Identifying Brain Tumors using Machine Learning

Rohan Tummala

12/11/22

Abstract

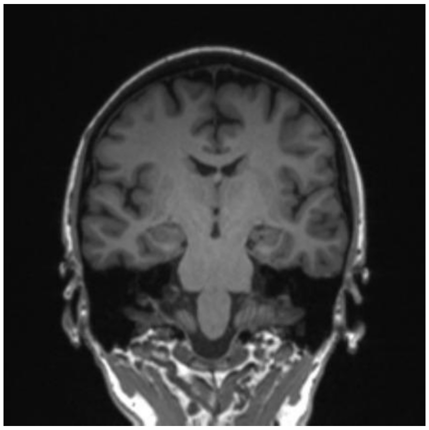
Through the rapid destruction of brain cells, malignant brain tumors can be one of the most dangerous forms of cancer that exist. Current methods for determining the presence and type of brain tumor in a given patient's MRI scan can oftentimes be inefficient and are prone for error. By using a machine learning algorithm, the error in these classifications is reduced significantly, and the process is made much more efficient. To approach this problem, we have implemented an Inception Resnet, which is a pretrained Convolutional Neural Network. The final model is able to determine the presence of a possible brain tumor as well as distinguish the type between three different classes. These classes include pituitary, meningioma, and glioma. Pituitary tumors are the most harmless of the three as they are noncancerous. The five year survival rate for a patient with a pituitary tumor is 97%. This contrasts greatly with both glioma and high grade meningioma tumors which are life-threatening [2]. It should be noted, however, that high grade meningioma tumors are very rare. Along with these three classes, a separate control class with no brain tumors was also used. After training the model on these four classes, the results were found to be impressive, as the final accuracy was at 96.7%.

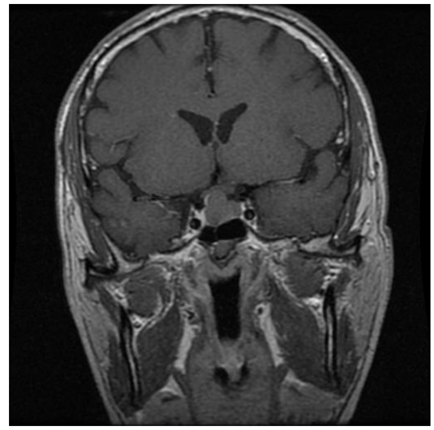
Introduction/Background

Brain cancer is one of the most potent and dangerous forms of cancer that can be, when left untreated, fatal. This is why it is important to detect tumors as quickly and efficiently as possible before the masses grow too large. Solving this research problem will be practically applicable in clinics where this tool could be used daily to determine the presence and type of tumor in a patient more efficiently. The current approach is to use an MRI to collect the brain images, which are human labeled [2]. Then a biopsy is performed to confirm the presence of a tumor [2]. The multiple stages in the existing methods of brain tumor detection allow for a higher chance of making an error in the diagnosis, and can increase the time needed to diagnose... The benefits of turning to a machine learning model to autonomously detect the presence and type of tumors in patients include being less resource intensive as well as being much more efficient and fast. Thus, we can conclude that using a machine learning algorithm, specifically the Inception Resnet model, could be an effective solution for this problem. Similar approaches have been taken to other image classification problems in healthcare**.** I hypothesize that the model will be able to detect the presence and type of tumor between three different possible classes—pituitary, meningioma, and glioma.

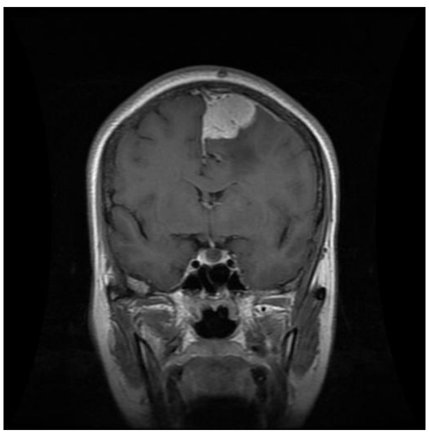
Dataset

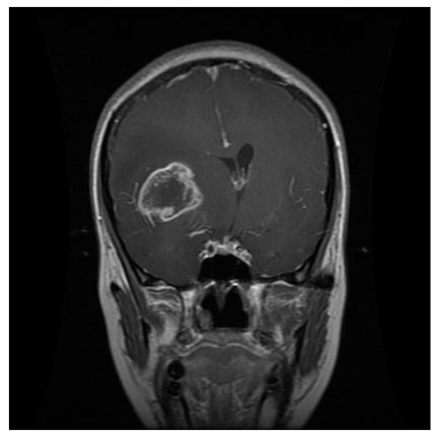
The dataset used in this model is a set of 5,952 images of MRI scans taken from a variety of angles, obtained from Sartaj Bhuvaji who is a computer scientist specializing in healthcare. MRI stands for Magnetic Resonance Imaging, and, as the name suggests, implements a large, tube shaped magnet that realigns water molecules in a patient's body [3]. Radio waves then cause the aligned atoms to produce signals, which are represented on the final image. The specific dataset used contains images that are labeled as belonging to one of four classes, three of them being types of cancer—pituitary, meningioma, glioma, and no tumor. Below is an example of each type of image:





No Tumor *Pituitary*





Meningioma *Glioma*

Figure 1: Images of the four different classes in the dataset

In total, there were 1621 glioma images, 574 meningioma images, 2000 images with no tumor, and 1751 pituitary images. These images were taken in three different orientations: sagittal (side), coronal (front), and transverse (top). The figures above are all in the coronal orientation. As can be seen in the figures, the more lethal glioma and meningioma tumors are much bigger whereas the pituitary tumor is mostly hidden. It is important to note that there were many variations in the dataset, namely with different images being of different quality and size.

Methodology

The model that was implemented to solve this problem was an Inception Resnet [4], a type of Convolutional Neural Network, or CNN. CNNs are generally used to solve image classification problems such as the one at hand, and utilize a deep neural network to do so [5]. The structure of a CNN includes convolutional, pooling, and fully connected layers. The convolutional layer utilizes small filters to scan through the given image and detect certain key features. Once this is done, the pooling layer downsamples the feature map given by the convolutional layer. Finally, the fully connected layer is responsible for predicting the final output.

Before this Inception Resnet could be implemented, it was necessary to preprocess the data in order to minimize the effects of over and under fitting. More specifically, in order to minimize overfitting, a randomized training and testing split of the data was used. In addition to this, a random set of the data was either sheared, zoomed into, or flipped so that the final model would be able to accurately predict the class of the image even under unusual conditions. Once preprocessing was completed, the next step was to decide on the best hyperparameters to maximize validation metrics. This was done through a table of different combinations of parameters, listed below:

A	B	C	D	E	F	G
	Dropout	Learning rate	# of steps	# of epochs	Training Acc	Testing Acc
1	0.5	1.00E-05	150	30	86%	73.30%
2	0.25	1.00E-04	125	31	97.50%	80%
3	0.35	1.00E-06	170	32	53.30%	33.30%
4	0.15	1.00E-03	130	31	91.67%	30%
5	0.2	1.00E-04	120	31	97.50%	73.30%
6	0.3	1.00E-04	130	31	95.83%	86.67%
7	0.3	1.00E-04	130	60	100.00%	80.00%
8	0.3	1.00E-04	130	120	100.00%	93.33%
10	0.3	1.00E-04	130	150	99.17%	96.67%

Figure 2: Hyperparameter Table

When training the model, we prevented overfitting by splitting the dataset into a training and testing set. Data was split into 80% training and 20% testing respectively. Through running the model with these different combinations, the last trial was found to yield the best testing accuracy, and thus the hyperparameters were finalized.

Results and Discussion

Using the hyperparameters from the last trial, the testing accuracy was found to be 96.67%. As expected, this value was a bit less than the training accuracy, which was 99.17%. The training and testing losses were 0.019 and 0.0596, respectively.

