Coursework Report – Detecting and Classifying Brain Tumours from a Dataset

Abstract

This report explores the use of deep learning techniques, specifically Deep Convolutional Neural Networks (DCNNs), for the classification of brain tumours from medical images. The primary goal of the project is to evaluate the effectiveness of DCNNs in accurately detecting and classifying images of brain tumours. The dataset consists of 224 images, divided into two categories: images with brain tumours and those without. Due to the small size of the dataset, data augmentation techniques were employed to artificially expand the dataset by creating 600 augmented images for both categories, resulting in a total of 1,334 images. A custom DCNN model was designed with convolutional, pooling, batch normalization, and fully connected layers to extract features and perform binary classification. The model used Binary Crossentropy as the loss function to evaluate the performance of the classification task.

Despite the application of data augmentation and various training techniques, the model achieved a maximum training accuracy of 73% and a validation accuracy of 71%. These results, while promising, are significantly lower compared to state-of-the-art models like MobileNetV2 and Yolov7, which have reported accuracy rates exceeding 99%. The primary limitation in this study was the small size of the dataset, which contributed to overfitting and poor generalization to new data. Future work suggests that using pre-trained models and expanding the dataset would likely improve performance. This project highlights the critical role of data quality and model selection in medical image classification and emphasizes the potential of deep learning for advancing early brain tumour detection in medical diagnostics.

Introduction

The purpose of this research is to demonstrate an understanding of deep learning while evaluating the effectiveness of machine learning for classifying brain tumors from medical images. The dataset consists of images with and without brain tumors, and the objective is to develop a deep learning model capable of accurately identifying and classifying these images.

For this task, a **Deep Convolutional Neural Network (DCNN)** has been chosen due to its ability to **automatically detect patterns in images**. CNN-based models are widely used in **object detection, facial recognition, and medical imaging**, making them well-suited for this classification problem. DCNNs extend the standard **Convolutional Neural Network (CNN)** by incorporating additional dense layers and advanced preprocessing, improving **feature extraction and generalization**.

CNNs leverage **feature extraction, spatial hierarchy, and parameter sharing** to recognize meaningful patterns while maintaining computational efficiency. By applying **convolutional layers** with small filters, the model can detect localized features such as edges, textures, and tumor structures. Techniques such as **pooling layers** help reduce dimensionality, improving model performance and reducing the risk of overfitting.

Alternative models, such as **Fully Connected Neural Networks (FCNNs) or Multi-Layer Perceptrons (MLPs)**, were not chosen as they treat input as **1D vectors**, losing spatial relationships between pixels—critical for medical image analysis. Additionally, **Random Forests and Decision Trees**, while effective for tabular data, are unsuitable for image classification due to their inability to capture complex visual patterns. Given these considerations, CNNs, and specifically DCNNs, are among the most effective deep learning approaches for this task.

There are many articles that have attempted the same or similar idea with different datasets as CNNs have already been used to detect brain tumours for example, one study showcased detecting brain tumours using the MobileNetV2 model to extract the features from the images and was reported to detect brain tumours with 99.8% accuracy [7] and correctly classify its types with 99.38% accuracy. Another model Yolov7 was used to detect the presence and precise locations of brain tumours within MRI scans and achieved a remarkable 99.5% accuracy using 2548 images of gliomas, 2658 images of pituitary, 2582 images of meningioma brain tumours, and 2500 images of non-tumours.

The structure of this report will describe the proposed methodology that was used to make the deep learning model (data preprocessing, model architecture, and training process) as well as the findings and results of the DCNNs attempt at classifying brain tumours; if successful, also mention how accurate were the results, did it predict correctly and how does it compare to other deep learning models that were used in comparatively in the same or similar tasks described in other articles or case studies.

Proposed Method

For this project, I have built a Deep Convolution Neural Network (DCNN) for the model of my choosing to classify brain tumours. To begin with, I used the Tensorflow quickstart for experts' template as the initial base design of my project, as it contained the base code for making a deep learning model with good structure and comments, making the program more visually appealing and more understandable to the programmer. Afterwards, I imported the libraries that I used to create the DCNN architecture.

The dataset that I used to create and train the model had only 224 images (150 'yes' images and 73 'no' images) to work with. I also decided to not use a pre-trained model as it will allow for full customization of the deep learning model and avoid the need for generalising the trained data as pre-trained models train from general images, not specific images only.

For image retrieval from the dataset, initially used images_dataset_from_directory to retrieve images, however, I quickly realized that it was difficult to for the model to learn since I had a very small dataset. This meant that my model was memorising the images instead of learning patterns.

