What's in Your Brain?: A CNN Approach to Tumor Detection

Daniel Cho, Kameel Dossal, David Han, Erica Song

Adapted from: "Brain Tumor Classification Using Convolutional Neural Network," by S. Das, O. F. M. R. R. Aranya and N. N. Labiba, 2019.

Introduction

We recreated and modified a convolutional neural network for classification of four categories of tumors (glioma, meningioma, pituitary, and no tumor) in MRI images. Survival rate greatly depends on the type of tumor, and this model can help doctors correctly classify tumors and provide the most fitting treatment as early as possible.

Methodology

We trained our model on the Brain Tumor Classification (MRI) dataset from Kaggle (source: https://www.kaggle.com/datasets/sartajbhuvaji/brain-tumor-classification-mri).

Preprocessing

The images, which were taken from different angles, were saved as JPG files of various sizes. We resized all images to 112x112, applied a Gaussian blur filter, and normalized the contrast with histogram normalization. We also randomly flipped the images horizontally and vertically before feeding them through the model.

Test set:
• 100 glioma tumors
• 115 meningioma tumors
• 105 no tumor
• 74 pituitary tumors

+

Training set:
• 826 glioma tumors
• 822 meningioma tumors
• 395 no tumor
• 827 pituitary tumors

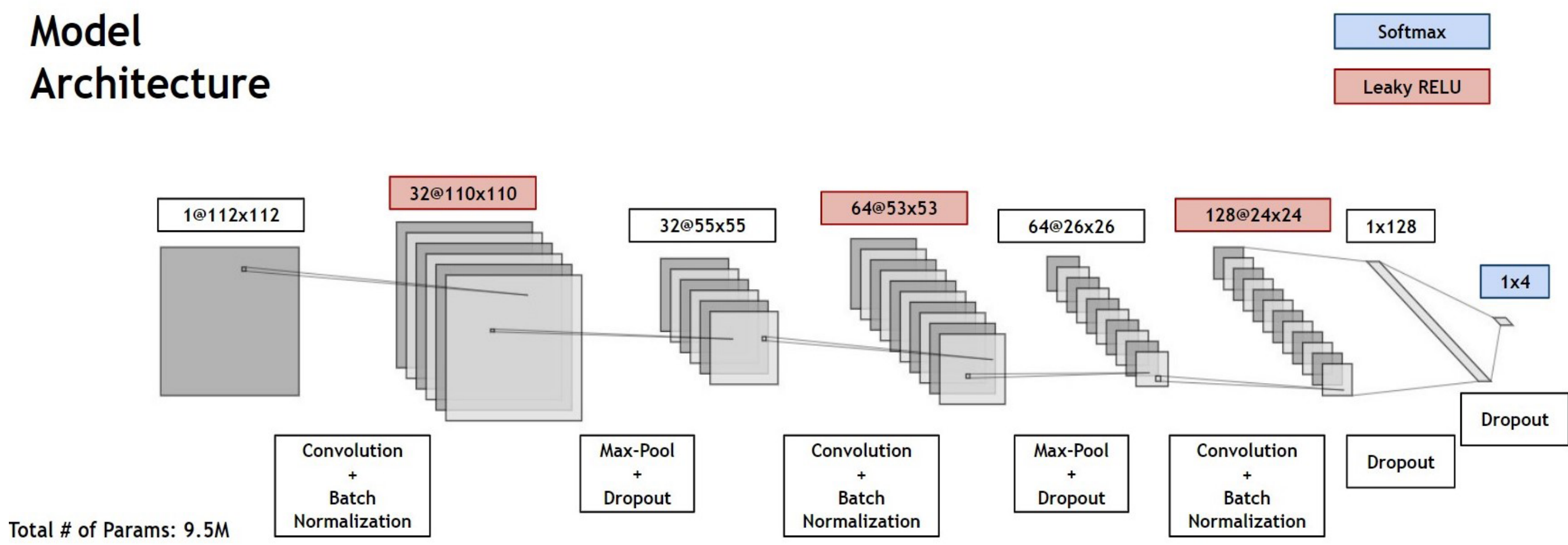### Model Architecture

Total # of Params: 9.5M

Figure 1. Our model used three convolutional layers with Leaky ReLU activation, batch normalization, max pooling, and dropout layers for training. We also used two linear layers with dropout; and ended with softmax activation. We used an Adam optimizer.

