A comprehensive review of using magnetic resonance imaging scans to detect the presence of MGMT methylation promoter in glioblastoma patients

MSc Digital Health

**Graduate School for Interdisciplinary Studies**

University of St Andrews

Submitted on: 13th August 2022

Submitted by: Harry Smith 210001720

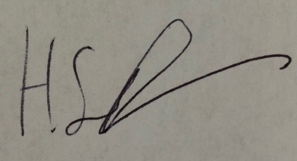
## Declaration

*I hereby certify that this dissertation, which is approximately 11,256 words in length, has been composed by me, that it is the record of work carried out by me, and that it has not been submitted in any previous application for a degree.*

*This project was conducted at me at the University of St Andrews from May 2022 to August 2022 towards fulfilment of the requirements of the University of St Andrews for the degree of Digital Health MSc in 2022 under the supervision of Professor Tom Kelsey.*

Harry Smith

Signed and dated:



13/08/2022

## Acknowledgements

Firstly, I would like to thank my supervisor, Tom Kelsey, who has provided essential advice and guidance throughout the whole project and the entire term on the Digital Health MSc course. I have learnt a huge amount from him and could have not wished for a more helpful supervisor.

I would also like to thank all my friends on the Digital Health MSc who have made the past year in St Andrews so enjoyable. The mix of backgrounds and nationalities was very special this year and I am grateful for the opportunity to study alongside some amazing people.

I would finally like to thank all my family and friends for their support throughout the dissertation process and the entire year. My love of Scotland and the east coast was distilled in me by my late father, and without this I wouldn’t have had the incredible St Andrews experience I have enjoyed so much.

## Abstract

Glioblastoma is an extremely aggressive form of brain cancer. Treatments for the disease vary and the type of treatment administered is dependent on the presence of the MGMT methylation promoter biomarker in the tumour itself. Traditionally, invasive surgery has been required to detect for the presence of this biomarker, but in the 2021 International Brain Tumour Segmentation Challenge it was proposed that this could be detected using medical imaging and computer vision. This paper assesses the landscape of attempts at performing this task and through the development of three vastly different models and novel data processing techniques looks at whether this challenge is currently achievable to obtain scores of clincial relevance. In summary, the results in this research show that it is not currently possible to achieve acceptable classification scores for the detection of the biomarker using multiparametric magnetic resonance imaging scans. However, while the results of the research do not succeed in this domain, it may be possible to use the same tools and techniques for alternative classifications of medical imaging. This paper also implements a system in the form of a web application for trained clincians to interact with these technical concepts to assess whether they may be of benefit in decision mmaking regarding patient well being.

Contents

Declaration i

Acknowledgements ii

Abstract iii

Chapter 1 – Introduction 1

1.1 Project Objectives 1

1.2 Report Structure 2

Chapter 2 – Literature Review 3

2.1 Terminology and Concepts 3

2.1.1 Glioblastoma 3

2.1.2 Genetic Biomarkers 4

2.1.3 MGMT Methylation Promoter 4

2.1.4 Multiparametric Magnetic Resonance Imaging Scans 4

2.2 Convolutional Neural Networks 5

2.2.1 Neural Networks 5

2.2.2 What are Convolutional Neural Networks? 7

2.2.3 Architectures – History of CNN designs to the present day 8

2.3 BRaTs and Related Work 10

2.3.1 The International Brain Tumor Segmentation Challenge (BRaTs) 10

2.3.2 Related Work 11

2.4 Summary and Research Proposition 11

2.4.1 Summary 11

2.4.2 Research Proposition 12

Chapter 3 – Methodology 13

3.1 Tools, Software and Resources 13

3.2 Data Visualisation 13

3.3 Data Pre-processing 15

3.3.1 Data Augmentation 15

3.3.2 Two-Dimensional Data Processing 17

3.3.2 Three-Dimensional Data Processing 18

3.4 Model Development 19

3.4.1 Two-Dimensional Model 19

3.4.2 ResNet 20

3.4.3 Three-Dimensional Model 21

3.4.4 Other Models 21

3.5 Application Development 22

Chapter 4 – Results 24

4.1 Competition Results 24

4.1.1 Developed Models 24

4.1.2 Competitors Models 25

4.2 Application Results 26

Chapter 5 – Discussion and Conclusion 28

5.1 Discussion 28

5.1.1 Strengths and Limitations 28

5.1.2 Future Development 29

5.2 Conclusion 30

References 31

## Chapter 1 – Introduction

Glioblastoma is the most frequent and aggressive form of brain tumour [1]. Prognosis is extremely poor with many patients dying within a year of diagnosis. The way the condition is treated is dependent on the genetics of the tumour itself with different individuals reacting to treatments in different ways. Reaction to treatment can be predicted by looking for the presence of the MGMT methylation promoter biomarker in the tumour. While currently invasive surgery is required to assess for the presence of this biomarker, the 2021 International Brain Tumour Segmentation Challenge was aimed at examining whether it was possible to detect the biomarker using conventional medical imaging in conjunction with innovative deep learning techniques.

1.1 Project Objectives

The overarching aim of this project is to provide a comprehensive assessment of whether using multiparametric magnetic resonance imaging can be used in combination with deep learning to assess for the presence of MGMT methylation promoter. This will be done through an assessment of the leading submissions of the International Brain Tumour Segmentation Challenge as well as the development of novel techniques to gauge the feasibility of this area of research. This will then be expanded upon to consider how this interdisciplinary area of medical research and computer science could be deployed for use in a clinical setting by users without a technical background in the form of a web application.

Thus, the primary objectives of the project are:

* Conduct a literature review to understand and explain what glioblastoma is and how it is treated. This will then examine the current state-of-the-art deep learning methods which could be used to approach the problem before providing an overview of the International Brain Tumour Segmentation Challenge and how it has been approached by other competitors.
* Develop and implement novel and tested data pre-processing techniques for multiparametric magnetic resonance imaging scans.
* Develop a selection of candidate deep learning models of varying complexity and assess them against unseen test data to detect for the presence of MGMT Methylation promoter.
* Develop and deploy a proof-of-concept web application for clinicians to utilise the data pre-processing techniques and models that have been developed and trained.

1.2 Report Structure

This report is designed to supplement the code and models developed which are available at: <https://github.com/smithharryh/Masters-Dissertation>. As such, the structure of this report is as follows:

* Chapter 2 – Literature Review: This chapter will provide a comprehensive overview of what glioblastoma, is and how it is treated, as well as explaining what biomarkers are and how the MGMT methylation promoter is currently detected. This leads on to an assessment of the deep learning landscape and a summary of the International Brain Tumour Segmentation Challenge.
* Chapter 3 – Methodology: This section outlines the process taken in developing the data pre-processing techniques, candidate models and web application. Here, the tools and technologies used in development are described as well as providing an in depth look at the differing models’ architectures.
* Chapter 4 – Results: This chapter displays the results achieved by the models in the International Brain Tumour Segmentation Challenge and explains the differences and similarities with other competition entrants. There is also a demonstration of the web application developed, showing the features that have been implemented.
* Chapter 5 – Discussion and Conclusion: The final section of this report analyses the results of the models in the wider context of the research area as well as looking at the strengths and limitations of the research. Potential further developments for the project are proposed before the project’s findings are summarised in a conclusion.

## Chapter 2 – Literature Review

This chapter will begin with an explanation of the key concepts relating to the project. Starting with glioblastoma itself, information is provided on what the illness is, who it primarily affects and the treatment available. Next biomarkers are examined to show how they can be used as a measure of disease and treatment before a brief explanation of what multiparametric magnetic resonance imaging scans are and how they are a useful medical imaging modality for use in this project. In conjunction an overview of convolutional neural networks will be put forward, while explaining what the state-of-the-art architecture research consists of in this area. Following this, there will be an examination of published attempts to use multiparametric magnetic resonance imaging scans to analyse the glioblastoma tumour’s genetic characteristics. The chapter will conclude by summarising the key findings and presenting a proposition of where the research presented in this paper may be of benefit in treating glioblastoma patients.

2.1 Terminology and Concepts

2.1.1 Glioblastoma

Glioblastoma is the most common and aggressive form of brain tumour [2]. When compared with incidence rates of other cancers glioblastoma may appear low at 3.4 people per 100,000 [3], but with a median survival rate of 15 to 23 months [4], prognosis is often extremely poor and carries with it the lowest long-term survival rate of malignant brain tumours [5]. Current research does not indicate any known specific causes either genetically or sociologically and presently no curative therapies exist for any age of patient [6].

