IT1244 Report

Justin Tan Min Shi , Han Yong Yi Jace , Sng Zhi Wen Collin , Ryan Justyn

Introduction

Cancer in the central nervous system (CNS) is ranked the tenth leading cause of global mortality, of which brain tumours account for up to 90% of the cases[1]. A brain tumour is characterised as an abnormal mass of cells in the brain or spinal cord that grow uncontrollably and can be classified as benign or malignant. While benign tumours often do not recur with surgical resection, malignant tumours possess the ability to infiltrate adjacent tissues and metastasise. Therefore, accurate tumour classification as benign or malignant is imperative to determine the optimal treatment. Medical imaging such as magnetic resonance imaging (MRI) are common tools for neuro-oncologists to examine and classify brain malignancies. In recent years, many deep learning algorithms have been developed to complement neuro-oncologists' diagnoses. In particular, Convolutional Neural Networks (CNN) have the advantage of automatically extracting and learning spatial hierarchies of features from MRI images. CNNs have achieved state-of-the-art classification performance in medical image analysis, but more explanation is needed to clarify the underlying mechanisms behind how the model learns[2,3]. AlexNet predicted unrecognisable images as certain objects with 99.99% confidence, such as labelling TV static as a motorcycle[4], suggesting that uninterpretable models may exhibit unpredictable behaviours. Hence, we attempted to address their black-box nature using gradient-based visualisation methods, such as Gradient-weighted Class Activation Mapping (GradCAM), to visualise what features (i.e., region of pixels) influenced the model's predictions[5].

Furthermore, a common issue in medical image analysis is limited data. CNNs are prone to overfitting when trained with small datasets. To overcome this, Anaya-Isaza *et al.* utilised image augmentation consisting of geometric flipping image transformation[6]. We expanded upon this by exploring other augmentation techniques, including image rotation, to boost our training data (**Supplementary Table 1**).

Lastly, various works attempted to improve the classification capabilities of CNNs[7]. Arsa *et al*. utilised VGG16-Random Forest Regressor (VGG16-RF) in Batik Classification. However, RFs have limited hyper-tuning capabilities.[8] Given our diverse training data, we chose CNN-Artificial Neural Network (CNN-ANN) and CNN-eXtreme Gradient Boosting (CNN-XGBoost) that offers more fine-tuning flexibility to improve generalisation.

Dataset

Our dataset comprises 231 MRI brain scans with tumours that are labelled benign or malignant. The images are either in grayscale or have RGB colour channels. While there is no optimal size for an image dataset, a typical training set size for image classifications is in the thousands[9]. To resolve this issue, we looked at transfer learning frameworks and data augmentation techniques on the training images. The images in the dataset also have varying resolutions, as shown in **Figure 1**. As most of the data were distributed about 200 pixels, we decided on a fixed resolution of 224 by 224 pixels as the input size.

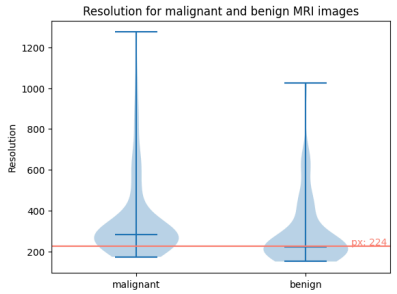

***Figure 1**. Distribution of the approximated resolution of images in the dataset. The red line shows pixel value of 224, which is the standard resolution from the ImageNet dataset.*

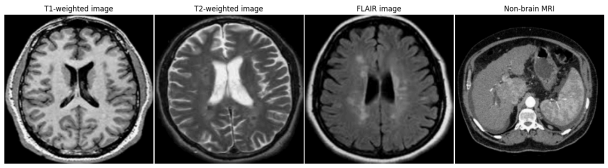

***Figure 2**. Different MRI modalities in the dataset.*

Additionally, the dataset also have MRIs of different modalities: T1-weighted, T2-weighted and Fluid attenuated inversion recovery (FLAIR) (**Figure 2**). These modalities differ primarily in longitudinal and transverse relaxation time during the measurement and different brightness and contrast for tissues and cerebral spinal fluid (CSF)[10]. Preprocessing alone cannot effectively isolate the brain

tissues from the CSF, hence we relied on the model to learn differences between the modalities. We also excluded an outlying non-brain MRI scan from our dataset ("malignant/248.jpg")

We then investigated the presence of duplicates and near-duplicates through the use of perceptual hashes using *phash* from the *imagehash* library. The function hashed the images and returned a 64-bit representation of the images[11]. We calculated pairwise hamming distance of these hashes and identified the maximum threshold, which yielded perceptually similar images across the two datasets. 15 benign and 24 malignant images were identified as duplicates with a pairwise distance threshold of less than 5 and removed from the dataset.

