Brain Tumor Segmentation with Attention-based U-Net

Tuofu Li[1, a, *, †], Javin Jia Liu[2, b, *, †], Yintao Tai[3, c, *, †], Yuxuan Tian[4, d, *, †]
[1]Computer Science, Dept. of Literature & Science, University of California (Davis), Davis, USA
[2]Statistics and Computer Science, Dept. of Science, McGill University, Montreal, Canada
[3]School of Engineering, College of Science & Engineering, University of Edinburgh, Edinburgh, United Kingdom
[4]Statistics and Computer Science, Department of Science, McGill University, Montreal, Canada
[*]Corresponding Authors：[a]javin.liu100@gmail.com, [b]y.tai-6@sms.ed.ac.uk,
[c]andy7093li@gmail.com, [d]yuxuan.tian@mail.mcgill.ca
[†] These authors contributed equally.

ABSTRACT

Brain tumors are a hazardous type of tumor, and they build pressure inside the skull when they grow, which can potentially cause brain damage or even death. Attention mechanisms are widely adopted in state-of-the-art deep learning architectures for computer vision and neural translation tasks since they enhance networks' ability to capture spatial and channel-wise relationships. We offer an attention-based image segmentation model that outlines the brain tumors in Magnetic Resonance Imaging (MRI) scans if present. In the paper, we mainly focus on integrating Squeeze-and-Excitation Block and CBAM into the commonly used segmentation model, U-Net, to resolve the problem of concatenating unnecessary information into the decoder blocks and attempt to locate the tumor boundaries. Our research clearly shows the application of the attention mechanism in U-Net, incorporates the Squeeze-and-Excitation with CBAM, and improves the performance in the brain tumor segmentation task. The model is delivered on an app with additional text to speech and chatbot features provided.

Keywords： Image segmentation, Brain tumor, Attention, U-net, Squeeze and Excitation Network, CBAM

1. INTRODUCTION

Brain tumors are the most common tumor out of all the primary central nervous system tumors. It accounts for 85% to 90% of the central nervous system tumors, as described by statistics collected on Cancer Nets1. MRI scans are used for tumor diagnosis by providing soft tissue contrast [1]. Convolutional Neural Network, which was constructed inspired by the structure of the human cortex, became the most accessible and accurate model for image classification tasks. One can apply such architecture to detect the presence of tumors in brain MRI scans. This paper would like to go beyond tumor classification and show where exactly the tumor resides.

Before the deep learning era, Edge-based and Region-based segmentation described in [1, 2, 3] were broadly applied for image segmentation. Later in 2014, Fully Convolutional Network-Based Semantic Segmentation (FCN) [4] was invented based on Convolutional Neural Network [5] to perform a pixel-wise classification of an image for semantic segmentation which was used as a benchmark for preceding segmentation models. Prediction suffered from coarse object boundaries using FCN until U-Net [6] was proposed to address the issue a year later. U-nets have been previously applied to the BRATS dataset [7], which stands for the Brain tumor segmentation dataset. It is a similar dataset [7] commonly used in the MICCAI conference where participants try to improve upon the segmentation of past years. The U-Net model originated from the EM segmentation challenge at ISBI 2012, and it won the challenge. See more in this paper [6]. In the modern-day and era, information and MRI scans are not as readily available as compared to other vision databases. U-net needs very few training samples to achieve good accuracy, and it runs quickly compared to other models. It performs consistently well at most medical segmentation tasks like the segmentation of neuronal structures in electron microscopic recordings, cell segmentation, and in our case, Brain tumor segmentation. These two reasons explain why U-net is one of the most popular segmentation techniques in the medical domain.

[1] "Brain Tumor: Statistics." n.d. Cancer.Net. https://www.cancer.net/cancer-types/brain-tumor/statistics.

Second IYSF Academic Symposium on Artificial Intelligence and Computer Engineering,
edited by Wei Qin, Proc. of SPIE Vol. 12079, 120790N · © 2021 SPIE
0277-786X · doi: 10.1117/12.2623112

Being bottlenecked by the ability to process information, a human can only focus on certain critical areas of an image yet still be able to interpret the given information appropriately. To simulate the human vision system, researchers have explored the many attention mechanisms [8, 9], implanting layers of learnable weights in deep neural network architectures to focus on the essential areas and suppress insignificant ones.