Extensive efforts have been made in an attempt to develop new treatments for glioblastoma patients, with little progress being made. The most common form of treatment for glioblastoma is a type of chemotherapy called Temozolomide. This can be administered orally and penetrates deeply into the central nervous system. It functions by methylating the DNA in order to prevent tumour proliferation by depleting the repair enzyme O6-methylguanine-DNA methyltransferase (MGMT) [7]. In short, if this repair enzyme is depleted, the cancer cells cannot split into 2 new cells and the cancer is unable to grow. Temozolomide is especially effective in patients whose tumour contains a methylated MGMT promoter, as this “renders the enzyme ineffective” [8].

The use of temozolomide is a treatment and not a cure. While no cure is readily available for glioblastoma, there is research underway currently to examine potential therapeutic options with the use of genomic analysis of the cancer itself at the forefront of drug development and discovery [9]. Furthermore, this genomic analysis can be used to identify genetic biomarkers to uncover new treatments and understand the impact they may have on given patients.

2.1.2 Genetic Biomarkers

Biomarkers are defined by the United Nations and International Labor Organisation as “any substance, structure, or process that can be measured in the body or its products and influence or predict the incidence of outcome or disease” [10]. More formally, Hulka et al state that biomarkers are “cellular, biochemical or molecular alterations that are measurable in biological media such as human tissues, cells, or fluids” [11].

While there are many biomarkers used for evaluating treatment and disease outcomes, in the case of glioblastoma, genetic biomarkers are particularly useful. These are mutations or polymorphisms in the gene which can be used to predict disease susceptibility, outcome, or treatment response [12]. To understand the potential effectiveness of using temozolomide for individual patients, the methyl-guanine methyltransferase (MGMT) gene with methylated promoter is examined.

2.1.3 MGMT Methylation Promoter

MGMT is a DNA repair enzyme that works by repairing the positions of the guanine which are alkylated by temozolomide [13]. This gene has been associated with increased survival duration in patients with glioblastoma who receive the alkylating agent temozolomide [14]. Patients with glioblastoma which did not contain methylated MGMT promoter did not appear to benefit from this conventional treatment, thus illustrating the importance in understanding its presence in every individual patient.

However, information on the status of MGMT in the tumour is only available if patients have segmental resection surgery where the gene can be accurately profiled. In addition, not all laboratories are able to analyse for the presence of MGMT methylation promoter and the time taken for this to occur may mean results are available after the patient has begun treatment. Therefore, to improve the treatment available to glioblastoma patients, it would be beneficial to know of the presence of the MGMT gene without the need for a surgical procedure.

Recent research has suggested that molecular profiling of the tumour may be possible using magnetic resonance imaging (MRI) scans, suggesting histologically similar tumours “demonstrate highly distinct imaging profiles on MRI” [15]. To develop this further, multiparametric MRI scans can be used to provide a greater volume of information from a single scan.

2.1.4 Multiparametric Magnetic Resonance Imaging Scans

Multiparametric MRI (mpMRI) is a unique type of MRI scan which can produce more detailed imaging than a traditional MRI. mpMRIs combine several MRI techniques into a single scan. Essentially it is a “method of trying to obtain an ideal three-dimensional (3D) image by combining T2-weighted (T2WI), diffusion weighted (DWI), dynamic contrast enhanced (DCEI) and, … MR spectroscopy (MRSI) images” [16]. Due to the volume of information available for analysis from an mpMRI scan, they are “routinely acquired for patients with suspected brain tumours” [17].

MRI scans have the advantage of being non-invasive for the patient as there is no ionising radiation and thus, they benefit from high patient acceptance. Combining this with the extra data available through mpMRI scans presents an opportunity for analysis of glioblastoma tumours without requiring surgery. Using computer vision algorithms to analyse mpMRIs can provide insights into the genetic makeup of the tumour, specifically whether it contains the MGMT methylation promoter gene.

2.2 Convolutional Neural Networks

2.2.1 Neural Networks

In recent times, neural networks have become synonymous with computers being able to execute tasks with superhuman ability. In reality, Neural Networks are a group of algorithms useful for “classification, clustering, pattern recognition and prediction in many disciplines”[18], which are loosely modelled on the structure of the human brain. Where in conventional programming, a human instructs a computer in how to perform a particular task, in neural networks, the computer “learns from observational data, figuring out its own solution” [19]. While the mathematical principles of artificial neural networks were first proposed in the 1940s by McCulloch and Pitts [20], it was not until the early 21st Century in which appropriate training algorithms were presented, and computational power was sufficient for these algorithms to be implemented in practise. At a high level, neural networks work by taking a large number of training examples and using these examples to develop a system which can perform as desired on new, unseen data.

More granularly, neural networks are composed of layers of nodes, weights and biases. Nodes are “simple linear or nonlinear computing elements that accepts one or more inputs [and] computes a function thereof” [21]. A node, or neuron, receives a signal, performs a function on this signal and presents an output. In order for a computer to be able to perform a calculation, the input and output of each node must be a real number. Nodes are arranged in layers, including the input layer where the data is passed to the network, and an output layer, where a prediction is made for either regression or classification. In between the input and output layer are a variable number of hidden layers, with each layer performing a different transformation on its signal to reach the desired output by the final layer. They are referred to as hidden layers as their inputs and outputs are hidden by an activation function.

The layers in a neural network are connected by weights, or edges. Weights are the parameters of a network which are updated by iterative training methods. These work in conjunction with biases which are the intercept terms of the linear components. The number of layers in the network is referred to as the depth, hence the name deep learning for models with a large depth, whereas the width refers to the number of nodes in a given layer [22].

Diagram

Description automatically generated

Figure 1 – Architecture of a neural network [23].

In order for neural networks to be able to make effective and accurate predictions on unseen data, they must go through a training process. This training process is an iterative, systematic methodology used to update the model’s weights so that when a new example is provided to the network, the weights are set so that an accurate prediction can be made. Put simply, training is the process of minimising a cost function by lowering and increasing the weights throughout the network. The challenge of training neural networks is to ensure they are fit optimally to the training data so that they can generalise for unseen data. If the model’s weights are adjusted such that it cannot generalise to test data as they are too specialised to the labelled training data, this is said to be overfitting. The inverse is said to be underfitting, and the essential quality of neural network training is ensuring a balance in between over and underfitting.

There are many different configurations of neural networks with the variables which determine the structure of the model known as hyperparameters. Different model architectures perform well on different tasks and on different data. For image-driven pattern recognition tasks, such as detecting the presence of MGMT methylation promoter in mpMRI scans, convolutional neural networks are commonly used.

2.2.2 What are Convolutional Neural Networks?

One limitation of traditional neural networks is they tend to struggle with the complexity of datasets with a large number of data points. Image data for example contains multiple channels for colour and dimensionality. This results in unwieldy, large networks which are difficult to train in a way which can allow them to generalise to unseen data [24].

Convolutional neural networks (CNNs) derive their names from a mathematical operation called convolution, a specialised form of linear operation which can help to address the problem of overfitting in artificial neural networks. CNNs are typically composed of three types of layers: convolutional, pooling and fully connected layers.

A convolution is an operation that “slides one function over another and measures the integral of their pointwise multiplication” [25]. In practise, this operation extracts patches from the input and applies the same transformation to every patch to create an output feature map [26]. In a CNN, the hidden nodes are replaced with a kernel convolution. Instead of the neurons being connected to every neuron in the previous layer, in CNNs, each neuron is only connected to a small region of the preceding layer [27], reducing the number of linkages. This architecture allows the network to concentrate on small, low-level features, or local patterns, and assemble them into larger features as the layers progress, looking for more complex patterns to extract, resulting in a hierarchical structure. This structure is common in real-world images and is one reason why CNNs have exhibited such positive performance in image recognition tasks.

Following a convolutional layer, a pooling layer is often introduced. The role of the pooling layer is to aggressively down sample and aggregate the convolutions resulting feature maps and, most commonly, output the maximum value in each channel [28]. This layer is necessary for two reasons. Firstly, to ensure that the features from the final convolutional area contain information about the whole of the input instead of only a small feature space. Secondly, to lower the total number of parameters used in the model and avoid the possibility of overfitting.

