# Distributed Deep Learning Framework for Brain Tumor Classification

**Anna Huang, Xinyi Lyu, Letitia Su, Vicky Yeh**
University of Washington
{annah503, xlyu7, suruoxi, shuyiyeh}@uw.edu

## Abstract

Brain tumors demand early detection for effective treatment, but relying on manual diagnosis via MRI scans is time-consuming and prone to error. The need to effectively and accurately identify brain tumors from Magnetic Resonance Imaging (MRI) images is paramount for patient care and survival rates. In this paper, we leverage recent advancements in medical imaging, particularly deep learning, to assess the effectiveness of previously trained Convolutional Neural Network (CNN) and Vision Transformers (ViT) models in accurately identifying and classifying brain tumors. Our goal is to determine suitable algorithms for this critical diagnostic task.

## 1 Introduction

Brain tumors represent a significant and severe category of cancer, where timely diagnosis is crucial for optimizing treatment outcomes. Magnetic Resonance Imaging (MRI) is a crucial tool in the diagnosis and management of brain tumors, providing detailed images of soft tissues within the brain in high spatial resolution. Brain tumors can vary significantly in their type, location, and aggressiveness, thereby necessitating tailored treatment approaches. For instance, Meningioma is the most common primary brain tumor, accounting for more than 30 percent of all brain tumors that may rely on treatment such as observation, surgery, radiation therapy, or drug therapy. On the other hand, Gliomas, specifically Astrocytoma is a fast-growing brain tumor that requires aggressive and more intensive treatments. Given the critical importance of accurately classifying brain tumors for patient care, our study aims to identify the most effective pre-trained models for classifying the different types of brain tumors, with an emphasis on speed and accuracy.

Our study focuses on case study of supervised learning methods that utilize labeled data from brain tumor MRI scans to train multiple deep-learning models. Leveraging the power of pre-trained CNNs and ViTs, we fine-tune and adapt these models to our specific task of brain tumor classification. Evaluation of these models is crucial, given the potential consequences of false negatives and false positives for patient care. Our evaluation metrics prioritize minimizing false negatives due to the significant impact on patient outcomes. The metrics include Accuracy, Recall, Precision, F2 Score, and Training Time. Additionally, we consider the computational costs and the Area Under the Receiver Operating Characteristic Curve (AUC-ROC) to comprehensively assess the model performance.

In summary, this study aims to answer key research questions regarding the classification accuracy and processing time for brain tumor MRI images and the effectiveness and limitations of CNNs and ViTs in multi-label brain tumor classification. Our methodology integrates several key processes in Fig. 1: standardizing and augmenting input MRI images to ensure consistency and improve model robustness; utilizing state-of-the-art Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) to extract meaningful features from the processed images; and employing a softmax layer to classify the features into distinct tumor categories, such as Astrocytoma, Glioblastoma, Oligodendroglioma, and others, thus enabling multi-label classification.

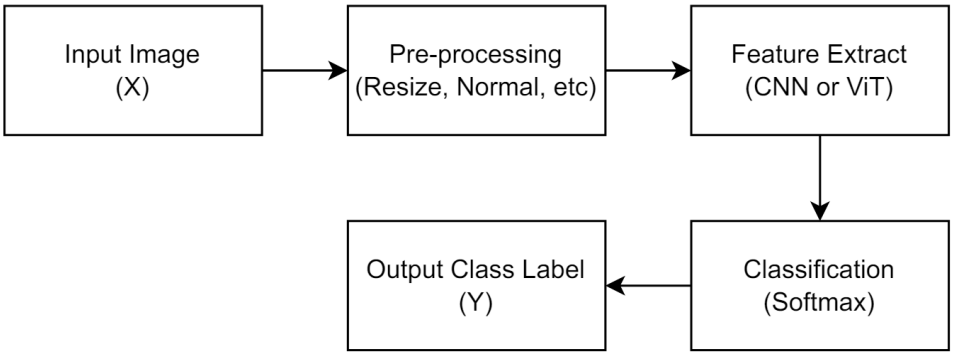



**Figure 1:** The framework that streamlines the process from image input to tumor classification.

## 2 Related Work

Brain tumor classification is a thoroughly studied topic, with numerous analyses and studies dedicated to identifying and categorizing different types of brain tumors.

Haq et al.[1] introduced DACBT, a Deep Learning Approach for brain tumor classification using MRI data in an IoT healthcare environment. The dataset consists of 233 enhanced images of brain tumor patients. The author proposes a deep learning approach to classify brain tumors using MRI images in IoT healthcare by leveraging pre-trained CNN architectures such as ResNet-50, VGG-16, Inception V3, DenseNet201, Xception, and MobileNet to improve accuracy results. Additionally, significant pre-processing, parameter fine-tuning, and data augmentation techniques enhanced performance with the ResNet-50-CNN model achieving 99.90% accuracy. Similar to our task, we will leverage some of the pre-processing and fine-tuning methods to improve our models in classifying brain tumors.

