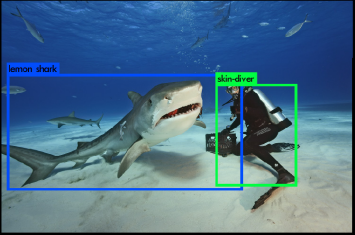
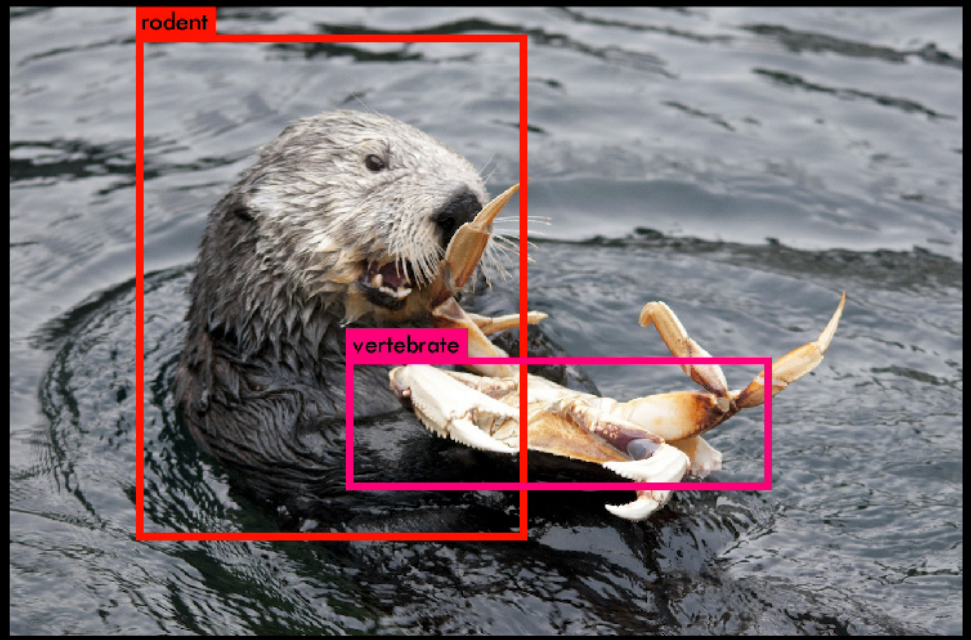
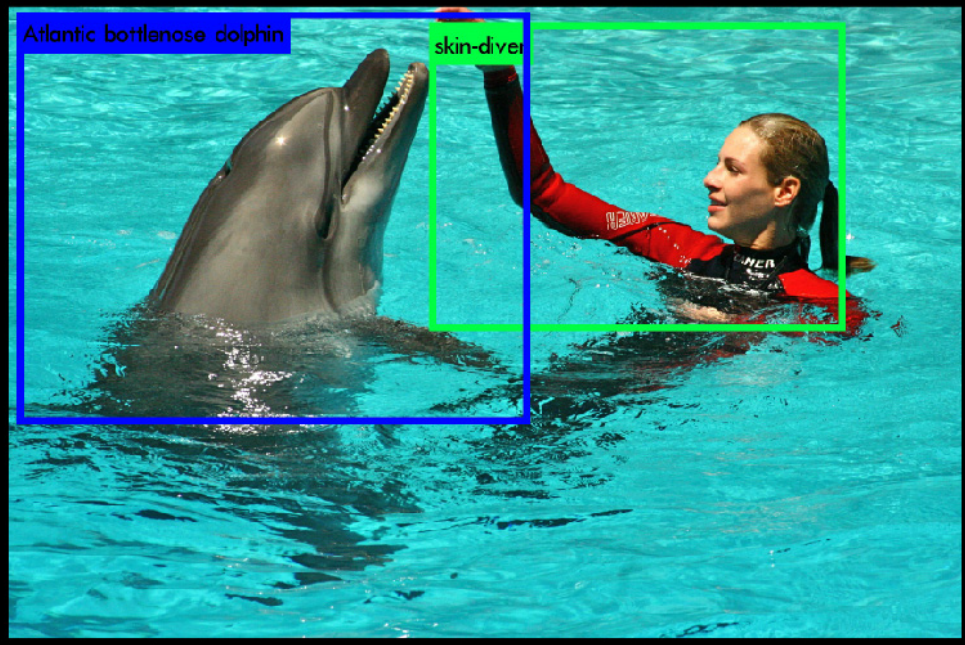
**YOLOv2**

