

Improved Glioma Grading using Deep Learning Techniques

Dr. Sreedevi Gutta and Sajjad Sabahuddin

February 28, 2024

1 Abstract

BACKGROUND AND PURPOSE: Gliomas are a sort of primary brain tumor that originates from glial cells. These cells are accountable for supporting the central nervous system of the human body. These tumors vary widely based on their nature, they could be either malignant or aggressive. Glioma grading is an essential part of diagnosis and treatment planning, as it can give critical information regarding the tumor’s characteristics and its behavior. The main objective of this study is to use MR images and develop an attention-based model to detect glioma and visually locate the tumor.

MATERIALS AND METHODS: The dataset includes a total of 285 patients with both high- and low-grade gliomas. The preprocessing steps applied on this dataset include interpolation to a standardized resolution of 1 cubic millimeter, alignment to a common anatomical template, and finally skull stripped. A pretrained VGG16 model with 2 attention modules is used for grade prediction. These attention modules extract intermediate features from the main architecture and predict the area of tumor by highlighting it. The proposed model’s performance is evaluated with other pretrained models like ResNet, DenseNet, MobileNet, and EfficientNet.

RESULTS: The proposed model was able to achieve an f1-score of 91.18%, demonstrating its robustness and capability to grade the tumor. Furthermore, the attention maps enabled detailed visualization of the tumor regions, enhancing the interpretability of the model’s predictions. The proposed model produced almost the same results as the pretrained model ResNet50 but with additional visualization of the tumor.

CONCLUSION: To conclude, the proposed model, leveraging its fundamental capability to automatically learn features, has proven remarkable effectiveness in glioma tumor detection and its classification. The integration of two attention maps into the VGG16 pretrained model has enhanced its capability of precisely focusing and detecting the tumor region, a feature that was not available in earlier models. This holds a promising advancement in disease diagnosis and medical imaging.

2 Dataset Description

The dataset was collected from “Multimodal Brain Tumor Segmentation Challenge 2018” hosted on a website of Perelman School of Medicine, University of Pennsylvania [1]. The following three preprocessing steps were considered before publishing the data online. Firstly, with the same anatomical template it is co-registered. Secondly, it is interpolated to the same resolution, which is 1 mm³. Lastly, it is skull-stripped, to remove the skull. This dataset has multimodal 3D brain MRI scans of 285 patients. Out of 285 patients, 210 belong to the glioblastoma (GBM/HGG) category and the rest 75 belong to LGG category. The scans are in Neuroimaging Informatics Technology Initiative (NIFTI) format, which has an extension of “.nii”. Each patient has 4 sequences, FLAIR, T1, T1CE, and T2 and each sequence has three types of orientations: namely axial plane, coronal plane, and sagittal plane [2]. When an MRI scan is viewed from top to bottom, that plane is referred to as axial plane [2]. Whereas the

plane from front to back is called coronal plane and from side-to-side plane is called sagittal plane [2]. In our case, each sequence has a shape of (240, 240, 155). Where the first 240 slices belong to sagittal plane and second 240 slices belong to coronal plane. The axial plane has 155 slices.

	HGG scans				LGG Scans			
	FLAIR	T1	T1CE	T2	FLAIR	T1	T1CE	T2
Sagittal								
Coronal								
Axial								

Figure 1: Scans of a patient with High Grade Glioma and Low-Grade Glioma. Each column corresponds to a sequence. The four sequences are FLAIR, T1, T1CE, and T2. Each row corresponds to a plane. The first, second, and third rows correspond to sagittal, coronal, and axial planes respectively.

3 Proposed Model - VGG16 with Attention

For the proposed model, we have adopted VGG-16 model [3], which has 16 layers in total. The convolution layers have a filter size 3×3 and stride = 1. The activation function is ReLU [3]. The maxpooling layer is of 2×2 filters and stride = 2 [3]. It also has 2 dense layers [3]. And lastly a softmax layer [3]. Right now, we have VGG16 as the backbone without any dense layers [3]. While analyzing an image or some MRI scan, we humans tend to focus only on the objects that are related to the task at hand [3]. For instance, dermatologists, who treat skin cancer, focus only on the lesion and not on irrelevant parts of the image like hair or background [3]. To copy this image exploration pattern, we are going to implement an attention module, which will calculate a spatial (pixel-wise) attention map [4]. Detailed illustration of the proposed network architecture is in figure 2.1 [4]. The gray blocks are the two attention modules (Refer to Fig.2.2 for details) [4]. One of the attention modules is pool-3 and the other is pool-4 [4]. The output which is the final feature vector is derived from concatenating the three feature vectors. These feature vectors were derived via global research pooling [4]. The final feature vector is passed to the classification layer, which is not in this figure [4].