Commonly, in a CNN architecture multiple convolutional and pooling layers are alternated between before a regular fully connected layer is added before an output layer to combine the different feature maps for prediction. However, since the original conception of the LeNet-5 CNN architecture in 1998, a range of designs have been proposed.

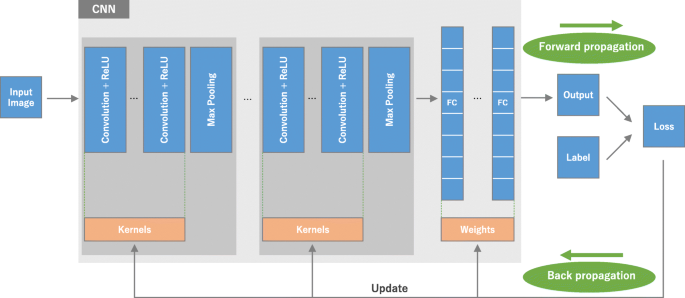


Figure 2. Simple CNN architecture [29]

2.2.3 Architectures – History of CNN designs to the present day

One of the original CNN architectures is known as LeNet-5. Created by Yann LeCun et al [30], it has been widely used for image recognition tasks and was originally proposed for handwritten digit recognition of the MNIST dataset (http://yann.lecun.com/exdb/mnist/) on which it received an error rate of 0.8%. While providing initially promising results, the architecture is now seen as primitive. The use of the square of the Euclidean distance between the input vector and the weight vector in the output layer to measure the probability of an image belonging to a given class tends to be slow to converge and does not penalise errors quickly. In more recent model designs, a cross-entropy cost function has led to improved results and training speeds [31].

As the early 21st Century complied with Moore’s Law [32] and brought improved processing speed and computational accessibility, neural network architecture design improvements began to gather pace. In 2010, the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) begun. This challenge was based around a hand-annotated dataset of 14 million images split across 20,000 classes. The goal was to accurately classify and detect objects and scenes from the dataset, and its annual running brought significant advancements in deep CNN architectures.

In 2012, the ILSVRC challenge was won using an architecture called AlexNet, which achieved an error rate of 15.3% [33]. The network was based on the original LeNet-5 but differed as it was much larger and deeper and was the first network to stack multiple convolutional layers. Furthermore, this network used the regularisation method of dropout, where a certain percentage of neurons are excluded when making predictions to reduce overfitting. The architects of this model also used data augmentation techniques such as flipping the images horizontally, in order to synthetically create a larger dataset.

In 2014, a new network called GoogLeNet won the challenge with an error rate of 7% [34]. The most significant advancement realised in this network architecture was a further increase in depth made possible by inception modules. Inception modules allow for parameters to be used more efficiently than in other architectures. This works by performing convolutions on an input with multiple different sizes of filter on the same level, and then concatenating the output for the result to be sent to the next level.

Diagram

Description automatically generated

Figure 3. Basic Inception Module architecture [35]

The inception module proposal is a form of the “network-in-network” proposed by Lin et al [36] and is utilised in GoogLeNet to create a 22-layer deep model.

The following year, the ILSVRC was won by Kaiming He et al using a Residual Network [37] (ResNet), which achieved an error rate of 3.6%. This was achieved as when developing the winning model, the researchers acknowledged that as the number of layers increases beyond a certain point, the error rate of the network does not improve on test or training data. This is due to a problem known as the vanishing gradient problem, whereby the gradients of the loss functions converge to zero and thus makes the network harder and longer to train [38]. To resolve this, the researchers implemented skip connections, where the network skips training several layers and connects directly to the output. This ensures the networks fits to the residual mapping of the signal, thus improving performance and training times.

Other minor developments have been made in model development since ResNet. In 2016, the Xception model was introduced which built on and improved the concept of inception modules [39], followed by the realisation of the Squeeze-and-Excitation network (SENet) in 2017 [40]. Both of these models expanded on the convolution operator central to CNNs, with the SENet achieving a 2.25% error rate in ILSVRC 2017, the final year of the competition. From this point, the research area has been in an age of refinement, not invention, with focus turning to making networks perform faster and with fewer parameters so that they are suitable for mobile devices. The introduction of ShuffleNet [41] and EfficientNet [42], achieved similar accuracy results while being significantly smaller and faster than previous models.

2.3 BRaTs and Related Work

2.3.1 The International Brain Tumor Segmentation Challenge (BRaTs)

Many of the state-of-the-art CNN model architectures were used in the leading RSNA-MICCAI Brain Tumour Radiogenomic Classification competition in 2021. One of the 8 leading submissions for the competition announced by the RSNA was based off a ResNet10 architecture. ResNet10 was proposed as a lightweight solution to improve performance of the residual network architecture without a loss in accuracy. ResNet10 consists of 10 convolutional layers in total and thus contains one-tenth the number of parameters when compared to ResNet18, a closely comparable version of the model [43].

Of the eight winning solutions, six have their highest scoring submission publicly available, four of which use a lightweight model such as a version of ResNet or EfficientNet. One of the models uses the UNet++ architecture [44]. This design is an extension of the U-Net model which relies on data augmentation to improve training results, rather than a larger number of samples [45]. The UNet++ version was created specifically for medical image segmentation to overcome 2 problems with traditional CNNs. Firstly, usually with CNNs, to determine the optimal depth of the network a search must be conducted over a given parameter space to understand at what point the model begins to overfit. Secondly, the use of skip connections as used in residual networks comes with some limitations, as they “impose an unnecessarily restrictive fusion scheme” [46]. UNet++ aims to address these concerns, but results from the BRaTs challenge suggest that while the network performs comparably well, there is no significant gain over other CNN architectures.

The final model in the top eight solutions does not use a neural network at all, instead using a support vector machine (SVM). SVMs are classification machine learning models, used to learn by example to categorise new instances of data. More formally, they are “a mathematical entity for maximizing a particular mathematical function with respect to a given collection of data” [47]. SVMs are powerful and versatile and thus are a popular machine learning algorithm. While this solution did score lower than the other known CNN submissions, it still achieved one of the highest levels of accuracy out of the 1556 submissions to the competition.

2.3.2 Related Work

Many attempts have been made to use non-invasive MRI scans to predict MGMT promoter methylation status in glioblastoma patients using machine learning methods with varying results. Chang et al [48] received an encouraging 83% accuracy in predicting MGMT promoter methylation status on a set of mpMRI scans obtained through the Cancer Imaging Archives (TCIA) and the Cancer Genome Atlas (TCGA) using a residual convolutional neural network.

In a similar study Korfiatis et al [49] achieved an accuracy of 94.95% on unseen test data using a ResNet50 model. This study also used minimal data pre-processing techniques and instead relied upon the complexity of the model, with the aim of acting as a proof of concept for how the computer vision technology could be implemented in clinical practise to reduce the need for distinct tumour segmentation. Here, the T2 MRI pulse regime is included in the dataset, which the researcher’s credit with improving the accuracy. However, it is of note that in this study, 78% of the total images used to test the system did not contain a tumour at all, and the data used for the study all came from the same academic centre. Therefore, it is possible that this high accuracy has been skewed by poor data collection methodolgy.

Despite the apparent success of using CNNs for the task, there are some compelling results which suggest that prediction in this manner may prove challenging to the degree of accuracy required for use in clinical practise. Han et al [50] used a bi-directional convolutional recurrent neural network on the TCIA brain scans to predict MGMT Methylation promoter status. The dataset was pre-processed to remove particularly noisy images and normalised across pulse regimes so that slice thickness was consistent. Furthermore, the data was augmented by rotating the images thus increasing the size of the data set 90-fold. Despite these methods used to combat overfitting and the sophisticated architecture of the network, the system achieved a patient level accuracy of 62%. However, the researchers note that despite a large number of scans, they only correspond to 21 patients, and conclude by suggesting that their results posit a link between MGMT methylation status and the characteristics of the tumour in the MRI. Further investigation is required using different architectures and data to legitimise the concept.

2.4 Summary and Research Proposition

2.4.1 Summary

Glioblastoma diagnosis comes with an extremely poor prognosis, with low survival rate and fast quality of life deterioration in patients. The most effective and common form of treatment is the use of the chemotherapy temozolomide, but this has only been proven effective in patients with the MGMT methylation promoter biomarker in the tumour itself. Where traditionally distinct tumour segmentation has been required to identify the presence of this biomarker, advancements in CNN architectures over the last decade have presented the opportunity of using mpMRI scans and machine learning techniques do this without surgery.