Tandel et al.[2] explore the role of ensemble deep learning for brain tumor classification across multiple MRI sequences, including T1-Weighted (T1W), T2-weighted (T2W), and fluid-attenuated inversion recovery (FLAIR). The study employs many pre-trained deep learning models, including AlexNet (8-layer), VGG16 (16-layer), ResNet 18 (18-layer), GoogleNet (22-layer), and ResNet 50 (50-layer), each thoroughly introduced with detailed parameters and layer information. We decide to reference the methodology by choosing 5 models for comparison, starting from those with highest accuracy and f-1 score, which are DenseNet, Inception, Xception, VGG and ResNet, respectively. However, their ensemble algorithm MajVot focus solely on predicted classes, potentially overlooking important information from the probability predictions of different models. Due to the weakness of ensemble of the same learning structure, we decided to include more structures such as Vision Transformers (ViTs) to increase the diversity of model type rather than just applying CNN models.

Tummala et al. [3] explore an ensemble of ViTs models for the automated classification of brain tumors from MRIs. They fine-tuned and evaluated various pre-trained ViT models (B/16, B/32, L/16, and L/32) and compared their performance for detecting three types of brain tumors: meningiomas, gliomas, and pituitary tumors. The ViT ensemble demonstrated exceptional classification accuracy, with results surpassing 98%, reinforcing the potential of these models in our project. However, this research was based on a relatively small dataset (3064 images) while ViTs generally require large datasets for optimal performance. Moreover, ViTs, especially in an ensemble, require significant computational resources for both training and inference. Thus, we need to compare both the effectiveness and efficiency between CNNs and ViTs on large dataset.

Xu and Ma [4] present a distributed convolutional network based on Spark (Dis-CNN) to streamline training complexity in image classification tasks. They address the training complexity issue with large-scale datasets. The Dis-CNN model is supposed to improve upon traditional CNNs by integrating random and k-means initialization methods for convolution kernel parameters. The authors experimented on the CIFAR-10 dataset showcasing the Dis-CNN's superiority in achieving higher accuracy and recall rates while exhibiting strong parallel performance. The evaluation results indicate that the Dis-CNN model outperforms traditional CNN approaches in terms of overall accuracy,

achieving an accuracy of 76.25% with fewer epochs compared to Rand-CNN and KMean-CNN. This paper provides useful insights into optimizing distributed training of CNNs on Spark, which could inform a similar distributed strategy for other brain tumor classification structures. Adapting distributed gradient descent to ViTs could potentially lead to more efficient training for large-scale medical image classification tasks.

## 3 Data Collection

The primary dataset for this project is the ReMIND [5] dataset provided by the National Institutes of Health (NIH). This dataset comprises 85,057 high-resolution brain MRIs from 114 subjects featuring various types of brain tumors, providing a rich source for training deep learning models to recognize and classify tumor margins and characteristics. For each patient, we have preoperative MRI sequences: native T1-weighted (T1), contrast-enhanced T1-weighted (ceT1), native T2-weighted (T2), and T2-weighted fluid-attenuated inversion recovery (T2-FLAIR). Each case typically contains multiple tumor segmentations created during the surgical planning stage, potentially facilitating the learning process.

Additionally, the diversity of the dataset, in terms of tumor types and severity grades in anonymous clinical metadata, presents an opportunity to develop robust models capable of handling a wide range of diagnostic scenarios, including brain tumor classification and severity prediction.

The data is accessible through the NIH's open-source portal, which ensures compliance with confidentiality and research standards.

### 3.1 Exploratory Analysis

Since the dataset contains both tumor types and severity grade labels, we started with a few exploratory analyses to assess the distribution of their classes and the potential relationships between the two variables.

| Brain Tumor Type | Count |
|---|---|
| Astrocytoma | 25477 |
| Atypical meningioma | 498 |
| DENT | 1251 |
| Ependymomas | 829 |
| Glioblastoma | 23548 |
| Glioma | 890 |
| Glioneuronal | 827 |
| Oligodendroglioma | 18571 |
| Papillary Glioneuronal | 829 |
| Others | 12337 |

| WHO Grade | Count |
|---|---|
| Not Assigned | 13227 |
| 1 | 2080 |
| 2 | 23969 |
| 3 | 18052 |
| 4 | 27729 |

**Table 1:** Distribution of Labels

There appears to be noticeable imbalance in brain tumor classes. Astrocytoma, Glioblastoma Oligodendroglioma and are the most common tumor types, whereas other types like atypical Meningioma and Ependymomas are relatively rare. The Others category represents a collection of shared metastatic features caused by cancers originated from other body parts. The distribution of severity based on WHO grades is also imbalanced, with a significant number of cases having no grade assigned and others predominantly falling under Grades 2 and 4. Therefore, our model training must carefully handle the skewed distribution by taking advantages of pre-processing techniques.

