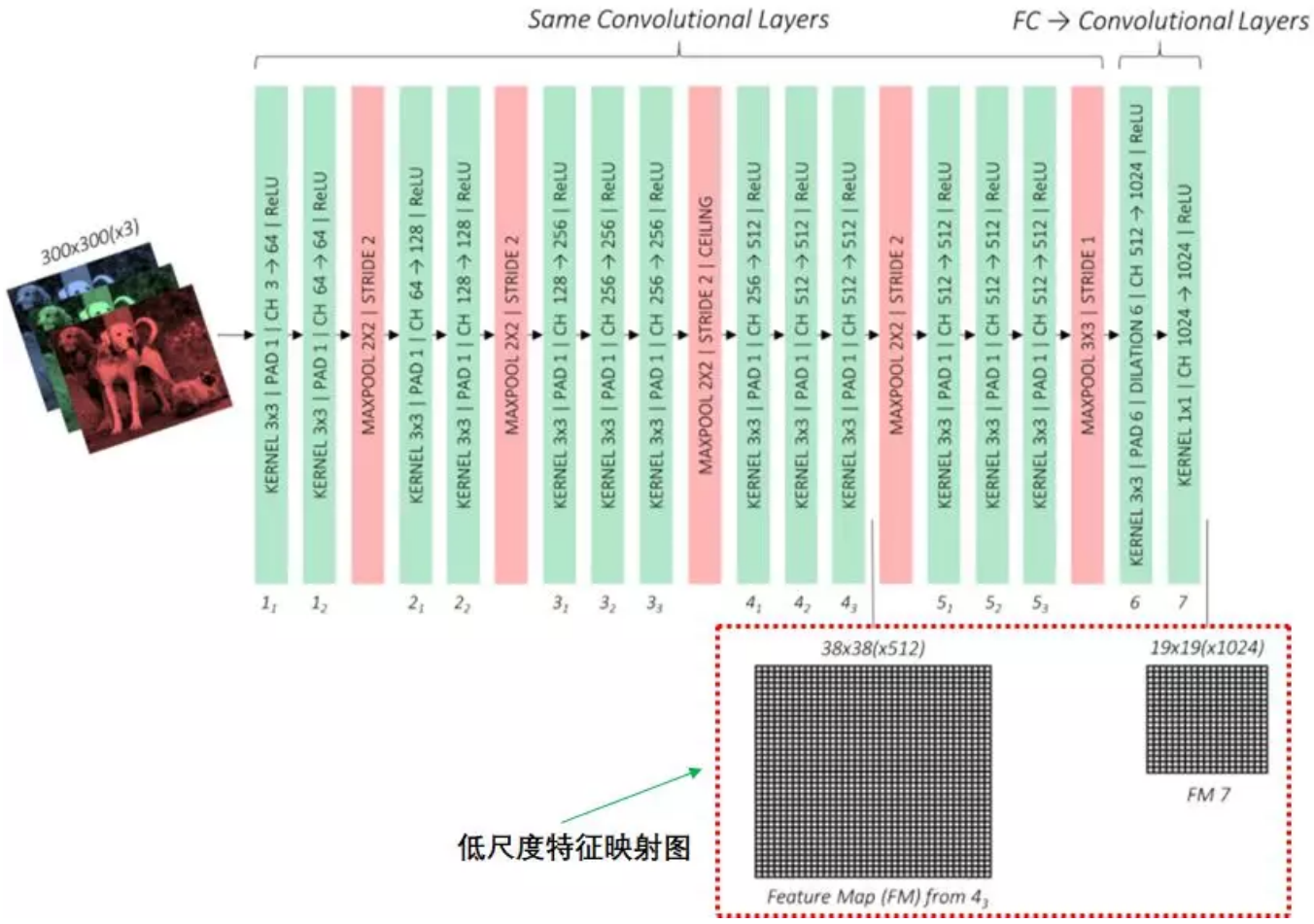
基础网络

基础网络的结构采用了VCG-16网络架构，VCG-16网络如下图：



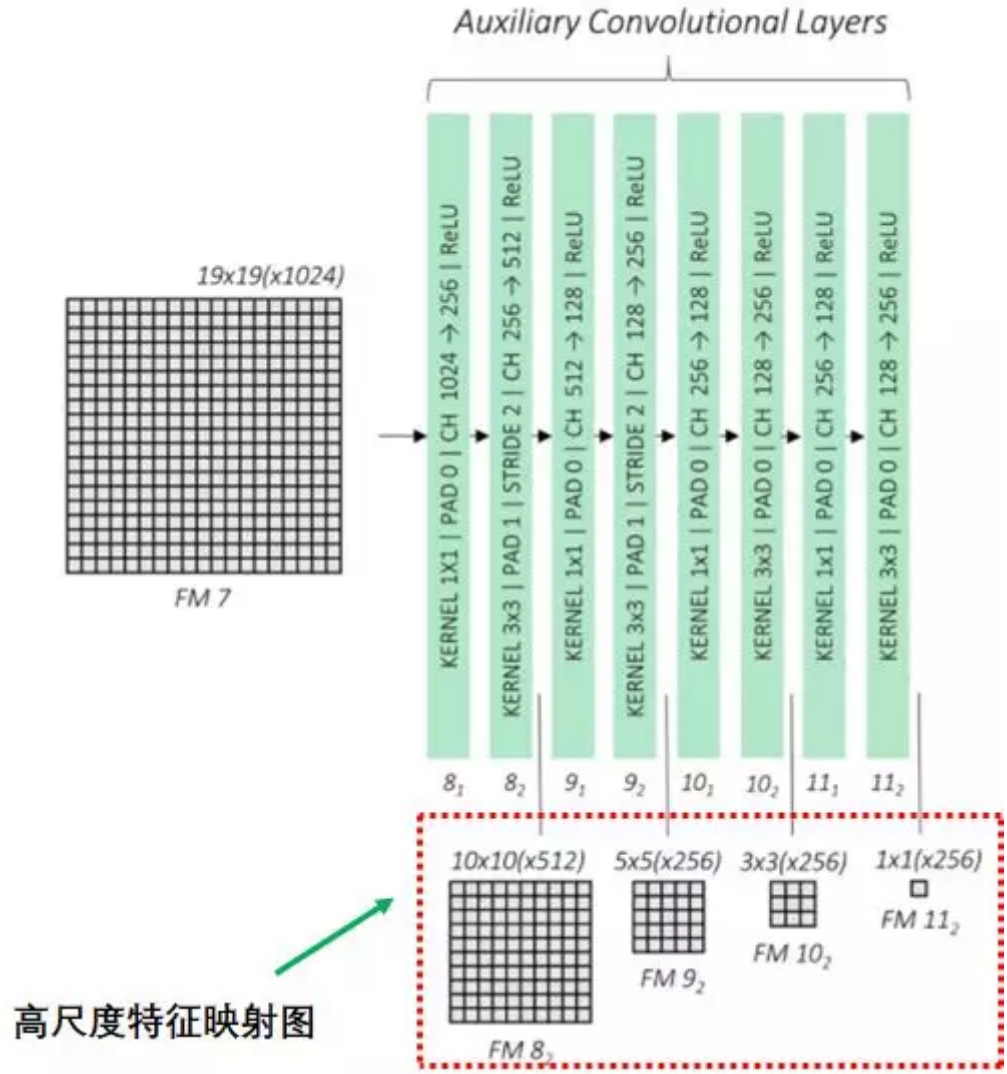
VCG-16网络包含了卷积层和全连接层（FC Layers），全连接层的任务用来分类，由于基础网络只需要提取特征映射图，因此需要对全连接层用卷积层代替，这一部分的参数和VCG-16网络的卷积层参数用迁移学习的方法获取。