This context survey has provided information on Glioblastoma, its prognosis and treatment. Separately, information has been provided on CNNs and how they work, as well as a summary of the major breakthroughs in their architectures. These two areas have been combined to examine the current approaches of using these technologies to examine biomarkers in brain scans.

2.4.2 Research Proposition

Some published literature suggests that a high degree of accuracy is available when crafting classifiers to perform this task, while accuracy in the BRaTS competition suggests the inverse. Every neural network and architecture has a near infinite number of hyperparameters and possibilities to explore and examine. This project looks to create novel ways of data processing and model training to understand the feasibility of the challenge of detecting MGMT methylation promoter in medical imaging, and to provide a comprehensive examination of both the results delivered by other competition participants and the models delivered in this report.

This will be expanded upon to create a proof-of-concept web application, which will show how it may be possible for clinicians to provide a folder of mpMRI scans in DICOM format and receive a probability of the MGMT methylator promoter status in a patient’s tumour. Combining these two aims will demonstrate both the potential of the technology and how it could be implemented into clinical practise.

## Chapter 3 – Methodology

This chapter outlines the methodology taken to build the classification models and deploy them for use on unseen data in the form of a simple web application. This begins with a presentation of the available open-source tools used to aid development, before examining and visualising the available data. Next, the different data pre-processing techniques that have been developed and implemented are discussed and an examination of the models trained is put forward with an explanation of how they have been assessed. Finally, the chapter concludes with a demonstration on how the web application was developed and the data pre-processing techniques and models implemented in it.

3.1 Tools, Software and Resources

As the data used for development was available on Kaggle, most of the development of the models was performed on the Kaggle IDE platform. This provided 37 hours of free GPU access per week, and 20 hours of free TPU access, which proved essential for training the models. Using the Kaggle platform meant that the 136.85 gigabytes of training and test data could be stored remotely.

Python was used to train the models and build the web application, and for data pre-processing and model development, a range of open-source python libraries were used:

* Pydicom – This was used to load the DICOM files and extract the image data from the metadata. This library also provided some fundamental pre-processing techniques.
* OpenCV – This package was used to resize and normalise the image data.
* Scikit-learn – This package was used to split the datasets into training and validation sets.
* TensorFlow – This library was the core behind the development and training of the machine learning models.
* Streamlit – A package which was used to create and host the web application, allowing for fast deployment of data science models.
* Python utility libraries – Numpy, Pandas and tqdm were all used for data analysis and visualisation purposes.
* Heroku – A platform used for deployment and hosting of the web application.
* Amazon SW3 – Amazon Web Services used for hosting the models’ weights for classification on the web application.

3.2 Data Visualisation

Before model development began, the available data was visualised, plotted, and examined to look for imbalances in the training set and understand the make-up of the dataset itself. The training data available consists of four types of mpMRI scans for 585 patients, whereas the test set contains 87 patient’s scans, each containing the four types. Each patient in the training set is labelled with either a 1 to represent the presence of MGMT Methylation Promoter, or 0 to signify its absence. The mpMRI types in both datasets are: T1-weighted (T1w), T2-weighted (T2w), T1wCE and Fluid Attenuated Inversion Recovery (FLAIR). The physics justifying the difference in the scans is outside of the scope of this report, however, in brief, each of the different types have differing pulse regimes resulting in difference of contrast in the resulting images. This can be useful in assessing different features of a given scan. Figure 4 shows an example frame from a given patient in the test set for each scan type.

A picture containing invertebrate, jellyfish, dark

Description automatically generated

Figure 4 – Example frame for each image type for Patient 00001.

Visualising each image shows that the plane type is not consistent across MRI types. As the scan types are available across the axial, sagittal and coronal planes, a combined voting mechanism may be of use to assess the output of the classification and make use of the different insights available from the different MRI types.

The labels of the training set are reasonably balanced, with slightly more patients (307) exhibiting the MGMT Methylation Promoter than those without it (278), as shown in Figure 5.

A picture containing logo

Description automatically generated

Figure 5 – Distribution of training labels.

In the Data Visualisation file, an example patient’s scans have been plotted and visualised across the 4 mpMRI types, revealing some illuminating observations.

Firstly, not all scan types have the same number of DICOM images per patient. For example, for patient 00000, 129 T1wCE, 400 FLAIR, 408 T2w and 33 T1w images are available. As the neural networks being developed will require the data to be input in the same shape, this will need to be addressed.

In conjunction, not all patients have the same number of scans for each type of mpMRI, suggesting that the slice time between scans is variable across patients. This could have an effect on training and evaluating the algorithm as patients will have different amounts of data available for use in classification. To resolve this, either the patients with minimal data could be removed and not used for training, the patients with an abundant amount of data could be sampled to acquire the same number of scans as the patients with less data, or the patients with less data could be padded to make up the difference.

Thirdly, it is apparent that there are a number of empty scans without any form of tissue that are present across all of the different pulse regimes. The occurrence of these black images appears to be stochastic and although it is most common that the middle of the scans contains the most useful data, this does not always seem to be the case.

After observing these features of the dataset and examining the context survey, the most common approach used by competition participants to address these concerns is to select a given number of samples from the middle of patient’s data. If the patient’s data is smaller than this number, then all of the samples are used, and the data is padded with blank images to match the required size. While some competition submissions favoured appending augmented scans to pad the data to be the correct shape, this appeared to have a negative impact on classification accuracy.

The final step in the data visualisation process was to examine the effect of applying the value of interest look up table (VOI LUT) function. The VOI LUT is a transformation of the pixel values into values which are meaningful and useful for the application by adjusting the window height and window width. This is used commonly when working with DICOM files.

3.3 Data Pre-processing

3.3.1 Data Augmentation

The data augmentation stage began with an examination of the techniques used by other participants in the BRaTS competition and the impact these had on results. Firstly, almost all entries used a VOI LUT application to transform each DICOM file into values more meaningful for use model development. The was applied when using the pydicom library (<https://pydicom.github.io>) to load the DICOM images, as the library contains a function for applying a VOI LUT function to a given file.

Next, it was important to normalise each image to allow for further data pre-processing techniques. Currently when loaded with the pydicom library, the data was available in Hounsfield Units (HU), but this created challenges when applying transformations on the data. For this reason, each image was normalised with each voxel being scaled to between 0 and 255.

It was necessary to address the problem of blank images in each patient in the dataset which may cause issues with classification. This was a simple problem after the image was normalised as if the maximum and minimum values for the pixels in the image were the same, then the image was classed as blank could be removed. However, it was not just blank images which could alter results. There were parts of the scan which contained only very small regions which could be viewed as brain, an example of which is shown in Figure 6.

A picture containing invertebrate, jellyfish, dark

Description automatically generated

Figure 6 – An example of a patient’s scan where only a small part of the brain is visible

While many participants had listed this as a potential problem when training the algorithm, few offered a solution. Most solutions elected to select a number of scans from the middle of the patient’s dataset to train the algorithm, hoping that this would contain the majority of useable data. However, this was no guarantee. To address this problem, the solution developed meant that when loading the data, a minimum image quality percentage could be specified. The algorithm then calculated this percentage as a number of pixels in the image. If the image contained more non-zero pixels than this amount, it would be used for training the algorithm. If not, the image was discounted from training. Through trial and error, 20% was found to be the most appropriate value to exclude blank images and include the highest amount of useable data. However, in future practise it may be of value to adjust this value depending on the given dataset and knowledge of a patient’s condition.

The next step in data preparation was to remove as much of the blank border around each scan as possible. As each step in a patient’s scans contained a variable amount of brain volume, removing the border was essentially to not skew the classification based on the size of the surrounding black area. Other participants had provided methodologies for doing this, some of which cropped edge sections of the brain in order to remove as much of the border as possible, where other solutions opted to crop only rows and columns in the image which contained all black pixels. Analysis of both showed that on balance the latter of these two options was optimal and tended to achieve more prosperous results, so the solution promoted in the user ‘YU4U’’s Kaggle notebook [51] was adapted and implemented.