### 3.2 Data pre-processing

The data are originally slices of 3D brain scans made by different MRI sequences, resulting in slightly different image structures. We started to clean the data based on their description to remove noises of empty or skull slices, crop as mush black area as possible, recenter and resize the images to derive a consistent format across the data.

The remaining pre-processing steps included normalization to enhance model training procedures. The standardization ensures that the model does not misinterpret variations in pixel intensity due to different imaging settings or machines as differences in pathological features.

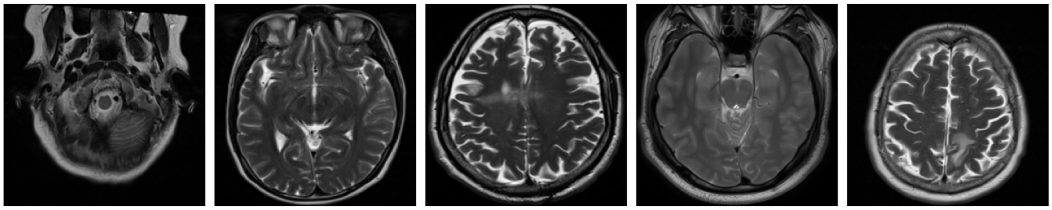
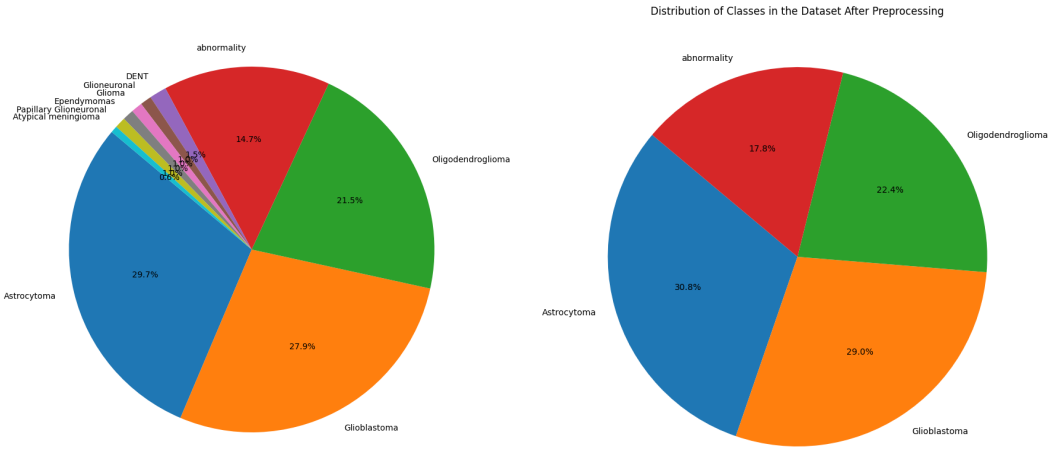


**Figure 2:** Sample Visualizations after pre-processing.

The processed dataset contains 78,560 images of 10 classes as shown by Fig. 3a, from 114 patients whose number of images vary from 399 to 1275. To resolve this extreme imbalance, we excluded the rare classes ($\leq 1.5\%$ of total) and down-sampled by randomly selecting 399 images per patient. Thus, we retained 42,693 samples, covering 4 classes in our experiments, as shown by Fig. 3b.




**(a)** Class distribution before processing

**(b)** Class distribution after processing

**Figure 3:** Class distribution after down-sampling.

Lastly, we randomly split the data into training (70%), validation (15%), and test (15%) sets by patients, so our experiments won't be biased by highly similar samples belong to the same patient. This separation ensures that the model is trained on one set of data, tuned with another, and finally evaluated on unseen data to assess its real-world applicability and performance.

## 4 Methodology

Our objective is to harness pre-trained models for transfer learning in multi-class classification of brain tumors. Specifically, we selected five CNN models and one advanced ViT model.

### 4.1 Model Description

**Pre-Trained CNN Models:**

- **VGG19**: The VGG model is notable for improving performance by increasing the network's depth. It employs an architecture with very small (3x3) convolution filters. The authors of

VGG discovered that beyond 16 - 19 layers, further increasing the depth did not yield significant improvements in performance. As a result, they developed the VGG19 model, which has 19 layers and is often used for transfer learning. This VGG19 model is utilized in our project.