基于VCG网络架构的基础网络如下图：



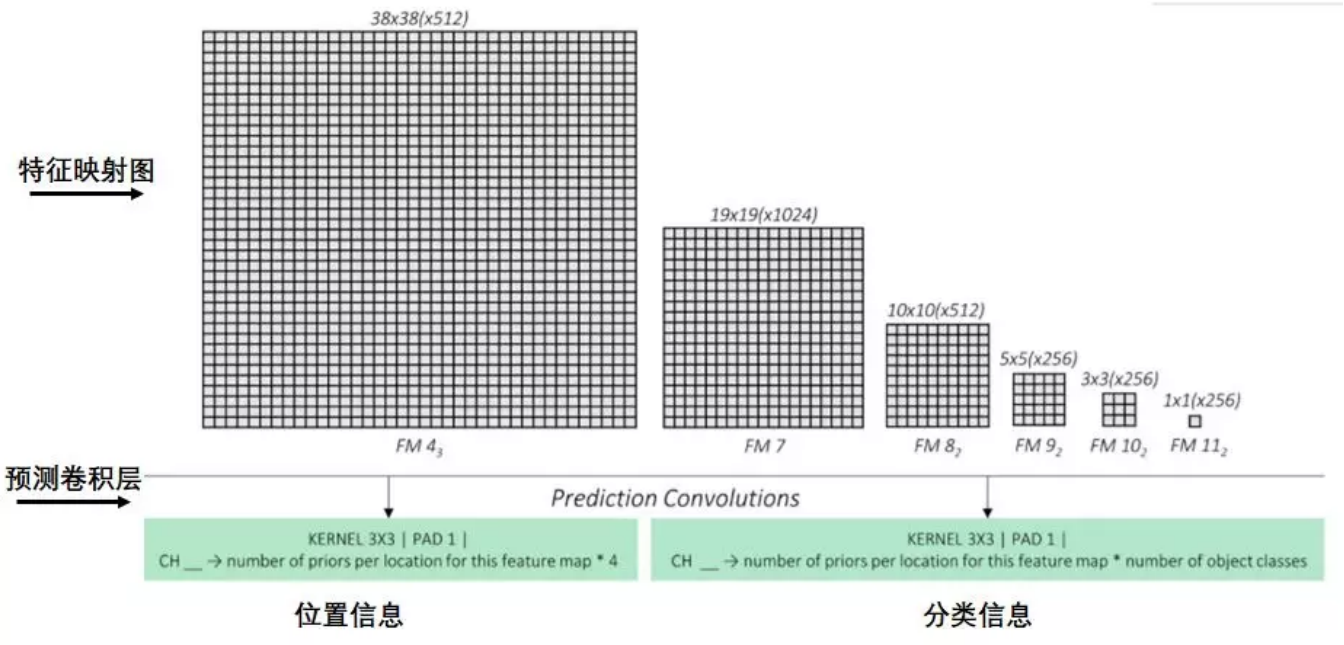
辅助卷积层

辅助卷积层连接基础网络最后的特征映射图，通过卷积神经网络输出4个高尺度的特征映射图：



预测卷积层

预测卷积层预测特征映射图每个点的矩形框信息和所属类信息，如下图：

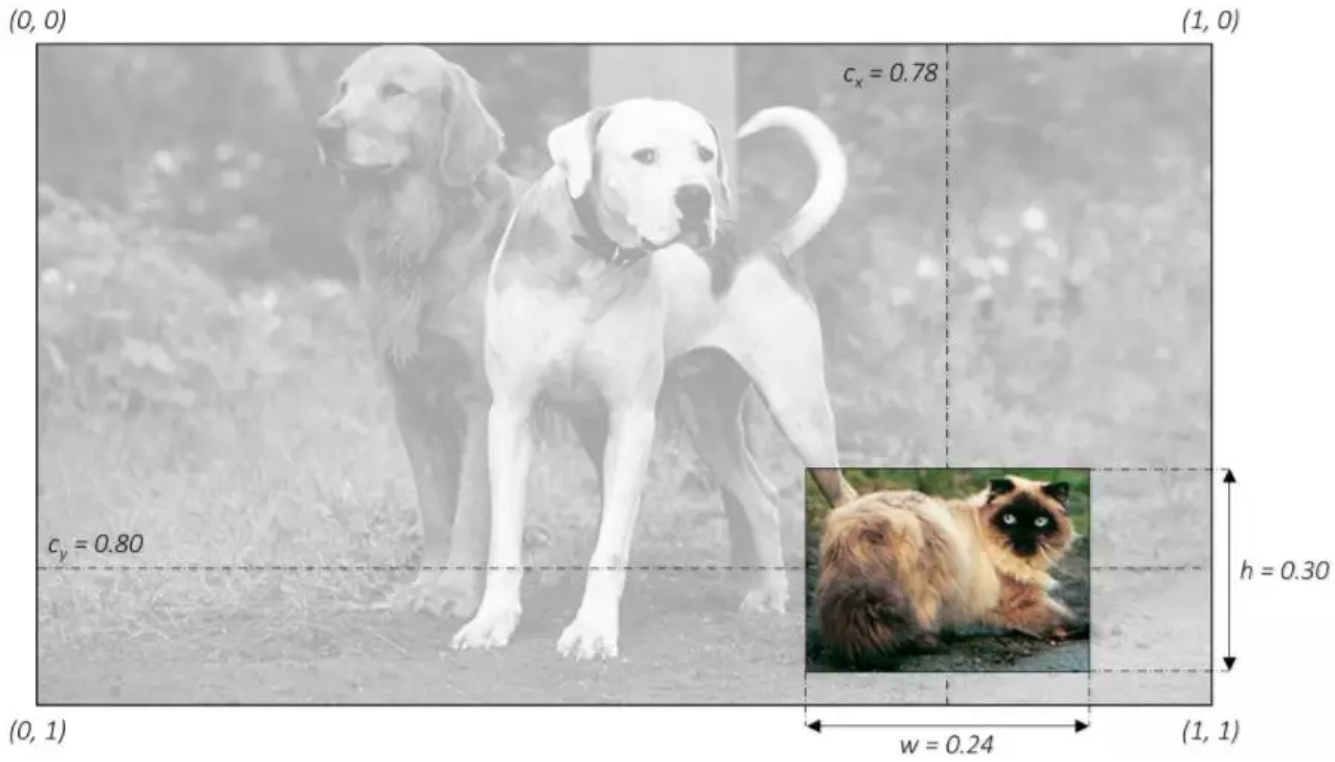


3.SSD网络中几个重要概念的详细解释

如何表示矩形框

我们用矩形框定位物体的位置信息和所属类，如下图：

常用四个维度表示矩形框信息，前两个维度表示矩形框的中心点的位置，后两个维度表示矩形的宽度和高度。为了统一，我们使用归一化的方法表示矩形框：



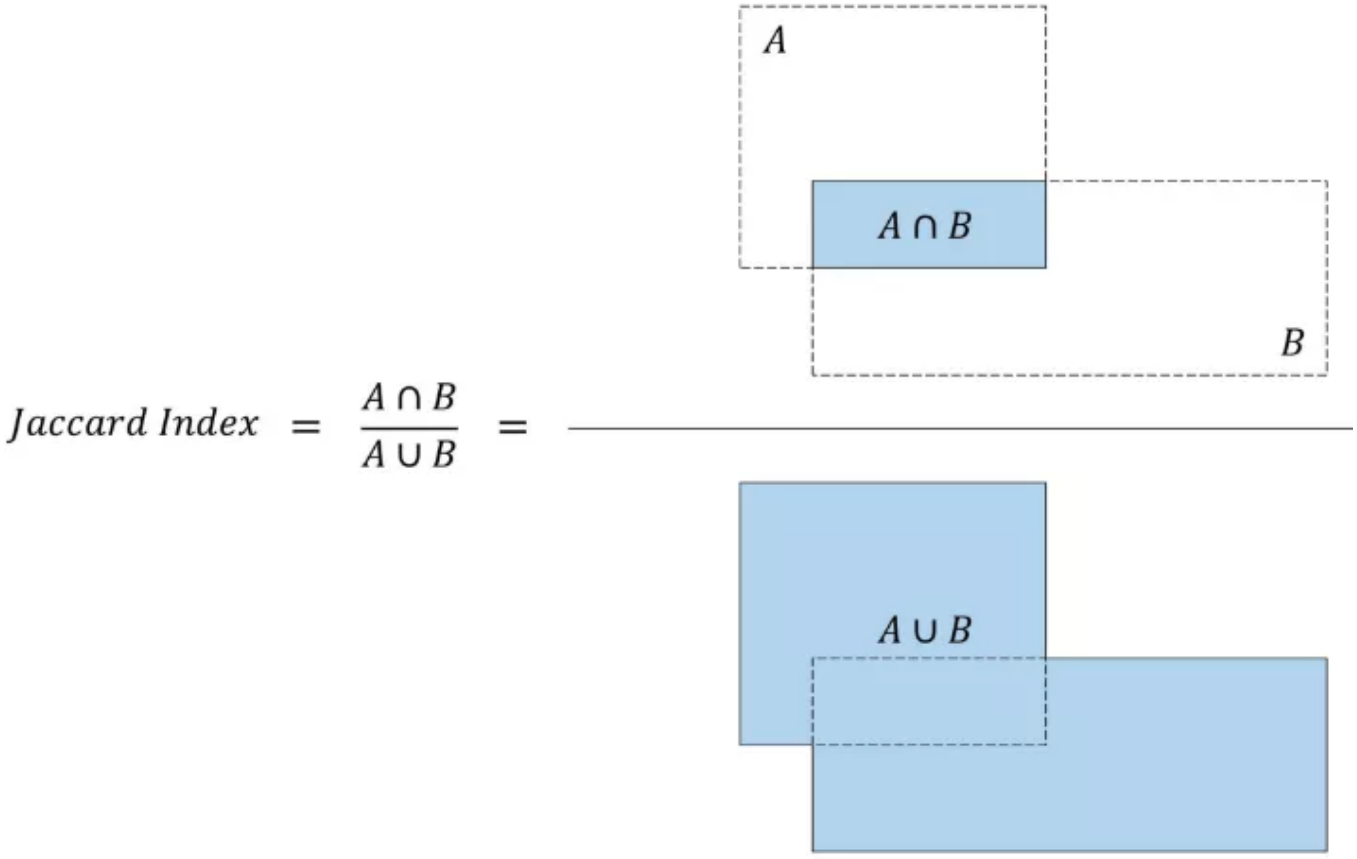
上图猫的矩形框为：（0.78，0.80，0.24，0.30）

如何衡量两个矩形框的重叠度