Instead, I switched to using ImageDataGenerator as it allowed me to not only retrieve the images, but also to add more images using data augmentation. After retrieving the images, data augmentation was applied to the training data to improve generalization by preventing overfitting and artificially increase the size of the dataset while keeping the model robust.

I've decided to create 600 augmentations of both the 'yes' and 'no' images to increase my current dataset from 224 images to 1334 images total for the dataset. This allows for the model to properly learn patterns and have more accurate results.

After, augmentation was complete, I merged all the augmented and non-augmented images from both datasets and afterwards, made a validation split of 80%-20% for training and testing (validation) respectively

Furthermore, normalisation was applied to both the trained data and the testing data to help further improve efficiency avoid data mismatching and be able to generalize further to produce more accurate results.

The proposed DCNN follows a structural pipeline that utilizes convolutional, pooling, batch normalization, fully connected layers to extract features and classify images. Each layer of the model uses a Conv2D layer with powers of 32 filters, a 3×3 kernel size, and uses ReLU activation function. Kernel regularizers are used to put a penalty on large weights to assist preventing overfitting. It is then accompanied by a MaxPooling2D layer to reduce spatial dimensions and then followed by batch normalization to stabilise training and improve convergence. Each convolutional block follows a multiple of 32 filters for each Conv2D layer (32 -> 64 -> 128). The extract features are flattened into a 1D vector in the flatten layer and then Dense (fully connected) layers with 256 -> 128 is included, followed by a 50% dropout layer to prevent overfitting. Finally, the final dense layers consist of one single neuron with sigmoid activation function for model binary classification.

Binary Crossentropy (BCE) was chosen as the desired loss function because the classification task involves binary labels (0 or 1). Since the model predicts whether a brain tumour is present (1) or absent (0) in an image, BCE is the most appropriate loss function. By minimizing this loss, the model improves its ability to correctly classify brain scans as having a tumour or not.

Binary Crossentropy measures the difference between the true labels and the predicted probabilities. It penalizes incorrect classifications more heavily when the model is confident but wrong. This helps in stabilizing training and improving classification performance.

Experimental Results

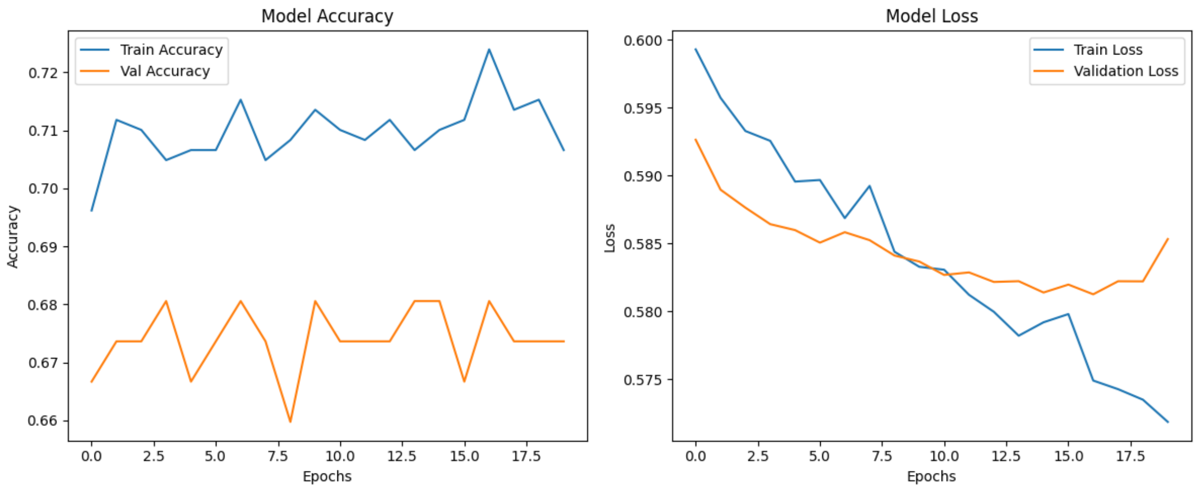
The hyperparameters help and utilise my model properly for example, I set the batch size for all the data to be batched into sets of 32 as it is the perfect balance between suitable for good accuracy and have decent efficiency at the same time. This is also helped by the fact that the size of the dataset I am using has <1000 data to work meaning there is no point having more batches as there really isn't much to train.