- **InceptionV3**: For previous models, the computational requirement grows as the number of layers in neural network increases. Inception networks were developed to enhance the capability of deep neural networks while efficiently using computational resources. Thus, InceptionV3 is ideal for image classification tasks, using constant computational budget even when the depth and width of the network increase.

- **DenseNet121**: DenseNet was designed to address the vanishing gradient problem, where gradients diminish as they propagate through many layers, leading to decreasing accuracy. In traditional deep neural networks with many layers, information of details will disappear as it traverses the network. DenseNet resolves this by implementing dense connections, ensuring that each layer receives direct input from all preceding layers. In this project, we utilize DenseNet with 121 layers.

- **Xception**: The Xception model, which stands for "Extreme Inception," builds on the ideas of Inception while replacing the standard Inception modules with depthwise separable convolutions, leading to more efficient computation and greater performance. This architecture is known for its high accuracy and efficiency. Thus, we use Xception to leverage its ability to handle intricate features in complex image classification tasks.

- **ResNet50**: The ResNet50 model is a 50-layer deep convolutional network known for its use of residual learning, which helps mitigate the vanishing gradient problem by allowing gradients to flow directly through the network using shortcut connections. This design enables the training of very deep networks without degradation in performance. ResNet50 is renowned for its robustness and high performance in various image recognition tasks, making it an excellent choice for the project.

**Pre-Trained Vision Transformer Models:**

- **ViT**: In brain tumor classification tasks, ViT offers several advantages. Firstly, its global attention mechanism allows it to analyze relationships between distant parts of an image, which is crucial when distinguishing subtle tumor characteristics. Secondly, ViT's patch-based processing reduces computational complexity compared to processing entire images, while still maintaining high classification accuracy. Additionally, ViT's ability to scale up effectively with larger datasets and higher resolution images makes it a strong candidate for medical imaging applications, where high-quality and large datasets are common. Therefore, we have chosen ViT, with the expectation that its transformer-based architecture will excel at capturing the complex and varied features present in medical imaging data.

### 4.2 Transfer Learning Implementation

We employed transfer learning to adapt the six pre-trained models for the task of classifying four distinct brain tumor types: Astrocytoma, Glioblastoma, Oligodendroglioma, and Abnormality. The tumor type labels were transformed from string identifiers into a four-dimensional one-hot encoding format, enhancing the model's ability to distinguish between the classes effectively.

For CNN models, we froze the early layers and fine-tune only the later layers, such as the last 10 layers. This fine-tuning focuses on adjusting the deeper, more specific layers of the network to our particular task, significantly reducing the risk of overfitting and the computational load during training. We also replaced the top layer with GlobalAveragePooling2D layer and added a Dense layer to transform the model output into four classes. We trained the CNN models on the entire dataset via Google Colab's A100 GPU.

For the ViT model, We adapted a pre-trained baseline vision transformer [6] from huggingface hub, with the MLP classification head initialized to 4 classes. We used a pre-trained

ViTFeatureExtractor to preprocess our images into a format acceptable by the ViT. However, due to the CUDA RAM limitation, we were only able to fit one-third of the dataset by downsampling the number of images per patient.

The training process was optimized using the Adam optimizer for CNNs and the AdamW optimizer for the ViT model to mitigate potential overfitting. For all models, we utilized categorical cross-entropy loss function, and a ReduceLROnPlateau learning rate scheduler. By leveraging the power of Adam optimization, we sought to efficiently minimize the categorical cross-entropy loss, which measures the difference between the predicted probabilities and the true labels. Through these configurations, we aimed to develop a robust and effective model capable of accurately identifying tumor types based on input image.

The parameters for each model, including batch size, number of epochs, and learning rate, were carefully chosen based on preliminary tests that gauged each model's response to different settings. We strategically selected and optimized model parameters to balance performance with the constraints imposed by limited computational resources.

## 5 Evaluation Methods

False negatives occur when the model fails to detect an actual tumor, potentially delaying treatment or leading to the absence of treatment altogether, thus allowing the brain tumor to progress, which can increase mortality rates. Additionally, false negatives provide false reassurance to patients who are erroneously told they do not have a tumor when they do. On the other hand, false positives arise when the model incorrectly identifies a tumor where none exists, potentially prompting unnecessary follow-up procedures and additional testing. In the context of brain tumor detection, false negatives are particularly concerning because they represent missed opportunities for early intervention and treatment, ultimately leading to more severe consequences for patients.

As a result, we are more concerned about false negatives and the metrics selected for evaluation are Accuracy, Recall, Precision, F2 Score, and Training Time.