SSD算法中有两处需要计算矩形框的重叠度，第一处是计算先验矩形框和真实矩形框的重叠度，目的是根据重叠度确定先验框所属的类，包括背景类；第二处是计算预测矩形框和真实矩形框的重叠度，目的是根据重叠度筛选最优的矩形框。

我们用Jaccard Index或交并比（IoU）衡量矩形框的重叠度。

交并比等于两个矩形框交集的面积与矩形框并集的面积之比，如下图：



损失函数算法

预测层预测了映射图每个点的矩形框信息和分类信息，该点的损失值等于矩形框位置的损失与分类的损失之和。

首先我们计算映射图每个点的先验框与真实框的交并比，若交并比大于设置的阈值，则该先验框与真实框所标记的类相同，称为正类；若小于设置的阈值，则认为该先验框标记的类是背景，称为负类。

然后预测层输出了映射图每个点的预测框，预测框的标记与先验框的标记相同。

预测框与真实框的损失函数等于预测框位置的损失与分类的损失之和。

1. 预测框位置的损失：

由于不需要用矩形框定位背景类，所以只计算预测正类矩形框与真实矩形框的位置损失：

我们用 nn.L1Loss函数计算矩形框位置的损失。

n1.L1Loss函数：

```
torch.nn.L1Loss(size_average=None, reduce=None, reduction='mean')
```

公式：

$$\ell(x, y) = L = \{l_1, \dots, l_N\}^\top, \quad l_n = |x_n - y_n|$$

其中N表示样本个数。

如果reduction不为'none'(默认设为'mean')，则

$$\ell(x, y) = \begin{cases} \text{mean}(L), & \text{if reduction} = \text{'mean'}; \\ \text{sum}(L), & \text{if reduction} = \text{'sum'}. \end{cases}$$

假设共有N个正类的预测矩形框，每个矩形框的位置为

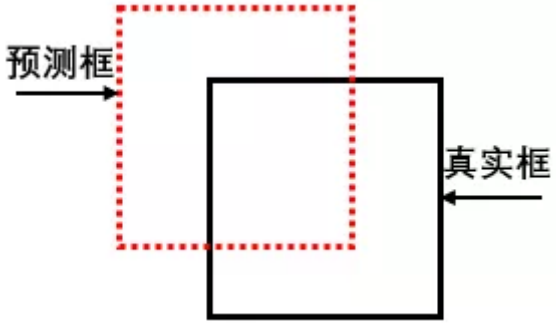
$$(x_{i\_min}, y_{i\_min}, x_{i\_max}, y_{i\_max})$$

