Brain tumor segmentation with U-Net and FC-DenseNets

Junchol Park
parkj.learning@gmail.com
https://github.com/parkjlearning/braintumorsegmentation
Springboard Mentor: Dipanjan Sarkar

Abstract

Semantic image segmentation predicts class labels for each pixel in the image. In medicine, semantic segmentation is widely used for medical image diagnostics, where machines can augment analysis performed by radiologists with efficiency and scalability. One active area of research is brain tumor segmentation, where deep neural networks are trained to predict where each pixel falls into tumorous tissue classes that were labeled by neuroradiologists. In this work we used a data set comprising clinically-acquired pre-operative multimodal MRI scans of glioblastoma and lower grade glioma, common types of brain tumors. This data set was preprocessed and labeled by neuroradiologists for BraTS 2020 competiton. We trained two state-of-the-art deep neural network models for image segmentation, U-net and FC-DeepNets and evaluated their performance.

Table of Contents

Introduction

What are in this image, and where in the image are they located? Semantic image segmentation answers both questions (**Fig. 1**). It is one of the key applications in image processing and computer vision, which has been widely used in numerous areas such as intelligent transportation and medical diagnostics. It takes RGB ($height \times width \times 3$) or grayscale (($height \times width \times 1$)) images and generates a segmentation map where each pixel contains a class label as an integer (**Fig. 2**). Since classification is conducted at the pixel level, segmentation is distinguished from the image-level classification.

Image segmentation is a key component of biomedical image analysis, where the desired output should include localization, i.e., a class label is supposed to be assigned to each pixel. Such applications range from labeling cell components (e.g. nucleus, cytoplasm, mitochondria etc.) in serial electron microscopic (EM) images to labeling pathological (e.g. tumor) tissues in MRI scans. One active research area focuses on brain tumor segmentation. Gliomas are the most common primary central nervous system malignancies. These tumors exhibit highly variable clinical prognosis, and usually contain heterogeneous subregions (e.g., edema (swelling in the periphery), enhancing and nonenhancing cores) [1]. To diagnose tumors semantic image segmentation can be applied to identify and localize tumorous subregions in MRI scans or volumes, which can augment analysis by significantly reducing time and labor required to run diagnostic tests.

State-of-the-art approaches for semantic image segmentation are powered by Convolutional Neural Networks (CNNs). One important challenge therein is a trade-off between localization and the use of context. For representation of context larger patches that require more max-pooling layers are used, and this reduces the localization accuracy while small patches allow the network to see only little context. More recent approaches proposed a classifier output that takes into account the features from multiple layers. Likewise, the main idea in a “fully convolutional network” [2] is to supplement a usual contracting network by successive layers, where pooling operators are replaced by upsampling operators. Hence, these layers increase the resolution of the output. For localization, high resolution features from the downsampling path are combined with the upsampled output (i.e. skip connections). A successive

convolution layer can then learn to assemble a more precise outcome based on the combined information. “U-Net” builds upon this with an important modification to have the upsampling part comprise a large number of feature channels, which allow the network to propagate context information to higher resolution layers [3]. As a consequence, the expansive path becomes more or less symmetric to the contracting path, and yields a u-shaped architecture (**Fig. 8**). With additional improvements in overlap-tiling and data augmentation U-nets excelled in various biomedical image segmentations like cell-tracking within EM stacks and labeling body parts in a 3D volume. Thus, here we first utilize a U-Net architecture to train a model for brain tumor segmentation using the data prepared for the BraTS 2020 competition. We then train another model using a more recently developed architecture, “fully convolutional DenseNets” (FC DenseNets), and compare performance of the two models.

Figure 1. Semantic image segmentation of street view in intelligent transportation.

Image credit: Semantic segmentation with deep learning by George Seif

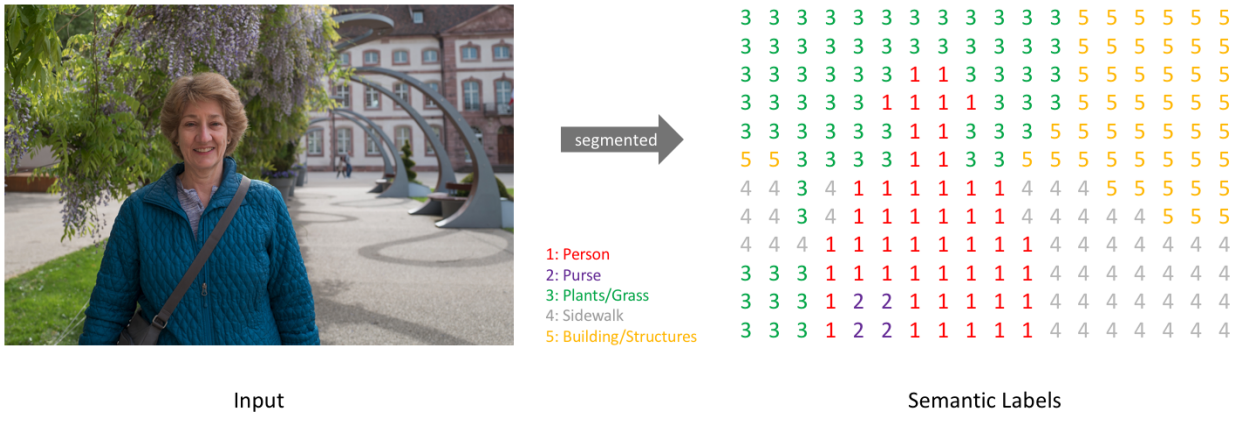

Figure 2. An example segmentation map.

