1.Finally we propose a method to jointly train on object detection and classiﬁcation. Using this method we train YOLO9000 simultaneously on the COCO detection dataset and the ImageNet classiﬁcation dataset.

提出一种联合目标定位与分类的训练方法，训练了COCO数据集与ImageNet数据集。

2We also propose a joint training algorithm that allows us to train object detectors on both detection and classiﬁcation data.

Our method leverages labeled detection images to learn to precisely localize objects while it uses classiﬁcation images to increase its vocabulary and robustness

我们还提出了一种联合网络，来同时训练检测数据和分类数据。（检测数据更难获取，考使用分类数据来弥补）

使用带标记的监测数据来定位物体。用分类数据来，提升检测器的样本量和鲁棒性。

3.提出YOLOv2的原因：

YOLO suffers from a variety of shortcomings relative to state-of-the-art detection systems. Error analysis of YOLO compared to Fast R-CNN shows that YOLO makes a signiﬁcant number of localization errors. Furthermore, YOLO has relatively low recall compared to region proposal-based methods. Thus we focus mainly on improving recall and localization while maintaining classiﬁcation accuracy

YOLO存在诸多缺陷，定位误差较大，召回率较低。我们在V2版本主要根据这两个方面进行改进。

4.YOLOv2：

Better、Faster、Stronger

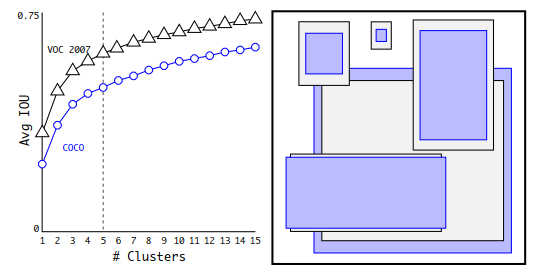
**Better**

**1.****Batch Normalization:**With batch normalization we can remove dropout from the model without overfitting. get more than 2% improvement in mAP.

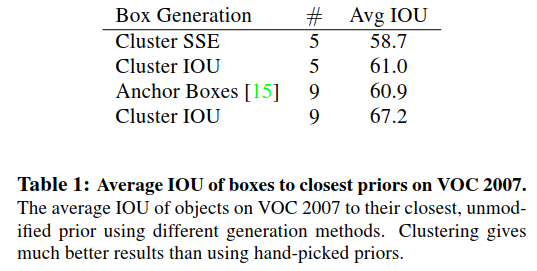
**2.High Resolution Classifier:**For YOLOv2 we ﬁrst ﬁne tune the classiﬁcation network at the full 448 × 448 resolution for 10 epochs on ImageNet.This gives the network time to adjust its ﬁlters to work better on higher resolution input. We then ﬁne tune the resulting network on detection. This high resolution classiﬁcation network gives us an increase of almost 4% mAP.

**3.Convolutional With Anchor Boxes:**①We remove the fully connected layers from YOLO and use anchor boxes to predict bounding boxes.②We also shrink the network to operate on 416 input images instead of 448×448. We do this because we want an odd number of locations in our feature map so there is a single center cell.③Using anchor boxes we get a small decrease in accuracy.YOLO only predicts 98 boxes per image but with anchor boxes our model predicts more than a thousand. Without anchor boxes our intermediate model gets 69.5 mAP with a recall of 81%. With anchor boxes our model gets 69.2 mAP with a recall of 88%. Even though the mAP decreases, the increase in recall means that our model has more room to improve.

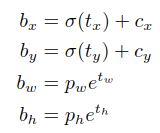
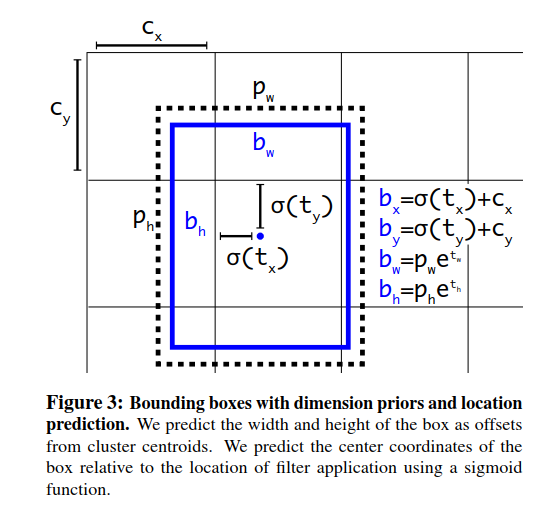
**4.Dimension Clusters:** Instead of choosing priors by hand, we run k-means clustering on the training set bounding boxes to automatically find good priors.



**Clustering box dimensions on VOC and COCO**.We run k-means clustering on the dimensions of bounding boxes to get good priors for our model. The left image shows the average IOU we get with various choices for k. We ﬁnd that k = 5 gives a good tradeoff for recall vs. complexity of the model. The right image shows the relative centroids for VOC and COCO. Both sets of priors favor thinner, taller boxes while COCO has greater variation in size than VOC.



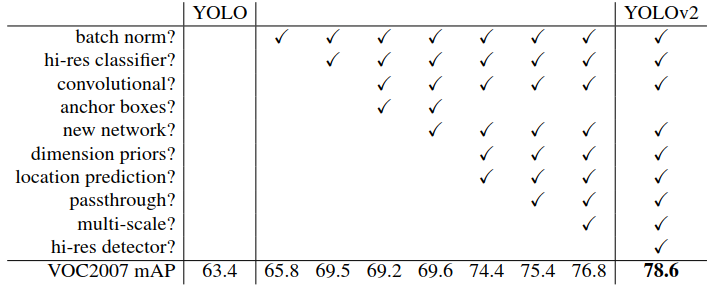
**5.Direct location prediction:**This formulation is unconstrained so any anchor box can end up at any point in the image,regardless of what location predicted the box.With random initialization the model takes a long time to stabilize to predicting sensible offsets.