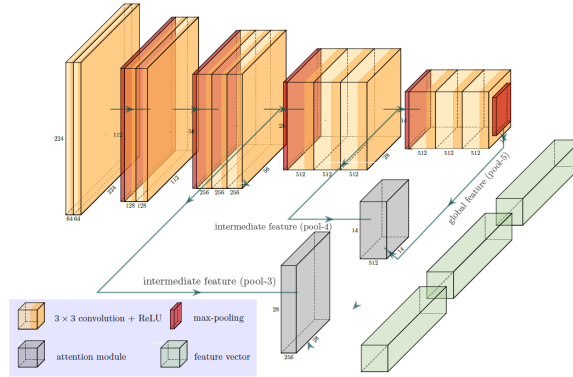


Figure 2.1: Detailed illustration of the proposed network architecture. The gray blocks are the two attention modules (Refer to Fig 2.2 for details) [4]. One of the attention modules

is pool-3 and the other is pool-4 [4]. The output which is the final feature vector is derived from concatenating the three feature vectors. These feature vectors were derived via global research pooling [4]. The final feature vector is passed to the classification layer, which is not in this figure [4].

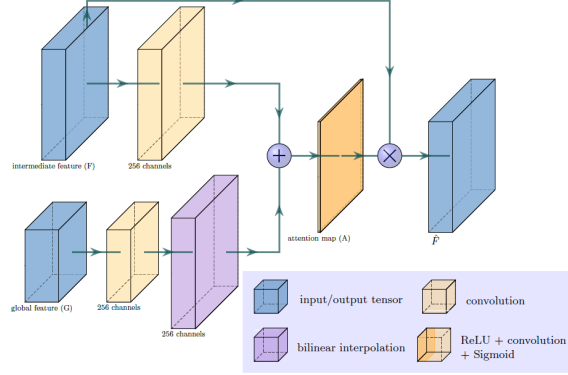


Figure 2.2: Illustration of the gray blocks from the two attention modules. Feature unsampling is performed with the help of bilinear interpolation when the spatial size of both intermediate and global features is unique [4]. The multiplication operation is “pixel-wise”, whereas the sum operation is element wise [4].

4 Fine Tuning for Deep Learning



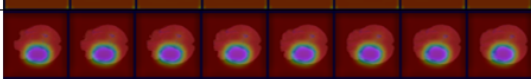
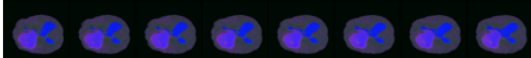

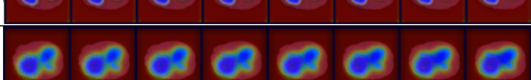
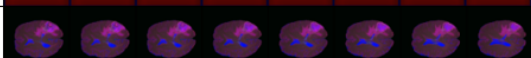
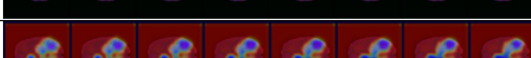
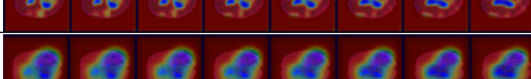
In our final stage of this process, we focused on fine-tuning hyperparameters like dropout rate, learning rate, and weight decay. The learning rate, which is a key parameter influencing the convergence of our model, was altered across four values: 0.01, 0.001, 0.0005, and 0.0001. As for the dropout rate, we explored a range of values from 0.2 to 0.5. Lastly, we adjusted the weight decay across the following range: 0.01, 0.001, and 1e-05.

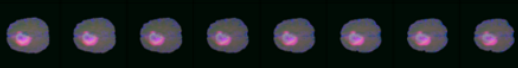
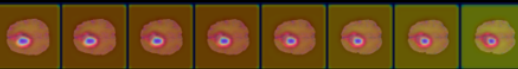
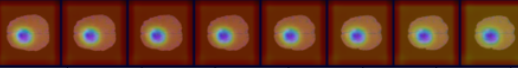
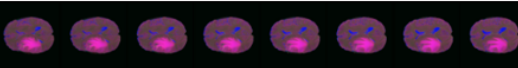

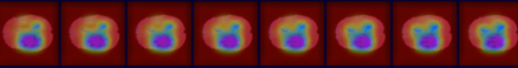
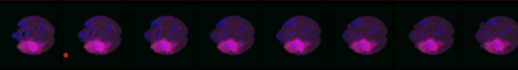
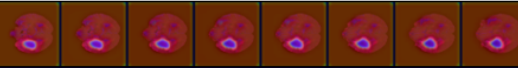
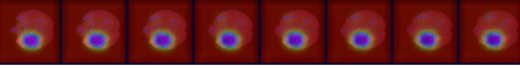
5 Results