An example segmentation map where each pixel contains a class label as an integer. For visual clarity, a fictive low-resolution feature map (right) was created for an example image (left).

Image credit: semantic-segmentation by Jeremy Jordan

Dataset

The MRI data are collected/preprocessed by the BraTS community for 2020 BraTS challenge (www.med.upenn.edu/cbica/brats2020). The dataset consists of 369 preoperative multimodal MRI scans of glioblastoma and lower grade glioma with pathologically confirmed diagnosis with accompanying ground truth labels by expert board-certified neuroradiologists. Each set comprises multimodal scans available as NIfTI files (.nii):

1. T1: T1-weighted, native image, sagittal or axial 2D acquisitions, with 1–6 mm slice thickness.
2. T1ce: T1-weighted, contrast-enhanced (Gadolinium) image, with 3D acquisition and 1 mm isotropic voxel size for most patients.
3. T2: T2-weighted image, axial 2D acquisition, with 2–6 mm slice thickness.
4. FLAIR: T2-weighted FLAIR image, axial, coronal, or sagittal 2D acquisitions, 2-6 mm slice thickness.

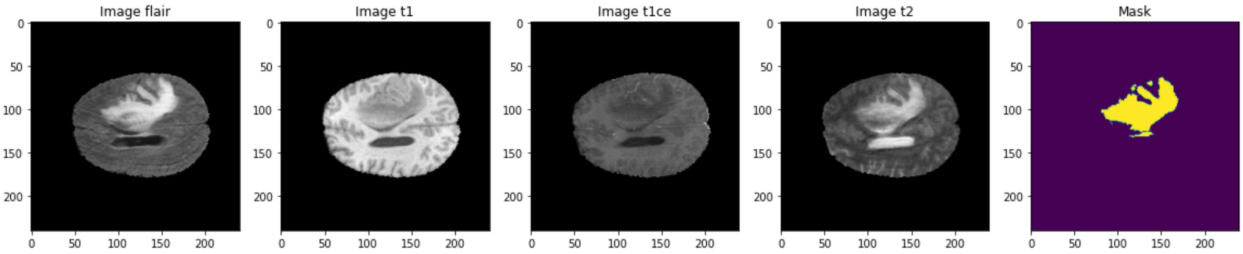

Figure 3. Example multimodal scans (Flair, T1, T1ce, T2) and the corresponding annotation (mask).

The dimension of each volume/modality is 240 (height) x 240 (width) x 155 (slices). Annotations (masks) are provided as a separate volume ('seg.nii' files) of the same dimensionality. Annotation values comprise the GD-enhancing tumor (ET - label 3), the peritumoral edema (ED - label 2), and the necrotic and non-enhancing tumor core (NCR/NET - label 1), and thus the segmentation would predict each pixel into four classes (**Fig. 4**):

- 'NOT tumor' (0)
- 'NECROTIC/CORE' (1)
- 'EDEMA' (2)
- 'ENHANCING' (3)

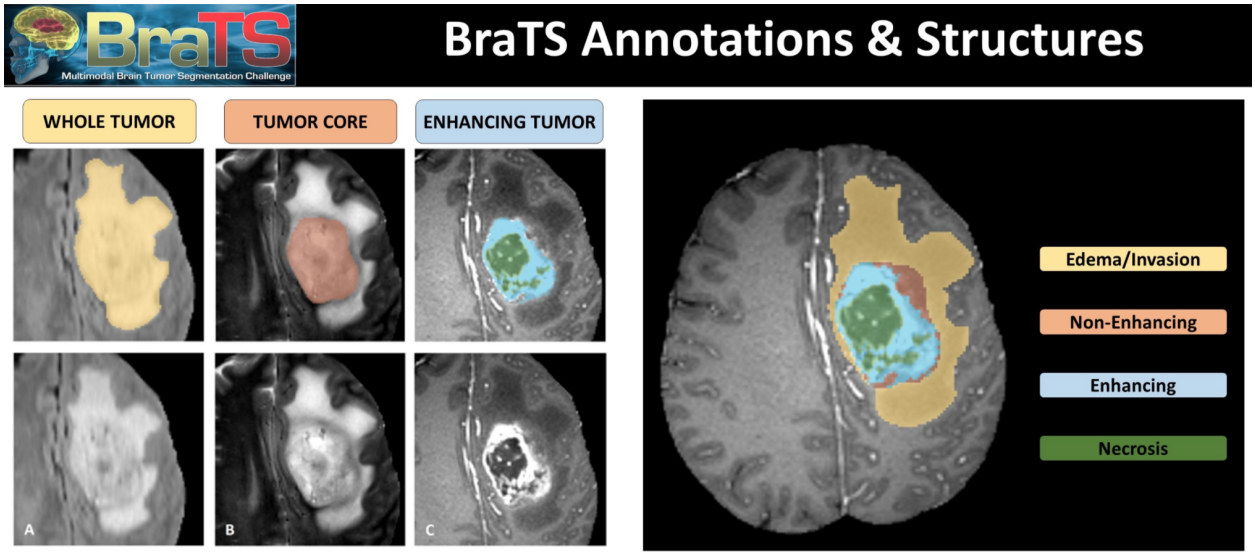

Figure 4. Annotations of the tumorous subregions.