**1输入输出**



输入：测试时416×416

输出：每张算法将每张图像分成S×S个grid cell，每个grid cell预测C个类别的和A个bounding box各自的五个参数，一张图片一共预测个值

在YOLOv2模型中S=13,A=3,C=9148，即预测个值

**2 Better←—**mAP更高

**2.1 Batch Normalization**

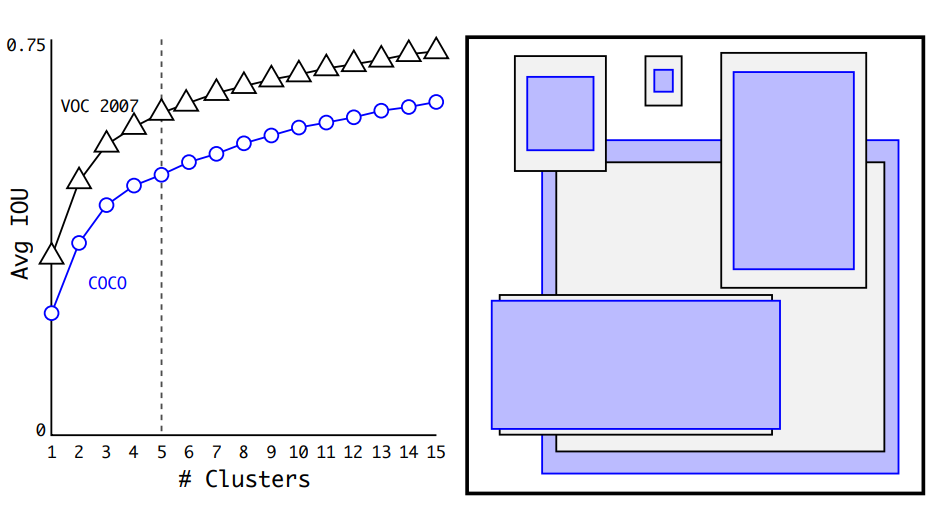
**2.2 High Resolution Classifier**

在图像尺寸Resize到448448的1000类的ImageNet图像分类数据集上训练10个epoch，以对在图像尺寸Resize到224224的该数据集上训练的原始分类器进行微调，使得模型更适合高分辨率输入

**2.3 Anchor**

确定宽高固定的先验锚框框，模型输出的是预测框相对于Anchor的偏移量bounding box，由与ground truth的 IoU最大的Anchor对应的bbox去负责检测物体