- **Accuracy** calculates the proportion of correctly classified observations (both true positives and true negatives) over the total number of observations.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

- **Recall** measures sensitivity by calculating the number of observations correctly classified as true positives over the total number of true positives and false negatives.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

- **Precision** quantifies the proportion of true positives among all positive predictions, computed as the number of true positives divided by the sum of true positives and false positives.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

- **F2 Score** is similar to the F1 Score but emphasizes recall (minimizing false negatives) over precision. It is calculated as (1 + $\beta^2$) * (precision * recall) / ($\beta^2$ * precision + recall), where $\beta$ is a parameter that controls the emphasis placed on recall ($\beta > 1$ favors recall over precision).

$$F_2 = \frac{(1 + \beta^2) * \text{precision} * \text{recall}}{(\beta^2 * \text{precision} + \text{recall})}$$

- **Training Time** The time taken to train the model on the training data.

- **Convergence (Training vs Validation Loss)** Convergence is when the training process stabilizes, as seen by consistent training and validation losses or a plateau in the graph over epochs, indicating the model has learned the data's patterns.

- **ROC-AUC curve** The Area Under the Receiver Operating Characteristic Curve measures a model's class separation ability across different thresholds by plotting True Positive Rate (TPR) against False Positive Rate (FPR). A higher AUC suggests better performance, summarizing the model's overall effectiveness.

# 6 Results and Discussions

## 6.1 Experiments with Individual Models

- **VGG19**: We trained the model for 20 epochs with a batch size of 16. The training accuracy reached 0.984 with the testing accuracy also reaching 0.931. The validation loss and accuracy are relatively stable from the loss and accuracy plots.

- **InceptionV3**: We trained the model for 30 epochs with a batch size of 8. The training accuracy is only 0.516 so far, with the testing accuracy around 0.477. From the loss and accuracy plots, we observed that although there was no overfitting, the value is much worse than other models. This may because the model was not fully converged yet, due to limited computational resources we were not able to train it further, but we expect the performance to be better while fully converge.

- **DenseNet121**: We trained the model for 30 epochs with a batch size of 32. The training accuracy reached 0.997, as well as the testing accuracy approached 0.871. However, the loss and accuracy plots show fluctuation in validation set, suggesting possible improvement with parameter adjustments. We have kept the results due to limited computational resources.

- **Xception**: We trained the model for 30 epochs with a batch size of 32. The training accuracy was 0.993, while the testing accuracy reached 0.884. From the loss and accuracy plots, we observed the validation loss and accuracy are both fluctuating, and expect the results can be further improved with more epochs and a smaller learning rate.

- **ResNet50**: We trained the model for 30 epochs with a batch size of 8. The training accuracy was 0.960, while the testing accuracy also reached 0.934. Addionally, from the loss and accuracy plots, we observed that this model is the most stable one because the training and validation line are closer to each other without significant oscillation.

- **ViT**: We fine-tuned the model for 50 epochs with a batch size of 32. The training accuracy reached 0.96 but validation accuracy converged to around 0.63 at best, while the testing accuracy only reached 0.58. From the plots, we observed significant fluctuations in both curves, indicating chances to further improve the performance with smaller learning rate.

## 6.2 Comparison of All Models

| Model | Type | Accuracy | | | Loss | |
|---|---|---|---|---|---|---|
| | | Training | Validation | Testing | Training | Validation |
| VGG19 | Pre-trained CNN | 0.984 | **0.940** | 0.931 | 0.052 | 0.299 |
| InceptionV3 | Pre-trained CNN | 0.516 | 0.475 | 0.477 | 1.111 | 1.177 |
| DenseNet121 | Pre-trained CNN | **0.997** | 0.879 | 0.871 | 0.027 | 0.371 |
| Xception | Pre-trained CNN | 0.993 | 0.886 | 0.884 | **0.025** | 0.397 |
| ResNet50 | Pre-trained CNN | 0.960 | 0.937 | **0.934** | 0.127 | **0.253** |
| ViT | Pre-trained ViT | 0.962 | 0.630 | 0.584 | 0.166 | 0.574 |

**Table 2:** Training, validation, and testing accuracy and loss of five pre-trained CNN models and one pre-trained ViT model, with the best score for each highlighted.

DenseNet121 achieves the highest training accuracy, VGG19 attains the highest validation accuracy, and ResNet50 secures the highest testing accuracy and the lowest validation loss. Overall, these

three models show comparable performance with very low losses in both training and validation. Similarly, Xception also demonstrates competitive accuracy and low loss, performing on par with the top models. In contrast, InceptionV3 and ViT show varying levels of performance, with InceptionV3 having moderate accuracy but higher loss values, and ViT displaying lower accuracy compared to the CNN models (see Table 2, Fig 5 and Fig 6).