Image credit: braTS 2020

Train, validation and test sets

The dataset was split into train (0.68), validation (0.12), and test (0.2) sets. The validation set was used for cross validation during the training. The trained model was then evaluated on the test set.

Features and target

MRI scans of two modalities (Flair, T1ce) were used as features to minimize redundancy. 100 out of 155 slices per volume were used for training. Slices were resized to 128 (height) x 128 (width) or 64 (height) x 64 (width) pixels for efficiency in computation. The dimension of input ('X') was (batch size x the number of slices) x 128 (height) x 128 (width) x 2 (modalities). The corresponding target ('Y') for X was constructed by applying one-hot encoding onto the annotation mask (**Fig. 3**). The dimension of target was (batch size x the number of slices) x 128 (height) x 128 (width) x 4 (classes).

Exploratory data analysis (EDA)

To better understand the dataset we conducted EDA and visualized the data in diverse ways. First, we visualize a volume (flair) of which 155 slices are rendered at different orientations (**Fig. 5**). Note that the tumorous subregions in the left hemisphere appear to be brighter with their higher pixel values. To emphasize differences across classes we snapshot 2D planes where tumorous subregions are widespread (**Fig. 6**). Our target comprises volumes with same dimensions that contain integer-valued annotation classes. We took snapshots of these labeled images, and confirmed pixel-by-pixel labeling of classes (**Fig. 7**).

Figure 5. Visualization (GIF) of 3D MRI volumes with different orientations.
Left: Sagittal slices, Center: Coronal slices, Right: Horizontal slices.

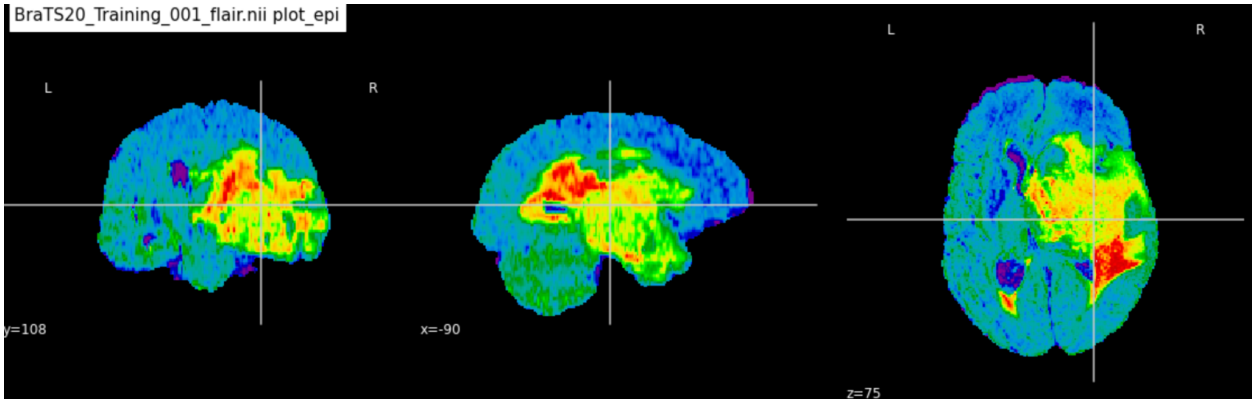

Figure 6. Visualization of a flair volume in coronal (left), sagittal (center), and horizontal (right) views.

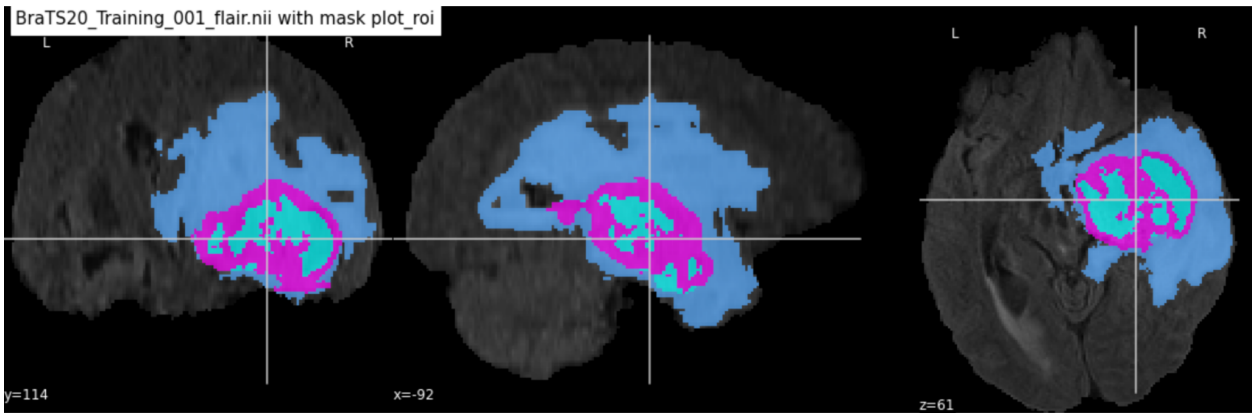

Figure 7. Visualization of annotated classes rendered on the volume.