Our dataset also had a mild class imbalance problem (62 benign and 129 malignant), which may result in our classifiers preferring the majority class (malignant)[12]. Thus, we explored minority oversampling methods, namely Synthetic Minority Over-sampling Technique (SMOTE) and image augmentation. However, we did not use SMOTE as it produced synthetic images with features across different modalities, which are not representative of typical MRI scans (**Supplementary Figure 1**). Instead, we utilised *Keras ImageDataGenerator*. **Figure 3** shows examples of augmented images using parameters defined in **Supplementary Table 1**.

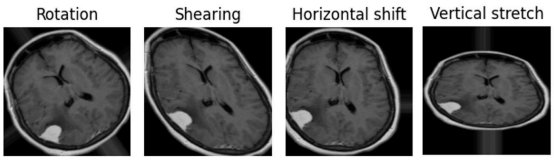

Figure 3. *Examples of added augmented images.*

Our data preprocessing involves separating the brain from the background using a binary threshold mask, morphological operations, cropping the images and augmenting the images to increase model robustness (**Figure 4**).

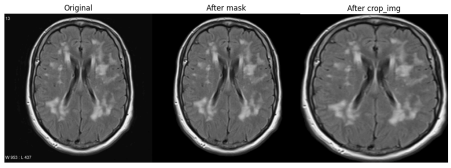

Figure 4. *Preprocessing flow of raw images using the opencv library.*

Methods

Figure 5 describes our schematic. We first preprocessed our data, removed duplicates and outliers, and augmented the training data. Afterwards, we trained the model with Stratified 5-fold cross validation and fine-tuned the hyperparameters. Finally, we analysed the model's predictions with GradCAM and evaluation metrics.

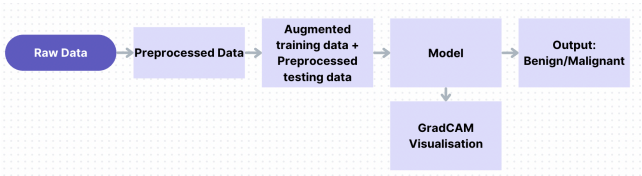

Figure 5. *Schematic of experimental process*

Baseline CNN model

Figure 6 describes the architecture of our baseline model. The single 32-filter convolution layer with 3333 filtering extracted low-level features i.e., edges. Batch normalisation was added to normalise the ReLU function in the first layer to speed up and stabilise the training process. Next, we expanded the convolutional layers with filters up to 128 to capture increasingly complex features. We then flattened the three-dimensional feature map and a dropout layer was used to prevent overfitting before a sigmoid output level to compute the probability that the image belongs to the positive (malignant) class.

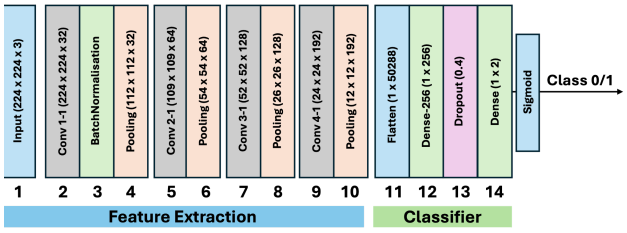

Figure 6. *Baseline CNN architecture.*

Transfer Learning

Most CNN feature extraction models have early layers comprising similar kernels resembling Gabor filters or colour blobs, which extract low-level features from the image, and are relevant for most image classification tasks. Hence, we used the initial layers of the pre-trained model by freezing them so that their weights do not change during training. Afterwards, we added new layers to fine-tune the model for tumour classification. Also, the frozen pre-trained layers act as a regularizer, reducing the model's chance of overfitting for small datasets. We leveraged VGG16's knowledge learned from the ImageNet dataset during pretraining to tackle limited data.. VGG16 is designed with stacked 3 x 3 convolutional filters for a good receptive field with added non-linearity and regularisation.[13]

In our VGG16-ANN model, we retained the pre-trained layers of VGG16 and attached an Artificial Neural Network (ANN) with a flatten layer, 2 densely connected layers and a dropout layer for regularisation (**Figure 7**). We trained ANN for tumour classification and fine-tuned other layers at a lower learning rate.