Hyperparameters	
Learning Rate	0.0001
Batch Size	128
Epochs	100

Results

The test accuracy on the plateaus at around 77% after 100 epochs. Our model currently struggles the most with classifying gliomas, only correctly identifying 22% of them. The model was fairly successful at classifying meningiomas (98% sensitivity), no tumor (100% sensitivity), and pituitary tumors (86% sensitivity).

$$Sensitivity = \frac{TP}{(TP + FN)}$$

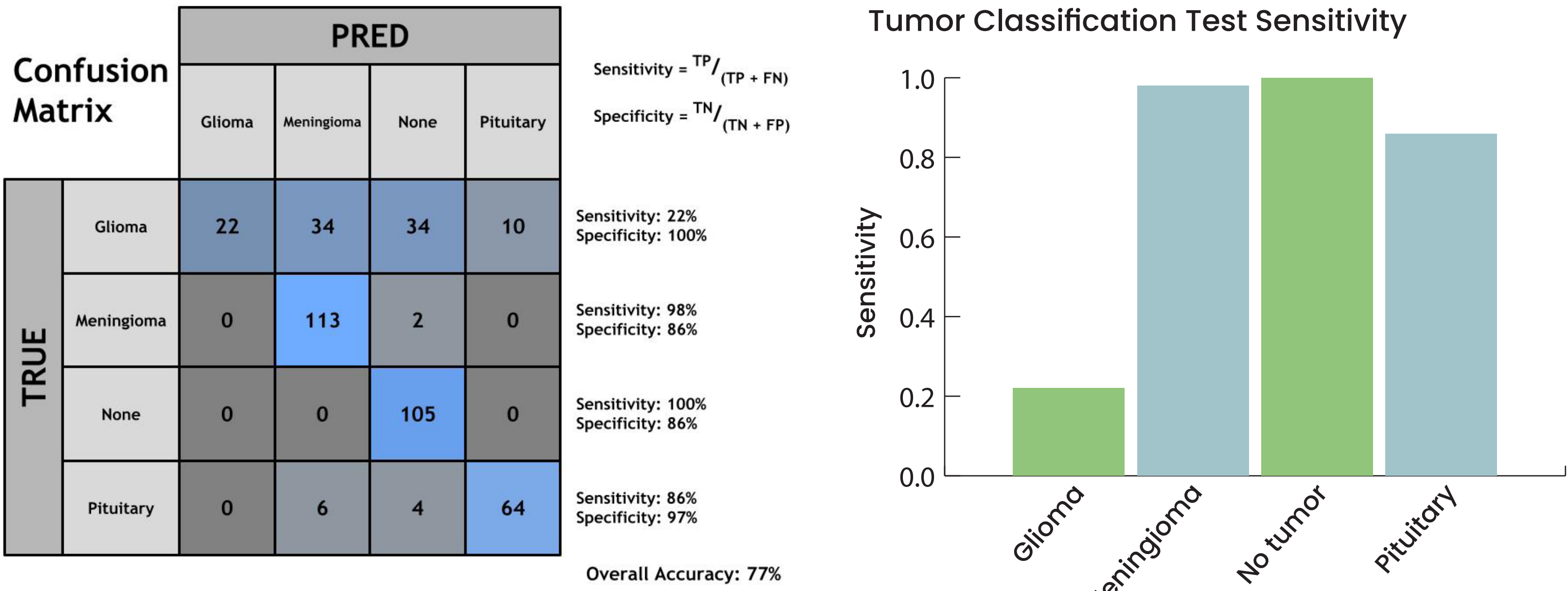$$Specificity = \frac{TN}{(TN + FP)}$$

Overall Accuracy: 77%

Figure 2. The confusion matrix above (left) shows the model's predictions for each type of tumor. This figure also shows calculations for sensitivity and specificity. TP and TN stand for true positive and true negative, respectively. FP and FN stand for false positive and false negative, respectively. Sensitivity measures the model's ability to recognize diseased individuals correctly. Specificity measures the model's ability to recognize correctly when an individual does not have a specific type of tumor. The sensitivity graph (right) compares the sensitivity of each type of tumor.