4和5一起，得到的提升：Using dimension clusters along with directly predicting the bounding box center location improves YOLO by almost 5% over the version with anchor boxes.

**6.Fine-Grained Features(passthrough):**Faster R-CNN and SSD both run their proposal networks at various feature maps in the network to get a range of resolutions. We take a different approach, simply adding a passthrough layer that brings features from an earlier layer at 26 × 26 resolution. This gives a modest 1% performance increase.

**7. Multi-Scale Training:**Instead of fixing the input image size we change the network every few iterations.Every 10 batches our network randomly chooses a new image dimension size.

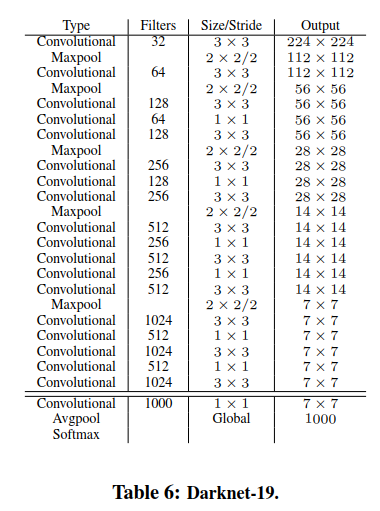


**Faster**

**1.Darknet-19:**We propose a new classification model to be used as the base of YOLOv2.Our model builds off of prior work on network design as well as common knowledge in the eld. The convolutional layers of VGG-16 require 30.69 billion floating point operations for a single pass over a single image at 224 × 224 resolution.This network is faster than VGG-16, only using 8.52 billion operations for a forward pass.

**2. Training for classification**

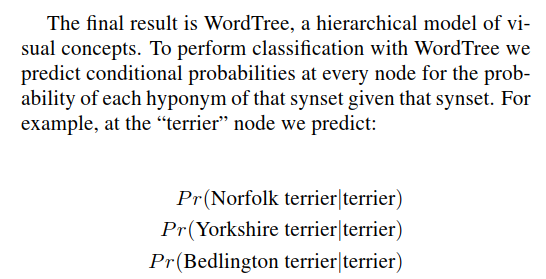
**3. Training for detection:**We modify this network for detection by removing the last convolutional layer and instead adding on three 3 × 3 convolutional layers with 1024 filters each followed by a final 1 × 1 convolutional layer with the number of outputs we need for detection.



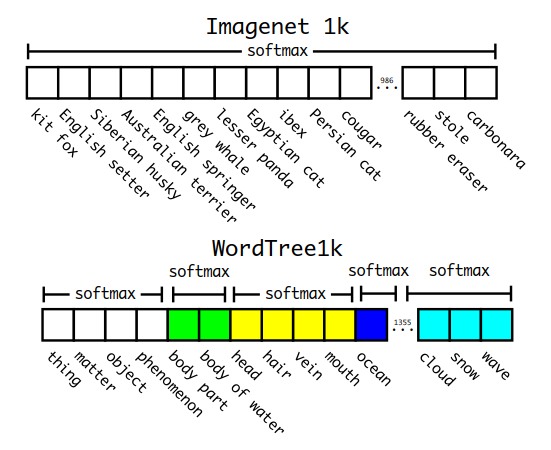
**Stronger**

We propose a mechanism for jointly training on classiﬁcation and detection data. Our method uses images labelled for detection to learn detection-speciﬁc information like bounding box coordinate prediction and objectness as well as how to classify common objects. It uses images with only class labels to expand the number of categories it can detect.

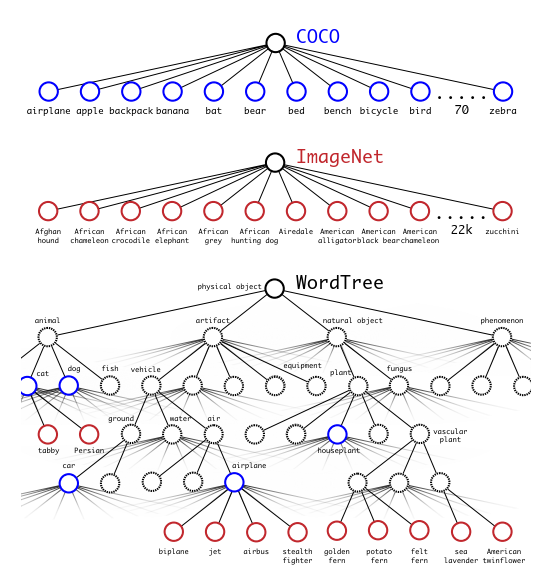
**Hierarchical classiﬁcation**



**Dataset combination with WordTree**

****

**Joint classiﬁcation and detection**

****

①When we analyze YOLO9000’s performance on ImageNet we see it learns new species of animals well but struggles with learning categories like clothing and equipment.

②New animals are easier to learn because the objectness predictions generalize well from the animals in COCO. Conversely, COCO does not have bounding box label for any type of clothing, only for person, so YOLO9000 struggles to model categories like “sunglasses” or “swimming trunks”.