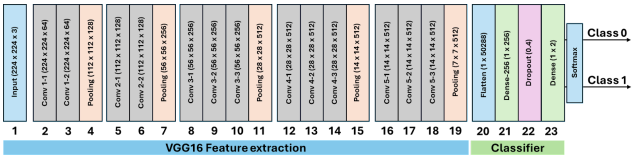

***Figure 7**. VGG16-ANN architecture.*

Meanwhile in our VGG16-XgBoost model, pretrained VGG16 is used as the feature extractor with a flattened output from the last pooling layer. Principal Component Analysis was used to reduce the number of unnecessary features, before XGBoost classification (**Figure 8**). XGBoost is an ensemble learning model which incorporates decision trees with each tree minimising the predecessor's error. Hyperparameters were optimised through GridSearchCV.

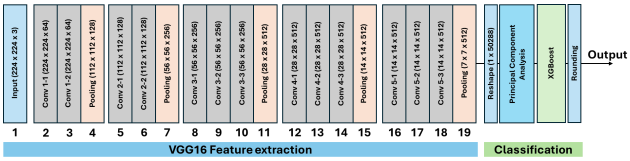

***Figure 8**. VGG16-XGBoost architecture*

Gradient-weighted Class Activation mapping

GradCAM is a gradient-based interpretability method introduced by Selvaraju et al.[14], which can be used to visualise the class decisions of CNN. The algorithm considers features from the last convolutional layer of the CNN, containing high level semantic information about the image. The algorithm computes the gradient of the class score with respect to the feature maps of the last convolutional layer, producing the weighted average of the gradients and feature map importance.

Results and Evaluation

We split our data into 80% training and 20% testing, and evaluated our model based on five metrics: precision, accuracy, recall, ROC AUC and F1 score. To fine-tune our models, we experimented with various parameters and observed the change to the stability and accuracy of the models.

Preliminarily, the baseline CNN did not focus on any particular area due to the lack of a corresponding heatmap, except for test image 2. This could suggest that its classification decisions were not based on the location of the tumour. For the VGG16 transfer learning models, the models could correctly identify the location of the tumour as seen from image leftmost of **Figure 9**. However, the model seemed to have learnt other features for classifying malignant task (e.g. edges of the skull from test images 2 and 3 and test image 5 for VGG16TF-15).

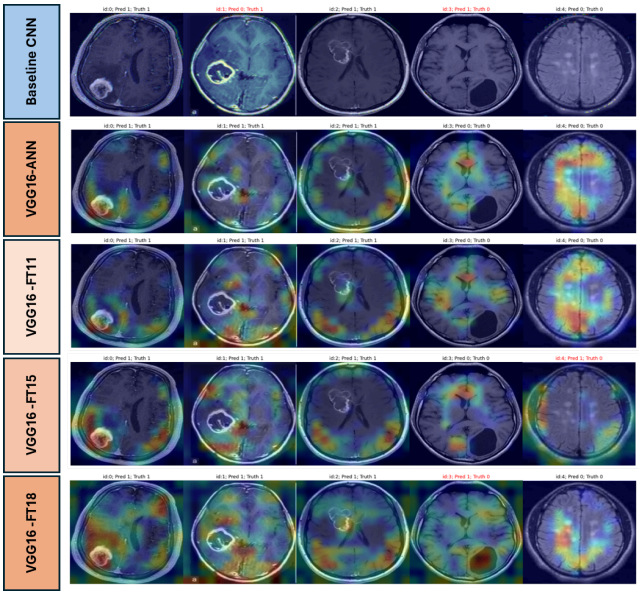

***Figure 9.** GradCAM visualisation to interpret the inner workings of the various CNN models. VGG16-FT models refer to the VGG16-ANN model with specific layers unfrozen and fine-tuned. Columns: Test images 1- 5 (left to right).*

Baseline CNN model

The baseline model demonstrated an ROC-AUC of 0.695 with a precision of 92.31%, indicating good prediction of the positive class (**Table 1**). The poor F1 score suggests that the simple architecture may not be enough to capture complex trends. To achieve this, we manually tuned the following hyperparameters to maximise the ROC AUC score: number of convolutional layers, learning rate, optimizer, batch size and dropout rate (See **Supplementary Table 2)**. With the best hyperparameters, **Supplementary Figure 2** shows the ROC AUC scores and loss during training and validation.