Magenta: NECROTIC/CORE, Cyan: ENHANCING CORE, Light blue: EDEMA, Gray: non-tumor.

U-Net model

Architecture

A conventional U-Net architecture was used for our model. A U-Net has symmetrical downsampling and upsampling paths (**Fig. 8**). Each downsampling block comprises a max pooling layer followed by two convolution layers doubling the number of feature maps of the previous block. Thus, more features and contextual information are learned through the downsampling/contracting path. The upsampling path serves a complementary process that comprises upsampling and two convolution layers, through which context information is propagated. Importantly, each upsampling block receives a skip connection from the resolution-matching downsampling block, which facilitates recovery of higher resolutions. The final output layer utilizes softmax activation for probabilistic multi-class classification. The full details of the U-Net architecture can be found here: U-Net_model.

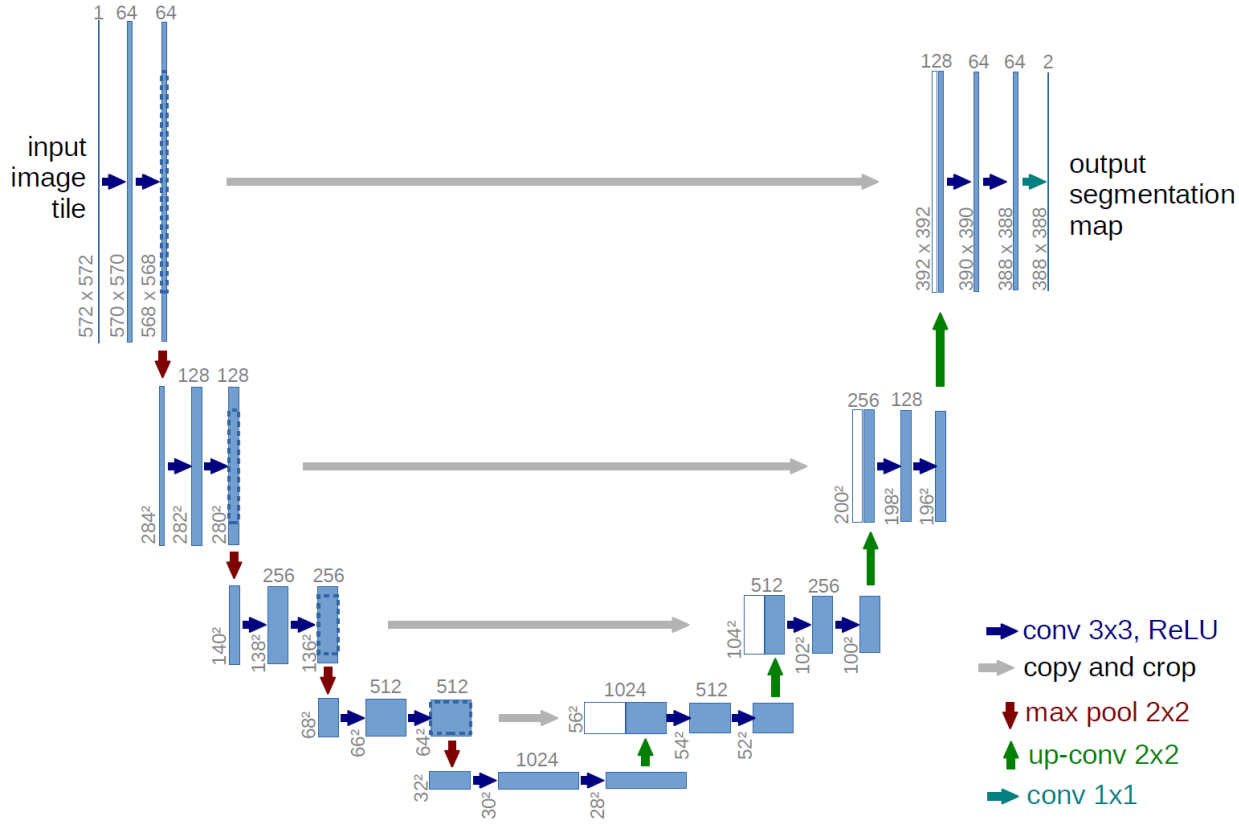

Figure 8. U-Net architecture.

