Lightweight Deep Learning Framework for Brain Tumour Classification

MSc Research Project
Data Analytics

Rohit Salvi
Student ID: x21208832

School of Computing
National College of Ireland

Supervisor: Paul Stynes

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Rohit Salvi
Student ID:	x21208832
Programme:	Data Analytics
Year:	2022-2023
Module:	MSc Research Project
Supervisor:	Paul Stynes
Submission Due Date:	14/08/2023
Project Title:	Lightweight Deep Learning Framework for Brain Tumour Classification
Word Count:	5266
Page Count:	19

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Rohit Narayan Salvi
Date:	17th September 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Lightweight Deep Learning Framework for Brain Tumour Classification

Rohit Salvi
x21208832

Abstract

The National Brain Tumour Society states that there are over 100 different forms of primary brain tumours, such as gliomas, meningiomas, pituitary and so on. Brain tumour diagnosis involves detecting the type of brain tumour and its severity. The challenge is to accurately identify and classify a brain tumour with limited computation. This research proposes a lightweight deep learning framework for brain tumour classification. The proposed framework combines a machine learning classification model and weight pruning optimization technique to detect brain tumours with limited computation. The classification model is implemented using Convolutional Neural Networks(CNN) and the magnitude-based weight pruning technique is used to optimise the classification model. A Brain Tumour Magnetic Resonance Imaging(BTMRI) dataset of 7023 MRIs representing 4 distinct classes of brain tumours namely – glioma, meningioma, pituitary and no tumour is utilised to analyse and evaluate a proposed framework. The results of the proposed framework are presented in this paper based on accuracy, sensitivity, specificity and loss function. Results of the proposed framework show an accuracy of 87.26% and loss 0f 0.39 after 25 epochs. The proposed framework is 4.65% more accurate and has 15% lower loss than the state-of-the-art CNN for multiclass brain tumour classification. This research shows promise for aiding patients in getting an early view of their tumour type.

1 Introduction

According to statistics from World Health Organization(WHO), brain tumours will be responsible for the death of 9.5 million patients globally in the upcoming decades(Veeranki et al.; 2023). Particularly in India, over 28,000 cases of brain tumours are reported each year and brain tumour is responsible for the death of 24,000 people annually in India[1]. In India, the high death rate from brain tumours is mostly due to a lack of specialised consulting. The majority of India's 1800 neurosurgeons reside in metropolitan areas, leaving 800 million Indians in suburban and rural areas with very little direct access to even primary general neurosurgeon(Ganapathy; 2022). The patient's chances of survival can be increased by early diagnosis of these brain tumours(Bayram et al.; 2023). Due to the different forms and structures of brain tumours, the diagnosis of the brain tumour is a challenging task. The diagnosis of brain tumours is heavily dependent on the manual review by radiologists or specialist doctors and is time-consuming. So to pace up the

[1]https://www.narayanahealth.org/diseases/brain-tumour

manual diagnosis process there are several Computer Aided Diagnosis(CAD) systems. The existing CAD systems used for the diagnosis of brain tumours have drawbacks such as less accuracy, maintaining large-scale data, high computational complexity and cost, high inference time and so on(Adarsh et al.; 2023). Thus, making CAD systems less reliable. Therefore, there is a need for a system that can accurately classify brain tumours with limited computation.

The aim of the research is to investigate to what extent a weight pruning technique reduces the loss function of a Convolution Neural Network(CNN) and accurately classify brain tumours with limited computation. To address the research question, the following specific sets of research objectives were derived.

1. Investigate the state of the art broadly around brain tumour classification.

2. Design a lightweight deep learning framework for brain tumour classification.

3. Implement a lightweight deep learning framework for brain tumour classification.

4. Evaluate a lightweight deep learning framework for brain tumour classification on the basis of accuracy, sensitivity, specificity and loss function.

The major contribution of this research is a lightweight deep learning framework that combines a machine learning classification model and weight pruning optimisation technique to classify brain tumours with limited computation. The proposed framework is validated using the BTMRI dataset which consists of 7023 brain MRIs, distinct into 4 classes of brain tumours namely: glioma, meningioma, pituitary and no tumour. The outcomes of the proposed framework are compared with the outcomes of state-of-the-art CNN for brain tumour classification.

The strategy of optimising CNN using the magnitude-based weight pruning technique will reduce the redundant weights of the network by setting them to zero. Thus, reducing the computational complexity. The reduced computational complexity will reduce the loss function of the network and will therefore improve the accuracy of the network. The strategy represents an efficient and reliable framework to aid patients in getting an early view of their tumour type.

This research discusses the deep learning models for brain tumour classification, pruning and optimization techniques in Section 2 related work. The research methodology is discussed in Section 3. Section 4 discusses the design specifications of the lightweight deep learning framework. The implementation of this research is discussed in Section 5. Section 6 presents the evaluation results and discusses research findings. Section 7 concludes the research and discusses future work.

2 Related Work

To accomplish the research objective, several existing related works around the research domain and deep learning techniques were critically reviewed and assessed. Reviewing existing related works helped in understanding key concepts, identifying methodological approaches for research, feasible frameworks for the implementation of research and validating the research findings by contrasting them with similar work done in the domain. The key findings after reviewing existing related works are presented in the current section of the research report. Section 2.1 summarises challenges in brain tumour classification, reviews different deep learning techniques implemented for brain tumour classification and discusses the further scope of improvement in existing works. Techniques for optimization of CNN are discussed in the 2.2

2.1 Deep Learning Techniques for Brain Tumour Classification

Adarsh et al. (2023) reviewed several machine learning approaches such as Support Vector Machine(SVM), Random Forest(RF), Extreme Learning Machine(ELM), Decision Tree(DT), Capsule network, etc. and deep learning approaches such as CNN, UNet, VGG16, Fuzzy learning and so on for brain tumour segmentation and classification for their survey. The survey discussed the pros and cons of each approach for brain tumour segmentation and classification. Most of the reviewed approaches for the survey used the 'Brain Tumor Image Segmentation Benchmark (BraTS)' dataset. The survey put forth that Magnetic Resonance Imaging(MRI) images are one of the pivot techniques in the diagnosis of a brain tumour. The survey highlighted that the irregular shape of the tumour and its distinction from its background is a major challenge in the segmentation and classification of brain tumours. The survey also stated that the accuracy, precision, specificity, sensitivity, dice coefficient and F1 Score are key metrics to validate the framework's robustness for brain tumour segmentation and classification. Collectively, the survey presented a key challenge, approaches that can be implemented for brain tumour segmentation and classification, and evaluation metrics that can be used to validate the framework.