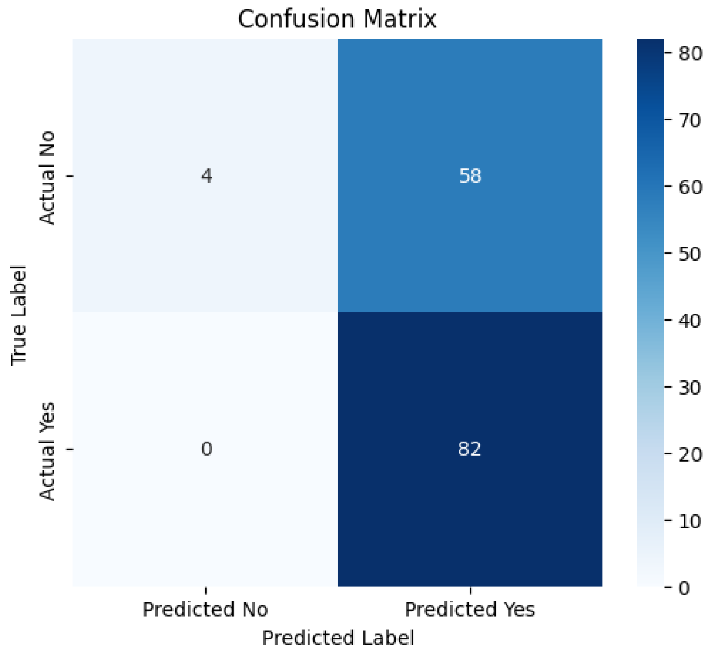
I also set the size of all the images to be trained on to be 256×256 to be consistent with the input data but to also to ensure higher accuracy when scanning an image. Finally, I ran 20 epochs as I manually tested to ensure the model could achieve high accurate results, while avoiding prolonged execution.

To assist with the execution of the model, early stopping and reduced learning rate was used to prevent overfitting, but to also make sure that the model was always improving. Suppose the model did not consistently improve, early stopping would stop the model from running, only keeping and restoring the best results/weights before the model started deteriorating, ensuring that the model used is the best-performing one on the validation data. It also saves time and performance making execution faster.

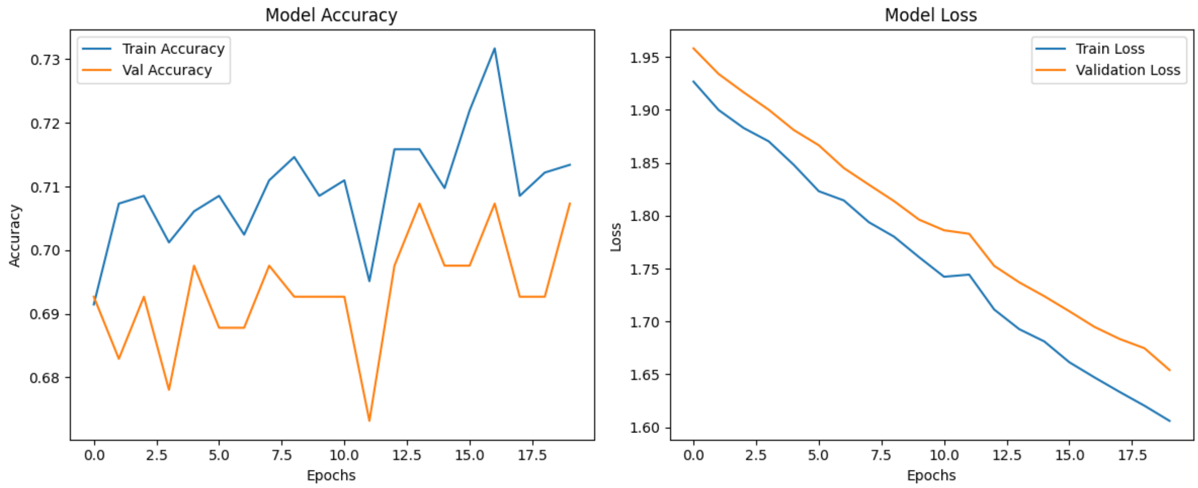
To properly diagnose and evaluate the performance of the model, graphs for the accuracy trend and the loss trend were constructed to show a good visualisation of how well the model had performed in correctly predicting the classification of images that contained brain tumours and the loss function (how far the predictions are from the true value) of the model's performance.