| Model | Type | Metrics | | | Training |
|---|---|---|---|---|---|
| | | Precision | Recall | F2 Score | Training Time (sec) |
| VGG19 | Pre-trained CNN | 0.931 | 0.931 | 0.931 | **1004** |
| InceptionV3 | Pre-trained CNN | 0.477 | 0.477 | 0.472 | 1740 |
| DenseNet121 | Pre-trained CNN | 0.871 | 0.871 | 0.871 | 1209 |
| Xception | Pre-trained CNN | 0.885 | 0.884 | 0.884 | 1113 |
| ResNet50 | Pre-trained CNN | **0.934** | **0.934** | **0.934** | 1936 |
| ViT | Pre-trained ViT | 0.260 | 0.229 | 0.215 | 3702 |

**Table 3:** Evaluation of Precision, Recall, F2 Score, and Training Time in seconds for five pre-trained CNN models and one pre-trained ViT model.

ResNet50 demonstrated the strongest performance across all metrics, closely trailed by VGG19 with around half of ResNet50's training time. With the highest F2 Scores, ResNet50 performed best at lowering false negatives. DenseNet121 and Xception displayed moderate performance, while InceptionV3 and ViT, with the latter having a longer training time, showcased comparatively lower scores across the metrics. (see Table 3).

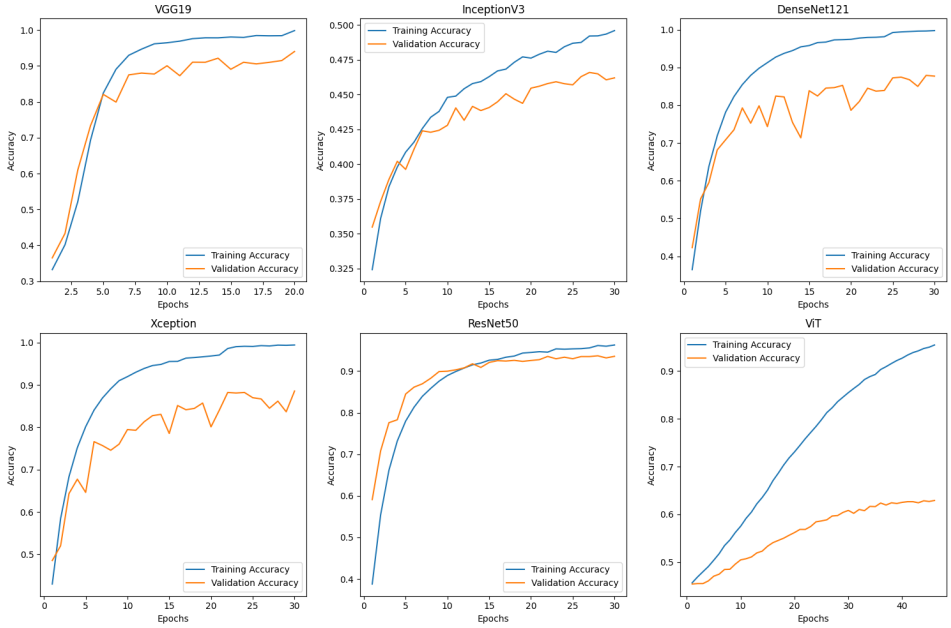



**Figure 4:** Training and validation accuracy curves for each model reveal consistently high accuracy for VGG19 and ResNet50, whereas the accuracy of other models diverges as number of epochs increase.