In our research, we investigate using two attention mechanisms – SE Block proposed by [8] that focuses solely on channel attention, and CBAM [9] that applies attention on both channel and spatial axis – to augment the segmentation ability of the U-Net model. Our research shows that the application of the attention mechanism on the concatenated channels of the U-Net can significantly improve the performance of the Brain Tumor MRI pixel-wise segmentation task in terms of IoU score [6]. In our experiments, the U-Net with a ResNet50 backbone and a coalesced attention mechanism that incorporates the Squeeze-and-Excitation and CBAM achieved the highest IoU score. We also explored the influence of different objective functions. In section 3, We compared the performance of Dice loss [10], Focal loss [11], and Focal+Dice loss in different models.

We also implemented an app using python's Tkinter library. This app allows the user to talk to a chatbot as well and play the doctor's message in their speaker using text to speech.

Currently, technology has changed our lives. Almost everyone who has a phone uses Facebook, and many medical apps exist to help provide a better service for patients. Two of the most popular apps include DynaMed Plus an app that allows healthcare workers to access medical information, and PEPID, an app that allows healthcare workers to search for possible diagnoses by inputting patient symptoms. Many of these are aimed to help doctors, but hardly any is aimed to help patients directly. We wanted to build an app like WebMD that allows the patients to self-diagnose their MRI scans at home to detect certain tumors. With the combination of help and confirmation from licensed healthcare professionals, this tool can greatly help patients to obtain a more accurate prediction.

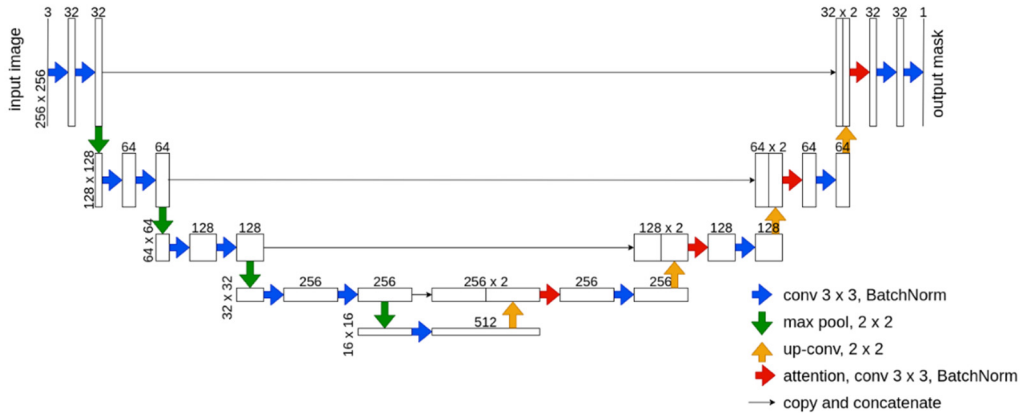

Figure 1. Attention-based U-Net Structure

NLP techniques are often used in translation apps to help users quickly translate a sentence and hear what it sounds like in another language. We also attempt to add these features to our healthcare app in hopes that it will help patients with language barriers better understand the app's contents.

2. METHOD

2.1 Dataset

The brain Magnetic Resonance Imaging (MRI) we used in this research is collected from the Cancer Genome Atlas (TCGA) and the Cancer Imaging Archive (TCIA) [12]. And all the images are provided by the following five institutions: Thomas Jefferson University (TCGA-CS, 16 patients), Henry Ford Hospital (TCGA-DU, 45 patients), UNC (TCGA-EZ, 1 patient), C

ase Western (TCGA-FG, 14 patients), Case Western – St. Joseph's (TCGA-HT, 34 patients) from TCGA LGG collection [12]. We regard each slice from a MRI series as an image, 3929 images in total. There are 3 channels in each image and a

corresponding mask channel, representing the abnormality's outline [12]. All mask channels are drawn by a medical professional [12]. All data is publicly available through Kaggle [12].

