**STUDY ON YOLOv3 ARCHITECTURE:**

**YOLO v3:**

1. YOLOv3 - An Incremental Improvement. This brought the fast YOLOv2 at par with best accuracies. YOLOv3 gives a MAP of 57.9 on COCO dataset for IOU 0.5.
2. The YOLO320 has same accuracy as the RetinaNet with ResNet50 backbone being 4x times faster. This makes YOLOv3 clearly very efficient for any general object detection use-case.

Ref.: <https://pjreddie.com/media/files/papers/YOLOv3.pdf>

**Bounding Box Prediction:**

1. YOLOv3 just like YOLOv2 uses dimension clusters to generate Anchor Boxes. Now as YOLOv3 is a single network the loss for objectiveness and classification needs to be calculated separately but from the same network.
2. YOLOv3 predicts the objectiveness score using logistic regression where 1 means complete overlap of bounding box prior over the ground truth object.
3. It will predict only 1 bonding box prior for one ground truth object (unlike Faster RCNN) and any error in this would incur for both classification as well as detection (objectiveness) loss.
4. There would also be other bounding box priors which would have objectiveness score more than the threshold but less than the best one, for these error will only incur for the detection loss and not for the classification loss.

**Class Predictions:**

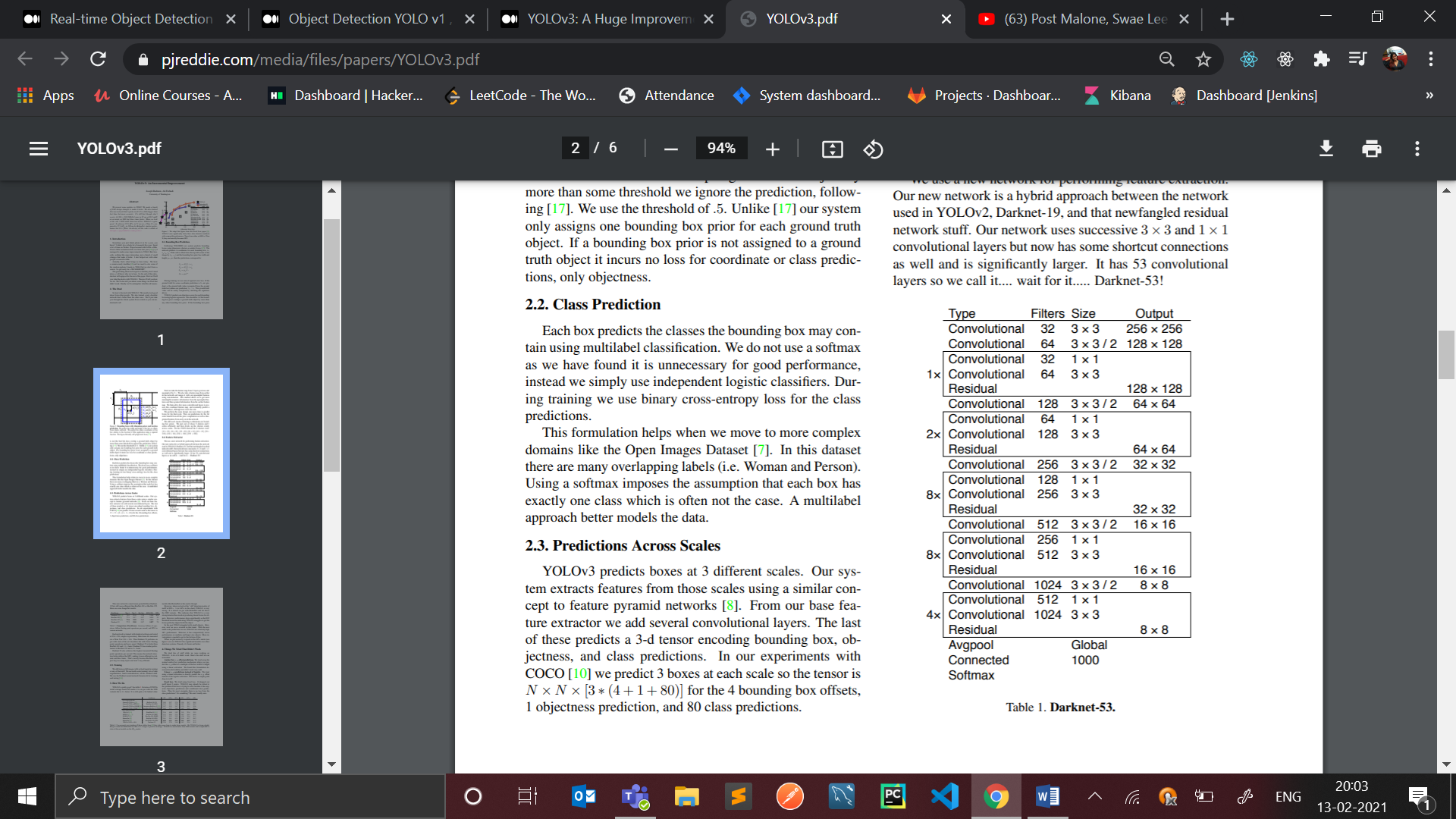
1. YOLOv3 uses independent logistic classifiers for each class instead of a regular softmax layer. This is done to make the classification multi-label classification.
2. What it means and how it adds value? Take an example, where a woman is shown in the picture and the model is trained on both person and woman, having a softmax here will lead to the class probabilities been divided between these 2 classes with say 0.4 and 0.45 probabilities. But independent classifiers solves this issue and gives a yes vs no probability for each class, like what’s the probability that there is a woman in the picture would give 0.8 and what’s the probability that there is a person in the picture would give 0.9 and we can label the object as both person and woman.