其中 i = 1,2,...,N

每个预测矩形框对应的正类真实矩形框的位置为：

$$(x_{ij\_min}, y_{ij\_min}, x_{ij\_max}, y_{ij\_max})$$

如下图的预测矩形框和对应的正类真实矩形框：



损失函数为：

$$L_{loc}=\frac{1}{N}\sum_{i=1}^{N}\left(|x_{i\_\min}-x_{ij\_\min}|+|y_{i\_\min}-y_{ij\_\min}|+|x_{i\_\max}-x_{ij\_\max}|+|y_{i\_\max}-y_{ij\_\max}|\right)$$

2.预测类的损失：

由第一节的损失函数介绍可知，大部分的预测矩形框包含了负类（背景类），容易知道一张图中负类的个数远远多于正类，若我们计算所有类的损失值，那么训练出来的模型会偏向于预测负类的结果。

因此我们选择一定数量的负类个数和全部的正类个数来训练模型，负类个数N_hn，正类个数N_p，负类个数与正类个数满足下式：

$$N\_hn = N\_p * 3$$

我们知道了负类个数，如何从数量庞大的负类中选择所需要的负类个数？本文采用了最难检测到负类的预测框作为训练的负类，称为Hard Negative Mining。

现在我们知道了如何选择负类，那么如何预测分类损失函数？关于多分类任务，我们常用交叉熵来评价分类损失函数。

若预测的类个数为K（包含了背景类），交叉熵公式如下：

$$H(p,q)=-\sum_{i=1}^{K}p(x_i)\log(q(x_i))$$

其中$p(x_i)$为真实类属于第 i 类概率，若属于第 i 类则$p(x_i)=1$；若不满足则$p(x_i)=0$。$q(x_i)$为预测类属于第i类的概率，每个先验框的预测类是一个1行K列的矩阵。

若交叉熵损失函数为CE Loss，预测类的损失为L_{conf}，有：

$$L_{conf} = \frac{1}{N\_p}\left(\sum_{N\_p} CE\ Loss + \sum_{N\_hn} CE\ Loss\right)$$