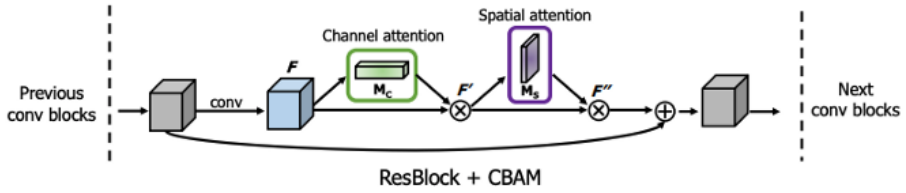

Figure 2. Channel-wise Attention Structure [13]

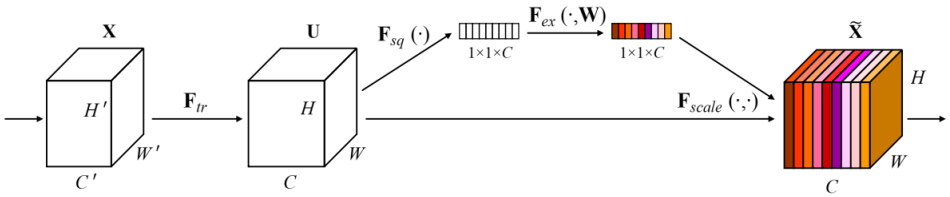

Figure 3. Mixed Attention Structure [14]

2.2 Model Structure

2.2.1 U-Net

U-Net [6] is a convolutional network proposed by Olaf Ranneberger, Philipp Fischer, and Thomas Bro back in 2015 for medical image segmentation. It is built upon the Fully Convolutional Network concept, which reduces the fully connected layers instead of using the convolutional layer as output and applies a concatenation mechanism to propagate the spatial information from the input image to the output. Thus, the U-Net is capable of pixel-wise classification tasks, especially in medical image segmentation.

There are two essential building blocks in U-Net, the contraction path and the expansion path. The contraction path is a multi-stage convolutional network. Each stage halves the image size and doubles the number of channels, forcing the convolutional layers to acquire knowledge about the high-level abstractions of input images.

The expansive path then applies transposed convolution to scale the channel height and width up by a factor of 2 at every stage and halve the number of channels. Nevertheless, the up-sampled channels lose precise spatial information through the previous down-sampling. Thus, we concatenate the up-sampled channels with corresponding channels in the contraction path to propagate the spatial information. Then we applied the attention mechanism to weigh and extract information that is more meaningful for our tasks. The attention mechanism is demonstrated in the next two subsections. Figure 1 illustrates the architecture and dimensions of each stage.

In the experiments, we also explored using ResNet50 [13], pre-trained on the ImageNet [15] as the contraction path to speed up the training and further improve the performance, as demonstrated in table 1. We removed the fully connected layers and treated the layers before each down-sample operation as one stage in the contraction path. The channels before each down-sample stage are used to concatenate to the corresponding expense stage channels.

2.2.2 Channel Attention

Features map from different channels in the contraction path incorporate different kinds of information, especially for the pre-trained ResNet50 network. Because the original task is image classification, some channel embedded information is unnecessary for our task, and some channels may depend on others. Thus, we applied the Squeeze-and-Excitation [8] to model the channel-wise dependence and weight channels according to the calculated attention vector.

$$SE(F) = \sigma\left(W_2\delta\left(W_1 AvgPool(F)\right)\right) \otimes F \quad (1)$$

Equation (1) describes the Squeeze-and-Excitation process, where W1∈ $\mathbb{R}^{\frac{C}{r}\times C}$, W2 ∈ $\mathbb{R}^{C\times\frac{C}{r}}$, σ is the sigmoid function, δ is the ReLU [16] function, and F represents feature maps. The Squeeze operation calculates the channel-wise statistics of F by a global average pooling function. To model the nonlinear channel-wise dependencies, we map the statics vector to a short vector by multiply the matrix W and apply the ReLU [16] function δ. Then we multiply to W2 and use a sigmoid function to extend the vector into the original dimension to get the attention of each channel. Finally, elementwise multiply each feature map with the corresponding weight in the attention vector SE(F). W1 and W2 are learnable parameters.