Model training

The input images and their corresponding segmentation maps are used to train the network with the stochastic gradient descent. Pixel-wise cross entropy loss function was used, which is calculated as the log

loss summed over all possible classes. $-\sum_{classes} y_{true} log(y_{pred})$. This scoring is conducted over all pixels and averaged. Adam was used as the optimization algorithm. The learning rate was set to be adjusted monitoring the loss calculated on the validation set, namely it was reduced as a factor of 0.2 in the absence of improvement in validation loss for two epochs. Training proceeded over 100 epochs. The model training data show convergence of loss function over training and validation data, and the evaluation metrics (accuracy, dice coefficient, and IoU) improved over the course of training epochs (**Fig. 9**).

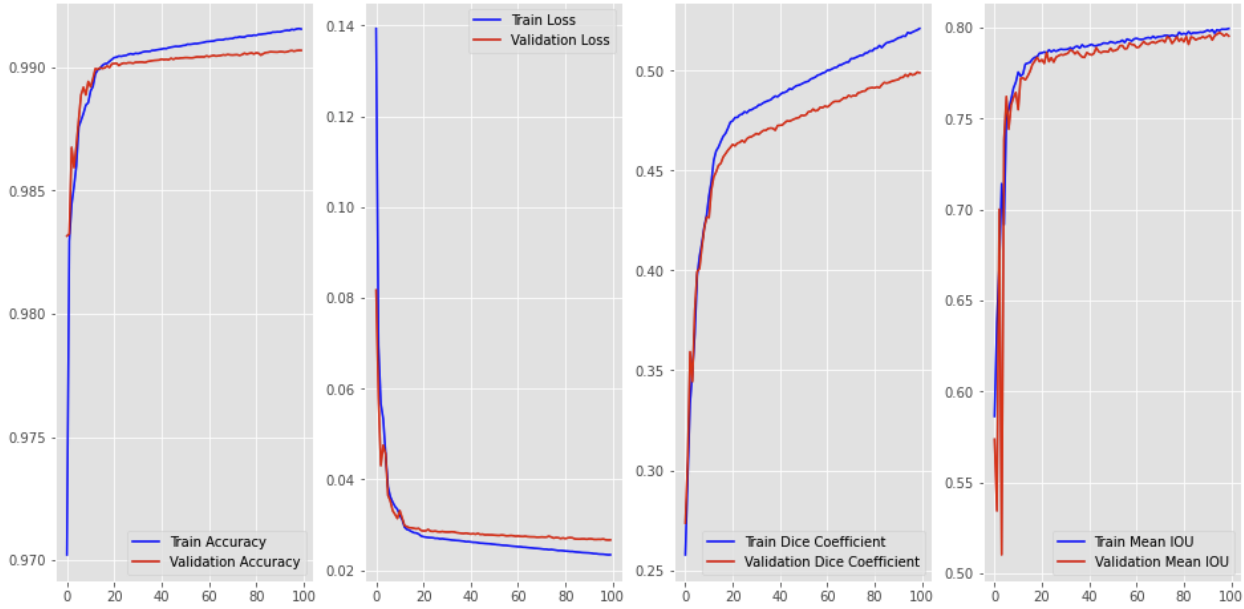

Figure 9. U-Net training log.

From left to right, accuracy, loss, dice coefficient, and IoU are plotted versus the train epoch.

Model evaluation

To assess model performance, we used the held out test set and visualized model predictions and ground truths in tandem for each class and observed that they largely overlapped (**Fig. 10**). We then quantified model performance using diverse metrics (**Fig. 11**).

- Accuracy: The percent of pixels that are correctly classified.
- Precision: The ratio of true positives to all predicted positives (true positives + false positives).
- Sensitivity: The ratio of true positives to all possible positives (true positives + false negatives).
- Specificity: The ratio of true negatives to all possible negatives (true negatives + false positives).

These four metrics are more susceptible to the class imbalance issue, i.e. the vast majority of pixels belong to the non-tumor class which may lead to exaggerated scores. In such cases, the two alternatives below provide more appropriate metrics.

- Intersection-over-Union (IoU): The ratio between the area of overlap and the area of union.
- Dice coefficient (F1 score): 2 * The Area of Overlap divided by the sum of the total number of pixels.

To examine model performance for each class, we also calculated dice coefficients per class.

- Dice coefficient NECROTIC/CORE
- Dice coefficient EDEMA
- Dice coefficient ENHANCING

Dice coefficients for each class provide more stringent evaluation of the model that is unbiased by the class imbalance issue. Segmentation of NECROTIC/CORE turned out to be more challenging perhaps due to their scarcity (**Fig. 11, right**). Model performed better for segmentation of ENHANCING and EDEMA subregions (**Fig. 11, right**).

