# Ensemble Learning in U-net

Alireza Shafaei Darestani *

University of Guilan
Alireza.Shafaei97@gmail.com

August 24, 2020

**Abstract**

Nowadays, the AI and the medical community are trying to solve important problems to help healthcare getting better and better. Unfortunately, there is a lack of data, which is vital for achieving better results; therefore, a new data generator has been designed that helps the model don't overfit. Even with a powerful network, the Dice coefficient value for the validation set is always more than the training set for any epoch in this medical task despite rare data.

Due to google colab time limitation, it is impossible to run this model continuously for more than 2 or 3 epochs. To address this issue, we saved the model after each epoch, and the saved model began the learning phase again with another google colab account. Despite this problem, the model dice coefficient and soft dice loss value show 0.97 and 0.01 respectively in the validation set after 20 epochs.

*Thanks to Pouya Pournasir and Dr. M. Aminian

# 1 Introduction

This is the first section.

U-net[1] is one of the best methods for segmentation tasks in medical imaging; however, as it gets deeper(1) the chance of overfitting grows. Not only memory usage is an important issue in tumor segmentation tasks with huge input shapes but also If we reduce the input shapes for memory purposes we may lose many important features and edges. We can tackle these two problems with the methods that are introduced in this short paper. For both segmentation and survival tasks, the model's data generator played a key role in solving the overfitting problem. In tumor segmentation tasks, the model must be generalized and the focus should be on the tumor not the brain's edges by the way. if the model pays attention to the tumor it can make better predictions.

# 2 Proposed Method

## 2.1 Segmentation Task

Preprocessing is always the best way to help the model learn better. In this step for the segmentation task, we sorted images by t1, t1c, t2, flair order to avoid model confusing. Because the input shapes are huge and unsupportable for memory, we defined a function to get subvolume from the main MR images that were inspired by AI in medical diagnosis[2], just like many other ideas behind this project. the shapes were 160, 160, 16, 4 but it is still unsupportable for colab's memory space. we decreased the size of the shapes by 120, 120, 16, 4. The difference is that in the training phase, the generator pass how many random slices we need per one training sample to the model. For example, it can generate 10 or more slices with 120, 120, 16, 4 shapes from one MR image. if the MR images are 369 samples, this model can learn with 3690 or more training data and will never get overfit. Now that the model can't get overfit we are able to use a powerful model. U-net is a powerful model for segmentation tasks but it can get better. We used U-net as the base model with one more argument named 'repeat'. The value for this argument can be a natural number like X and it can combine X U-nets together to get more powerful. It seems that U-nets in this method are having a competition with each other to make more changes in the final result after each batch. The U-net that gets better results will be more powerful in the next batch.

## 2.2 Survival Task

In the survival task, the main idea is that we should train the model based on the most important part of the images. By using what the segmentation model has predicted with the MRI input, the model learns how to predict the chance of survival of the patient. To clarify, the survival model takes an 80,80,64 shaped input cube, which is the main tumor predicted by the segmentation model. In the pre-processing part, we analyze the whole segmentation label and extract 10 or more cubes, which have the maximum sum of the voxels. Then, the generator chooses one of them as the model input. Same as the segmentation task, the training set is not constant and changes by epochs, so it is impossible to get overfit. We pass this input to a res-net50 and the output layer connected to a dense-net.

# 3 Results

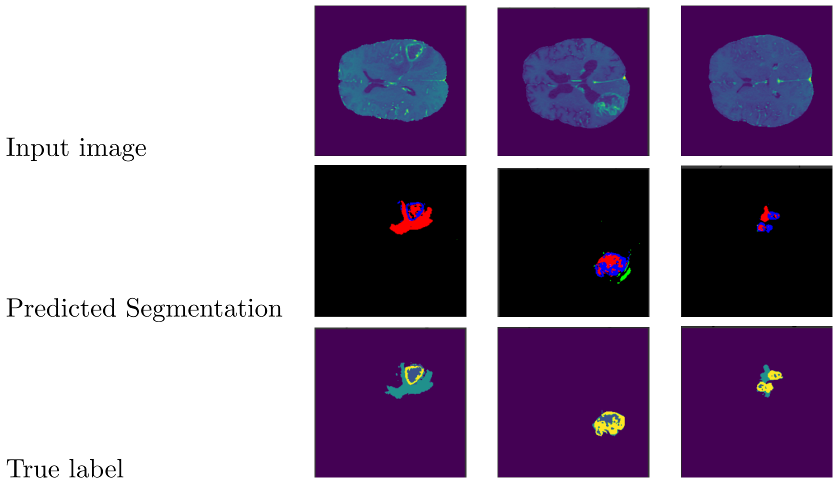


Figure 1: Labeled Data Prediction

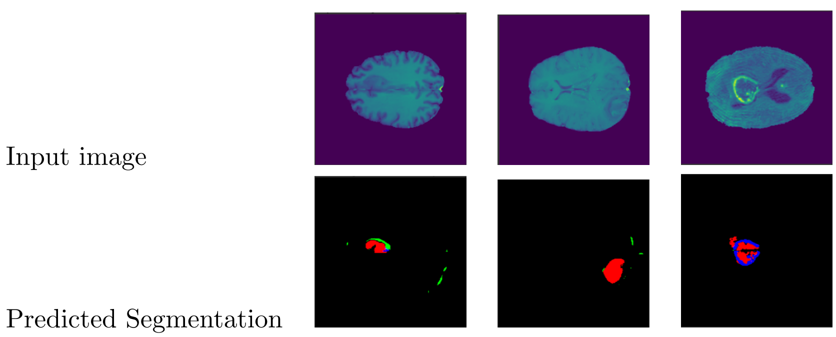


Figure 2: Unlabeled Data Prediction

# 4 References

[1]U-Net: Convolutional Networks for Biomedical Image Segmentation Olaf Ronneberger, Philipp Fischer, Thomas Brox

[2]Coursera AI for Medicine pranav rajpurkar, Andrew Ng