A survey by Gottipati and Thumbur (2023) reviewed traditional strategies and effective deep-learning techniques for brain tumour segmentation and classification. The survey discussed the importance and effectiveness of the MRI imaging method in the field of automated medical diagnosis of disease. Brain tumours can be segmented and classified using - Manual Methods, Semi-Automatic Methods and Deep Learning Methods. The survey mentioned that manual methods for segmentation and classification by experts are highly vulnerable to errors and are time-consuming. Whereas the outcomes from semi-automatic methods for segmentation and classification of brain tumours vary at each instance. Because software segmenting and classifying brain tumours in semi-automatic methods require human inputs for computation. These inputs may vary from person to person and with time. Thus, semi-automatic methods are less reliable. The survey discussed several fully automated supervised and unsupervised deep learning techniques for brain tumour segmentation and classification. The survey spotlighted that feature selection and generating a probabilistic map from features is a crucial task in the segmentation and classification of brain tumours. Additionally, the survey highlighted that CNN has the in-built potential to effectively recognise the key insights or features from the image dataset, making it one of the popular architectures for image processing and classification.

A detailed assessment of CNN techniques for brain tumour classification by Xie et al. (2022) is a state-of-the-art survey of CNN-based deep learning techniques for brain tumour classification. The survey reviewed 83 research of CNN techniques for brain tumour classification in detail from the year 2015 to 2022. To briefly summarize the key findings from the survey are - it discussed and highlighted the importance of a number of target classes in classification problems, the influence of a training dataset on results, the significance of data pre-processing and augmentation for classification, and the impact of CNN architecture on brain tumour classification. The survey pointed out that transfer learning with fine-tuning on pre-trained CNNs performs effectively with limited data or when data training is expensive. Additionally, the survey highlighted the scope of improvement of CNN architecture in terms of the volume of training data required, long training time, high hardware requirements and black box nature of CNN making architecture less explainable and trustworthy. Overall, the survey provided the perks and cons of CNN and factors to be considered while implementing CNN architecture.

Özkaraca et al. (2023) implemented the dense CNN architecture for brain tumour classification. Researchers discussed and pointed out that the traditional approaches for image classification require abundant pre-processing of data and pre-trained models may fail to achieve significant results in the healthcare sector. Researchers studied the three different models namely - simple CNN, VGG16 and ResNet; identified deficiencies in each of them and eliminated their deficiencies by proposing a dense CNN architecture. The BTMRI dataset was used to build and validate dense CNN architecture. In order to validate the robustness of dense CNN architecture, K-fold validation was applied. The research evaluation showed significant performance of dense CNN architecture for brain tumour classification with an accuracy of 95-97% after applying K-fold validation. However, the long processing time was the drawback of dense CNN architecture. The scope of improvement in the dense CNN architecture for brain tumour classification drives this research motivation and objective to implement optimised CNN for brain tumour classification. The dataset used to build a lightweight deep learning framework is taken from the research of dense CNN architecture for brain tumour classification. As a part of a current research experiment, it was tried to implement dense CNN architecture for brain tumour classification and the plan was to further optimise it. However, the experiment was not able to achieve significant results as mentioned in the report on dense CNN architecture for brain tumour classification. Because the report on dense CNN architecture for brain tumour classification lacks clarity about the pre-processing of MRIs, activation function used in the CNN architecture and CNN layers configuration in detail. Also, the processing time taken by dense CNN architecture was not stated in the report.

2.2 Pruning and Optimization Techniques for Deep Learning Network

A survey by Kulkarni et al. (2022) summarized different approaches of pruning for optimizing deep learning networks. The techniques discussed in the survey can be overviewed in the Figure 1. The objective of optimization techniques is to reduce the computational cost of the network, decrease the model size, speed up the model and make the model power efficient. There are four approaches for optimizing deep learning networks namely - Parameter Search, Parameter Decomposition, Parameter Quantization and Parameter Removal. In the parameter search approach, a student model with lesser parameters is generated from the parent model. The parameter search approach requires extensive fine-tuning and is time-consuming. In the parameter decomposition approach, the weight matrix connecting two hidden layers is decomposed into smaller matrices. In the parameter quantization approach, memory sizes of parameters are reduced by reducing the precision of parameters. Thus, the network will consume less memory. However, there is a high chance of loss of information while using the parameter quantization approach. In the parameter removal approach, the redundant parameters of the network are dropped. The parameter removal approach has further methods such as Node Pruning, Weight

Pruning, Channel Pruning, Connection Pruning, Layer Pruning and Filter Pruning depending on the component of the network that is removed. Overall the survey provided an overview of different approaches that can be applied for the optimization of deep learning networks.

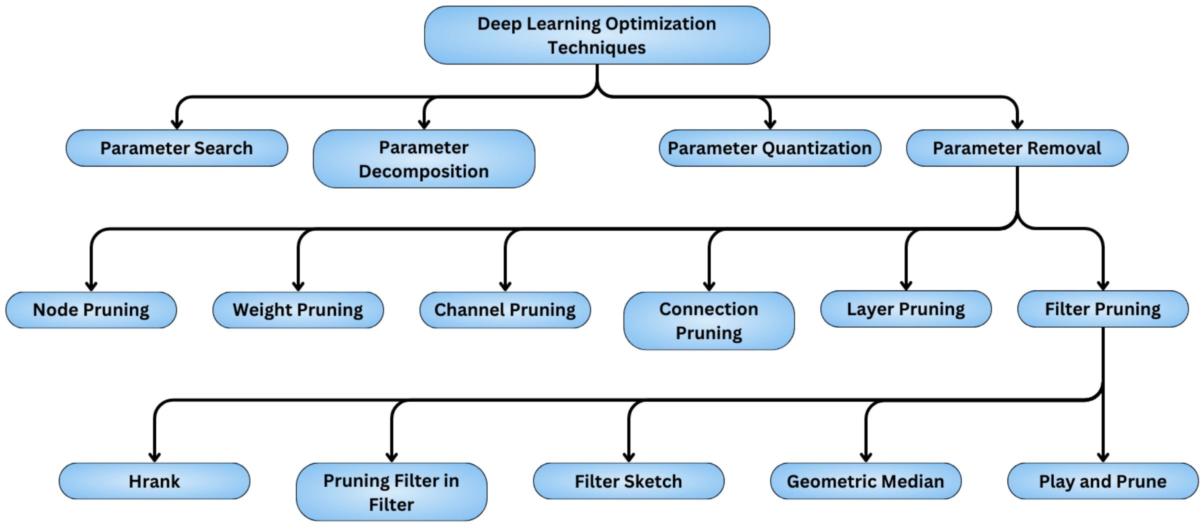


Figure 1: Deep Learning Optimization Techniques

Wang et al. (2022) implemented efficient CNN architecture for the classification of garbage waste. The research highlighted that the lightweight CNN model has less computational complexity and can be easily deployed on small devices or embedded systems to address the image classification problem across various domains. Initially, the research implemented the weighted CNN for garbage waste classification. Then the weighted CNN model was compressed by using the weight pruning technique to generate a lightweight CNN for garbage waste classification. Later, the lightweight CNN model was optimised to improve its performance. After evaluating the lightweight CNN model for garbage waste classification it was noticed that the accuracy of the lightweight CNN model for garbage waste classification was better than the state-of-the-art AlexNet Model for garbage waste classification. However, there was a slight reduction in the accuracy of the lightweight CNN model compared to the accuracy of weighted CNN for garbage waste classification. The research of lightweight CNN for the classification of garbage waste provided a direction for optimizing the CNN using the weight pruning technique and provided a guideline for an approach to be taken for optimizing CNN.