VGG16-ANN model

To understand the effects of fine-tuning from different convolution layers of VGG16, we froze the first *n* layers and trained the rest of the convolutional layers at a lower learning rate. We recorded the mean loss and model

performance in **Figure 10** for StratifiedKFold for k = 5. We can see a drop in loss and increase in performance between layers 12 and 15. This could be due to the use of generalised filters[15] in the first 3 convolutional blocks (up to layers 11). Hence, fine-tuning the model by unfreezing layers before 11 may interfere with these weights causing drop in performance. As such, we propose three VGG16 transfer learning models–-namely VGG16TF-11, VGG16TF-15 and VGG16TF-18. For each model, ANN is first trained for 15 epochs, before fine-tuning from layers 11, 15 and 18 onwards for 7 epochs. An early stopper was also used to monitor changes in validation loss to combat overfitting with a patience of 3 epochs. A baseline VGG16-ANN was also trained with 15 epochs and early stopping, without fine-tuning. Batch size and learning rate were manually tuned.

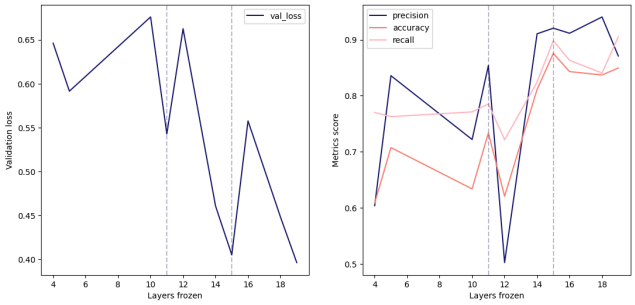

***Figure 10**. Model performance after transfer learning*

Referring to **Supplementary Figure 3**, the learning mainly occurred in the initial training epochs as the classifier was trained at a higher learning rate while less improvements were seen during fine-tuning. With early stopping, there were no big deviations between the train and test losses suggesting no significant overfitting issue. Generally, the VGG16 transfer learning models performed well for the task across the different number of fine-tuning layers with ROC AUC and F1 score above 0.85 (**Table 1**). Fine-tuning of the later layers did not bring significant improvements to the baseline VGG16-ANN model.

VGG16-XGBoost model

Since XGBoost needs a lot of data to function, we augment the train images up to a total of 1200 images. Then, after passing all images into VGG16, PCA creates a clear division of the data points (See **Supplementary Figure 4)**, thus PCA can be utilised to reduce the number of features used. Referring to **Supplementary Figure 5**, 450 and 250 is the number used in the PCA. To learn more about the effect of fine-tuning, GridSearchCV is dissected into 6 independent sections and evaluates its increase in performance. XGBoost models performed relatively well compared to Baseline-CNN and has enhanced F1 score when coupled with PCA preprocessing (**Table 1**).

Table 1: Table of performance metrics of all the models used.

Model	Precision	Accuracy	Recall	ROC-AUC	F1 Score
Baseline CNN	0.9231	0.5641	0.4286	0.7630	0.5854
VGG16-ANN[1]					
Baseline VGG16-ANN	0.8929	0.7692	0.8065	0.8744	0.8475
VGG16TF-11	0.8571	0.7692	0.8276	0.8685	0.8421
VGG1GTF-15	0.8929	0.7692	0.8065	0.9119	0.8475
VGG16TF-18	0.8929	0.7692	0.8065	0.8928	0.8475
VGG16-XGBoost					
Base Model	0.8181	0.7692	0.6923	0.7692	0.7500
Model with PCA (450 components)	0.7692	0.7692	0.7692	0.7692	0.8462
Model with PCA (250 components)	0.7692	0.7692	0.7692	0.7692	0.8462