The below table contains all the results from the models we have explored. Starting with ResNet 5 experiments were conducted on the complete dataset. We have used five metrics Confusion Matrix [5], Accuracy [6], Precision [7], Recall [8], and F1-score [9]. For our research we are more focused on the f1-score because it provides a balanced measure of precision and recall. F1-score studies both the ability to predict all positive cases, i.e. recall, and the ability to accurately detect positive cases, i.e. precision. The proposed model was able to record an f1-score of 91.18% with only 8 misclassifications from the total of 57 test patients. The hyperparameters which produced these results are as follows: Learning Rate = 0.01, Dropout Rate = 0.4, and Weight Decay = 1e-05.

No.	Model Names	Confusion Matrix	Accuracy (%)	Precision (%)	Recall (%) (Sensitivity)	F1-score (%)				
1	ResNet50	<table><tr><td>8</td><td>7</td></tr><tr><td>1</td><td>41</td></tr></table>	8	7	1	41	85.96	85.42	97.62	91.18
8	7									
1	41									
2	MobileNetV2	<table><tr><td>4</td><td>11</td></tr><tr><td>6</td><td>36</td></tr></table>	4	11	6	36	70.18	76.59	85.71	80.85
4	11									
6	36									
3	DenseNet201	<table><tr><td>8</td><td>7</td></tr><tr><td>6</td><td>36</td></tr></table>	8	7	6	36	77.19	83.72	85.71	84.70
8	7									
6	36									
4	EfficientNet_B2	<table><tr><td>6</td><td>9</td></tr><tr><td>7</td><td>35</td></tr></table>	6	9	7	35	71.93	79.55	83.33	81.48
6	9									
7	35									
5	Proposed Model VGG16 with Attention	<table><tr><td>8</td><td>7</td></tr><tr><td>1</td><td>41</td></tr></table>	8	7	1	41	85.96	85.42	97.62	91.18
8	7									
1	41									

6 Visualizations

Patients		
1	Input Images	
	Pool-3 Attention	
	Pool-4 Attention	
2	Input Images	
	Pool-3 Attention	
	Pool-4 Attention	
3	Input Images	
	Pool-3 Attention	
	Pool-4 Attention	

4	Input Images	
	Pool-3 Attention	
	Pool-4 Attention	
5	Input Images	
	Pool-3 Attention	
	Pool-4 Attention	
6	Input Images	
	Pool-3 Attention	
	Pool-4 Attention	

As we examine the above visualizations of 6 different patients, we can notice that the attention layers were able to detect the tumor from the input images. The model is coloring the tumor blue. The rest of the image, that is irrelevant, is either colored red or any another color. The gray blocks in figure 2.1 are the pool-3 and pool-4 attentions which produce the above visualization outputs.

References

- [1] <https://www.med.upenn.edu/sbia/brats2018/data.html>
- [2] Padmanaban, Sriramakrishnan Thiruvankadam, Kalaiselvi T., Padmapriya Thirumalaiselvi, M. Sivasakthivel, Ramkumar. (2020). *A Role of Medical Imaging Techniques in Human Brain Tumor Treatment*. 8. 565-568. 10.35940/ijrte.D1105.1284S219.
- [3] Ibtesam, A. (n.d.). *Image Classification with Attention*. Paperspace. <https://blog.paperspace.com/image-classification-with-attention/>
- [4] Yan, Y., Kawahara, J., Hamarneh, G. (2019). *Melanoma Recognition via Visual Attention*. In: Chung, A., Gee, J., Yushkevich, P., Bao, S. (eds) *Information Processing in Medical Imaging. IPMI 2019. Lecture Notes in Computer Science()*, vol 11492. Springer, Cham. https://doi.org/10.1007/978-3-030-20351-1_62
- [5] Simplilearn. (2023). *What is a Confusion Matrix in Machine Learning?*. <https://www.simplilearn.com/tutorials/machine-learning-tutorial/confusion-matrix-machine-learning>
- [6] Srivastava, Niharika. (2023). *Training, Validation Accuracy in PyTorch*. <https://www.e2enetworks.com/blog/training-validation-accuracy-in-pytorch>
- [7] <https://torchmetrics.readthedocs.io/en/stable/classification/precision.html>
- [8] <https://torchmetrics.readthedocs.io/en/stable/classification/recall.html>
- [9] https://torchmetrics.readthedocs.io/en/stable/classification/f1_score.html