2.2.3 Mixed Attention (CBAM)

The CBAM [9] model focuses on increasing focus on essential features by using attention modules. We apply both channel attention and spatial attention modules to form the CBAM. To do so, we use max pooling and average pooling layers. The max-pooling selects the maximum pixel values of the batch, while the average pooling of a batch selects the average value of the pixels. We concatenate these two pooling layers and apply convolution to them to produce a spatial attention map. The spatial attention tries to make the CNN focus on the most critical parts of an image complementary to the channel attention. Hence, we can combine both spatial attention and channel attention in one model (CBAM). In a CBAM, we first apply channel attention, and then we apply spatial attention. We then add the output to the previous convolutional layer to get the finalized result. Equation (2) and (3) illustrates the overall process. From the feature map F, we apply the channel attention and get the channel attention map Mc. We do element-wise multiplication of Mc and the original feature map to get F'. We then apply spatial attention to get the spatial attention map. The spatial information is the cross multiplied by the info from the channel attention and added to the value from the previous conv blocks, as shown in Figure 3.

$$\begin{aligned} F' &= M_C(F) \otimes F \\ F'' &= M_s(F') \otimes F' \end{aligned} \tag{2}$$

$$\begin{aligned} M_s(\mathrm{F}) &= \sigma\left(f^{(7\times7)}([AvgPool(F); MaxPool(F)])\right) \\ &= \sigma\left(f^{(7\times7)}\left(F^s_{avg}; F^s_{max}\right]\right)\right) \end{aligned} \tag{3}$$

$$\begin{aligned} M_c(F) &= \sigma\left(MLP\left(AvgPool(F)\right) + MLP\left(MaxPool(F)\right)\right) \\ &= \sigma\left(W_1\left(W_0\left(F^c_{avg}\right)\right) + W_1\left(W_0(F^c_{max})\right)\right) \end{aligned} \tag{4}$$

2.2.4 Loss Functions

a) Focal Loss

Applying traditional loss functions such as Binary Cross Entropy Loss on the Brain MRI Segmentation dataset may lead to a problem because only a small portion of the images may contain a tumor. In other words, the dataset is a highly unbalanced dataset for any pixel-wise loss calculation. Training with mostly negative pixels can be inefficient. In fact, overwhelming the gradient with easily classified pixels that do not contribute much to the training process may lead to degeneration of models.

$$FL(p_t) = -\alpha_t(1-p_t)^{\gamma} log(p_t) \tag{5}$$

Focal loss [11] described in equation (5) extends the Binary Cross Entropy Loss with two extra hyperparameters added. α value is a weighted coefficient multiplied by each ground truth class during loss computation. We can address the unbalanced problem by assigning a greater alpha to the rare classes. The gradient for a confident prediction is lower as γ increases, meaning that the model can focus on a more complex set of pixels with a steeper loss function, as demonstrated in Figure 4.

b) Dice Coefficient

Dice Coefficient (F1 score) [17] can be interpreted as a weighted average of the precision and recall. Figure 5 demonstrate the intuition of the dice coefficient. Let TP denote true positive, FP denotes false negative, and FN denotes false negative. Then for image segmentation, the Dice Coefficient is calculated using the formula:

$$Dice = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{6}$$

The derivation of the formula is shown below:

$$F_1 = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} = \frac{2}{\frac{TP+FP}{TP} + \frac{TP+FN}{TP}} = \frac{2 \times TP}{2 \times TP + FN + FP} = Dice \quad (7)$$

2.2.5 Metrics: IoU Score

Intersection over union is defined by the intersection between the predicted foreground and the ground truth foreground, divided by their union. As illustrated in Figure 5, the IoU score is calculated as:

$$IoU = \frac{TP}{FP \cup FN \cup TP} \quad (8)$$

It is an intuitive way to evaluate the overlapping area. We use this metric to show the performance of the model.