**2.3.1 Dimension Clustering**

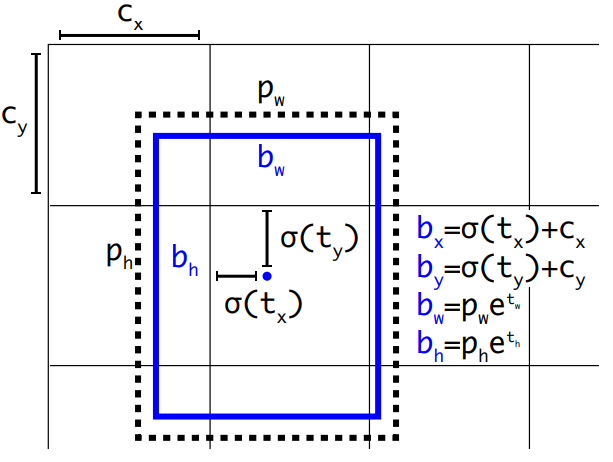


对训练集的bounding box标签运行k-means聚类，某anchor box到聚类中心的距离为：

聚类与bounding box标签的位置和的对应的类别都无关，计算两框IoU时假定二者中心重合

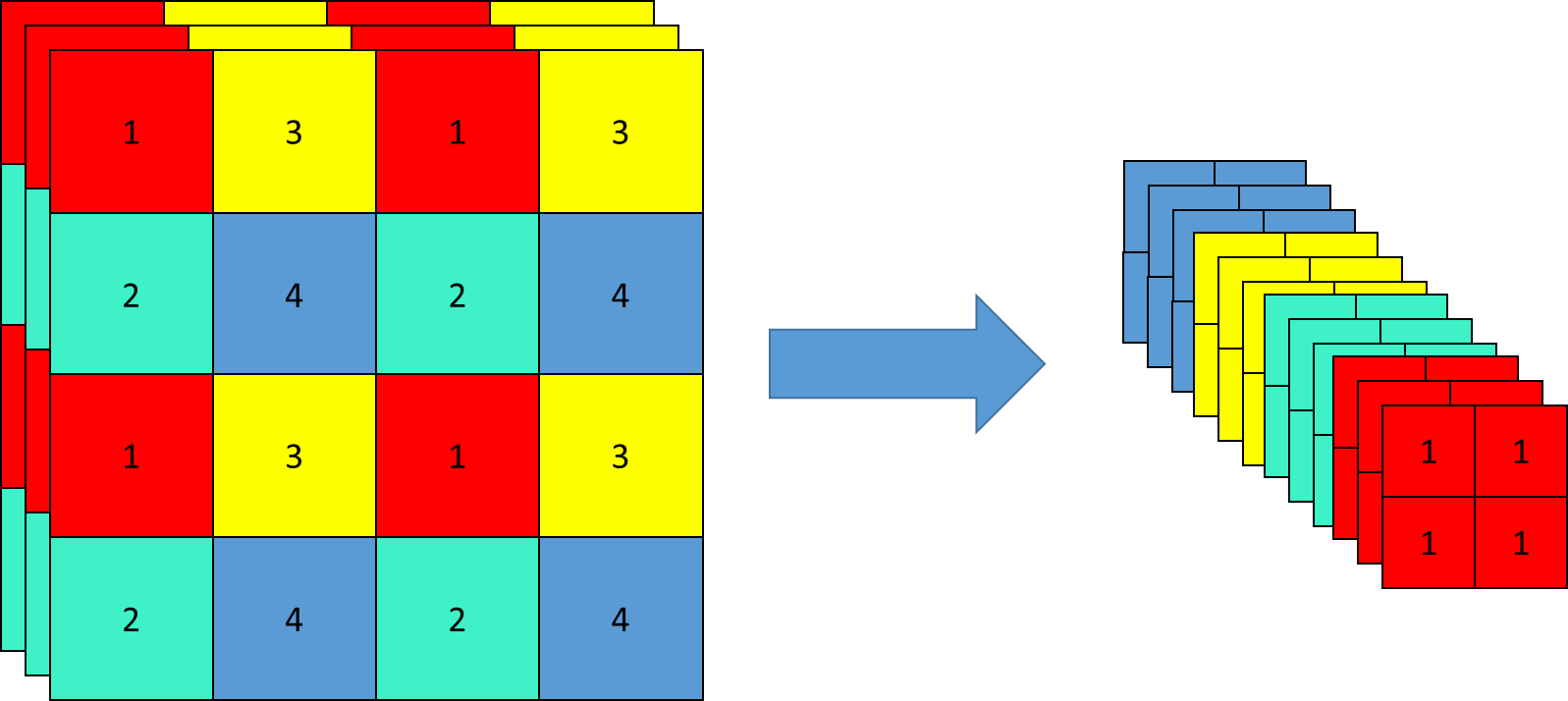
聚类数k越多，平均IoU就越大，但模型的复杂性也就越大，通过左图权衡得到k=5，也即每个grid cell有5个Anchor

取其聚类中心所对应的宽高，如右图，作为Anchor的先验  
**2.4 Direct location prediction**



：grid cell左上角的坐标；：Anchor的宽高

**2.5 Fine-Grained Features**



特征图通过如图所示拆分，得到一个PassThrough层，该层再与特征图经过卷积池化等后生成的新特征图拼接，便融合不同粒度的特征

**2.6 Multi-Scale Training**

通过多尺度训练，每10个batch随机选取一次输入图片尺度

YOLOv2卷积模型下采样32倍，因此用32的倍数等作为可选的图像输入维度

**3 Faster←—**速度更快

**3.1 Darknet-19 classifier**

Conv2d

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Filters | Size/Stride | Output |
| Convolutional | 32 | 3×3 | 224×224 |
| Maxpool |  | 2×2/2 | 112×112 |
| Convolutional | 64 | 3×3 | 112×112 |
| Maxpool |  | 2×2/2 | 56×56 |
| Convolutional | 128 | 3×3 | 56×56 |
| Convolutional | 64 | 1×1 | 56×56 |
| Convolutional | 128 | 3×3 | 56×56 |
| Maxpool |  | 2×2/2 | 28×28 |
| Convolutional | 256 | 3×3 | 28×28 |
| Convolutional | 128 | 1×1 | 28×28 |
| Convolutional | 256 | 3×3 | 28×28 |
| Maxpool |  | 2×2/2 | 14×14 |
| Convolutional | 512 | 3×3 | 14×14 |
| Convolutional | 256 | 1×1 | 14×14 |
| Convolutional | 512 | 3×3 | 14×14 |
| Convolutional | 256 | 1×1 | 14×14 |
| Convolutional | 512 | 3×3 | 14×14 |
| Maxpool |  | 2×2/2 | 7×7 |
| Convolutional | 1024 | 3×3 | 7×7 |
| Convolutional | 512 | 1×1 | 7×7 |
| Convolutional | 1024 | 3×3 | 7×7 |
| Convolutional | 512 | 1×1 | 7×7 |
| Convolutional | 1024 | 3×3 | 7×7 |
| Convolutional | 1000 | 1×1 | 7×7 |
| Avgpool |  | Global | 1000 |
| Softmax |  |  |  |