3 Methodology

For ensuring the validity, reliability and credibility of the research finding the research methodology is a crucial part of the research process. The research methodology has four steps tailoring Knowledge Discovery in Databases(KDD) as outlined in the Figure 2.

The first step is data collection. An open-source BTMRI dataset is accessible on Kaggle was used to analyse and evaluate the lightweight deep learning framework. The BTMRI dataset is a blend of three datasets namely- figshare, SARTAJ and Br35H datasets(Nickparvar; 2021). The BTMRI dataset has 7023 MRIs representing 4 distinct classes

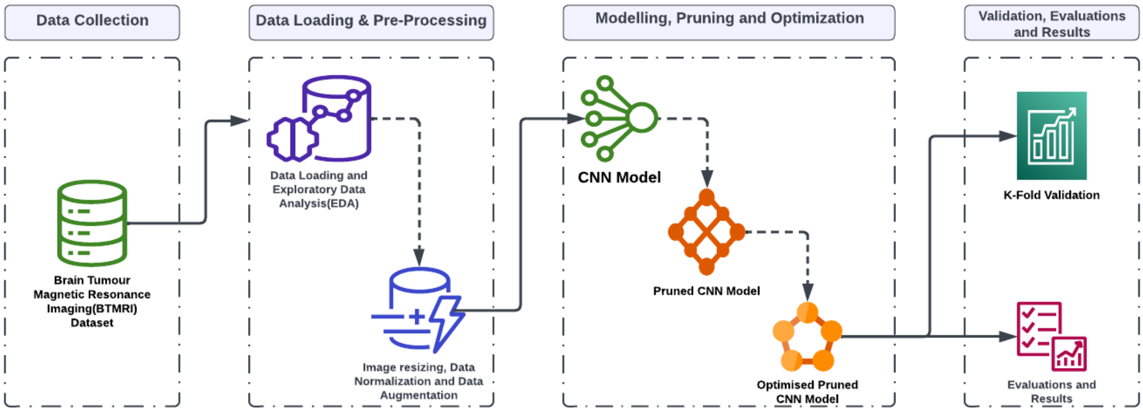


Figure 2: Research Methodology

of brain tumours namely – glioma, meningioma, pituitary and no tumour. Further, the count of each distinct class is - 1621 MRIs of glioma, 1645 MRIs of meningioma, 1757 MRIs of pituitary and 2000 MRIs of no tumour. The MRIs of each distinct class of the BTMRI dataset can be glimpsed in Figure 3.

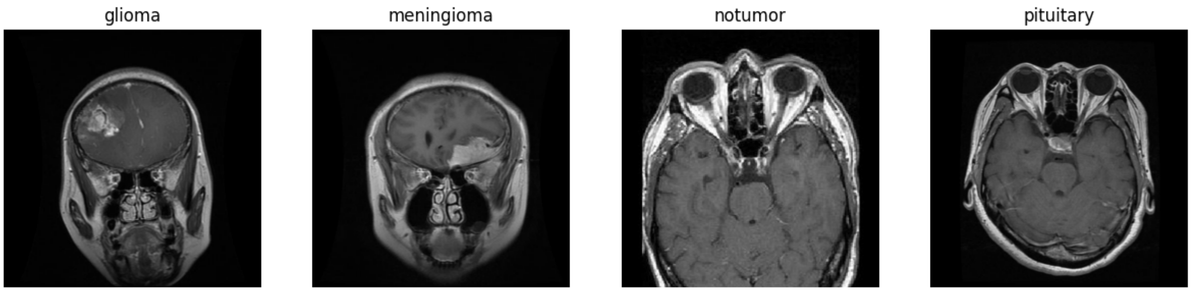
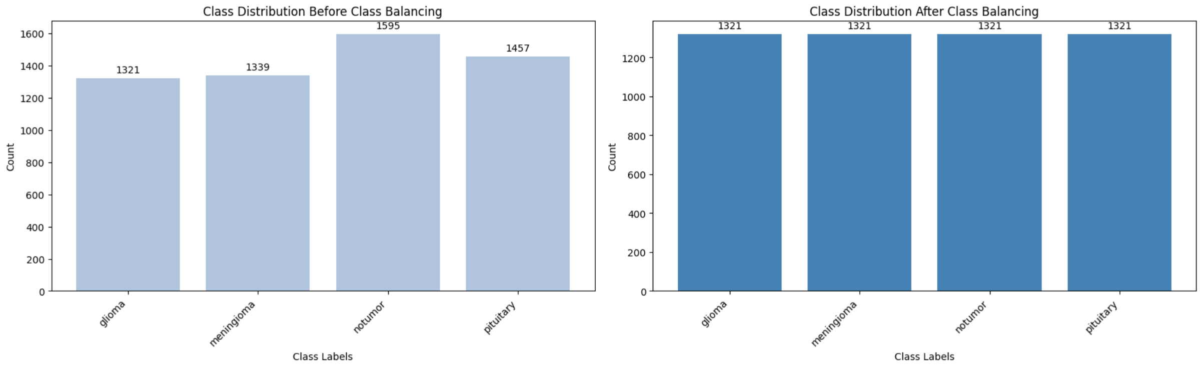


Figure 3: Sample MRIs of each class in BTMRI dataset

The second step, data loading and pre-processing involves loading the data from the directories, exploring the data in different classes, handling class imbalance and data augmentation. After loading the data from the directories, exploratory data analysis(EDA) was performed. In EDA it was identified that there was a class imbalance in the training dataset. As mentioned by Longadge and Dongre (2013), handling a class imbalance is of utmost importance in medical applications for the accurate classification of disease and having a bias-free framework. In the training dataset, there are ample numbers of MRIs and the difference in the number of MRIs in the major and minor classes was small. So undersampling technique was applied. The undersampling technique fastens up the model training process and improves performance(Van Hulse et al.; 2009). The class distribution before class balancing can be observed in Figure 4a and the class distribution after class balancing can be observed in Figure 4b. Also, the MRIs in the BTMRI dataset are of varying sizes. So the MRIs were pre-processed to have constant image size. The survey by Shorten and Khoshgoftaar (2019) concluded the usefulness of the data augmentation technique for enhancing the datasets and building robust models. So the few MRIs were randomly horizontally and vertically flipped to enhance data quality through the data augmentation technique. Also, the MRIs were converted into PyTorch tensor format for ease of processing by the framework.