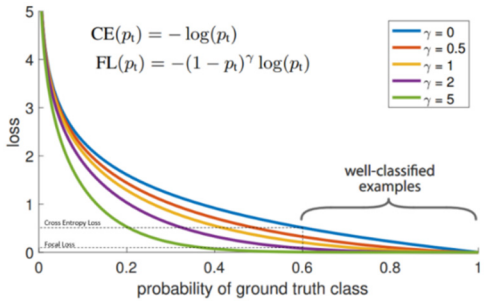

Figure 4. Focal Loss [7]

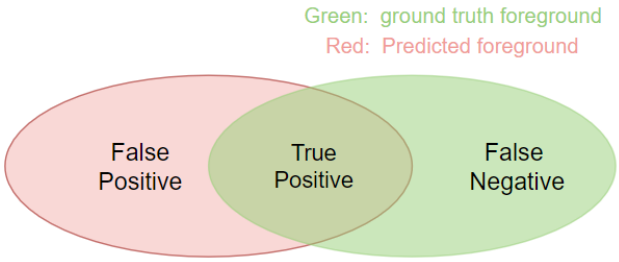

Figure 5. Dice Coefficient and IoU Intuition

2.2.6 Model Deployment

a) Chatbot

We built our chatbot based on the SpaCy NLP library and Rasa Open-Source Platforms.

b) Pipeline

This pipeline is used to train our chatbot. First, we import the spaCy NLP Library. This loads the pre-trained spaCy model, and then we added a tokenizer and a featurizer. The DIET Classifier is used for intent classification and entity extraction, and the response selector selects the corresponding output for our chatbot. The fallback classifier filters out all the irrelevant inputs.

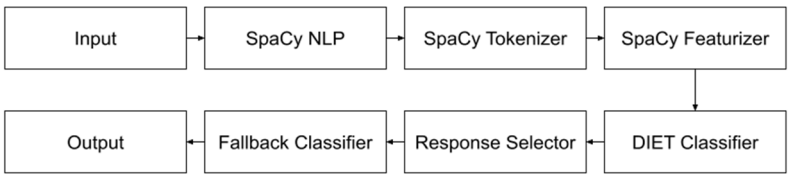

Figure 6. NLP Training Pipeline

c) Training Set

The Model needs to be trained with three sets of data. NLU (Natural Language Understanding) data are pairs of strings of intents and questions. For example, "Good morning" and "Hello" can be labeled as "greeting". Stories are a series of actions based on intentions. Responses are the actions defined in stories.

d) Chatbot Deployment

We merged the chatbot with the tumor classification model by filtering out some intents to trigger the image classification function. For example, when the chatbot classifies an intent as "Ask for Symptom", it triggers the program to receive an image as input, and feed the input into the classification model, and finally set the classification result as an output message.

e) User Interface

We used the *tkinter* app to build the app. As shown in Figures 7 and 8, the interface has a text area for the user to interact with the chat bot. There is also a speak button for the bot to speak the text inside the text box. We implemented this using the *googletrans* library and *gtts* library. To upload an image, a patient asks the bot the question, "How is my symptom?" The bot will allow the user to upload an image by dragging it to their local directory, putting the filename into the text space, and then press the add button. This will upload the image to our U-Net model and using the model to diagnose the input image. The chat box will then return the image that the user input and the image we get after image segmentation. It will also tell the user if they have a brain tumor.

