Brance AI Applied Researcher Task

Shubham Hazra

May 10, 2023

1 Introduction

The task was to create an AI module that classifies whether a creative has image and text overlap or not. Basically given an image with an object and some text the model should be able to predict if some of the text overlaps with the object.

2 Assumptions Made

So I looked at the test images provided and all of them had a single person and some English text. So based on this I made the following assumptions:

1. The object of significance is a person (Can be multiple people)
2. The text is in only English

3 Approaches

The very first and simple solution that comes to mind is to have a vision model that localizes both the object in question and the text and gives us some sort of an idea as to their locations so that we can easily check if these locations have some overlap and declare the text to be overlapping if it is.
I tried two different methods for this localization. For both the methods only the model and method of localizing the person changes while the text recognition models remains the same. I installed a library called **easyocr**. The reference is here: https://pypi.org/project/easyocr/1.6.2/. The two different methods are:

1. Using object detection to get rectangular bounding boxes for the object
2. Use semantic segmentation to get pixel-wise mask for the object

For the rest of this report I am going to use the following image as the sample image that I tested my models on first:

Figure 1: Sample Image

The bounding boxes that **easyocr** gives on the sample image visualized:

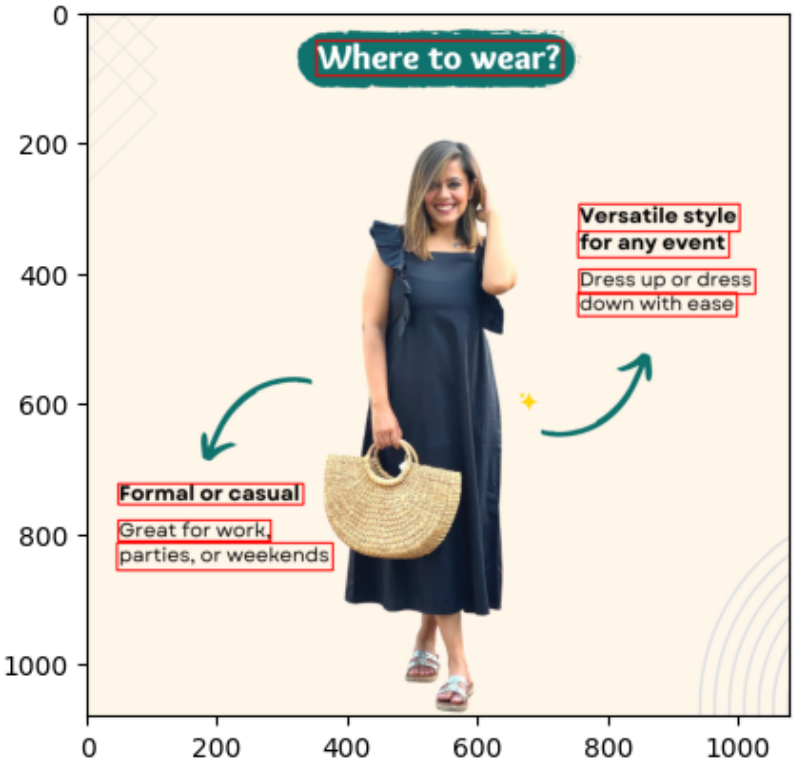

Figure 2: Easyocr predictions

3.1 Method 1

For this I used the **yolov3** object detector to get bounding boxes of the person class. Reference for the code are here and here.
After running the detector on the sample image I get the following output:

Figure 3: YoloV3 prediction

Well after this it was a simple matter of writing a function to check if any of the bounding boxes given by the yolov3 overlapped with that outputted by easyocr. Running the final predict function on all the test images gave the following output:

Figure 4: Predictions on test images

You can already see that it does pretty well on test images. However it gets one wrong i.e It predicts no overlap instead of overlap because the bounding box provided by the yolov3 model does not contain the entire person.
So I tried with bounding boxes 1.3 times bigger than that outputted by the yolov3 model and got the following result:

Figure 5: Predictions on test images with a multiplier of 1.3

This was able to give the right prediction on the image it was failing earlier on however it failed on another image by giving overlap instead of no overlap because of the bigger bounding box.

3.1.1 Drawbacks and Improvements

The very major drawback is that the model returns rectangular bounding boxes which is not able to contain the exact shape of the object, here a person. So text placed very close to the person will give overlaps if it is within the box even if it does not touch the person. So we must look for a better solution.
An improvement can be to use better object detectors that are trained to give entire bounding boxes i.e. bounding boxes that cover the entire object.

3.2 Method 2

For this I used pytorch's **DeepLabV3** segmentation model to get the pixel-wise mask of the person class. Reference for the code are here and here.
However the model gives masks for all the classes that it was trained on but I am only interested in the person class. So I wrote a small function to get on the mask for the person running which on the sample image gives the following output:

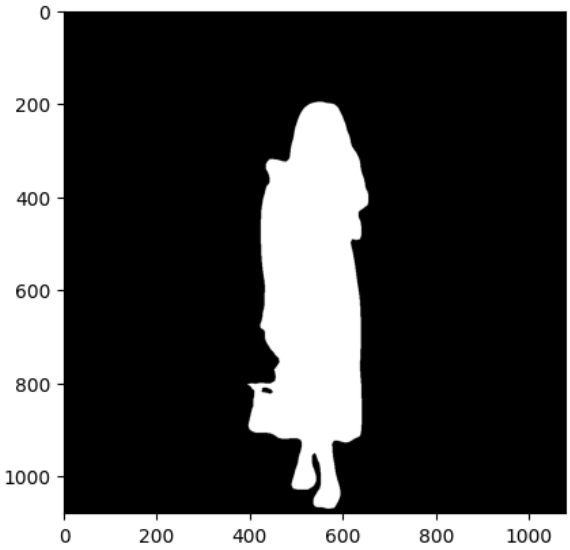

Figure 6: DeepLabV3 prediction

Well after this I wrote a function to check if any of the bounding boxes given by the easyocr overlapped with the mask predicted by DeepLabV3. Running the final predict function on all the test images gave the following output:

Figure 7: Predictions on test images

Lo and behold! It is **100% accurate** on the test images. Visualizing the masks and the text bounding boxes:

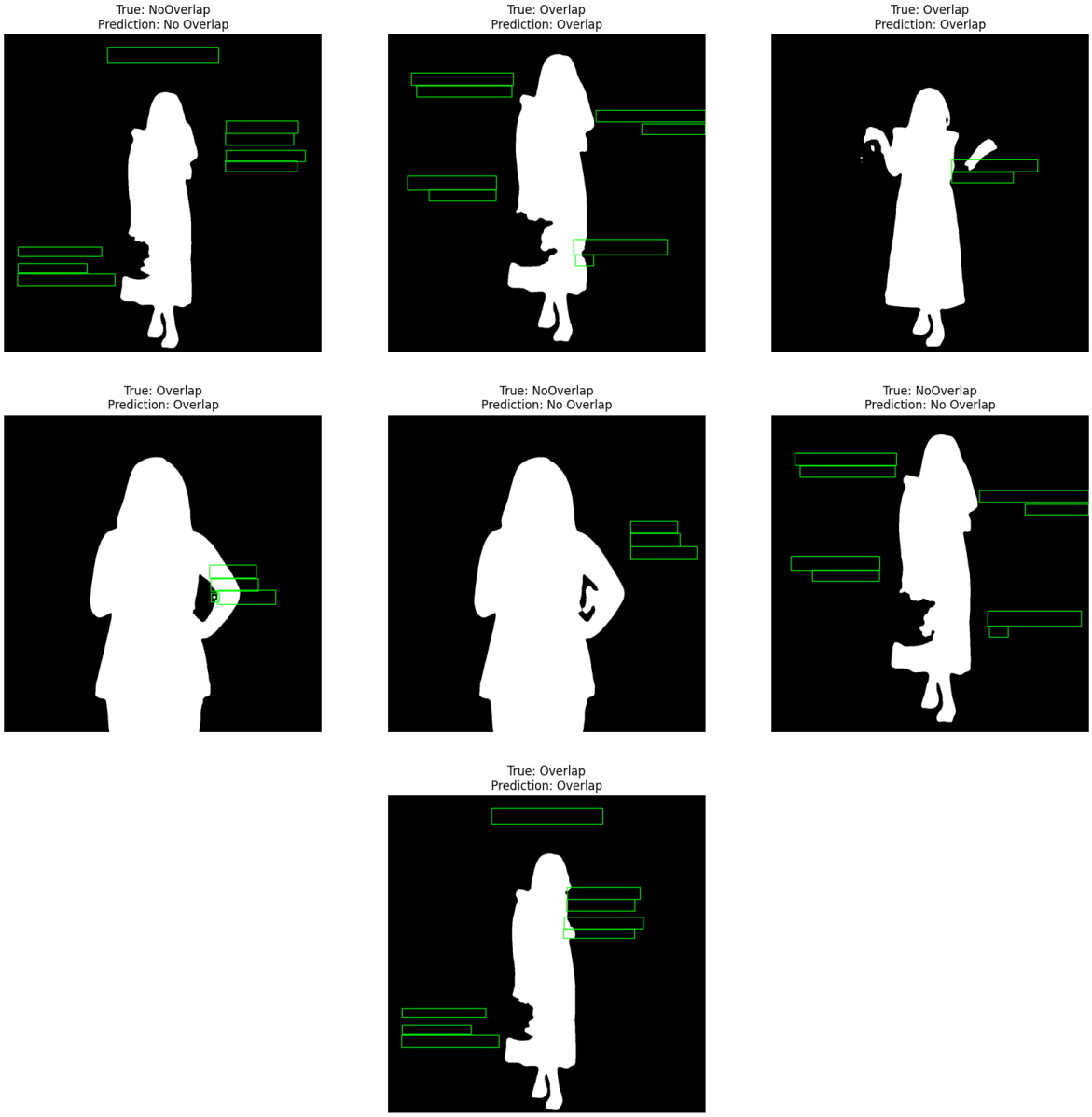

Figure 8: Masks visualized on test images

This is clearly better than method 1 as it does not face the issue of the predicted location of the object being larger than the object itself. i.e It is correctly able it to predict the shape of the onject.

3.2.1 Drawbacks and Improvements

Figure 9: Sample image and the predicted mask

One drawback is that if you look closely in the above image. The predicted mask does not capture the bag that the lady is holding and any text that overlaps with the section of the bag not predicted will not be considered as an overlap.
I only used the mask of the person of the assignment however we can pass in an argument to also get the output mask of the bag and this issue will be resolved.

3.3 Conclusion

The conclusion to draw is that the second method i.e. using segmentation masks is the better method and gives very good results on the test dataset. It can be improved by passing exactly what the items of importance are going to be e.g. person,bag,chair etc. so that we can extract exact masks and provide accurate predictions. Also we should know which language the text is going to be in.