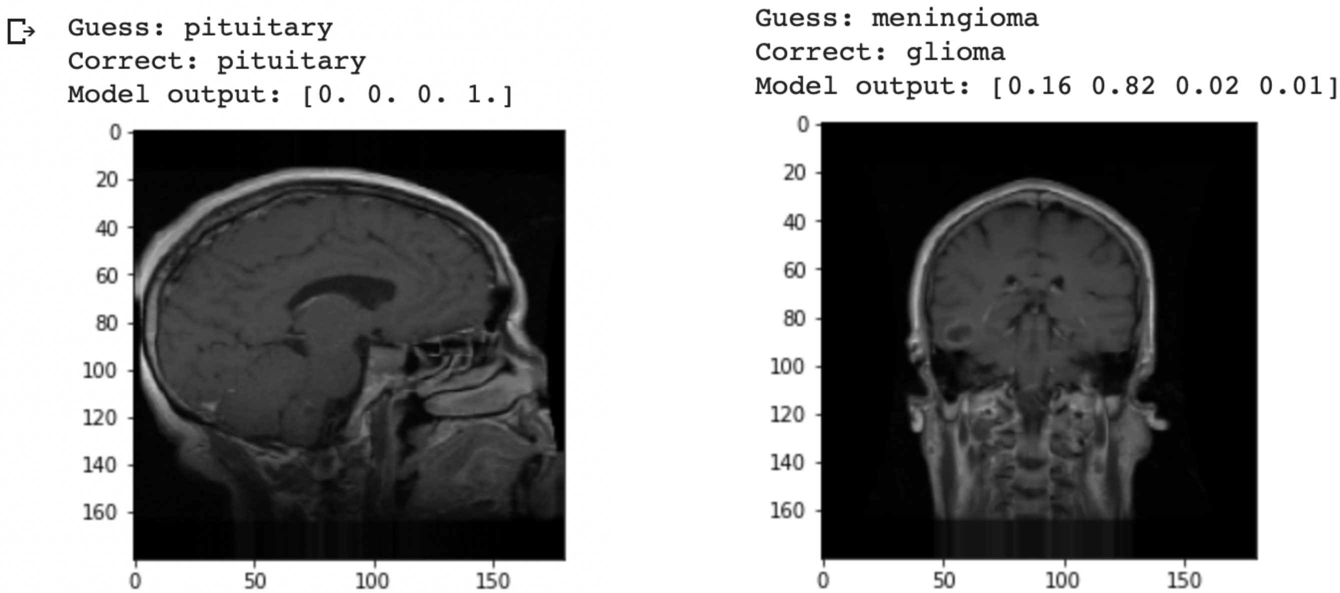


Figure 3: Model prediction of given images

Figure 3 shows an example of both a correctly and incorrectly classified image. In the example on the left, the model correctly predicts with 100% certainty that a pituitary tumor is present. However, on the right, the model is relatively unsure of its prediction of meningioma, and is ultimately wrong, as the image actually contains a glioma tumor. This is a good way to check the model on a case by case basis to see how it classifies individual images in the dataset. Next, I looked at the loss and accuracy functions, which are shown in Figure 4.

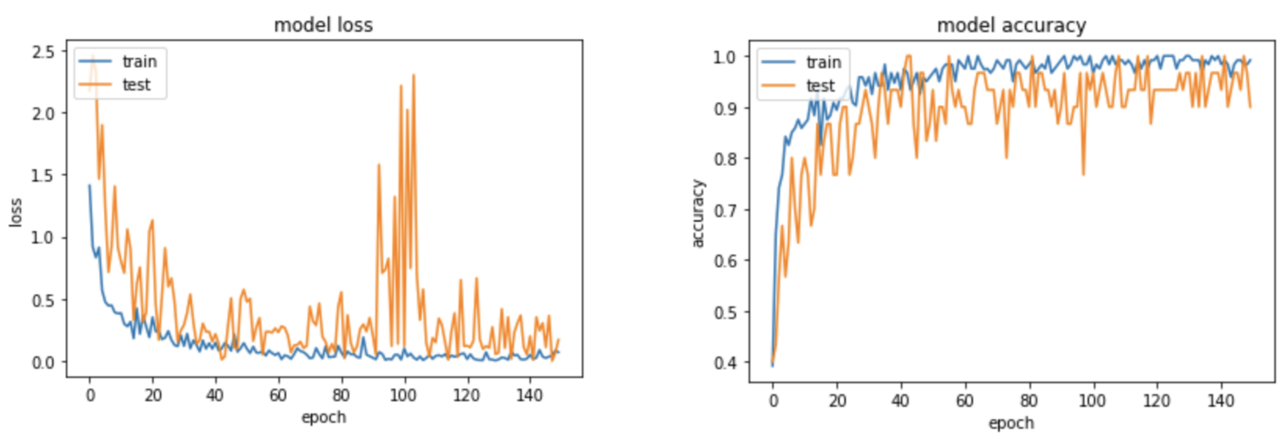


Figure 4: Model loss and accuracy graphs

The graph on the left in Figure 4 depicts the model's loss curve going down exponentially as the number of epochs increases. There is a spike around epoch 100 as the model changes its approach to classifying the images, ultimately leading to a lower loss, meaning the model is quite accurate. This loss graph corresponds with the model's accuracy graph shown on the right in Figure 4. As the loss curve decreases, the model's accuracy increases, which makes sense as it becomes better at correctly classifying given images in the dataset.