Darknet-19是YOLOv2的分类器，有19个卷积层，5个最大池化层，

Convolutional

LeakyReLU

BN

决策部分采用global average pooling，对每个channel求平均，舍弃了全连接层，同时也使得输入图像尺寸不受限制

训练时，在1000类的ImageNet图像分类数据集上，将图像Resize到224224训练分类器，然后仍在此数据集上，将图像Resize到448448对模型进行微调

**3.2 Training for Detection**

26×26×64

Convolutional 64 1×1

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Filters | Size/Stride | Output |
| Convolutional | 32 | 3×3 | 416×416×32 |
| Maxpool |  | 2×2/2 | 208×208×32 |
| Convolutional | 64 | 3×3 | 208×208×64 |
| Maxpool |  | 2×2/2 | 104×104×64 |
| Convolutional | 128 | 3×3 | 104×104×128 |
| Convolutional | 64 | 1×1 | 104×104×64 |
| Convolutional | 128 | 3×3 | 104×104×128 |
| Maxpool |  | 2×2/2 | 52×52×128 |
| Convolutional | 256 | 3×3 | 52×52×256 |
| Convolutional | 128 | 1×1 | 52×52×128 |
| Convolutional | 256 | 3×3 | 52×52×256 |
| Maxpool |  | 2×2/2 | 26×26×256 |
| Convolutional | 512 | 3×3 | 26×26×512 |
| Convolutional | 256 | 1×1 | 26×26×256 |
| Convolutional | 512 | 3×3 | 26×26×512 |
| Convolutional | 256 | 1×1 | 26×26×256 |
| Convolutional | 512 | 3×3 | 26×26×512 |
| Maxpool |  | 2×2/2 | 13×13×512 |
| Convolutional | 1024 | 3×3 | 13×13×1024 |
| Convolutional | 512 | 1×1 | 13×13×512 |
| Convolutional | 1024 | 3×3 | 13×13×1024 |
| Convolutional | 512 | 1×1 | 13×13×512 |
| Convolutional | 1024 | 3×3 | 13×13×512 |

Convolutional 1024 3×3

PassThrough Layer

13×13×256

Convolutional 1024 3×3

13×13×1024

13×13×1280

Convolutional 1024 3×3

13×13×1024

13×13×125

Conv2d 125 1×1

在20类的PACCAL VOC2012目标检测数据集上训练检测器

**4 Stronger←—**类别更多

**4.1 Hierarchical classification**

WordNet的有向图结构转化为树结构WordTree：

将WordNet中的“physical object”作为根节点

首先添加与根节点间只有一条路径词

再添加到与根节点间有多条路径的词，其路径取最短的

在分类时，计算每个同义词集的下义词的给定该词集条件下的概率如对“terrier”结点，计算

*Pr*(Norkfolk terrier|terrier)

*Pr*(Norkfolk terrier|terrier)

*Pr*(Norkfolk terrier|terrier)…

通过将某节点到根节点的条件概率相乘得到该结点的绝对概率

*Pr*(Norkfolk terrier) = *Pr*(Norkfolk terrier|terrier)

该条件概率是模型的直接输出值

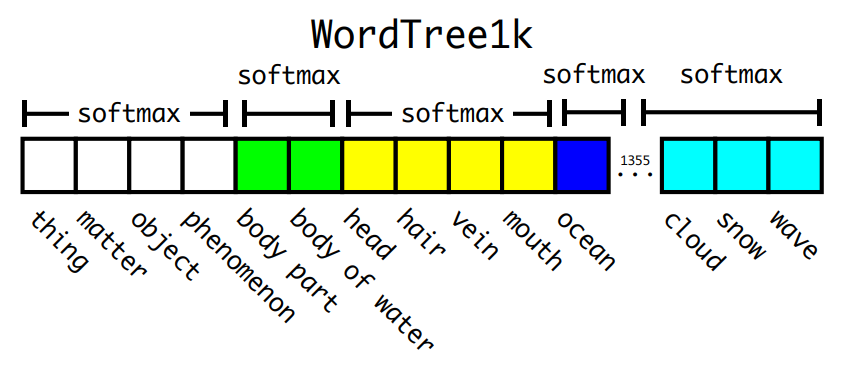
\* *Pr*(terrier|hunting dog)