**Predictions across scales:**

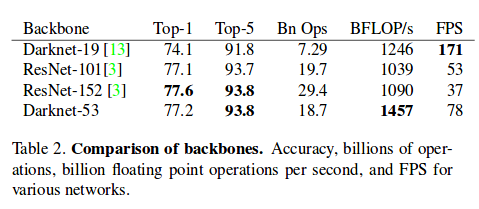
1. YOLOv3 predicts boxes at 3 different scales.
2. It extracts features from those scales using a similar concept to feature pyramid networks. From our base feature extracted we add several convolutional layers. The last of these predicts a 3-d tensor encoding bounding box, objectness, and class predictions.
3. We take the feature map from 2 layers previous and upsample it by 2× . We also take a feature map from earlier in the network and merge it with our upsampled features using element-wise addition. This method allows us to get more meaningful semantic information from the upsampled features and finer-grained information from the earlier feature map. We then add a few more convolutional layers to process this combined feature map, and eventually predict a similar tensor, although now twice the size. We perform the same design one more time to predict boxes for the final scale. Thus our predictions for the 3rd scale benefit from all the prior computation as well as fine-grained features from early on in the network.
4. YOLOv3 gains the ability to better predict at varying scales using the above method. The bounding box priors generated using dimension clusters are divided into 3 scales, so that there are 3 bounding box priors per scale and thus total 9 bounding box priors.

**Feature Extractor:**

1. YOLOv2 used Darknet-19 as its backbone feature extractor, YOLOv3 uses a new network- Darknet-53! Darknet-53 has 53 convolutional layers, its deeper than YOLOv2 and it also has residuals or shortcut connections.
2. Darknet-53 also achieves the highest measured floating point operations per second. This means the network structure better utilizes the GPU, making it more efficient to evaluate and thus faster.







**What improved?**

1. The average precision for small objects improved, it is now better than Faster RCNN but Retinanet is still better in this.
2. As MAP increased localization errors decreased.
3. Predictions at different scales or aspect ratios for same object improved because of the addition of feature pyramid like method. And, MAP increased significantly.