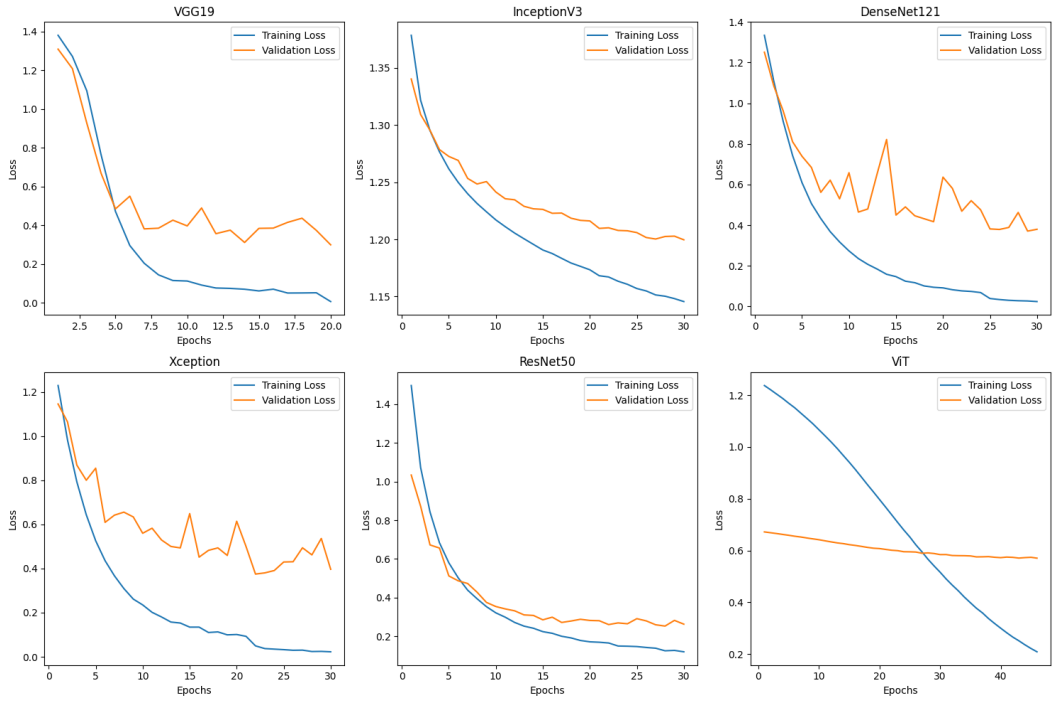



**Figure 5:** Training and validation loss curves for each model. DenseNet121, and Xception exhibit significant overfitting with validation loss diverging upward, while InceptionV3 and VGG19 show mild overfitting. ResNet50 performs the best, while ViT demonstrates the poorest performance.

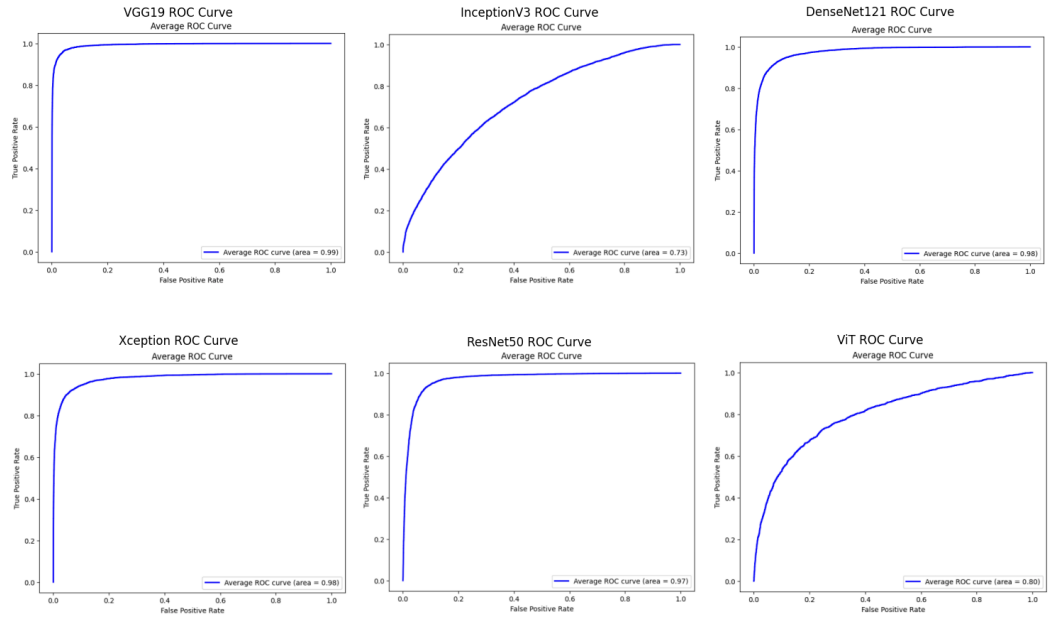



**Figure 6:** ROC-AUC curves for each model. VGG19 achieves the highest AUC score at 0.99, followed by DenseNet121 and Xception at 0.98, ResNet50 at 0.97, ViT at 0.80, and InceptionV3 with the lowest score at 0.73.

In conclusion, VGG19 and ResNet50 are the top models for brain tumor classification via MRI scans, with exceptional training and testing accuracy (0.984/0.931 for VGG19, 0.960/0.934 for ResNet50). VGG19 achieves impressive results with the best computational efficiency (1004 seconds), while

ResNet50 has better convergence (see Figure 5). ViT, the only pre-trained Vision Transformer model, performs poorly, suggesting it is challenging to adapt to brain tumor classification tasks.