\*…

\**Pr*(mammal|animal)

\* *Pr*(animal|physical object)

在分类时，假定*Pr*(physical object)=1，并采用multi-label，即若作为训练样本的子节点为1，则其父节点也为1

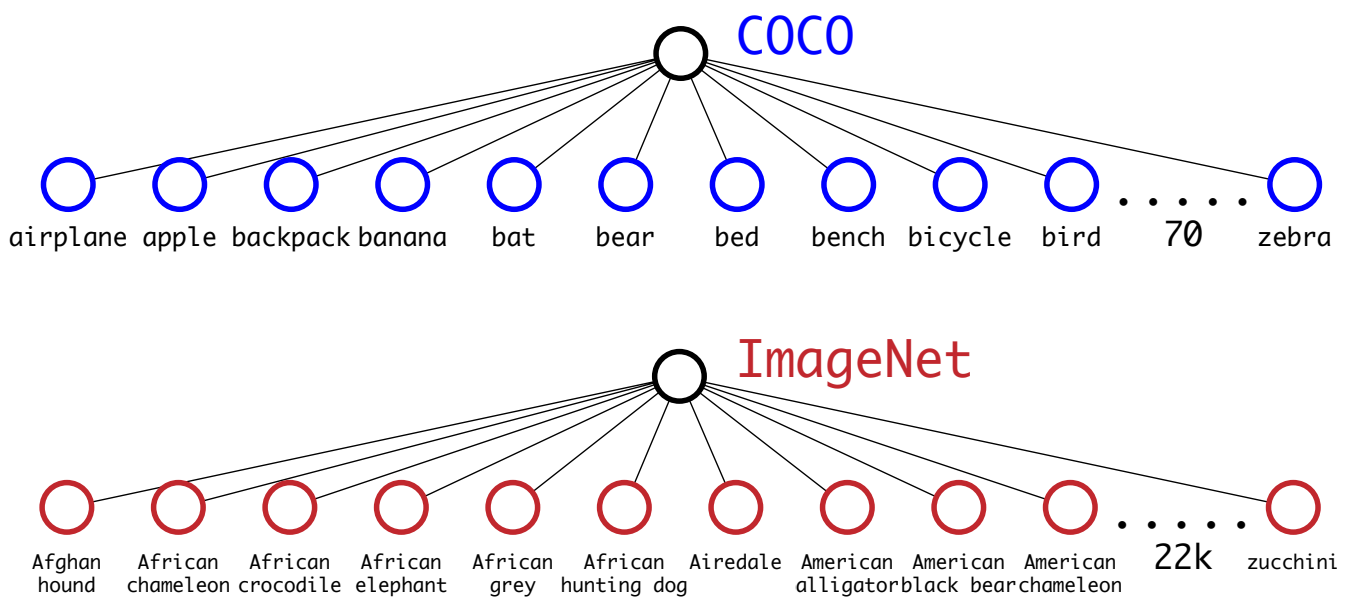


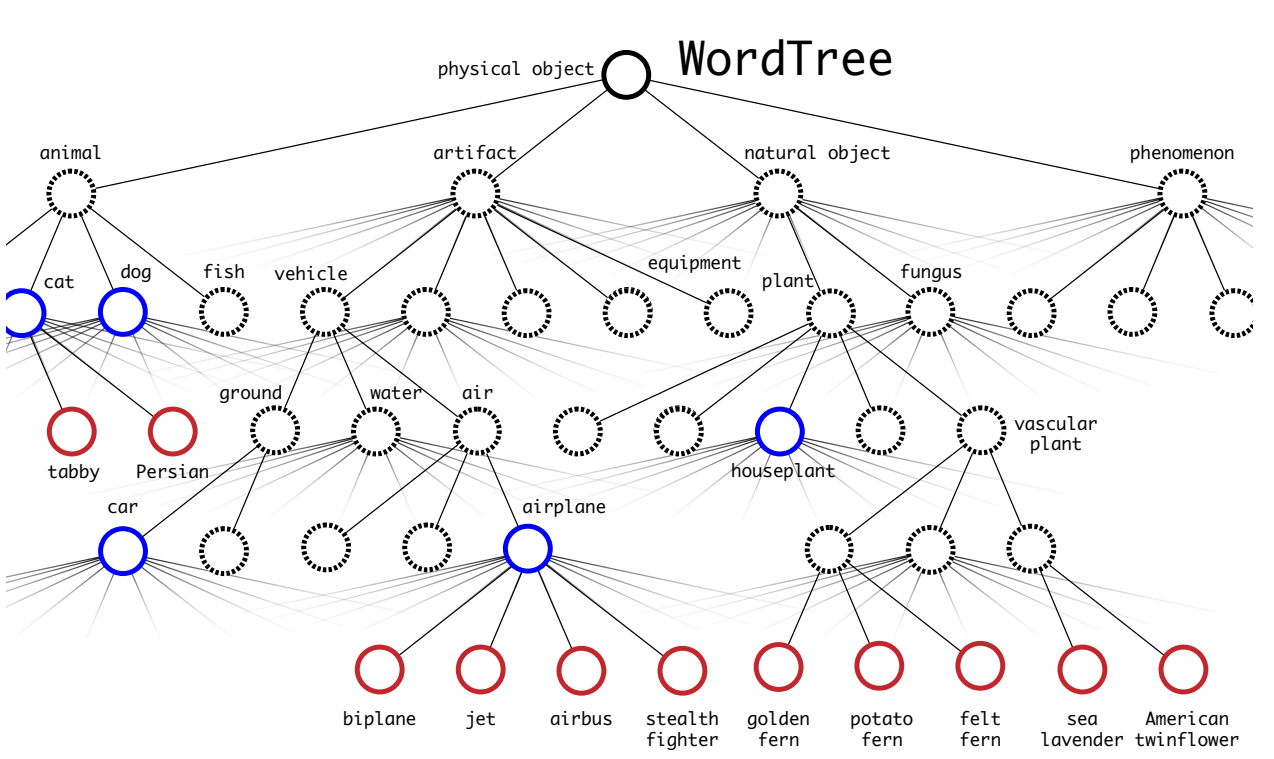
为验证该方法，使用ImageNet图像分类数据集中1000个类别并添加全部中间结点，共1369个类别，建立WordTree1k，用Darknet-19计算条件概率，再算出绝对概率后用对并列下义词做softmax运算，与multi-label做损失，训练模型参数

采用WordTrue后，准确率反而略微下降了，这可能是因为标签相关的物体在视觉上并不相似，视觉上相似的却不在同一分支下

WordTree同样适用于检测，将置信度作为*Pr*(physical object)，向下遍历这棵树，在每个分支处得到置信度最高的结点，并该结点置信度与原有置信度做乘法，直到置信度小于某阈值，预测物体类别

**4.2 Dataset combination with WordTree**





把ImageNet图像分类数据集，80类的COCO目标检测数据集和入未包含进来的ImageNet目标检测数据集中的类别都映射到WordTree，再加入一些中间结点，共9418种类别