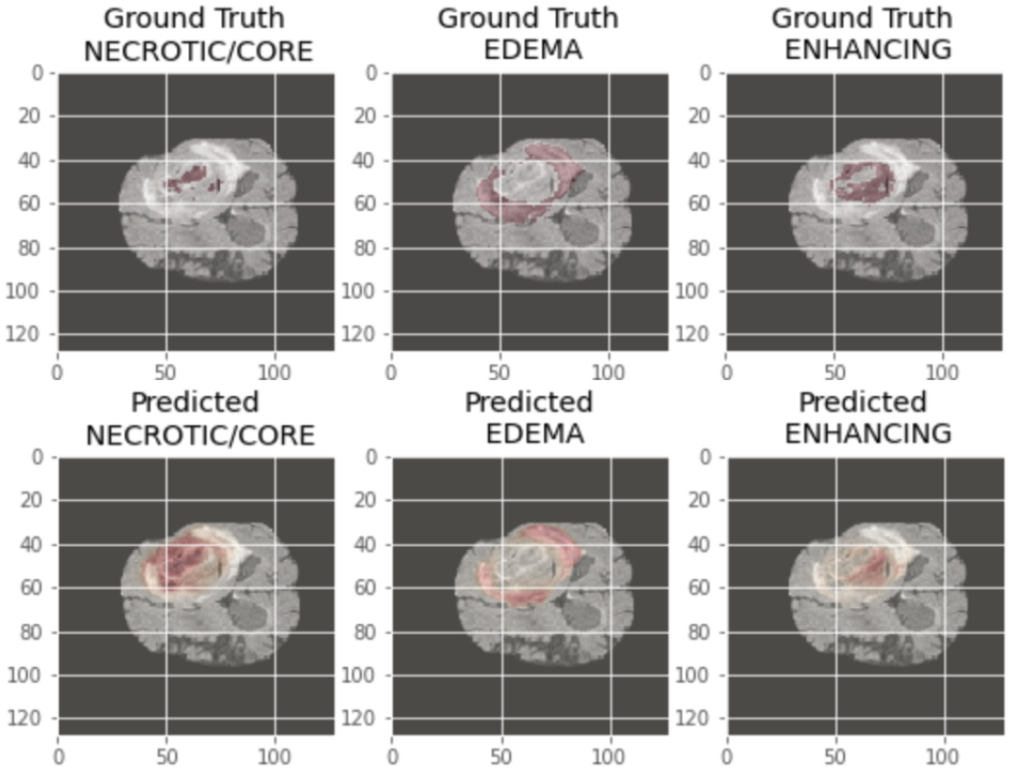

Figure 10. Visualization of ground truths (top row) and model predictions (bottom row).

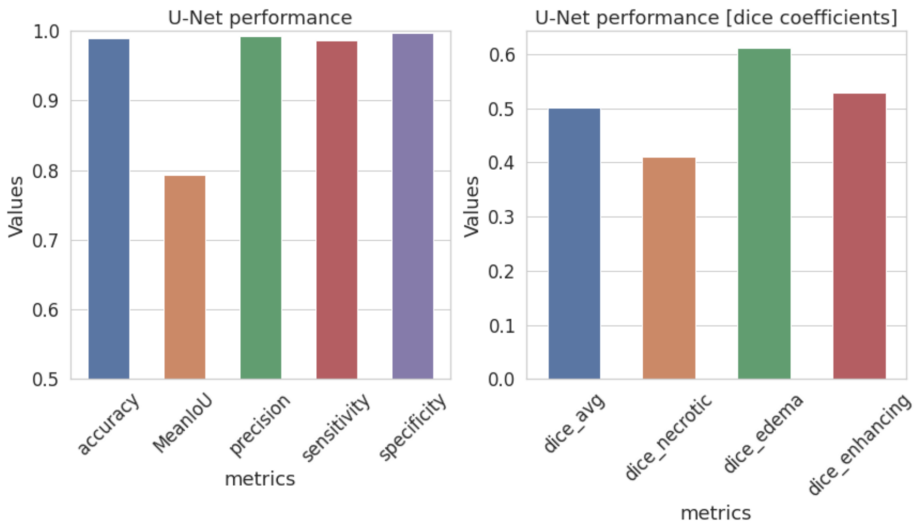

Figure 11. U-Net performance on diverse metrics.

Gross metrics in the left plot were averaged across all classes. On the right, dice coefficients (F1 score) were calculated for each class to compare model performance across different classes.

FC-DenseNets model

Architecture

The architecture of the full-convolutional DenseNets (FC-DenseNets) is similar to that of the U-Net in the sense that it comprises downsampling (contracting) and upsampling (expanding) paths as well as the skip connections from the downsampling to the upsampling path (**Fig. 12**) [4]. The downsampling path incorporated three dense blocks (**Fig. 12,** right), in which all feature outputs are iteratively concatenated in a feedforward fashion to facilitate the feature reuse. Each dense block comprises several repetitions of a series that consists of batch normalization, activation, convolution and dropout. The last layer of the downsampling path is referred to as a bottleneck which is a dense block.

The upsampling path comprises convolution, upsampling operations (transposed convolutions), and skip connections. Transition up modules upsample the previous feature maps, which are then concatenated to the ones coming from the skip connection to form the input of a new dense block. The last dense block summarizes the information contained in all the previous dense blocks at the same resolution. Note that some information from earlier dense blocks is lost in the transition down due to the pooling operation. Nevertheless, this information is available in the downsampling path of the network and can be passed via skip connections. Hence, the dense blocks of the upsampling path are computed using all the available feature maps at a given resolution, leading to a very deep FC-DenseNets.