## 7 Limitations and Future Work

Despite the advancements in Vision Transformer (ViT) models, there are notable limitations that impacted the generalization performance of our brain tumor classification task:

- **Computational Resources**: ViTs are computationally intensive, requiring more memory and processing power for both training and inference compared to CNNs. This increased demand can be a limiting factor when working with high-resolution medical images and extensive datasets, as it may lead to longer training times and the need for more powerful hardware.
- **Overfitting**: Due to the complexity of ViT models and the limited size of our dataset, the model showed signs of overfitting. While the training accuracy was high, the validation and test accuracies were much lower, indicating that the model was not generalizing well to unseen data. Regularization techniques, such as dropout and data augmentation, were applied, but they were not sufficient to fully mitigate the overfitting issue.
- **Lack of Inductive Bias**: ViTs do not inherently encode spatial hierarchies as CNNs do. This means that they may struggle to capture local patterns and structures within the images unless the dataset is large enough to learn these relationships from scratch. Brain tumor images often require understanding of fine-grained details and local structures, which CNNs are naturally better at due to their architectural design.
- **Feature Extractor Tuning**: While ViTs can benefit from pre-trained feature extractors, the feature extractor itself might not be fine-tuned to the specific nuances of medical imaging. Our initial approach used a pre-trained feature extractor without further fine-tuning, which may have limited the model's ability to adapt to the specific characteristics of brain tumor images.
- **Complexity of Hyperparameter Tuning**: ViTs have more hyperparameters that need careful tuning compared to CNNs. The optimal settings for these hyperparameters can vary significantly depending on the dataset and the specific task, making it challenging to find the best configuration without extensive experimentation.

Due to the complex nature of ViT, our sample size and computational resources are limited for it to learn the intricacy to differentiate different brain tumors. While for the CNN models, after extensive data pre-processing as to fine-tune the models with sufficient and balanced data, the results show significant improvement, with models like VGG19 and ResNet50 demonstrate great potential in accurately classifying four major brain tumor types with only brain MRI images.

In the future, we aim to extend our classifications to include more than the current four types of brain cancers, incorporating even the rare ones by up-sampling those classes. Building on the success of our models in differentiating between the four major types of brain tumors, we plan to extend their capabilities to also identify the severity of the tumors. With additional time and computational resources, we believe the Vision Transformer model can be further adapted and improved for brain tumor classification tasks.

## 8 Conclusion

This study has effectively demonstrated the application of deep learning models, particularly pre-trained Convolutional Neural Networks and a Vision Transformer, for the multi-class classification of brain tumors using MRI images. Our experiments reveal that models like VGG19 and ResNet50 offer robust performance and high accuracy, proving to be valuable in the medical imaging domain. Despite facing challenges such as limited computational resources which particularly affected the Vision Transformer's performance, our findings are promising. Overall, we anticipate our findings to contribute to brain tumor diagnostics, providing rapid, accurate insights that can assist healthcare professionals and inform treatment decisions, ultimately leading to better patient outcomes.

**Team Member Contributions:**

Anna Huang: Problem formulation, tabulating final results, running tests, writing up the report of evaluation methods and results

Xinyi Lyu: Problem formulation, crawling the data, preliminary data analysis, plotting graphs during data analysis, running tests, writing up the report

Letitia Su: Balancing and preparation of experiment datasets, coding up the ViT algorithm, running experiments, analyzing limitations and improvements in report

Vicky Yeh: Coding up the CNN algorithm, running tests, writing up the report of methodology and results

## References


[1] Amin ul Haq, Jian Ping Li, Shakir Khan, Mohammed Ali Alshara, Reemiah Muneer Alotaibi, and CobbinahBernard Mawuli. Dacbt: Deep learning approach for classification of brain tumors using mri data in iot healthcare environment. *Nature News*, Sep 2022. URL `https://www.nature.com/articles/s41598-022-19465-1`.

[2] Gopal S. Tandel, Ashish Tiwari, Omprakash G. Kakde, Neha Gupta, Luca Saba, and Jasjit S. Suri. Role of ensemble deep learning for brain tumor classification in multiple magnetic resonance imaging sequence data. *Diagnostics*, 13(3):481, Jan 2023. doi: 10.3390/diagnostics13030481.

[3] Sudhakar Tummala, Seifedine Kadry, Syed Ahmad Chan Bukhari, and Hafiz Tayyab Rauf. Classification of brain tumor from magnetic resonance imaging using vision transformers ensembling. *Current Oncology*, 29(10):7498–7511, 2022. ISSN 1718-7729. doi: 10.3390/curroncol29100590. URL `https://www.mdpi.com/1718-7729/29/10/590`.

[4] Xu, Jiangfeng and Ma, Shenyue. Image classification model based on spark and cnn. *MATEC Web Conf.*, 189:03012, 2018. doi: 10.1051/matecconf/201818903012. URL `https://doi.org/10.1051/matecconf/201818903012`.

[5] The cancer imaging archive (tcia)_2024. *The Cancer Imaging Archive (TCIA)*, Jan 2024. URL `https://doi.org/10.7937/3RAG-D070`.

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *Advances in Neural Information Processing Systems*, 33:2020–2033, 2020. URL `https://doi.org/10.48550/arXiv.2010.11929`.