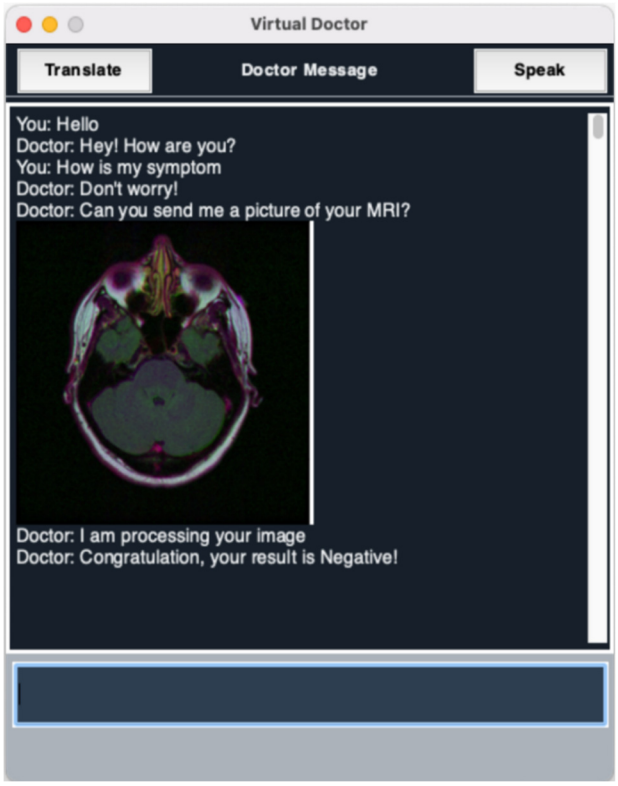

Figure 7. Negative Diagnosis

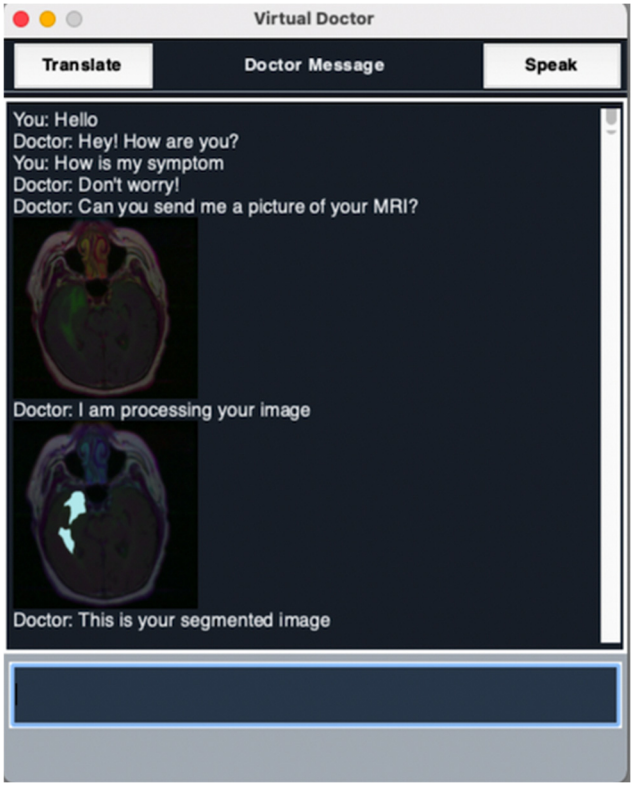

Figure 8. Positive Diagnosis

f) Life-long Learning

Due to the amount limit of the numbers of the neural network model being data-hungry, it's necessary to collect new data and retrain the model during the lifetime of the diagnosis system. One convenient way is to ask permission of using the user's data and let professional doctors annotate these new and add them to the training data set for retraining. The price of hiring professional doctors is high, it deserves to explore how much the model prediction can help doctors annotate data. Also, the possibility of applying semi-supervised learning on new un-annotated data is worth talking about in the future.

3. EXPERIMENTS AND RESULTS

We split the dataset into training set and test set with a 4:1 ratio, and applied transfer learning for training with ResNet50 pretrained on ImageNet as the model's backbone. Our training system is publicly available at: https://github.com/TYTTYTTYT/Attention_Based_U-Nets.

Various methods were explored for training, and all performances were measured using the IoU score. We analyzed the impact of loss function by comparing two models with different loss functions, including Focal Loss, Dice Coefficient, and their combination, respectively. We also tested a model with CBAM, SENET and both CBAM and SENET attention modules.