其中N_P和N_hn分别为正类、负类个数。

总损失函数为预测类损失和预测位置损失之和，记为L，有：

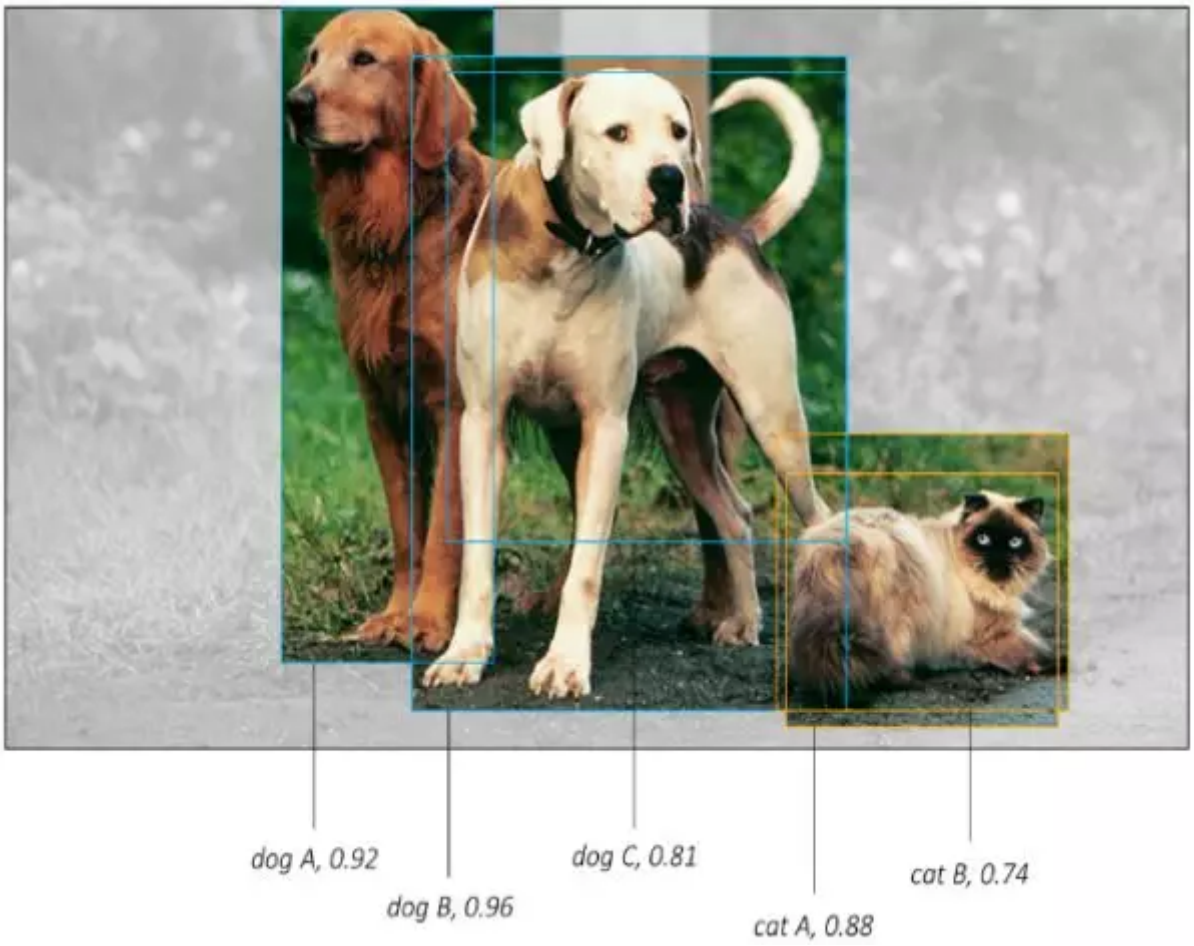
$$L = L_{conf} + \alpha \cdot L_{loc}$$

α常设置为1，或者也可作为待学习的参，SSD论文中设置α等于1。

4.SSD网络结构如何定位目标

前面介绍通过先验框和真实框的交并比来分类，若交并比大于阈值则为正类（包含某个特定物体的类），若交并比小于阈值则为负类（背景类）。

预测框与先验框的个数相等，若有多个相同正类的预测框的交并比很大（如下图），如何选择最优的预测框？



上图的五个预测框预测了三只狗和两只猫，三只狗的交并比如下表：

IoU	dog B	dog A	~~dog C~~
dog B	-	0.1	0.6
dog A	0.1	-	0.05
~~dog C~~	0.6	0.05	-

设置阈值为0.5，因为预测dog B的分数最大（0.96），且dog B和dog C的交并比大于阈值，因此一致dog C的预测框。由于dog A与其他预测框的交并比小于阈值，因此保留dog A的预测框。即狗的输出结果为两个。