**4.3 Joint classification and detection**

形成WordTree后即可开始进行联合训练，使用COCO目标检测数据集和ImageNet数据集中最常见的9000类作为训练集

对用于检测的图像，正常传播损失

对用于分类的图像，找到13×13×3×(5+9418)的输出中提取概率最高的类别，传播相应标签和其上层的分类损失，并且假定对应的 bbox框与ground truth的IoU至少是0.3，并传播损失

验证：

YOLOv2可以检测9000多种类别，故也称为YOLO9000

在200类的ImageNet目标检测数据集上，尽管只有44类是在COCO目标检测数据集中出现过的，mAP也不低

**5损失函数**

负责检测物体的预测框的分类误差

最大IoU小于阈值的预测框的置信度误差

最大IoU大于阈值且负责检测物体的预测框的置信度误差

负责检测物体的预测框的定位误差

Anchor与预测框的位置误差

符号说明：

：将bbox与所有ground truth中心点重合，计算 IoU并取最大值，判断其是否小于Thresh

：是否为前12800次迭代

： bbox是否负责检测物体，每个grid cell最多有一个ground truth**，**由IoU> Thresh且同一grid cell中与ground truth的 IoU最大的Anchor对应的bbox去负责检测物体；IoU> Thresh但非最大的Anchor忽略其损失

：bbox置信度

：bbox中心点坐标和宽高

：bbox类别

：Anchor中心点坐标和宽高

：Anchor与标注框的IoU

：标注框中心点坐标和宽高

：标注框类别