[1]For VGG16-ANN models, baseline model refers to VGG16 convolutional blocks attached with custom classifier. The other VGG16TF models refer to specific layers that were fine-tuned after preliminary training. For VGG16TF-11, VGG16TF-15, VGG16TF-18, first 10, 14 and 17 layers were frozen during the fine-tuning phase.
Abbreviations: *ROC-AUC* Receiver Operating Characteristic

Discussion

We explored three classification methodologies and incorporated GradCAM with the baseline CNN and the VGG16-ANN. We also explored different augmentation strategies to tackle class imbalance and small dataset size. For tumour classification, false negatives can be costly while false alarms (false positives) are highly disruptive. We hence paid particular attention to F1 score, which is a harmonic mean between precision and recall rates. As such, we conclude that the VGG16-ANN model is best suited for brain MRI binary classification.

Since we were unable to track gradient changes of difference in output across VGG16 and XGBoost, this was not applied to VGG16-XgBoost. Henceforth, other model agnostic methods like SHAP (SHapley Additive exPlanations) could be explored. Furthermore, our models did not achieve similar performance to models proposed by other works about CNN with accuracies of above 90%[2]. This could be attributed to the models not focusing on the tumour region to make classifications. This means that other features in the MRI scans, such as the skull, may disrupt the model's ability to classify the tumour. Hence, future work includes exploring tumour segmentation strategies before classification.

References

1. Brain tumour: Statistics (2023) Cancer.Net. Retrieved 30 Mar 2024 at: https://www.cancer.net/cancer-types/brain-tumour/statistics
2. Gupta, M., Sharma, S. K., & Sampada, G. C. (2023). Classification of Brain tumour Images Using CNN. Computational intelligence and neuroscience, 2023, 2002855. https://doi.org/10.1155/2023/2002855
3. Krishnapriya, S., & Karuna, Y. (2023). Pre-trained deep learning models for brain MRI image classification. Frontiers in human neuroscience, 17, 1150120. https://doi.org/10.3389/fnhum.2023.1150120
4. Nguyen, A., Yosinski, J., & Clune, J. (2015). Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 427-436).
5. Zhang, Y., Tiňo, P., Leonardis, A., & Tang, K. (2021). A survey on neural network interpretability. IEEE Transactions on Emerging Topics in Computational Intelligence, 5(5), 726-742. doi: 10.1109/TETCI.2021.3100641
6. Anaya-Isaza, A., Mera-Jiménez, L., Verdugo-Alejo, L., & Sarasti, L. (2023). Optimizing MRI-based brain tumour classification and detection using AI: A comparative analysis of neural networks, transfer learning, data augmentation, and the cross-transformer network. European journal of radiology open, 10, 100484. https://doi.org/10.1016/j.ejro.2023.100484
7. Karrar N., Mohamed, F., Adnan M., Saba, T., Bahaj S., Kadhim K., & Khan A. (2023). Brain Tumor Classification and Detection Based DL Models: A Systematic Review. IEEE Access, https://doi.org/10.1109/access.2023.3347545
8. Fatima, S., Hussain, A., Amir S., Ahmed S., & Aslam S. (2023). XGBoost and Random Forest Algorithms: An In-Depth Analysis.
9. Research Gate (2020). How to determine the adequate number of data sets required for convolutional neural network? Retrieved from https://www.researchgate.net/post/How_to_determine_the_adequate_number_of_data_sets_required_for_convolutional_neural_network#:~:text=100%20number%20of%20images%20is,effective%20generalization%20of%20the%20problem
10. Preston, D. O. (2006). MRI Basics. Retrieved from https://case.edu/med/neurology/NR/MRI%20Basics.htm
11. Krawetz, N. (2011). Looks Like it. Retrieved from https://www.hackerfactor.com/blog/index.php?/archives/432-Looks-Like-It.html
12. Google Developers (2023). Handle imbalanced data. Retrieved 6 Apr 2023, from https://developers.google.com/machine-learning/data-prep/construct/sampling-splitting/imbalanced-data
13. Simonyan, K., & Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition (2014). https://doi.org/10.48550/ARXIV.1409.1556.
14. Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. 2017 IEEE International Conference on Computer Vision (ICCV), 618–626. https://doi.org/10.1109/ICCV.2017.74
15. Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? https://doi.org/10.48550/ARXIV.1411.1792