(a) Class Distribution Before Class Balancing (b) Class Distribution After Class Balancing

Figure 4: Data Balancing

The third step is modelling, pruning and optimisation. It is a crucial step for validating the research objective. The architecture of CNN for brain tumour classification by Soewu et al. (2022) was considered as a base CNN model for this research. Then the base CNN model was pruned using a weight pruning technique. Later, the weight-pruned model was optimised to improve the model performance.

The fourth step, Validation, Evaluations and Results involves evaluating the performance of the lightweight deep learning framework using accuracy, sensitivity, specificity and loss function. Additionally, to further verify the robustness of the framework K-fold cross-validation technique is applied.

4 Design Specification

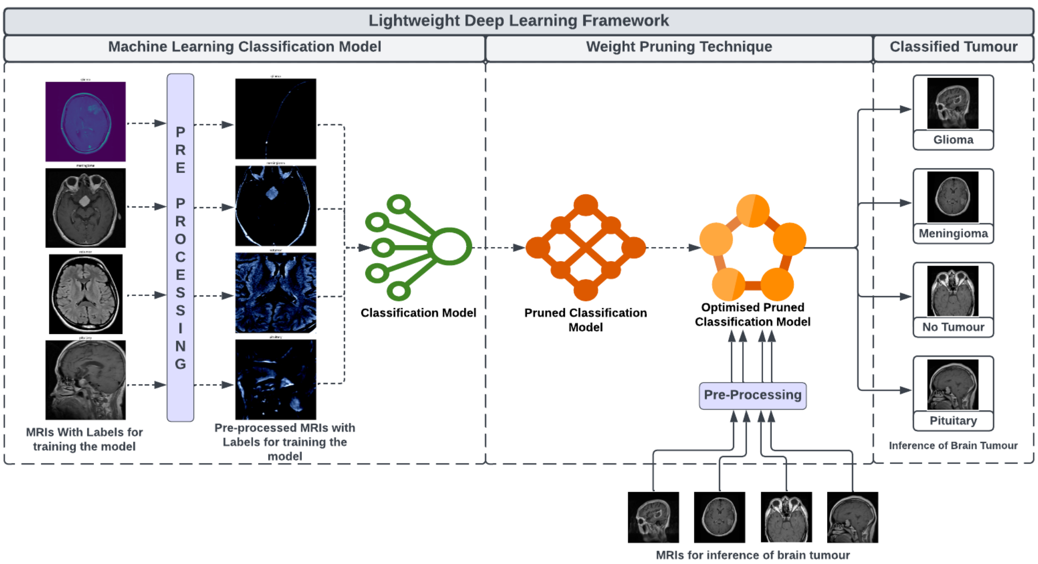


Figure 5: Lightweight Deep Learning Framework Architecture

The lightweight deep learning framework architecture combines a machine learning classification model and weight pruning techniques as shown in the Figure 5. The components of the machine learning classification model include the pre-processing unit and machine learning classification model as discussed in Section 4.1. The weight pruning technique is discussed in Section 4.2

4.1 Machine Learning Classification Model

The machine learning classification model is fed in with the raw MRIs with labels which are used to train the model for brain tumour classification. The raw MRIs with labels are processed and transformed to improve their quality and further classification. Post pre-processing, the processed MRIs with labels are fed into the classification model. CNN is the machine classification model utilised in the framework. CNN is trained on the pre-processed MRIs with labels to classify the brain tumour.

4.2 Weight Pruning Technique

Post fabrication of the machine learning classification model the redundant weights in the network are pruned using the magnitude-based weight pruning technique. The magnitude-based weight pruning technique sets the redundant weight of the network to zero. Thus reducing the computation complexity and improving the model performance. The pruned machine learning classification model is fine-tuned to identify the optimal pruning ratio. Post identifying the optimised ratio, the optimised pruned classification model is generated for the classification of brain tumours. The optimised pruned classification model is fed with MRIs for inference of the type of brain tumour in it.

5 Implementation

The lightweight deep learning framework was implemented by using PyTorch. The web-based interactive computing platform - Jupyter Notebook, was used for the execution of the research framework. Python(Version: 3.10.7) programming language was used to develop the research framework. As stated by Stančin and Jović (2019), Python is relatively easy to use and provides a free and open-source large number of robust libraries for each aspect of data science and machine learning. The details of implementation are audited in the following subsections.

5.1 Environmental Setup

The Jupyter Notebook(Version: 6.4.12) was set up in the system with the following specifications.

- **Operating System:** Windows 10 Home Single Language(Version: 22H2).
- **Processor:** Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1.80 GHz.
- **Storage:** 2TB.
- **RAM:** 8GB(Extendable 20.4GB Virtual Memory).
- **Graphical Processing Unit:** NVIDIA GeForce MX150.

The MRIs dataset was imported from the kaggle(Nickparvar; 2021) and stored in the system. Additionally, libraries and packages for supporting the execution of the research framework were installed and imported into the Jupyter Notebook. The Figure 6 represents a few packages and libraries used in the execution of the research framework. PyTorch's core libraries, such as 'torch' and 'torchvision', provide support for deep learning operations, core functionalities and various utilities for loading images, image transformation, creating datasets, and so on, which were used for implementing the research framework. The 'OS' module is used to interact with the operating system and directory operations for fetching and loading data. The 'NumPy' library is utilised for array operations. To keep the count of elements in the list 'counter' from the 'collection' class was imported. The 'matplotlib' library is used for various graphical visualizations of research and 'seaborn' for various statistical visualization of data. The 'scikit-learn' library provides support for various operations such as calculating framework accuracy, creating confusion matrix, optimization tasks, k-fold validation and generating classification reports. For reproducibility of research experiments and results the seed of the various operations involving random number generators is set to constant. The path to the directories containing MRIs dataset was set in the execution environment.

Figure 6: Incorporated Libraries and Packages for Research Implementation

5.2 Data Processing

To enhance the data quality and improve the robustness of the model the data is preprocessed. The BTMRI dataset is subdivided into training and testing sets by default. The training data is resized using RandomResizedCrop() class to improve generalization whereas testing data is resized using Resize() class. Also, the training data was randomly horizontally and vertically flipped using RandomHorizontalFlip() and RandomVerticalFlip() classes. Additionally, the images from the training and testing set were converted to PyTorch tensor for further processing with tensor using ToTensor() class and normalise using Normalize() class. The training and testing set had sub-directories of target classes in it. So ImageFolder() class was used to load the data from the organised directory structure. Further, the DataLoader() class was used to load the data into batches for training and testing the framework.