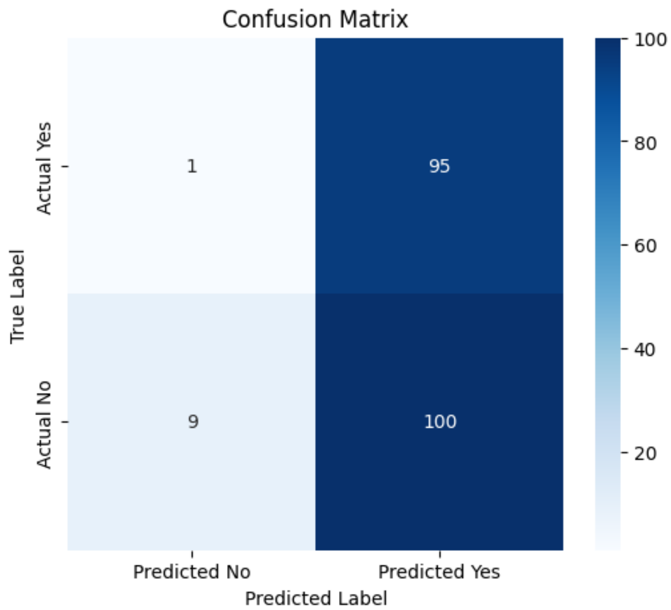
My initial performance of my model did not perform that well due to the model accuracy for the both the training accuracy and the model, with my highest training accuracy being approximately 72% and my highest value accuracy being 68%.





After attempting to improve my model multiple times, I managed to reach where I had improved the model loss of my data, however there was a still a bit of disparency between my training accuracy and value accuracy as the highest training accuracy was 73% and the highest value accuracy was approximately 71%





The overall performance of my model is ok, but it is not great for properly predicting and classifying brain tumours in a dataset. Comparatively to other studies, that use pre-trained models as mentioned previously, MobileNetV2 and Yolov7 were able to achieve accuracy scores above 99%, my highest accuracy that model was able to reach was approximately 80%

Summary

In conclusion, while the deep convolutional neural network (DCNN) model developed for classifying brain tumours showed some promising results, the overall performance was limited by several factors, particularly the small size of the dataset. With only 224 images, the model faced challenges such as overfitting and poor generalization to new, unseen data. Despite the application of data augmentation, the accuracy achieved (around 80%) is significantly lower compared to state-of-the-art models like MobileNetV2 and Yolov7, which have achieved accuracy rates exceeding 99%.

The primary limitation of this approach was the insufficient data, which hindered the model's ability to learn meaningful patterns and generalize well. Moving forward, leveraging a pre-trained model, such as VGG16 or ResNet50, would likely improve performance, as these models are trained on large, diverse datasets, enabling them to capture more complex features. Additionally, expanding the dataset with more annotated images would be essential for enhancing the model's generalization capabilities and achieving higher accuracy.

This project highlights the importance of both data quality and model choice in medical image classification tasks. Future efforts should focus on augmenting the dataset and experimenting with pre-trained models to overcome the limitations observed in this study. By addressing these challenges, deep learning models can play a crucial role in the early detection and classification of brain tumours, potentially improving medical diagnostics and patient outcomes.

References

There are no sources in the current document.

References

Introduction

https://www.researchgate.net/profile/Rohit-Thakur-9/publication/346017211_IMPORTANCE_OF_SPATIAL_HIERARCHY_IN_CONVOLUTION_NEURAL_NETWORKS/links/5fb63022a6fdcc6cc649fdb0/IMPORTANCE-OF-SPATIAL-HIERARCHY-IN-CONVOLUTION-NEURAL-NETWORKS.pdf - Importance of spatial hierarchy

https://www.tensorflow.org/tutorials/quickstart/advanced?_gl=1*1js2pta*_up*MQ..*_ga*NTk1NTM3NjU0LjE3NDE3ODI5MDM.*_ga_W0YLR4190T*MTc0MTc5NzE0MS4zLjAuMTc0MTc5NzE0MS4wLjAuMA..

Methodology

Experimental Results

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9763653

References for other alternate models or frameworks used and their results

https://keras.io/api/applications/mobilenet/ - Keras

https://bmcmedimaging.biomedcentral.com/articles/10.1186/s12880-024-01476-1 [7]

https://pmc.ncbi.nlm.nih.gov/articles/PMC10453020/#:~:text=In%20this%20research%2C%20we%20addressed,99.5%25%20accuracy%20in%20our%20analysis. [8]

https://academic.oup.com/biomethods/article/9/1/bpae080/7903126

Declaration

- I have used GenAI tools for developing ideas
- I have used GenAI tools to help me understand key theories and concepts
- I have used GenAI tools to identify trends and themes as part of my data analysis