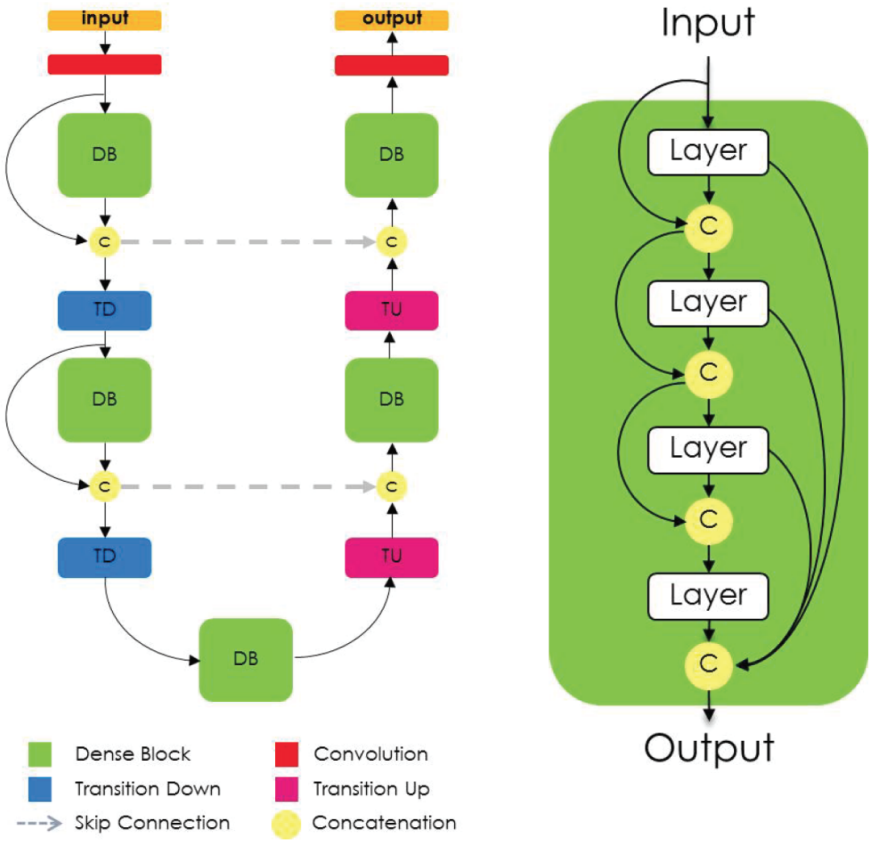

Figure 12. FC-DenseNets architecture.

Model training

The training scheme used for U-Net was applied to train the FC-DenseNets. However, FC-DenseNets was very deep with much more feature maps than the U-Net, and we faced an out-of-memory issue while using the same GPU (NVIDIA TESLA P100) with 16GB memory. Thus, we had to further downsample the images to 64 (height) x 64 (width) pixels. After resizing we could avoid the memory issue. The training log showed that the train loss converged with improvements in evaluation metrics (accuracy, dice coefficient, IoU) on the train set. However, unlike the case of the U-Net, we observed neither the convergence of loss nor improvements in model evaluation metrics on the validation set (**Fig. 13**).

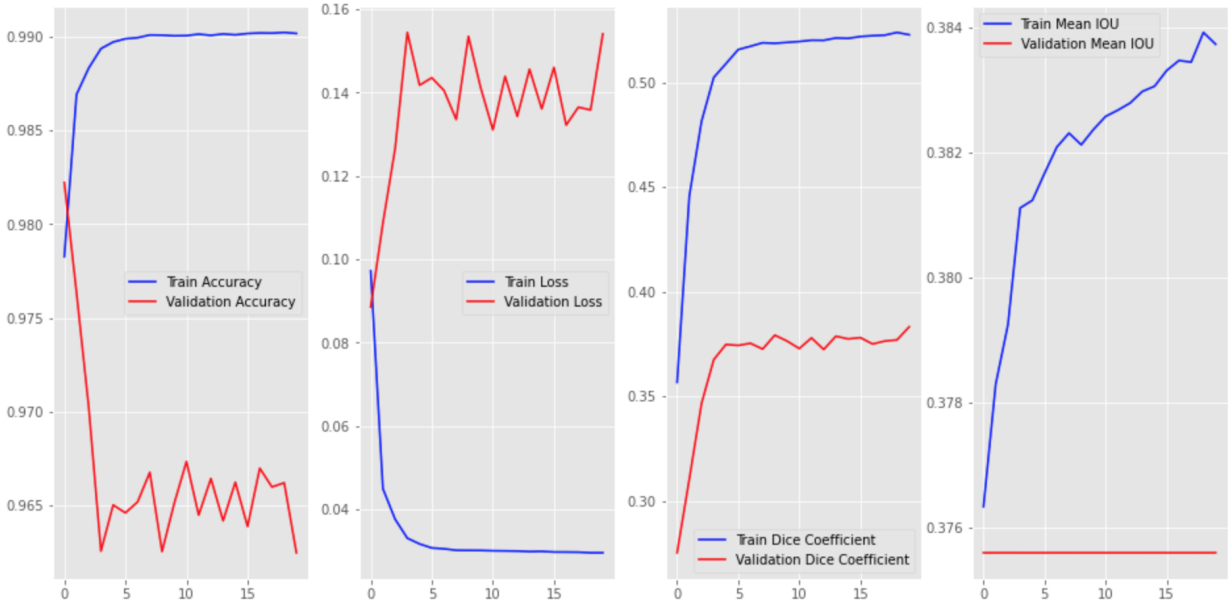

Figure 13. FC-DenseNets training log.