5.3 Modelling, Pruning and Optimization

After processing of data, the data was fed into the classification model. CNN model defined using the PyTorch framework was a classification model. The convolutional layers were defined using the function Conv2d() from the 'torch.nn' library. For downsampling the spatial dimensions of feature maps, MaxPool2d() from the 'torch.nn' library. For the classification of brain tumours based on the extracted features, fully connected layers were defined using the Linear() function from the 'torch.nn' library. ReLU was used as an activation function for all three convolutional layers and for the first fully connected layer. The cross-entropy loss is the loss function used in this research and the optimizer that is applied to minimize the loss function is the Adam optimizer. The layout of the CNN model is tabulated in Table 1. Then the model was trained on the training set.

Layer (type)	**Output Shape**	**Param #**
Conv2d-1	[-1, 16, 256, 256]	448
MaxPool2d-2	[-1, 16, 128, 128]	0
Conv2d-3	[-1, 32, 128, 128]	4,640
MaxPool2d-4	[-1, 32, 64, 64]	0
Conv2d-5	[-1, 16, 64, 64]	4,624
MaxPool2d-6	[-1, 16, 32, 32]	0
Linear-7	[-1, 256]	4,194,560
Linear-8	[-1, 4]	1,028

Table 1: Layout of the CNN Model

Post-training the CNN model the redundant weights in the CNN model were pruned using a magnitude-based pruning technique to reduce computational complexity and improve model performance. The replica instance of the CNN model is generated along with its trained parameters. The pruning function was defined to prune the model. The pruning function was fed with the replica instance of the CNN model and the pruning ratio as the input. The pruning function identified all the weight parameters in the replica instance of the CNN model and calculated the threshold value using the pruning ratio. Then the pruning mask of each weight parameter was calculated based on whether its absolute value is more than that of the threshold or not. Lastly, the weights of the replica instance of the CNN model are pruned by element-wise multiplication of weights and its calculated pruning mask. The pruning function returns the pruned model as the outcome of the processing. Later, the pruned model was trained on the training set.

Once the pruned model was trained then the hyperparameter optimization was performed through Bayesian optimization. The 'skopt' library was used to perform hyperparameter optimization. The objective function was defined that accepted the list of parameters as input. From the list of parameters, the pruning ratio was extracted, pruned model with extracted pruning ratio was generated, trained and evaluated. The search space was defined with the help of the 'Real' class in 'skopt'. A search space calculated a pruning ratio with the help of a log uniform function with a low value of 0.01 and a high value of 0.5. The process of calculating the prune ratio using search space and evaluating its performance using objective function was iterated to 10 calls. The gp_minimize() function from 'skopt' was used to perform Bayesian optimization. The optimal pruning ratio was obtained from the result of the gp_minimize() function and the optimized pruned model was generated with the help of the optimal pruning ratio. Then the optimized pruned model was trained and evaluated.

6 Evaluation

The aim of the research is to investigate to what extent a weight pruning technique reduces the loss function of a Convolution Neural Network(CNN) and accurately classify brain tumours in a timely fashion. To validate the research finding a series of experiments were conducted. The outcomes of each experiment in the research are evaluated on the basis of the accuracy of the model, sensitivity and specificity of each class calculated based on the confusion matrix and loss function of the network. The performance of the model in each experiment was evaluated using a testing dataset. The testing dataset has 300 MRIs of glioma tumours, 306 MRIs of meningioma tumours, 300 MRIs of pituitary tumours and 405 MRIs with no tumour. The outcome and discussion of each experiment are as follows.

6.1 Experiment 1: CNN Model for Brain Tumour Classification

The aim of the first experiment was to implement a CNN model for brain tumour classification. The architecture proposed by Soewu et al. (2022) was referred for implementation of the CNN model. The CNN model of the first experiment was built using the PyTorch framework and the BTMRI dataset whereas the CNN architecture by Soewu et al. (2022) was implemented using Keras deep learning library of TensorFlow framework and Br35H[2] dataset. The accuracy of a CNN model implemented in the first experiment for brain tumour classification was 82.60%. The confusion matrix of the CNN model implemented in the first experiment for brain tumour classification is depicted in Figure 7. Table 2 summarises the results of the first experiment in terms of the sensitivity and specificity of each class. Figure 8 depicts the plot of the loss function of the CNN model implemented in the first experiment. It was observed that the value of the loss function for the CNN model implemented in the first experiment begins with 1.1 and over the number of epochs it decreases. After 25 epochs the loss of the model was approximately 0.4.

Table 2: Sensitivity and Specificity of Each Class for CNN Model

Class	Sensitivity	Specificity
Glioma	77.00%	96.93%
Meningioma	85.29%	84.08%
No Tumor	94.07%	96.91%
Pituitary	70.00%	99.11%

The CNN model implemented in the first experiment was the base CNN model for the further experiments that were performed as a part of this research.