Finally, once all transformational techniques had been applied, each image was resized and scaled to between 0 and 1 to be passed to the input layer of the neural network. Differing image sizes were tested with 64 by 64 pixels determined to be the optimal trade-off between memory capabilities and improved results.

Other participants opted to use different data generation pre-processing techniques, such as rotating and flipping the available images to create a larger dataset. While there may have been value in artificially transforming the different MRI types into the other planes, it was shown that there was not a significant improvement in competition score in performing this data augmentation. Furthermore, the memory and computational resource limitations dictated by the competition organisers suggested that this was not a valuable use of resources.

Two categories of model type were to be developed, two-dimensional and three-dimensional, and the data would need to be processed and structured differently for both model types.

3.3.2 Two-Dimensional Data Processing

The first attempt in loading the data for the two-dimensional models was a simple loop to append each image for each patient to a list with a corresponding list for the given label. Performing this acted as inspiration for many of the data pre-processing techniques as loading all images of each patient took up too much memory and made model training impossible.

The next attempt was to use a python generator to yield batches of data to the model for training, with the aim of improving memory management and allow for efficient batching of the data. While this improved performance it ruled out the ability to drop blank scans from dataset as the location of the blank scans was stochastic and thus a consistently sized yield could not be guaranteed.

The following iteration involved using the TensorFlow data API to load and batch the data. This had the advantage of providing the best possible performance when using TensorFlow to train and develop models as it loads the data into a graph-based structure of tensors. However, this made performing the necessary data pre-processing steps extremely challenging and un-wieldy as only TensorFlow operations could be performed on the data due to the graph structure TensorFlow uses to house the data, ruling out the use of the pydicom and numpy libraries. While this would have been possible if the scans were in a conventional image format such as a PNG, the tools available in TensorFlow were not best suited to the DICOM format.

In the end, the most beneficial methodology was to create a custom data generator class which contained methods for each pre-processing technique. The data was then pre-processed before being stored in memory and passed to the model. This was limited to a degree by memory constraints, but this did not have a significant impact on development and came with the advantages of being able to use all the data pre-processing techniques that had been developed and the data could be batched efficiently for training the model.

3.3.2 Three-Dimensional Data Processing

In the two-dimensional data model, each step in a scan is considered to be independent. This means that there is no relationship between each DICOM scan and thus all patients are viewed as a homogenous body, instead of as being distinct from one another. To address this, the data could be modelled in a three-dimensional way, where the scans for each patient are separate from each other.

At first a similar approach was taken as when developing a data loader for the two-dimensional data, including the use of a python generator and the TensorFlow data API. However, the same problems soon became apparent. Furthermore, it was apparent that loading the data in as similar a way as possible between two-dimensional and three-dimensional versions would be a significant benefit when making the web application.

However, there were some marked differences when developing the two ingestion methods. Firstly, a uniform number of scans was needed to be taken from each patient when developing the three-dimensional model to ensure that the shape of data been passed to the model was consistent across patients. This ruled out the option of removing blank scans from a given patients dataset, and some patients would need to be padded with blank scans to ensure a consistent approach. After examining a random sample of patients and considering the approach of other competition entrants, it was determined that 64 scans should be extracted from the middle of each patient’s data. This number allowed for the maximum amount of meaningful data to be utilised while blank padding would only be required for a small number of patients. It is worth noting that this approach has some significant flaws, as the model is unable to learn anything from the blank scans and potentially useful insights could be missed. However, this was by far the most popular methodology chosen by competition participants and other solutions which were tested were either ineffectual or made no improvements on the model’s performance.

Despite these differences, the three-dimensional data loader utilised the same methodology for applying the VOI LUT function, resizing and normalising the image and removing the surrounding black border from the image.

Once complete, both data loading methods and the pre-processing methods were moved to a data processing utility file which could be used with little modification for loading data for classification on the web application, and the development of all the models.

3.4 Model Development

3.4.1 Two-Dimensional Model

The first model developed was a two-dimensional convolutional neural network. This laid a lot of the groundwork which other models could then develop on. The core structure of the model drew inspiration from Francois Chollet’s architecture [52]. It is a simple structure with two-dimensional convolutional layers alternated with max-pooling layers, stacked on top of each other. The model uses the ReLu activation function for each convolutional layer and has a dense output layer with the sigmoid activation function for the binary classification problem.

A random search was conducted to determine and set the optimal hyperparameters for the model to achieve the best performance. This included setting the batch size, and while a batch of 8 and 16 both did not exhaust all memory requirements, a batch size of 8 resulted in the best score. Performance was measured with the area under the curve (AUC) score on a separate validation set. AUC was used as a performance metric as this was also used in the assessment of models for the competition.

The model was set to run for 20 epochs, but this was monitored by two call-back methods. The first was a checkpoint call-back, used to save the weights at the epoch which gave the best AUC score on the validation dataset. An early stopping call-back was also used to monitor the validation AUC and stop the model’s training if the score did not improve for 10 epochs. If compute resources and time was limitless this would not have been necessary or useful, but as GPU usage on Kaggle was capped at 37 hours per week it was important not to unnecessarily use this resource. It was also evident that peak AUC was often achieved within the first 10 epochs of training, so often continuing training beyond this point did not improve the model’s predictive performance.

As with the other models developed, four separate neural networks were created, one for each of the MRI types. Training time for each model was variable but, due to the low level of model complexity, did not exceed three minutes. They were then all assessed and evaluated for individual performance before the results were compiled to illustrate the effectiveness of the multiparametric input.

To assess the models, an inference notebook was used. This is a common technique used in competitions on Kaggle where the model is trained, and the weights saved before a separate notebook loads the weights and assesses the test data to provide a competition score. This eliminates the unnecessary use of compute resources and speeds up the scoring process. Once the two-dimensional model had been trained and assessed, verifying the effectiveness of the two-dimensional data input pipeline, a more complex model was developed using the ResNet architecture.

3.4.2 ResNet

Development of the ResNet model was conducted in a similar way to the two-dimensional convolutional model. The model was setup by using the ResNet50 architecture as a base. This model contained 25.6 million parameters with a depth of 107 layers. Although more complex version of the ResNet architecture were available with more parameters and better performance on the ImageNet database, ResNet50 was an optimal trade-off between sufficient model complexity to achieve good predictive performance, but not too much complexity which slowed down training and was too resource intensive. For the base model, both a trainable version, meaning that the weights can be adjusted, and an untrainable version, where the weights are fixed, were tested with the untrainable version resulting in the best performance.

A custom output was appended to the base model, with a global average pooling layer feeding into a dense layer of 128 neurons which connected to an output layer of one neuron and the sigmoid activation function.

As with the two-dimensional convolutional model, binary cross-entropy was used as a loss function with the Adam optimizer. However, during development of this model, it became apparent that the default Adam learning rate of 0.001 was too large for the model to be able to learn from the training data. To counteract this problem, the initial learning rate was set to be 0.0001 with an exponential decay reducing the learning rate as training progressed. This allowed for the model to make smaller adjustments when learning and thus could learn from the provided data. Testing this method of dynamically decreasing learning rate on the initial two-dimensional convolutional model dramatically improved predictive performance and thus a decaying learning rate was used for all model training.

The same batch size and call-backs were used as with the two-dimensional convolutional model, with training time averaging around 11 and a half minutes across the four models.

3.4.3 Three-Dimensional Model

The three-dimensional model took inspiration from the keras guide for CT scan classification [53], using a similar architecture and structure. Here three-dimensional convolutional layers are alternated with three-dimensional max pooling layers followed by batch normalisation, to normalise the input data between layers. In each layer the number of neurons in the convolutional layer doubles. Four sections constructed like this are followed by a three-dimensional global average pooling layer and a dense layer. This model also contains a 30 percent dropout layer, where 30 percent of the inputs are randomly set to zero with the aim of reducing overfitting. A random search suggested that a 30 percent dropout was optimal. The model concludes with a sigmoid activated output layer.

Despite using early-stopping call-backs, a dropout layer and a significant amount of training data it was noticeable how quickly this model began to overfit. For example, with the FLAIR data, the model achieved an AUC on the training data of over 0.9 after only seven epochs, while correspondingly the validation AUC was concerningly lower at 0.43. Despite all efforts, the complexity of the model appeared to cause it to struggle to generalise on unseen data, and despite performing better on the other MRI types, overfitting was clearly an issue which was difficult to address.