While the accuracy measurement above is a good way of getting a broad overview of model performance, using a confusion matrix allows us to more closely inspect the results for each class. Thus, a confusion matrix was generated and is shown below in Figure 5, where each row represents the true label of the data, and each column represents the predicted label from the model output:

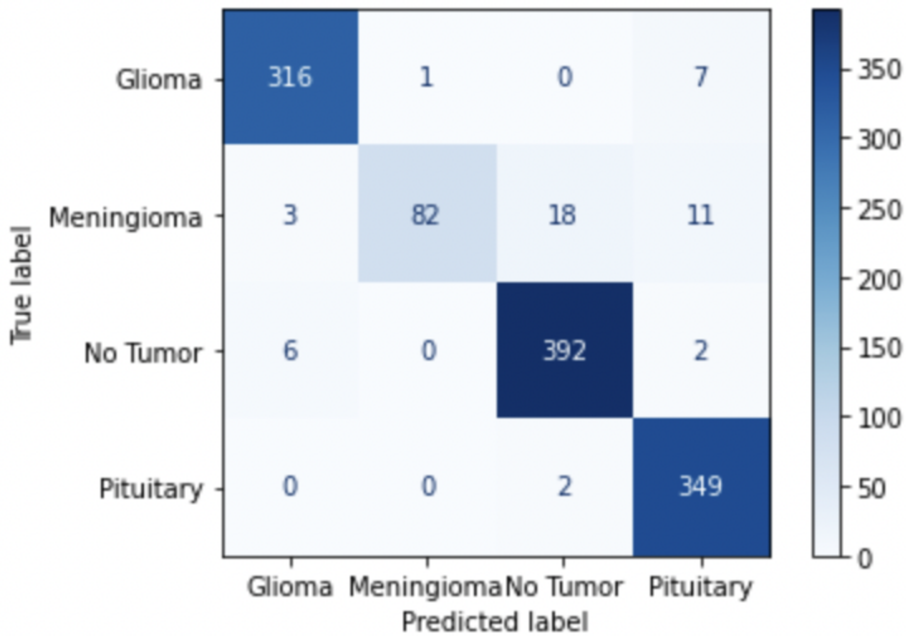


Figure 5: Confusion matrix of the trained Inception Resnet Model

Based on this confusion matrix, it can be seen that the model performed best on determining if a given image either contained a glioma or pituitary tumor, or no tumor at all. However, in the meningioma class, the model only correctly predicted 82 out of 114 images that contained a meningioma tumor. This low accuracy is most likely a result of the rarity of high grade meningioma tumors in the given dataset. There were only 114 meningioma images in the dataset compared to the large amounts of data points in the other classes, which explains this result.

It is also important to note that the model did not have many false negatives, which is beneficial in that it means there were not many cases where an image contained a tumor but was classified as having no tumor. Based on these results, the model has a recall of 95% (392/412), which indicates that only 2% of any kind of cancer cases were not detected. Other sources of possible error could come with the quality of the dataset and that there were some images that were not in the best quality or were taken at obscure angles.

Conclusions

By training an Inception Resnet CNN model on our MRI scan dataset, we have created an accurate brain tumor classification tool that can be implemented in practice. The model's high accuracy values means that it can be trusted as a reliable indicator of the presence and type of tumor in a brain MRI scan. However, it should be noted that the model shows some weakness when classifying images that contain a meningioma tumor, and that the model has a recall rate of 95%.

In the future, it could be beneficial to incorporate other, more high quality datasets to see if the model's accuracy can be improved even further. These other datasets could also be taken from different methods of scanning a brain that do not involve using an MRI, such as a CT scan. In addition, other forms of CNNs could be tested on the dataset to again see if the accuracy can be improved. It could also be beneficial to compare our model's accuracy to the accuracy of a traditional diagnosis procedure, in order to better understand the real-world application of AI models to medical diagnosis.

Acknowledgments

I would like to acknowledge my mentor, Nic Thibodeaux, for helping me throughout the project in fixing various bugs and problems with the code.

References

[1] Huang, J., Shlobin, N., Lam, S., DeCuypere, M., 2021, "Artificial Intelligence Applications in Pediatric Brain Tumor Imaging: A Systematic Review", World Neurosurgery, pp. 99-105, https://www.sciencedirect.com/science/article/pii/S1878875021015655?scrlybrkr=e93f630c

[2] McFaline-Figueroa, J.R., Lee, E.Q., 2018, "Brain Tumors", The American Journal of Medicine, pp. 874-882, https://www.sciencedirect.com/science/article/abs/pii/S0002934318300317

[3] Mambray, M.C., Barajas, R.F., Cha, S., 2015, "Modern Brain Tumor Imaging", Synapse, pp. 8-23, https://synapse.koreamed.org/articles/1059671

[4] Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. 2016, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning". arXiv, 1602.07261. Retrieved from https://arxiv.org/abs/1602.07261v2

[5] Rawat, W., & Wang, Z. 2017, "Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review". Neural Comput., 29(9), 2352–2449. doi: 10.1162/neco_a_00990