[2] https://www.kaggle.com/datasets/ahmedhamada0/brain-tumor-detection

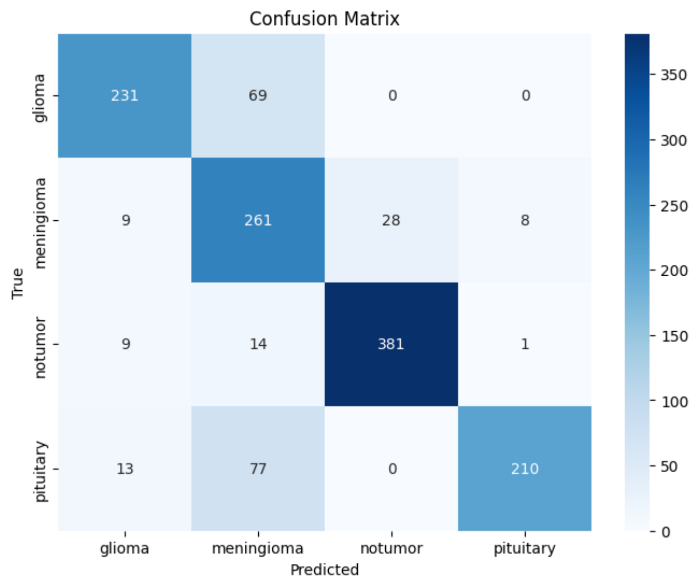


Figure 7: Confusion Matrix of a CNN Model

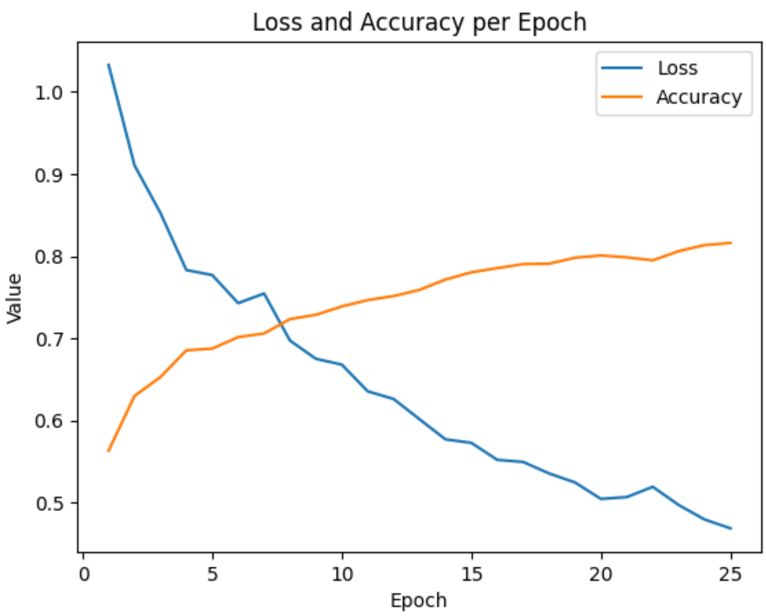


Figure 8: Loss and Accuracy per Epochs of a CNN Model

6.2 Experiment 2: Pruned CNN Model for Brain Tumour Classification

The aim of the second experiment was to prune the base CNN model using a magnitude-based weight pruning technique. As discussed and concluded by Wang et al. (2022), if redundant weights are cautiously identified then the model accuracy will not be impacted. In the medical domain, the model accuracy must not be compromised. So weight pruning technique was elected. The pruning ratio for this experiment was set to 0.25. Post-pruning, training and evaluating a pruned model, the pruned model demonstrated an accuracy of 86.34%. Thus, there was an increase in accuracy by roughly around 4% after pruning the CNN model. The confusion matrix of the pruned CNN model implemented in the second experiment for brain tumour classification is depicted in Figure 9. By comparing the confusion matrix in Figure 7 and Figure 9, it can be noticed that model prediction improved after pruning the model. The summary of accuracy, sensitivity and specificity of each class for the pruned CNN model is tabulated in Table 3. The plot of the loss of the network over each epoch can be observed in the Figure 10. It was spotted that the value of the loss function for the pruned model was initially around 0.46 which gradually decreased to around 0.39 after the completion of 25 epochs.

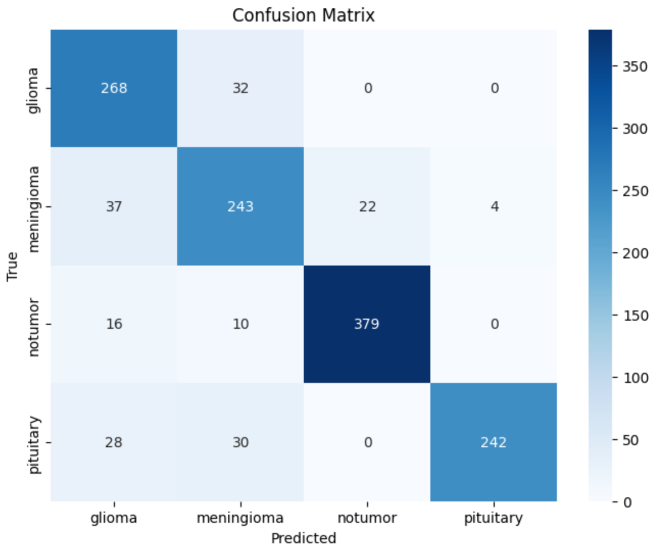


Figure 9: Confusion Matrix of a Pruned CNN Model

Table 3: Sensitivity and Specificity of Each Class for Pruned CNN Model

Class	Sensitivity	Specificity
Glioma	89.33%	91.99%
Meningioma	79.41%	92.84%
No Tumor	93.58%	97.57%
Pituitary	80.67%	99.60%

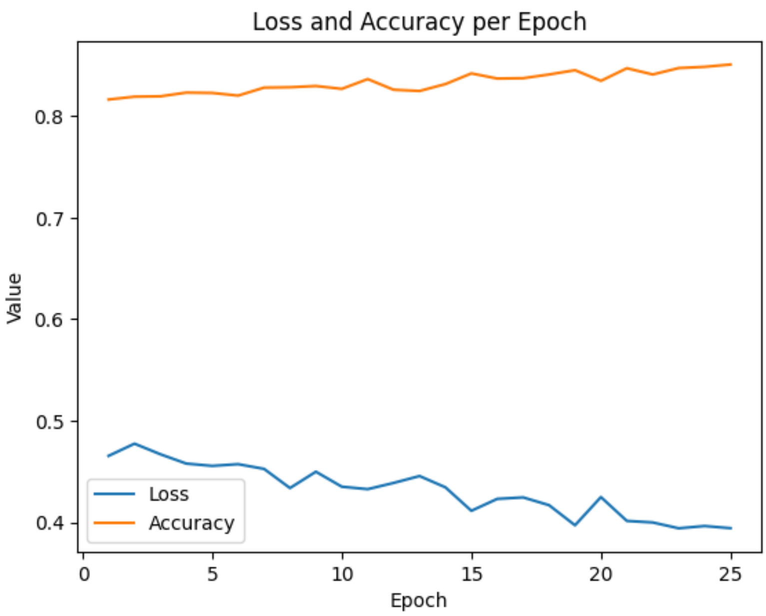


Figure 10: Loss and Accuracy per Epochs of a Pruned CNN Model

6.3 Experiment 3: Optimizing Pruned CNN Model for Brain Tumour Classification

The aim of the third experiment was to prune the base CNN model with an optimal pruning ratio. In the second experiment, the pruning ratio was randomly decided. But in the third ratio optimal pruning ratio was identified using the Bayesian optimization technique. After identifying the optimal pruning ratio using the Bayesian optimization technique, the base CNN model was pruned with the optimal pruning ratio to generate the optimal pruned CNN model. The pruning technique was the same, as executed in the second experiment. The accuracy of the optimal pruned CNN model was 87.26% which was slightly better than the pruned model implemented in experiment 2. The confusion matrix of the optimally pruned CNN model is shown in Figure 11. The optimal pruned CNN model showed better prediction results. The performance report in terms of sensitivity and specificity of each class for optimal pruned CNN model is recorded in Table 4. Additionally, the loss of the network in each epoch can be seen in the Figure 12. It was observed that the loss of the optimal pruned model drastically decreases over a few initial epochs. In the first epoch of training the optimal pruned CNN model, the loss was around 0.65 and after the completion of 25 epochs, the loss of the model fell to 0.39.