We also examined the accuracy and loss for the training and test sets over time and concluded that our model was overfitting the data because the training loss decreased to a very low value, while the test accuracy stagnated. We created a second model with only one convolutional layer, instead of three and with increased dropout rates. However, the final test accuracy was only about 70%. Because the nature of the problem, we prioritized accuracy in choosing our final model. Thus, we decided to use the overfit model due to its higher accuracy on the test set.
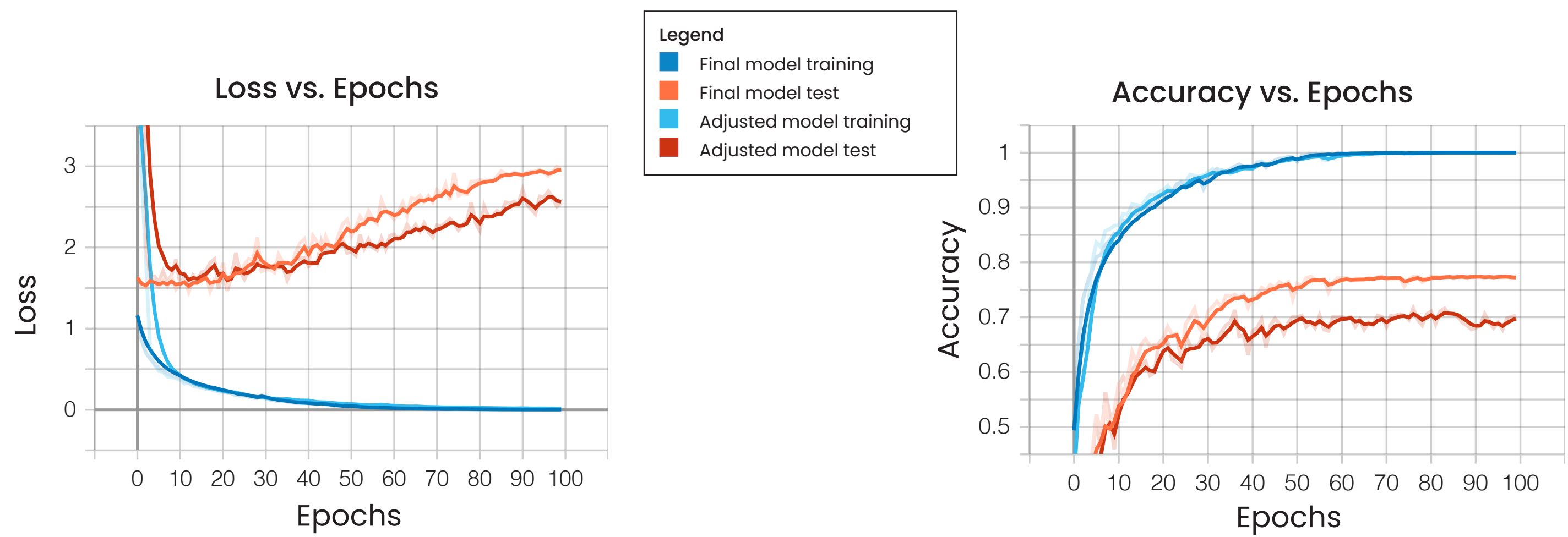

Figure 3. The figure above plots the training and test loss and accuracy against epochs for both our final model and the adjusted model, where we further tried to prevent overfitting. Our attempts to prevent overfitting reduced the test loss in the adjusted model, but the test accuracy was also lower in the adjusted model, compared to the final model. The training loss and training accuracy followed similar trends for both models.

Figure 4. The figure above samples some correctly and incorrectly labeled images from the test set for each tumor type. The top row of images were correctly classified. The second row contains incorrectly classified images (if any existed) with the model's predicted label below.

Discussion

Through this project, we gained experience fine-tuning a deep learning model to perform well on image processing. We learned new techniques to standardize images by contrast, such as histogram equalization, and we learned more about brain tumors in general.

Our final model seems to be overfitting to the training data, so on top of the original model, we increased the dropout rate, adding batch normalization, and randomly flipping images. We could reduce the complexity of the model by removing convolutional layers. We could also experiment with more random transformations on the images before training and tune the dropout rate, kernel size, and stride further. Another strategy is adding weight regularizers to the linear layers to penalize large magnitude of weights.

To further improve the accuracy of the model, we could try gathering more MRI images of brain tumors to train on. Having an even more equal images in each of the four categories would reduce the model's bias towards any one category. We might also want to choose images that would not be greatly distorted by resizing during preprocessing.

In the future, this model could be used to help healthcare professionals make more informed decisions about the identity of a brain tumor based on MRI imaging, saving time and money for patients. Some models may have better accuracy than doctors in recognizing very small tumors .

LIME Explainer

In the medical field, trust and accuracy are both important for patients when receiving diagnoses. According to a 2004 study by David H. Thom et al., "62 percent of patients in the highest quartile of trust reported that they always took prescribed medication and followed their doctor's recommendation, compared with just 14 percent of patients in the lowest trust quartile." Models like the CNN we built give little insight into the features they are using to learn and make decisions, which drives distrust in artifical intelligence.