3.4.4 Other Models

Throughout the course of development, different model types were explored with varying results. Variations of EfficientNet and MobileNet were developed and both models initially appeared unsuited to the task requiring exceptionally low learning rates in order to gain any insight from the training data. Furthermore, some of the larger versions of ResNet and EfficientNet failed entirely, as was consistent with other competition entrants [54]. When learning did occur, both models consistently scored marginally beneath the developed ResNet model. Therefore, it was not deemed necessary or useful to persistent in working with these model types.

An interesting model type explored was to extrapolate the architecture of the ResNet model to work with three dimensional convolutions and input data, thus benefitting from the published ResNet architecture and being able to consider patient’s data as a homogenous entity. Other similar approaches were used by participants in the BRaTS competition. Unfortunately, the complexity of this model type was prohibitively expensive in terms of GPU allocation for training. While some of the scores at the higher echelons of the leaderboard utilised a three-dimensional transfer learning approach, this was certainly not universal. In addition, as discussed in the results and discussion sections of this paper, the scores achieved by all participants in the competition were extremely stochastic, ergo it was deemed unnecessary to persist with this development if it would not significantly improve on the other models developed nor guarantee a high competition score or clinical utility. Instead, resources were better spent on developing the web application which would demonstrate how clinicians would be able to use the trained models and data pre-processing techniques to classify a patient’s mpMRI scan.

3.5 Application Development

The purpose of the web application was to allow the user, considered to be a clinician, to upload a patient’s mpMRI scans and select which data pre-processing options to use. The user can then select one of the three trained models to evaluate for the presence of MGMT methylation promoter in the tumour. Thus, the application provides the clinician with a comprehensive method for analysing the scans. For this project, the application serves as a proof-of-concept, showing what could be developed for clinicians to access and use technical data science tools easily and efficiently with no coding or development knowledge.

The web application is build using the Streamlit Python library (<https://streamlit.io>). Streamlit provides a fast and robust framework for building data science applications, which takes control of all front-end development. Due to this, only two files are used to create the web application. The first file is the data processor, which contains a class for two-dimensional and three-dimensional data loading. The code is almost identical to that required to load data for training the model with minor adjustments to remove any labelling functionality needed in training.

The second file required is used to set up the application, make the prediction and display it to the user. This file contains three functions: a main function to create option boxes, dropdowns and markup for the use; a predict function which extracts all of the data from a zip file provided by the user, loads it using the data processor file and makes a prediction which is then displayed visually; a clean-up function which removes all temporary files and prepares for a new prediction to be made. Here, the user can provide between one and four of the different types of MRI that models have been trained for. The weights for the desired models are then loaded from an Amazon Web Services SW3 bucket for the model to make a prediction. The predictions for each MRI type are then combined and averaged before being displayed to the user.

The application itself is hosted using Heroku. Heroku allows for the hosting of small apps with low levels of traffic in many languages for free. The service has a GitHub plugin allowing for quick and easy deployment. Heroku manages dependencies through a requirements.txt file and integrates with AWS SW3 for deployment of static files such as the model’s weights. The application is available at: <https://dissertation-hs.herokuapp.com>.

As the application works as a proof-of-concept testing of the functionality has been limited. The application functions well for known inputs, but there is no data or input sanitisation to ensure the data and selections provided by the user are compliant with the way the application works. This was beyond scope for the needs of the project at this time but would be required if the application was to be used more widely or deployed for clinical use.

## Chapter 4 – Results

This chapter contains a summary of the results each of the different model achieved in the BRaTS competition. These results are contrasted with the published achievements of others in the competition. Next, the resulting web application is presented with a brief summary of the functionality available.

4.1 Competition Results

4.1.1 Developed Models

Overall, all models tested and developed achieved underwhelming scores in the BRaTS competition, although not inconsistent with other competitors. Table 1 shows the output of all scores across the three models which were developed in their entirety.

|  |  |  |  |
| --- | --- | --- | --- |
| Model Type | MRI Type | Private Score | Public Score |
| ResNet | T1wCE | 0.50021 | 0.62737 |
| T1w | 0.50845 | 0.53329 |
| T2w | 0.51446 | 0.55761 |
| FLAIR | 0.55566 | 0.60835 |
| Combined | 0.543 | 0.64006 |
| Two-Dimensional Convolutional | T1wCE | 0.45028 | 0.54228 |
| T1w | 0.54631 | 0.54598 |
| T2w | 0.50055 | 0.61522 |
| FLAIR | 0.56595 | 0.59883 |
| Combined | 0.50259 | 0.60729 |
| Three-Dimensional Convolutional | T1wCE | 0.45843 | 0.55496 |
| T1w | 0.55741 | 0.49682 |
| T2w | 0.55294 | 0.54492 |
| FLAIR | 0.52269 | 0.57505 |
| Combined | 0.54835 | 0.57082 |

Table 1 – Model results for the BRaTS 2021 Competition. The public score is the AUC calculated using 22% of the test data, whereas the private score is calculated with the remaining 78%. This approach is used to order the Kaggle leaderboard based on the model which performs best on unseen data.

Each model was tested using each of the MRI types, before the predictions for each of the types were combined and averaged to create the combined score. The public score is calculated using 22% of the available test data, with the private score accounting for the remaining 78% of data. Therefore, analysis of the private score is best for examining predictive performance. Both private and public scores relate to the models AUC score on unseen test data.

The poor performance of all models is clearly of note. The three-dimensional convolutional model’s performance was marginally better than the ResNet, which itself was 0.04 better than the two-dimensional convolutional model. However, the models scores are only slightly greater than the 0.5 score which is the equivalent of random guessing. Furthermore, there is a level of stochasticity in the models’ predictions and re-running training or scoring on the test data again could have resulted in a shakeup of model scores. This was observed in the development process, where both the ResNet and three-dimensional convolutional models would alternate between achieving the best score. They were, however, consistently stronger than the simplest model.

From the results, it is clear that all 3 models performed poorest on the T1wCE MRI type, but it is difficult to extrapolate any solid conclusions regarding this as general performance was poor.

4.1.2 Competitors Models

Comparing the results of the models to fellow competitors revealed similarities. The best performing model of the competition was only able to achieve an AUC of 0.62 on the private dataset with the model’s author claiming that this was not due to sophistication of development but purely chance; if the model was run again the score may not be as high. This belief was consistent across the top performing models. Furthermore, where it is usually common in Kaggle competitions for top performing models on the public leaderboard to similarly be positioned high on the private leaderboard, this was not the case in the BRaTS competition. The winning models achieved a relatively poor score on the public leaderboard before the private one was revealed. The overwhelming conclusion from all models developed in this paper and the participants at the higher end of the leaderboard was that the models were struggling to gain any insight from the training data and could not improve significantly on a random guessing approach and thus could not provide clinical utility.

To expand on this, it is worth noting that none of the individual patient’s predictions in the winning submission provided a greater than 58 percent or less than 43 percent probability of the biomarker being present. The individual predictions of this model were not anywhere near to what would be required for use in clinical practise.

4.2 Application Results

While the results of the developed models could be viewed as disappointing, the application’s development brough much greater benefit. Figure 7 shows a screenshot of the design of the web application.

Graphical user interface, application, Teams

Description automatically generated

Figure 7 – Screenshot of the developed web application after a classification has been made.

The web application contains a panel for the user to select which model and data pre-processing options they would like to use when analysing a patient’s mpMRI scans. This includes where to apply the VOI LUT transformation, remove the surrounding black border from a patients scan and drop the blank images. There is also a slider to set the percentage of pixels which cannot be blank for the image to be used in prediction. The default value for this is 20 percent, as this was the level used to train the models.

In the main window of the application, there is an input file for a clinician to upload a zip file containing a nested structure of mpMRI files. As the purpose here is to serve as a proof-of-concept application, the required structure of the uploaded file is rigid, and there is no input sanitisation or checks to ensure the correct formatting. Information on the required formatting, as well as all code used in training the models and development of the application is available at: <https://github.com/smithharryh/Masters-Dissertation> in the project’s “README”.

Once the clinician has uploaded the patient’s files and clicked predict, a progress bar shows that the prediction is in progress. The speed of the prediction is variable as there are external variables such as internet speed used to load the weights from the AWS SW3 bucket, the varying complexities of the models and the number of MRI types given by the user. Typically, prediction for a standard data pre-processing configuration and the ResNet model takes approximately 45 seconds for all four MRI types.