Table 4: Sensitivity and Specificity of Each Class for an Optimised Pruned CNN Model

Class	Sensitivity	Specificity
Glioma	84.33%	95.94%
Meningioma	82.03.41%	92.64%
No Tumor	93.83%	96.69%
Pituitary	86.67%	97.82%

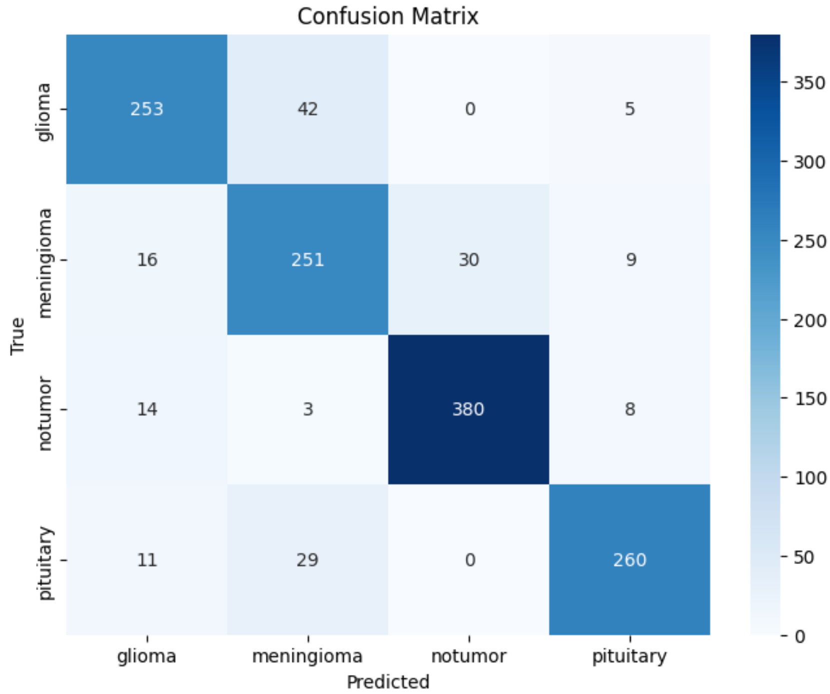


Figure 11: Confusion Matrix of an Optimised Pruned CNN Model

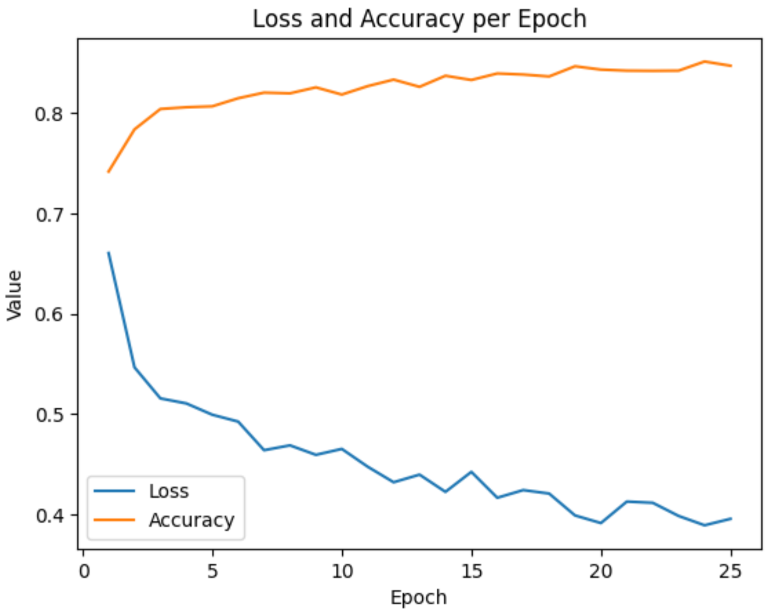


Figure 12: Loss and Accuracy per Epochs of an Optimised Pruned CNN Model

6.4 Experiment 4: K-Fold Cross-Validation

The aim of the fourth experiment was to validate the robustness of the optimal pruned CNN model using the K-fold cross-validation technique. The number of folds for k-fold validation was set to 5. After running the optimal pruned CNN model through k-fold cross-validation, the mean accuracy of the model was 86.46%. Thus showcasing the robustness of the optimised pruned model

6.5 Discussion

The objective of the research was to investigate the effectiveness of the lightweight deep learning framework for the classification of brain tumours in comparison with the existing state-of-the-art CNN model. The state-of-the-art CNN model with which the research lightweight deep learning framework is compared is 'Convolutional Neural Networks for MRI-Based Brain Tumor Classification' by Soewu et al. (2022). The state-of-the-art CNN model was implemented for the binary classification of brain tumours on the Br35H dataset. However, state-of-the-art showed an accuracy of 82.60% on the BTMRI dataset for multi-class brain tumour classification problems. Highlighting the robustness of state-of-the-art by showcasing decent results on vivid datasets. From the Figure 13, it can be observed that the accuracy of the lightweight deep learning framework proposed and implemented in the research is 87.26%. Thus, showcasing better results than the state-of-the-art CNN model on the BTMRI dataset.

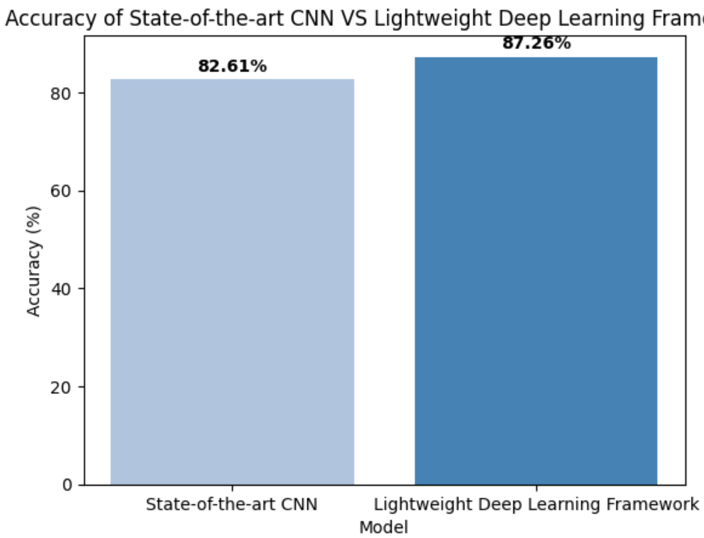


Figure 13: Accuracy of State-of-the-art CNN model and Lightweight Deep Learning Framework

The visual of keen interest is a plot of loss per epoch between a state-of-the-art CNN model and a lightweight deep learning framework for brain tumour classification. It can be observed from the plot in Figure 14 that the loss of a lightweight deep learning framework is comparatively lower than the loss of the state-of-the-art CNN. Thus, the results address the research objective that weight pruning techniques effectively reduce the loss function by approximately 0.5 and improve the accuracy of the framework.