From left to right, accuracy, loss, dice coefficient, and IoU are plotted versus the train epoch.

Model evaluation and comparison between the two models

The performance of FC-DenseNets was evaluated on the test set using the same metrics described above. As expected from the relatively poor performance on the validation set during training, the FC-DenseNets underperformed the U-Net on all metrics (**Fig. 14**). This could be taken as a case where more is less (more layers, feature maps, reuse of features through recursive concatenations). However, we also note that performance of the FC-DenseNets could be improved by modifying the model architecture, tweaking the hyperparameters and/or using less downsizing of images etc., although such endeavors are out of the scope of this project.

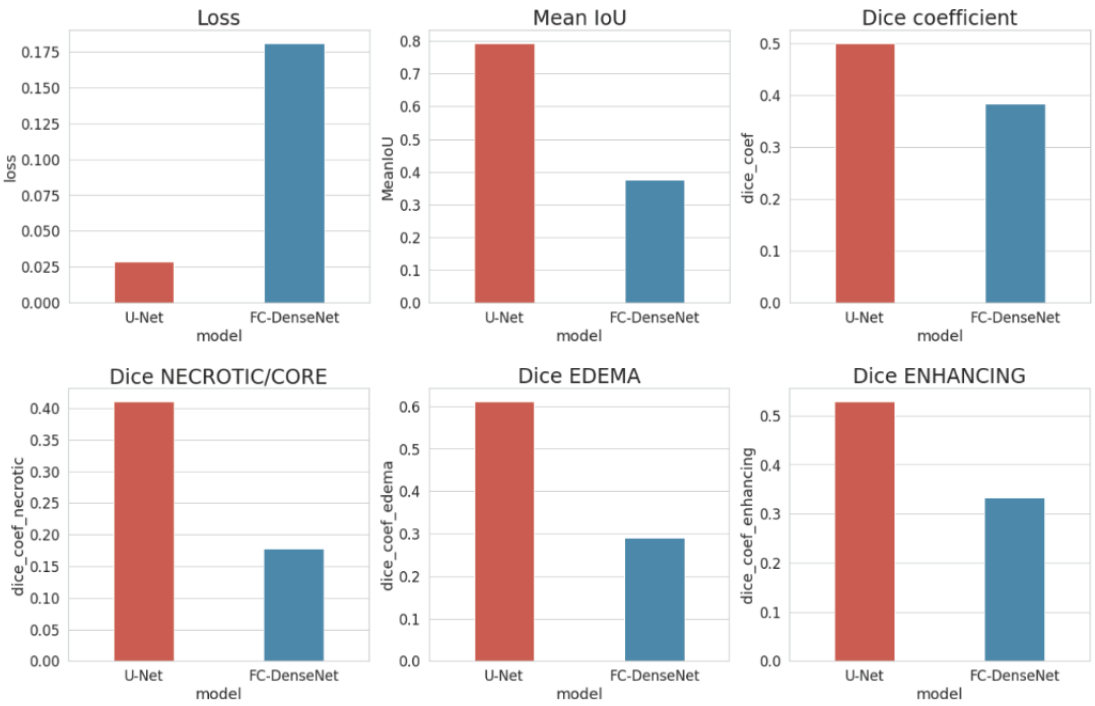

Figure 14. FC-DenseNets performance measured on diverse metrics in comparison to U-Net.

Note that the U-Net outperformed the FC-DenseNets on all metrics we measured.

Conclusion

We tackled a brain tumor detection task by end-to-end training two of the state-of-the-art convolutional neural networks for semantic image segmentations. The U-Net model had a relatively simple architecture, and it could be relatively easily implemented and trained on a single NVIDIA GPU (TESLA P100) with 16GB memory. The U-Net outperformed the more complex and deeper FC-DenseNets at least on the versions of the models we configured. Admittedly, there are many model parametric, hyper-parametric, and schematic (e.g. postprocessing) spaces we have not exhaustively explored that might improve model performance, which would be an important direction of future studies.

Reference

1. Bakas, S. *et al.* Advancing The Cancer Genome Atlas glioma MRI collections with expert segmentation labels and radiomic features. *Scientific Data* vol. 4 (2017).
2. Long, J., Shelhamer, E. & Darrell, T. Fully convolutional networks for semantic segmentation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015)

doi:10.1109/cvpr.2015.7298965.

3. Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv* (2015).
4. Jegou, S., Drozdzal, M., Vazquez, D., Romero, A. & Bengio, Y. The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2017) doi:10.1109/cvprw.2017.156.