Once prediction is complete, the user is provided with a probability for the presence of the MGMT methylation promoter biomarker in the patient’s scans, rounded to two decimal places, while in the background the extracted MRI files are deleted from the server. This was required to keep server costs as low as possible, thus multiple predictions would require the user to repeatedly upload the file.

## Chapter 5 – Discussion and Conclusion

This chapter aims to analyse and review the results achieved in development of both the deep learning models and the web application and consider the context of classifying the MGMT methylation promoter biomarker more generally. Furthermore, this chapter will consider what future developments could be made which might prove useful to the field before concluding with a summary of the project’s findings.

5.1 Discussion

5.1.1 Strengths and Limitations

As presented, all results realised both in this project and in the competition were significantly below the level required for the development of a classification system that could be used clinically. Following the closure of the competition, the organisers attempted to spell out why this may be the case [55]. One suggestion provided is an explanation that the reserved, private, test dataset included patients’ data from organisations which were not represented in the training set, so generalisation may have been scuppered by this. However, this does not explain why the validation AUC remained consistently low throughout development, so while this may have been a contributing factor, it is certainly not the sole reason for universally low competition scores.

A more likely reasons for the surprising results was that the competition itself was a leap into the unknown. The BRaTS competition had never been run using MRI data, let alone the use of more complex mpMRI. Most of the data was not publicly available prior to the challenge running and thus had not been suitably assessed by clinicians for its viability in the challenge.

Despite all this, perhaps the most illuminating reason for the low AUC scores by competitors was the enormity of the challenge faced. The predicted feature to be identified in the challenge was not humanly visible, another first for the BRaTS competition. Knowledge of the biomarkers presence in the scan came from molecular analysis of biopsy specimens, not human clinical assessment. The competition itself was driven by discovering whether using advanced deep learning techniques, an area that has grown so significantly in recent years, could detect the presence of a biomarker in an image that a human could not. Unfortunately, with current tooling, techniques, and research, this is not the case. However, despite this underwhelming conclusion, when considering the development undertook in this project for the model construction and application development, there is reason for optimism.

Firstly, while it is important in research to discover what it is possible, it is equally important to understand the challenges and obstacles which pave the path to discovery. The literature review of this report cited some sceptical literature suggesting that the classification of MGMT methylation promoter given MRI scans was possible, yet the results of this challenge crucially show that “there is much work to do to assess whether we can use medical imaging to reliably forecast genomic features of cancer” [56]. Additionally, this report has identified the need, and areas, for further research into where the discrepancies are in the competition and other published research on the matter.

Secondly, the techniques developed for this project and the competition as a whole are transferable to other domains. The technology to classify mpMRIs used in the models is robust, and some of the image pre-processing techniques and data pipelines are both novel and improvements on existing methodology. While biomarker identification may be beyond the scope of what is possible, the classifier that has been created may transfer successfully to a similar yet distinct domain.

Finally, the development and publication of the web application acts as a successful proof of concept for abstracting the technical methodology of data pre-processing and model training, allowing a non-technical clinician to use technical tooling and methodology to come to informed judgements on patient imaging.

5.1.2 Future Development

While this projecte acted as a comprehensive examination of the techniques available for the classification of biomarkers in mpMRIs, there is still scope for future development and research. As aforementioned, it is important to research the disparity in the competition’s poor results and that of other published literature where the results were much more optimistic. This may lead onto a breakthrough of an as yet untried method or could find flaws in the conducted research. Either way, it would be beneficial to develop this understanding to direct the research in this area.

Technically, there are a number of additions which could prove useful if adapting the program created for use in a different mpMRI based research domain. First, using a weighted average to combine the predictions may improve the accuracy of the classification model. While in the present iteration there is no bias when averaging the models, it may be beneficial to give more credence to the model with the greatest test accuracy. This is unlikely to improve the score achieved in the BRaTS competition significantly, but if the model were applied to a different use case this could make an incremental difference.

The model could also be trained on a greater range of data. While it was not deemed beneficial to apply overly extensive data transformations in this case to synthetically generate more data, where there is more evidence that a successful model could be developed it may be worth, for example, transforming each MRI type into the sagittal, coronal, and axial planes. Furthermore, as discussed, one reason why the competition results were not successful was that the test data came from organisations which were not represented in the training data. Understanding the difference in the two datasets as well as using a broader sample of MRI types may improve the performance in another domain.

It is worth observing that there is no evidence that the technical additions noted here would improve the score in the BRaTS competition and this report is comprehensive in showing that the biomarker classification problem is beyond the scope of currently available deep learning techniques. However, these additions could improve the classification accuracy if the model was applied to a more robust domain with greater potential for positive results.

Finally, it would be beneficial to work in conjunction with physicians in the development of the web application. The proof-of-concept application developed here serves its purpose, but if this were to be developed further it would be useful to understand the needs of clinicians themselves and where they see value in the product.

5.2 Conclusion

This project and report have provided a comprehensive approach at using advanced deep learning techniques and innovative data processing methods to assess mpMRIs for the presence of MGMT methylation promoter. In line with the results found in the BRaTS competition of 2021, the accuracy of the models developed was not of a high enough standard to be considered for clinical use in the near future. The all-encompassing assessment presented in this report demonstrated why this is the case and where future research in this area could be pointed to better comprehend the differences in speculative published literature and achieved results.

Furthermore, the deployment of the web application shows how technical deep learning research can be housed in an informative user interface for clinicians to be able to work with different models and data pre-processing techniques. Further development and testing in this area would be beneficial as outlined in this report, but the concept is sound and could be of benefit for future clinical practise.

Overall, this research acts as a comprehensive assessment of the techniques available for mpMRI classification and this report explains why there are flaws in the BRaTS competition of 2021. The project also offers a template for integrating deep learning classification models into a user facing application targeted at clinicians.

## References

[1] Ohgaki, H., & Kleihues, P. (2007). Genetic Pathways to Primary and Secondary Glioblastoma. The American Journal of Pathology, 170(5), 1445–1453. <https://doi.org/10.2353/ajpath.2007.070011>

[2] Thakkar, J. P., Dolecek, T. A., Horbinski, C., Ostrom, Q. T., Lightner, D. D., Barnholtz-Sloan, J. S., & Villano, J. L. (2014). Epidemiologic and Molecular Prognostic Review of Glioblastoma. Cancer Epidemiology Biomarkers & Prevention, 23(10), 1985–1996. <https://doi.org/10.1158/1055-9965.epi-14-0275>

[3] Ferlay, J., Soerjomataram, I., Dikshit, R., Eser, S., Mathers, C., Rebelo, M., Parkin, D. M., Forman, D., & Bray, F. (2014). Cancer incidence and mortality worldwide: Sources, methods and major patterns in GLOBOCAN 2012. International Journal of Cancer, 136(5), E359–E386. <https://doi.org/10.1002/ijc.29210>

[4, 5, 9] Shergalis, A., Bankhead, A., Luesakul, U., Muangsin, N., & Neamati, N. (2018). Current Challenges and Opportunities in Treating Glioblastoma. Pharmacological Reviews, 70(3), 412–445. <https://doi.org/10.1124/pr.117.014944>

[6] Wirsching, H. G., & Weller, M. (2016). Glioblastoma. Malignant Brain Tumors, 265–288. <https://doi.org/10.1007/978-3-319-49864-5_18>

[7] Chua, J., Nafziger, E., & Leung, D. (2019). Evidence-Based Practice: Temozolomide Beyond Glioblastoma. Current Oncology Reports, 21(4). <https://doi.org/10.1007/s11912-019-0783-5>

[8] Newlands, E., Stevens, M., Wedge, S., Wheelhouse, R., & Brock, C. (1997). Temozolomide: a review of its discovery, chemical properties, pre-clinical development and clinical trials. Cancer Treatment Reviews, 23(1), 35–61. <https://doi.org/10.1016/s0305-7372(97)90019-0>

[10] Strimbu, K., & Tavel, J. A. (2010). What are biomarkers? Current Opinion in HIV and AIDS, 5(6), 463–466. <https://doi.org/10.1097/coh.0b013e32833ed177>

