**AI to explore the Brain Tumor dataset**

Yuyang Zhu(yz3691) and Tianle Zhu(tz2434)

Medical Image Analysis

04/14/2020

**Outline**

1. **Recap project**
2. **Implementation and progress of the project** 
   1. **Data preprocessing**
   2. **Object detection - Yolo**
3. **Challenges and state-of-the-art solutions**
4. **Exploration and discussion**

# 1. Recap project

Recently, we changed our objectives for the Medical image analysis project. The main goal of this project now is to explore Brain MRI datasets (include Brain Tumor dataset and Brain MRI Segmentation dataset). We did data segmentation by using UNet in Brain MRI Segmentation dataset, classification by using Vgg16 and ResNet50 in the Brain Tumor dataset. Currently, we explore object detection by using You Only Look Once (YOLO) in Brain MRI datasets during Milestone three.

Brain MRI segmentation dataset contains brain MRI images together with manual FLAIR abnormality segmentation masks. It contains 4568 brain MRI images for 110 patients from the Cancer Imaging Archive (TCIA). The Brain Tumor dataset consists of 155 brain MRI images with a brain tumor and 98 images without a brain tumor. Therefore, we have 4821 brain MRI images in the Brain MRI dataset after combines the Brain Tumor dataset and the Brain MRI Segmentation dataset.

# 2. Implementation and progress of the project

## 2.1 Data Preprocessing:

YOLO requires rectangular masks to do the classification task, so we relabeled the mask in the Brain MRI Segmentation dataset and provided the mask to the Brain tumor dataset. We manually labeled the masks by using Visual Object Tagging Tool (VoTT). For the Brain MRI Segmentation dataset, we labeled the images by approximating it to the original masks( irregular shape) and give it appropriate rectangular contours; For the Brain Tumor dataset, it didn’t provide masks, so we have to label them by manually spotting the abnormal part. The abnormality is usually in a different color and an abnormal shape. In the meantime, we transformed the format of images from .tif to .jpeg in order to meet the requirement of VoTT.

We did data augmentation for both datasets, which includes rotation up to 40 degrees, width and height shift up to 20%, a slight shearing, slight zooming, and horizontal flip. We created one extra image for each image in the dataset. We also tried filtering the Brain MRI Segmentation data before putting them into the training set.

In the Brain MRI Segmentation dataset, we have brain MRI images for 110 patients. Each patient has a sequence of scan images. Images for each patient all start with their skull image and end with images from the top of the head, which rarely contain any useful masks in them, and it is going to deteriorate the model’s training process. Thus, it is reasonable to rule out these images in the training set. Basically, we ruled out the first 15% and the last 15% image for each patient. This is an approximate threshold (obstacle) because the pictures are continuous and we cannot confidently say which part should be excluded.

## 

## 2.2 Object detection -- YOLO

YOLO trains the network by using a model with 24 convolutional layers, followed by two fully connected layers. The last convolution layer outputs a tensor with shape (7, 7, 1024) and then the tensor is flattened. It will train for two stages. On the first stage-trained in large batch size, while on a second stage with smaller batch size, with each about 50 epochs. As a result, it is incredibly painful to train it on a large dataset. Thus, we did the mini sample test using the Brain Tumor dataset (300 images) and then trained it on the Brain Segmentation MRI dataset. We initialized the model with pre-trained darknet weights.

# 3. Challenges and Solutions

## 3.1 Challenges

YOLO is a user-oriented program and the training process is already packaged. So, there are only minor issues we need to adjust such as changing the input and output suffix.

There are issues when we are labeling images. We manually labeled these images, and some parts of it might be erroneous (because some masks can barely be seen in the image). we could take a closer look by drawing an outline of the mask in the image, as what we did in the UNet segmentation, or we could discard them.

YOLO imposes strong spatial constraints on bounding box predictions since each grid cell only predicts two boxes and can only have one class. YOLO model struggles with small objects that appear in groups such as tiny brain tumor cannot be detected. Since our model learns to predict bounding boxes from data, it struggles to generalize to objects in new ratios.

3.2 Solutions

After we did the object detection on the Brain MRI Segmentation dataset by using YOLO, we decided to compare YOLO outputs with UNet outputs. From the loss function plot, we obtained that YOLO uses more flexible learning rates and batch sizes, which reduces the loss function in a smoother way.

We did a manual scoring for the accuracy of test results (whether the prediction is correct). As a result, UNet has a 91.125% accuracy on the test set(160 images), and YOLO has a 90.625% accuracy, which is a very similar result.

In the images U-net or YOLO failed to predict correctly, most of them are having very tiny or indistinguishable labels so both U-net and YOLO went wrong, and other than that U-net is most likely to make false negative predictions, while YOLO is more likely to make false-positive predictions and rarely have false negatives(1 in 160)

Another thing we are trying to do to evaluate the models is to compare the generalization results, currently, we only have a Brain tumor and Brain MRI dataset, so what we are trying to do might be using a model trained on Brain MRI dataset to predict the Brain tumor dataset, and evaluate the performance of the two models.

We trained the current model on Brain MRI Segmentation dataset, but the Brain tumor dataset actually shares very similar features with the Brain MRI Segmentation dataset, so whether the model can be generalized to a similar dataset could be a good measurement of the model goodness.UNet performs poorly on the generalized dataset. It only predicts 20% percent of the abnormal parts and has about 30% of the false-positive rate. It looks like it fails to find features like abnormal distensions in the data.

YOLO does better than UNet when we are generalizing the dataset. However, it still exists a false-positive rate, but it has a much higher true positive rate than U-net. Which shows that YOLO is a more generalized model.

# 4. Exploration and Discussion

Since our YOLO model struggles with small objects such as cannot detect minor brain tumor, we explore other detection systems to deal with this problem. After searching the key similarities and differences between YOLO and several top detection frameworks, we find that R-CNN can help us to remedy this shortage.

R-CNN and YOLO have some similarities. Each grid cell proposes potential bounding boxes and uses the convolutional feature to score those boxes. Comparing to YOLO, R-CNN has a shortage in speed such as the resulting system is very slow which taking more than 40 seconds per image.

R-CNN uses region proposals instead of sliding windows to find objects in images. It extracts about the 2000 bottom\_up region proposals. Then, it uses the convolutional network to extract features for each proposal. After that, it classifies each region using class-specific linear SVMs. Finally, a linear model adjusts the bounding boxes, and non-max suppression eliminates duplicate detections. The differences might deal with the problem that the YOLO model struggles with small objects.