ENEL 645 – Data Mining & Machine Learning

Automated Object Classification in Images with Machine Learning

Project Report

Adam Kerr (10146363)

December 19, 2019

Approach

Two approaches were taken during this project; object classification using cropped annotations and object detection.

Object Classification

The first approach was to use the annotation data to obtain the bounding boxes for each labelled object in the image and use this data to train a convolutional neural network. Instead of building a network from scratch, it seemed more favorable to use an existing network architecture that has been proven to be effective. Keras provides a variety of architectures that have been implemented using the framework's components, as well as pre-trained weights for these models, which are very suitable for transfer learning. From the available architectures, Xception [1] was chosen because of its high performance and relatively low number of parameters which would help in training the model faster.

An analysis of the bounding boxes indicated that the average dimensions of the annotations was approximately 42 and this was used as the input size for the network. All of the annotations, once cropped were then resized to 42x42. Although the aspect ratio is not preserved when resizing the images like this, it was the quickest approach and provided good results regardless of the distortion.

When importing the pre-trained model, the final layers were removed and the input layer modified to match the new input dimensions. A dense hidden layer was added to the model, as well as the final output layer matching the number of classes required. Now the network could be trained on the provided data.

Object Detection

In addition to the above method, an object detection approach was also explored. One of the popular techniques for this is using the "You Only Look Once" (aka YOLO) [2] network architecture. This method is capable of locating and classifying multiple objects within an image. The latest improvement to this model is known as YOLOv3 [3] and was selected for its improvements over its predecessors. Keras unfortunately does not provide any pre-trained object detectors and YOLOv3 was not implemented using the Keras framework. Fortunately, a Keras port of the network does exist [4], and the provided scripts were used to train and evaluate the model on the DIOR dataset. Given the complexity of the model, the training time per epoch was quite long, and although it was trained for as long as possible, it didn't seem that it had completely trained to convergence.

Evaluation

Additional Data

25 additional satellite images were taken from Google and annotated for additional validation. The images contained 193 instances of airplanes, vehicles, and ships in a similar perspective as the DIOR dataset.

Xception

The Xception model performed very well, with a test accuracy of 94%. The confusion matrix indicates that the majority of the errors come from the model classifying vehicles as ships, and to a lesser degree ships as vehicles. On the 193 annotations in the 25 additional images the model is able to accurately classify almost 98% of the images, only misclassifying 4.

YOLOv3

The YOLOv3 model is more difficult to evaluate due to the nature of object detection classification. One standard way of doing this is the mean average precision (mAP) of the model. With an IOU threshold of 0.5 and a minimum classification confidence of 50% the mAP score on the DIOR dataset was 37.02 and on the additional images it was 66.42. Another way of evaluating the model would be to use the recall of contained classes in the images, that is the percentage of images where the network correctly predicts that an instance of the class exists in the image. This recall was 71.3% for airplanes, 78.6% for ships, and 72.4% for vehicles. Examining the predictions on the additional 25 images shows that the model does a relatively good job in detecting and classifying objects, although it does struggle with locating several of the ships as well misclassifying others as vehicles. Nonetheless, it correctly detects at least one instance of each class in all of the images that the class is present.

References

[1] Chollet, F. (2016). *Xception: Deep Learning with Depthwise Separable Convolutions*. [online] arXiv.org. Available at: https://arxiv.org/abs/1610.02357 [Accessed 19 Dec. 2019].

[2] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2015). *You Only Look Once: Unified, Real-Time Object Detection*. [online] arXiv.org. Available at: https://arxiv.org/abs/1506.02640 [Accessed 19 Dec. 2019].

[3] Redmon, J. and Farhadi, A. (2018). *YOLOv3: An Incremental Improvement*. [online] arXiv.org. Available at: https://arxiv.org/abs/1804.02767 [Accessed 19 Dec. 2019].

[4] GitHub. (2018). *experiencor/keras-yolo3*. [online] Available at: https://github.com/experiencor/keras-yolo3 [Accessed 19 Dec. 2019].