[11] Hulka, B. S., & Wilcosky, T. (1988). Biological Markers in Epidemiologic Research. Archives of Environmental Health: An International Journal, 43(2), 83–89. <https://doi.org/10.1080/00039896.1988.9935831>

[12] de Vries, B., Haan, J., Frants, R. R., van den Maagdenberg, A. M., & Ferrari, M. D. (2006). Genetic Biomarkers for Migraine. Headache: The Journal of Head and Face Pain, 46(7), 1059–1068. <https://doi.org/10.1111/j.1526-4610.2006.00499.x>

[13] Weller, M., Stupp, R., Reifenberger, G., Brandes, A. A., van den Bent, M. J., Wick, W., & Hegi, M. E. (2009). MGMT promoter methylation in malignant gliomas: ready for personalized medicine? Nature Reviews Neurology, 6(1), 39–51. <https://doi.org/10.1038/nrneurol.2009.197>

[14] Hegi, M. E., Diserens, A. C., Gorlia, T., Hamou, M. F., de Tribolet, N., Weller, M., Kros, J. M., Hainfellner, J. A., Mason, W., Mariani, L., Bromberg, J. E., Hau, P., Mirimanoff, R. O., Cairncross, J. G., Janzer, R. C., & Stupp, R. (2005). MGMT Gene Silencing and Benefit from Temozolomide in Glioblastoma. New England Journal of Medicine, 352(10), 997–1003. <https://doi.org/10.1056/nejmoa043331>

[15] Diehn, M., Nardini, C., Wang, D. S., McGovern, S., Jayaraman, M., Liang, Y., Aldape, K., Cha, S., & Kuo, M. D. (2008). Identification of noninvasive imaging surrogates for brain tumor gene-expression modules. Proceedings of the National Academy of Sciences, 105(13), 5213–5218. <https://doi.org/10.1073/pnas.0801279105>

[16] Demirel, H. C., & Davis, J. W. (2018). Multiparametric magnetic resonance imaging: Overview of the technique, clinical applications in prostate biopsy and future directions. Türk Üroloji Dergisi/Turkish Journal of Urology, 44(2), 93–102. <https://doi.org/10.5152/tud.2018.56056>

[17] Thakur, S. P., Doshi, J., Pati, S., Ha, S. M., Sako, C., Talbar, S., Kulkarni, U., Davatzikos, C., Erus, G., & Bakas, S. (2020). Skull-Stripping of Glioblastoma MRI Scans Using 3D Deep Learning. Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries, 57–68. <https://doi.org/10.1007/978-3-030-46640-4_6>

[18] Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Mohamed, N. A., & Arshad, H. (2018). State-of-the-art in artificial neural network applications: A survey. Heliyon, 4(11), e00938. <https://doi.org/10.1016/j.heliyon.2018.e00938>

[19] Nielsen, M. A. (2015). Neural networks and deep learning. <http://neuralnetworksanddeeplearning.com>

[20] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics, 5(4), 115–133. <https://doi.org/10.1007/bf02478259>

[21] Stegemann, J. A., & Buenfeld, N. R. (1999). A Glossary of Basic Neural Network Terminology for Regression Problems. Neural Computing & Applications, 8(4), 290–296. <https://doi.org/10.1007/s005210050034>

[22] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning (Adaptive Computation and Machine Learning series) (Illustrated ed.). The MIT Press.

[23] Wang, S. C. (2003). Artificial Neural Network. Interdisciplinary Computing in Java Programming, 81–100. <https://doi.org/10.1007/978-1-4615-0377-4_5>

[24, 27] O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458.

[25, 31] Géron, A. (2022). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems (3rd ed.). O’Reilly Media.

[26, 52] Chollet, F. (2021). Deep Learning with Python, Second Edition (2nd ed.). Manning.

[28] Sun, M., Song, Z., Jiang, X., Pan, J., & Pang, Y. (2017). Learning Pooling for Convolutional Neural Network. Neurocomputing, 224, 96–104. <https://doi.org/10.1016/j.neucom.2016.10.049>

[29] Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. Insights into Imaging, 9(4), 611–629. <https://doi.org/10.1007/s13244-018-0639-9>

[30] Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278–2324. <https://doi.org/10.1109/5.726791>

[32] Mack, C. A. (2011). Fifty Years of Moore’s Law. IEEE Transactions on Semiconductor Manufacturing, 24(2), 202–207. <https://doi.org/10.1109/tsm.2010.2096437>

[33] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84–90. <https://doi.org/10.1145/3065386>

[34] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9). <https://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Szegedy_Going_Deeper_With_2015_CVPR_paper.html>

[35] DeepAI. (2020, June 25). Inception Module. <https://deepai.org/machine-learning-glossary-and-terms/inception-module>

[36] Lin, M., Chen, Q., & Yan, S. (2013). Network in network. arXiv preprint arXiv:1312.4400. <https://arxiv.org/pdf/1312.4400.pdf>

[37] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778). <https://arxiv.org/abs/1512.03385>

[38] Ide, H., & Kurita, T. (2017). Improvement of learning for CNN with ReLU activation by sparse regularization. 2017 International Joint Conference on Neural Networks (IJCNN). <https://doi.org/10.1109/ijcnn.2017.7966185>

[39] Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1251-1258). <https://arxiv.org/abs/1610.02357>

[40] Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7132-7141). <https://arxiv.org/abs/1709.01507>

[41] Zhang, X., Zhou, X., Lin, M., & Sun, J. (2018). Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 6848-6856). <https://arxiv.org/abs/1707.01083>

[42] Tan, M. &amp; Le, Q.. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning, in Proceedings of Machine Learning Research 97:6105-6114 <https://proceedings.mlr.press/v97/tan19a.html>.

[43] Gong, J., Liu, W., Pei, M., Wu, C., & Guo, L. (2022). ResNet10: A lightweight residual network for remote sensing image classification. 2022 14th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA). <https://doi.org/10.1109/icmtma54903.2022.00197>

[44,46] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2020). UNet++: Redesigning Skip Connections to Exploit Multiscale Features in Image Segmentation. IEEE Transactions on Medical Imaging, 39(6), 1856–1867. <https://doi.org/10.1109/tmi.2019.2959609>

[45] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. Lecture Notes in Computer Science, 234–241. <https://doi.org/10.1007/978-3-319-24574-4_28>

[47] Noble, W. S. (2006). What is a support vector machine? Nature Biotechnology, 24(12), 1565–1567. <https://doi.org/10.1038/nbt1206-1565>

[48] Chang, P., Grinband, J., Weinberg, B., Bardis, M., Khy, M., Cadena, G., Su, M. Y., Cha, S., Filippi, C., Bota, D., Baldi, P., Poisson, L., Jain, R., & Chow, D. (2018). Deep-Learning Convolutional Neural Networks Accurately Classify Genetic Mutations in Gliomas. American Journal of Neuroradiology, 39(7), 1201–1207. <https://doi.org/10.3174/ajnr.a5667>

[49] Korfiatis, P., Kline, T. L., Lachance, D. H., Parney, I. F., Buckner, J. C., & Erickson, B. J. (2017). Residual Deep Convolutional Neural Network Predicts MGMT Methylation Status. Journal of Digital Imaging, 30(5), 622–628. <https://doi.org/10.1007/s10278-017-0009-z>

[50] Han, L., & Kamdar, M. R. (2017). MRI to MGMT: predicting methylation status in glioblastoma patients using convolutional recurrent neural networks. Biocomputing 2018. <https://doi.org/10.1142/9789813235533_0031>

[51] YU4U. (2021, August 14). Normalized Voxels: Align Planes and Crop. Kaggle. <https://www.kaggle.com/code/ren4yu/normalized-voxels-align-planes-and-crop>

[53] Zunair, H. (2020, September 23). Keras documentation: 3D image classification from CT scans. Keras. <https://keras.io/examples/vision/3D_image_classification/>

[54] Baba, F. (2021). RSNA-MICCAI Brain Tumor Radiogenomic Classification | Kaggle. Kaggle. <https://www.kaggle.com/competitions/rsna-miccai-brain-tumor-radiogenomic-classification/discussion/281347>

[55, 56] Mongan, J. (2021). RSNA-MICCAI Brain Tumor Radiogenomic Classification | Kaggle. Kaggle. <https://www.kaggle.com/c/rsna-miccai-brain-tumor-radiogenomic-classification/discussion/284024>