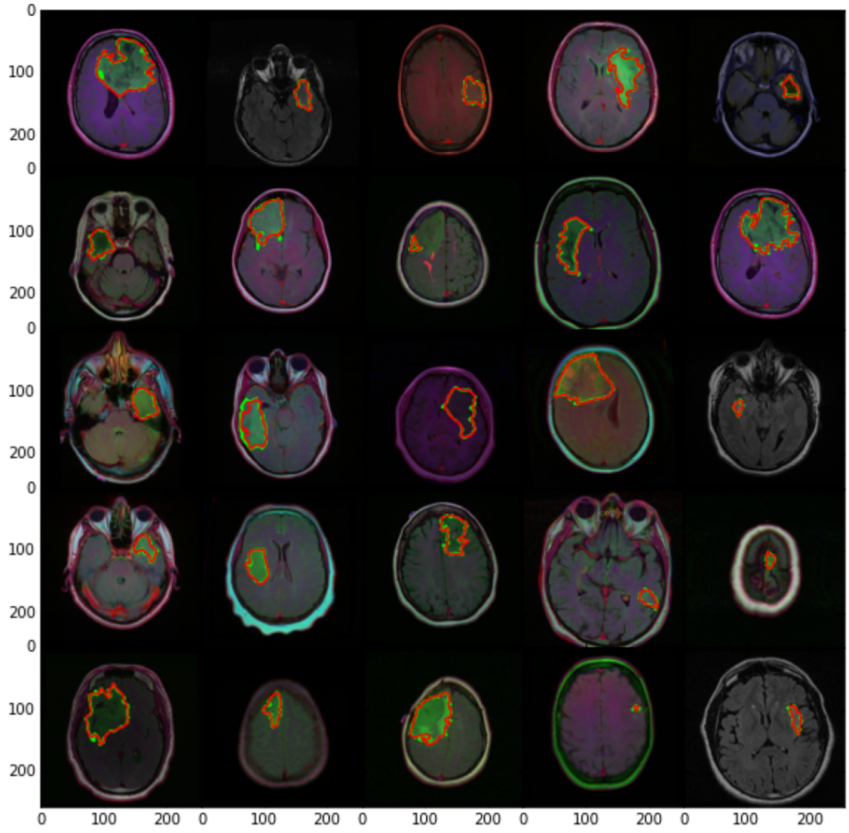

Figure 9. This figure compares the result of the predicted masks (area circled in red) with the ground truth masks (circled in green)

The CBAM ResUNet structure got an IoU Score of 84.76 per cent with focal. We got a lower IoU Score with a focal loss of around 84.23 per cent from the SENET model and the best IoU Score of 85.42 per cent by combining the SENET model with the CBAM model.

With the Dice loss, we get an IoU Score of 83.87 per cent. Since the randomly initialized weights of the decoder could be unstable in the early stage of training, we froze the decoder for the first 5 epochs trying to add more stability to it. Still, it did not offer a significant boost to the IoU score.

Considering that the dataset is relatively small for image segmentation tasks, we tried training the model with data augmentation on the fly using Keras ImageDataGenerator. However, no significant improvement was made.

Among all methods explored, the model with SENet and CBAM with Focal loss performed the best and achieved an IoU Score of 85.42 per cent. Table 1 shows the IoU score obtained using various loss functions, and Figure 9 shows the prediction result of our trained model. We also compared the effects of the loss function. We found that that the model performed better using focal loss when compared to dice loss.

For the model with SENet, we got an IoU score of 83.65 per cent with dice loss and 84.23 per cent with focal loss. For the model with CBAM, we got an IoU score of 83.76 per cent with dice loss and an IoU score of 84.76 per cent with focal loss.

To test the effect of adding the ResNet as the backbone, we compare our ResUNet model's performance with the original U-Net results. Using the dice loss with SENet, the U-Net got an accuracy of 82.74 per cent, and the ResUNet got an accuracy of 83.65 per cent. Similarly, the ResUNet also performed better than the U-Net with focal loss. ResUNet got an IoU score of 84.23 per cent with SENet, which is higher than UNet, which got an IoU score of 78.72 per cent. The ResUNet also performed better than the UNet using CBAM, as shown in table 1.

This proves that our ResUNet performs better than the original UNet.

Table 1. Validation IoU scores corresponding to models with combination of Backbones, attention mechanism, and loss functions. Note: ResUNet means a U-net with Resnet50 as backbone.

Method	IoU Score
U-Net: Dice loss	81.35%
U-Net: Focal loss	79.20%
U-Net: Focal + Dice loss	82.55%
U-Net with SENet: Dice loss	82.74%
U-Net with SENet: Focal loss	78.72%
U-Net with SENet: Focal + Dice loss	82.89%
U-Net with Complete SENet: Dice loss	82.75%
U-Net with Complete SENet: Focal loss	81.24%
U-Net with Complete SENet: Focal + Dice loss	83.43%
ResUNet: Dice loss	79.58%
ResUNet: Augment + Focal + Dice loss	83.73%
ResUNet With SENet: Dice loss	83.65%
ResUNet With SENet: Focal loss	84.23%
ResUNet with CBAM: Focal loss	84.23%
ResUNet with SENet + CBAM: Focal	85.07%
ResUNet with SENet + CBAM: Dice loss	**85.82%**
ResUNet with SENet: Focal + Dice loss	**85.22%**
ResUNet With SENet + CBAM: Focal + Dice loss	**85.42%**

4. FUTURE WORK

From manually examining the images, we discovered that the image qualities are rather spotty. Some images are dominated by the third channel, while others are merely blank images that do not contribute to model learning. Although deep neural networks are robust against minor variations, it is still important to manually go through the test set and ensure that it only consists of data from the desired distribution.