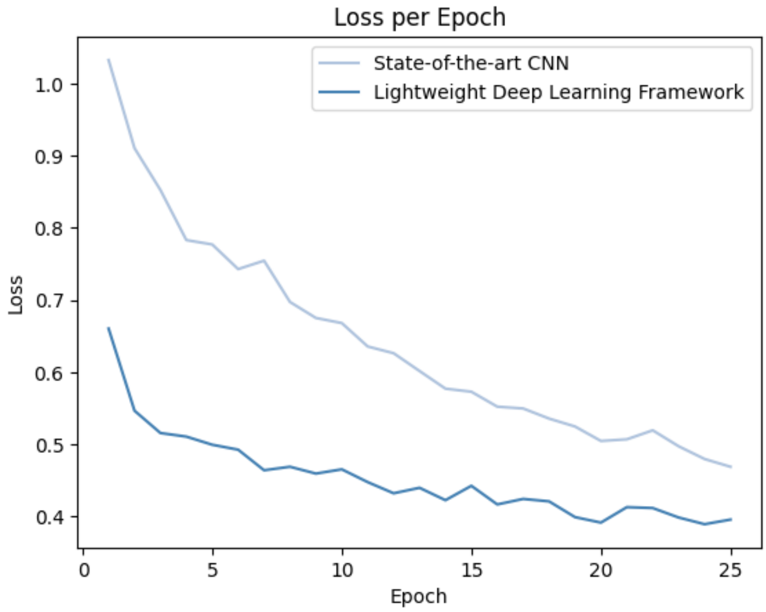


Figure 14: Loss per Epoch of state-of-the-art CNN model and Lightweight Deep Learning Framework

7 Conclusion and Future Work

The aim of this research was to present an efficient and reliable framework to aid patients in getting an early view of their brain tumour type. To fulfil the research objective, a lightweight deep learning framework that combines a machine learning classification model and weight pruning optimization technique was implemented in the research. The BT-MRI dataset was used to train and validate the implemented research framework. The quality and the diversity of data were enhanced through the data augmentation technique while pre-processing the data. The CNN model was built and trained to classify brain tumours. A weight pruning technique was used to optimise the CNN model and the optimal pruning ratio was calculated using Bayesian optimization technique. Then, the CNN model was pruned with the optimal pruning ratio. In this way, a lightweight deep learning framework was fabricated for brain tumour classification which can be used by patients to get an early view of their tumour type. The lightweight deep learning framework shows better results in terms of accuracy than state-of-the-art CNN for the BTMRI dataset. Also, the loss of a lightweight deep learning framework was low compared to the loss of state-of-the-art CNN. The framework is able to infer brain tumours in nearly about 7 seconds. However, it was observed that the time taken for calculating the optimal pruning ratio was high. Nearly above a couple of hours. Although the sparsity of the framework was high. It was observed that there was no significant reduction in the model size.

Further, there is a scope to fine-tune the model to improve its accuracy. Additionally, the framework can be enhanced for detecting the severity of the brain tumour and possibly attempted for diagnosis of other medical abnormalities such as lung cancer, breast cancer, PCOS and so on. As their diagnosis is done through MRIs.

Acknowledgement

I would like to express my heartfelt gratitude to my supervisors and mentors, Prof. Paul Stynes, Prof. William Clifford and Prof. Eugene McLaughlin for their consistent guidance, support and encouragement throughout my research project. I am grateful for the support provided by NCI faculty and staff members throughout my journey of pursuing Master's in Data Analytics. Last but not least, I would like to take this opportunity to thank my parents, family, especially my brother and friends who always believed in me and motivated me. Their constant support has been a driving force behind my consistent efforts.

References

Adarsh, K., Manohari, R. et al. (2023). A short literature review of brain tumour segmentation and classification framework: Challenges issues and future scope, *2023 7th International Conference on Trends in Electronics and Informatics (ICOEI)*, IEEE, pp. 1354–1360.

Bayram, H. Y., Bingol, H. and Alatas, B. (2023). Automated classification of brain tumor disease with a novel cnn relief and svm-based deep hybrid model., *Traitement du Signal* **40**(2).

Ganapathy, K. (2022). Neurosurgery in india: an overview, *World Neurosurgery* **79**(5-6): 621–628.

Gottipati, S. B. and Thumbur, G. (2023). A review on segmenting and classification of brain tumor using deep learning methods, *2023 8th International Conference on Communication and Electronics Systems (ICCES)*, IEEE, pp. 1115–1121.

Kulkarni, U., Hallad, S. S., Patil, A., Bhujannavar, T., Kulkarni, S. and Meena, S. (2022). A survey on filter pruning techniques for optimization of deep neural networks, *2022 Sixth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, IEEE, pp. 610–617.

Longadge, R. and Dongre, S. (2013). Class imbalance problem in data mining review, *arXiv preprint arXiv:1305.1707* .

Nickparvar, M. (2021). Brain tumor mri dataset.
URL: *https://www.kaggle.com/dsv/2645886*

Özkaraca, O., Bağrıaçık, O. İ., Gürüler, H., Khan, F., Hussain, J., Khan, J. and Laila, U. e. (2023). Multiple brain tumor classification with dense cnn architecture using brain mri images, *Life* **13**(2): 349.

Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning, *Journal of big data* **6**(1): 1–48.

Soewu, T., Singh, D., Rakhra, M., Chakraborty, G. S. and Singh, A. (2022). Convolutional neural networks for mri-based brain tumor classification, *2022 3rd International Conference on Computation, Automation and Knowledge Management (ICCAKM)*, IEEE, pp. 1–7.

Stančin, I. and Jović, A. (2019). An overview and comparison of free python libraries for data mining and big data analysis, *2019 42nd International convention on information and communication technology, electronics and microelectronics (MIPRO)*, IEEE, pp. 977–982.

Van Hulse, J., Khoshgoftaar, T. M. and Napolitano, A. (2009). An empirical comparison of repetitive undersampling techniques, *2009 IEEE international conference on information reuse & integration*, IEEE, pp. 29–34.

Veeranki, P. S. L., Banavath, G. L. and Devi, P. R. (2023). Detection and classification of brain tumors using convolutional neural network, *2023 7th International Conference on Trends in Electronics and Informatics (ICOEI)*, IEEE, pp. 780–786.

Wang, Y., Zhao, G. and Li, X. (2022). A weight pruned convolutional neural network for garbage image classification, *2022 15th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, IEEE, pp. 1–6.

Xie, Y., Zaccagna, F., Rundo, L., Testa, C., Agati, R., Lodi, R., Manners, D. N. and Tonon, C. (2022). Convolutional neural network techniques for brain tumor classification (from 2015 to 2022): Review, challenges, and future perspectives, *Diagnostics* **12**(8): 1850.