LIME explainers help alleviate some of the mystery surrounding complex deep learning models by visualizing the model's focus in images. We built a LIME explainer to map the superpixels most relevant to the model predictions.
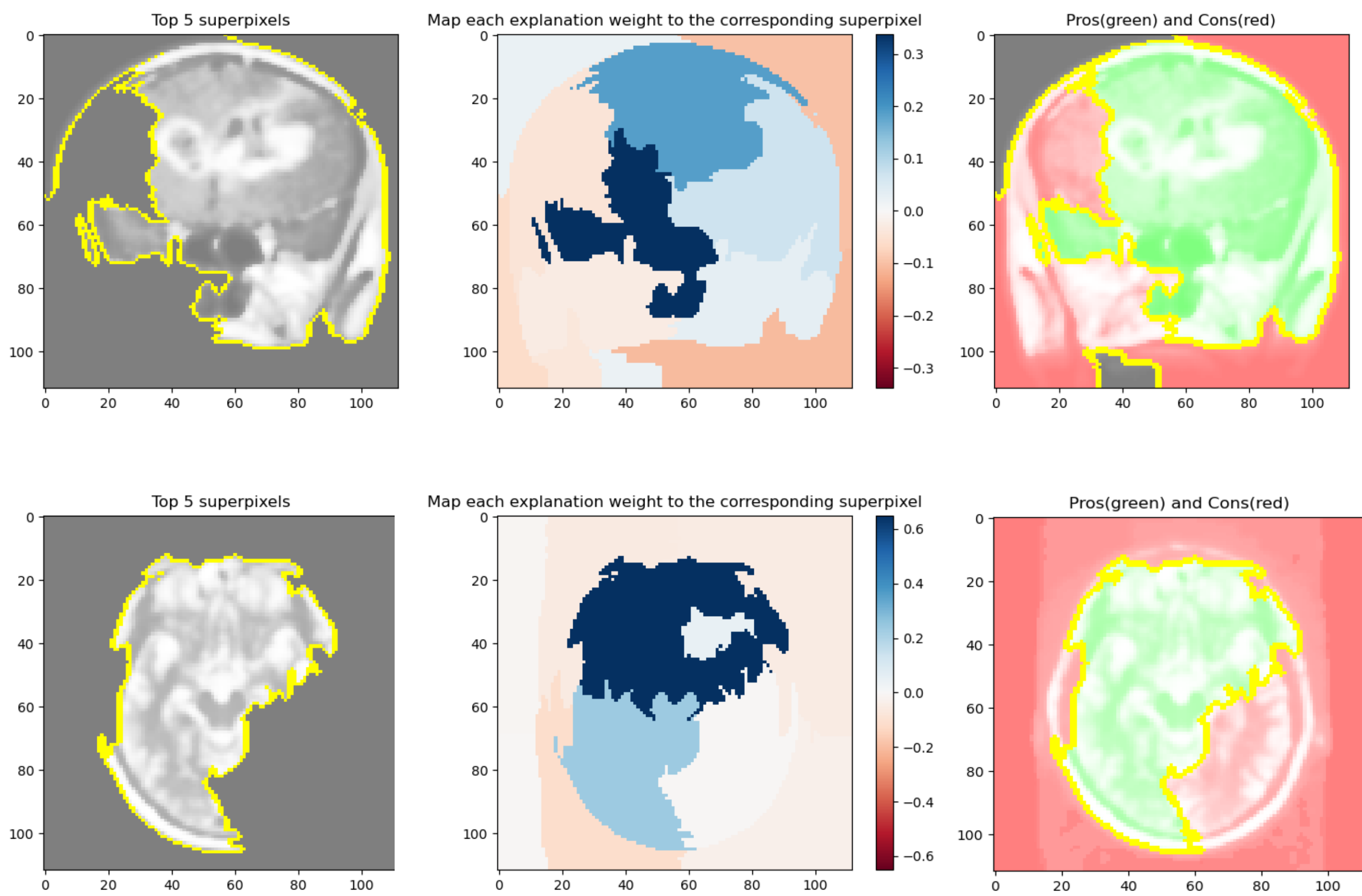

Figure 5. The images above were generated by a LIME explainer. The top row corresponds to a glioma tumor in the test set. The bottom row corresponds to a meningioma tumor from the test set. The superpixels highlighted have the most influence on the final label.

From the LIME explainer, we can see that the model does not perfectly pinpoint the tumor and the noise in the surrounding areas of the brain can negatively impact the prediction. In some images, regions of the image outside the brain and head also contributed to the prediction, even though they should be irrelevant.

Sources

Das, S., Aranya, O. F., & Labiba, N. N. (2019). Brain tumor classification using Convolutional Neural Network. 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT). https://doi.org/10.1109/icasert.2019.8934603

Sartaj Bhuvaji, Ankita Kadam, Prajakta Bhumkar, Sameer Dedge, & Swati Kanchan. (2020). Brain Tumor Classification (MRI) [Data set]. Kaggle. https://doi.org/10.34740/KAGGLE/DSV/1183165

Thom, D. H., Hall, M. A., & Pawlson, L. G. (2004). Measuring patients' trust in physicians when assessing quality of care. Health Affairs, 23(4), 124–132. https://doi.org/10.1377/hlthaff.23.4.124