Also, we would like to expand our model to be able to do image segmentation on multiple datasets. Perhaps the user uploads an image of a kidney tumor. We would like to be able to diagnose that too and provide the patient with a variety of options. Furthermore, we would like to improve the app and make it iOS, Android, and web accessible to use it anywhere they would like.

There are two parts that we plan to improve for the user interface. First, we are considering adding a speech-to-text function that allows users to communicate with the virtual doctor with no text required. Second, we'd like to add a function that exports the patients' diagnosed results and segmented MRI images to their local storage for our future version.

5. CONCLUSION

We built an app to diagnose Brain Tumor scans and do image segmentation of the brain MRI scans. In the experiments, we found that using Focal loss and the Dice coefficient as our loss function provides a more accurate and consistent model. Furthermore, we also designed an application that can offline diagnose and segment the brain tumor effectively.

The app with its chatbot features can respond to the patient's requests, and the patient can upload an image by asking the chatbot, "What are my symptoms?" in which the chatbot will allow the user to input a file and run the U-net on the inputted image and give the final segmented image as a result. The proposed method, as well as its application, has great potential in brain tumor segmentation and wishes to be utilized in the actual medical field. With its text to speech features, this can help people better understand their diagnosis.

We sought to improve the performance of a basic U-net model structure. We used Resnet-50 for the model of our backbone and experimented with attention modules in our model as an attempt to increase our model's performance. In the end, the best performance was achieved by using both SENET and CBAM in the same model, and it gave an IoU score of 85.42%.

Our method can handle brain tumor segmentation with a satisfying performance, which is potential in real world application.

REFERENCES

[1] S. Bauer, R. Wiest, L. Nolte, and M. Reyes, "A survey of mri-based medical image analysis for brain tumor studies," Physics in Medicine & Biology, 58(13): R97, 2013.

[2] M. Wani and B. Batchelor, "Edge-region-based segmentation of range images," IEEE Transactions on Pattern Analysis and Machine Intelligence, 16(3):314–319, 1994.

[3] P. Yushkevich, J. Piven, H. Hazlett, R. Smith, S. Ho, J. Gee, and G. Gerig, "User-guided 3d active contour segmentation of anatomical structures: significantly improved efficiency and reliability," Neuroimage, 31(3):1116–1128, 2006

[4] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3431–3440, 2015.

[5] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, "Object recognition with gradient-based learning, " In Shape, contour and grouping in computer vision, pages 319–345. Springer, 1999.

[6] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.

[7] B. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, et al, "The multimodal brain tumor image segmentation benchmark (brats)," IEEE transactions on medical imaging, 34(10):1993–2024, 2014.

[8] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-Excitation Networks," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.

[9] S. Woo, J. Park, J. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module," Computer Vision – ECCV 2018, 2018, 3–19.

[10] Tustison, N. J., and J. C. Gee, "Introducing Dice, Jaccard, and other label overlap measures to ITK," Insight J 2 (2009).

[11] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," In Proceedings of the IEEE international conference on computer vision, pages 2980–2988, 2017.

[12] M. Buda, A. Saha, and M. A. Mazurowski, "Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm," Computers in Biology and Medicine, 109:218–225, 2019.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, pp. 770-778, 2016

[14] T. H. Phan, "Resolving Class Imbalance in Object Detection with Weighted Cross Entropy Losses," arXiv:2006.01413v1 [cs.CV], 2006

[15] J. Deng, W. Dong, R. Socher, L. Li, K. Li and F. Li, "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248-255, doi: 10.1109/CVPR.2009.5206848.

[16] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Bolzmann Mechines," in ICML, 2010.

[17] L. M. DeAngelis, "Brain tumors," New England Journal of Medicine, 344(2):114–123, 2001.