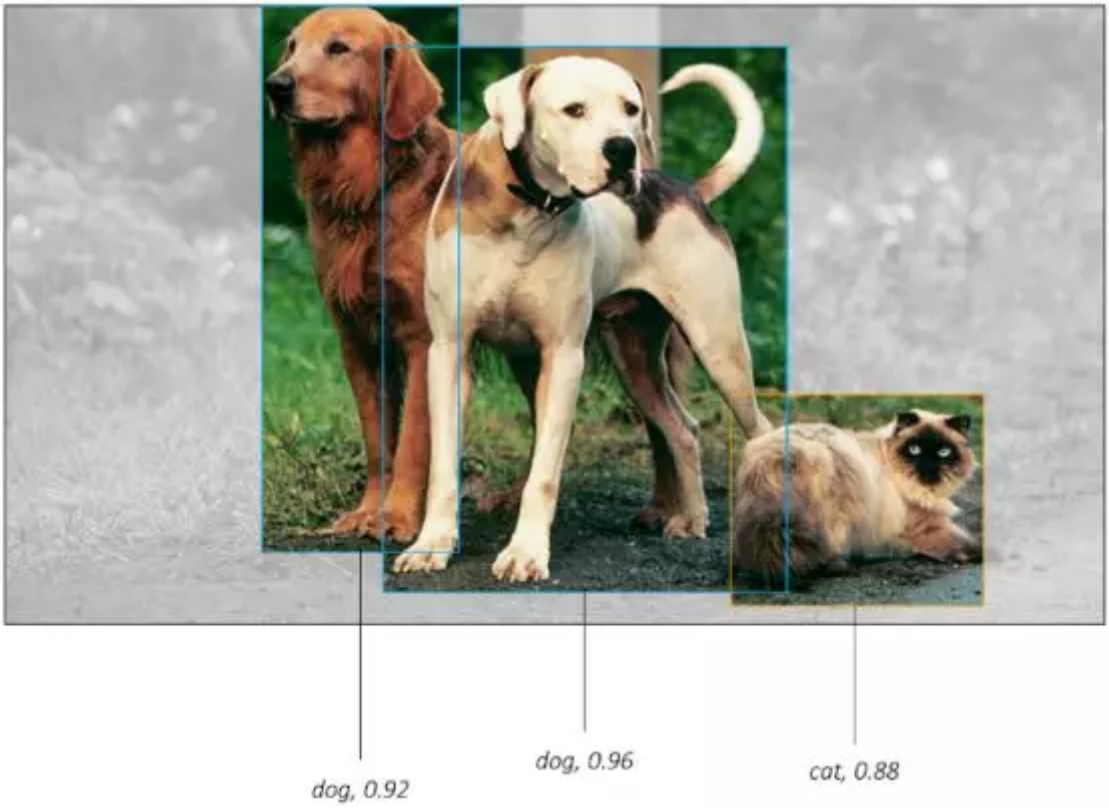
猫的预测矩形框如下表：

IoU	cat A	~~cat B~~
cat A	-	0.8
~~cat B~~	0.8	-

同理，由于cat A的预测分数最高，且cat B与cat A交并比大于阈值，因此抑制cat B预测框。

上述方法称为非极大值抑制（Non-Maximum Suppression）。

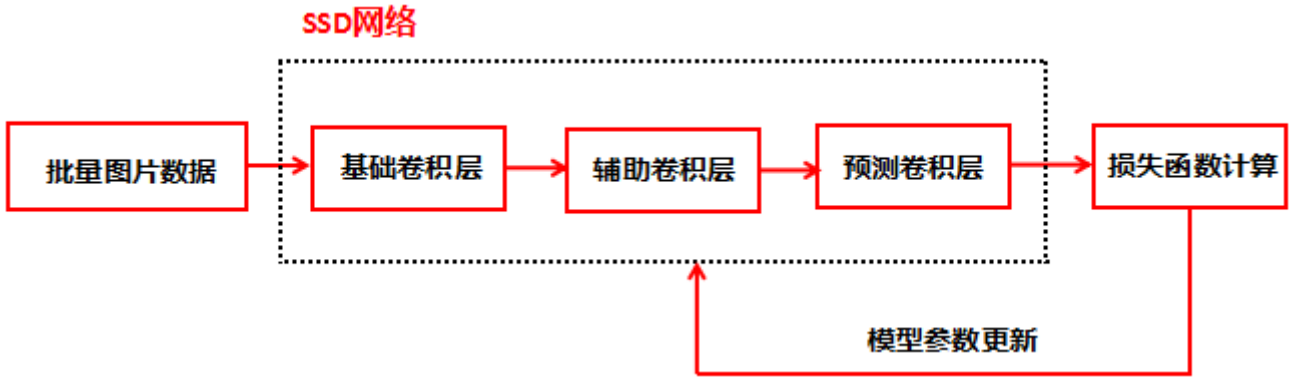
根据非极大值抑制方法，猫狗的预测框如下图：



5.SSD网络的算法流程

介绍了SSD网络结构以及理解该网络所需要的基础概念，基于这些知识，下面介绍SSD网络的算法流程。

训练阶段：



预测阶段

6.小结

本文介绍了SSD算法框架及原理，由于算法细节较多以及篇幅的关系，小编选择了几个非常重要且设计很巧妙的细节进行介绍，更详细内容的链接https://github.com/sgrvinod/a-PyTorch-Tutorial-to-Object-Detection，对于英文不好的同学，可参考该文帮助理解，若有不懂欢迎交流。

推荐阅读

520 页机器学习笔记！图文并茂可能更适合你，文末附下载方法

李航老师《统计学习方法》(第2版）课件分享，文末附下载

Github | 吴恩达新书《Machine Learning Yearning》完整中文版开源

经典好书 | 141页的《Deep Learning with PyTorch》开源书籍

400页《TensorFlow 2.0 深度学习算法实战》中文版教材免费下载

欢迎扫码关注：