Healthcare Workflows and Machine Learning

Luke Edgecombe
Robotics with Industrial Application
Heriot-Watt University
Edinburgh, Scotland
Email: le4005@hw.ac.uk

Chinyere Ihuoma Uwa
Applied Cybersecurity
Heriot-Watt University
Edinburgh, Scotland
Email: ciu4000@hw.ac.uk

Jahnavi Makaraju
Applied Cybersecurity
Heriot-Watt University
Edinburgh, Scotland
Email: jm4042@hw.ac.uk

Sarjjana Venkataramana
Applied Cybersecurity
Heriot-Watt University
Edinburgh, Scotland
Email: sv4016@hw.ac.uk

***Abstract*—some text**
***Keywords*—Medicine, Machine learning, Classification, Optimisation, Image analysis**

I. Introduction

Some text introducing the concepts - classifiers - image classifiers - convolutional NN - model refinement

In this report, a selection of differing machine learning (ML) techniques are explored. Each technique was chosen based on its area of application. The subject area of medicine was chosen for exploration, and a selection of datasets are explored with different features and formats: (i) a collection of brain magnetic resonance imaging (MRI) images that exhibit three cancer-based pathologies; (ii) a tabular dataset containing patient demographic and healthcare-related information related to stroke pathology; (iii) ?.

Stroke is a huge problem to global health; it remains one of the most common causes of death and disability [1]. The use of ML on stroke datasets is becoming a common practice because it makes it much easier for doctors to spot risks sooner and make better choices on the best treatment option for the patient [2]. The goal was to test different ML models to see which ones are best at classifying data and how unbalanced data impacts accurate predictions.

To approach the ML problems, an understanding of previous literature on the topic is required for an effective workflow. Academic papers were reviewed to gain an effective skill set and background knowledge required to begin experimentation.

II. Literature

ML as a concept is the basis of the field of artificial intelligence (AI); the primary purpose of ML is the generation of algorithms that are able to learn. In this regard, research has focused on the optimisation of these systems by generation of new techniques and identifying the most optimal combinations of pre-existing software/hardware [3].

ML is a popular tool in stroke research and its used to help spot problems earlier and assess patients more accurately. Large-scale research clearly shows that age, blood pressure, heart conditions, and blood pressure all play a role in how stroke affects a patient, hence the use of predictive modelling is needed to identify issues and intervene as early as possible [1]. Traditional statistics methods, like logistic regression are popular in healthcare because they're easy to interpret and identify factors that contribute to a patient's risk.

A. Logistic Regression

According to Aboong [4], logistic regression effectively points out the main stroke risk factors, proving it's a great tool for analysing organised medical datasets. Heo et.al [2] developed a machine learning model for predicting stroke outcomes, and their work showed how effective machine learning methods work better than the usual scoring tools. A big challenge, though is that stroke datasets usually suffer from a severe class imbalance and this can negatively affect a model's performance. Using techniques such as Synthetic Minority Over-sampling Technique (SMOTE) to balance the data has proven to improve the model's ability to find rare conditions in medical datasets [5]x.

The MRI image set relies on the classification ML technique; the specifics of this are introduced here.

B. Image Classifiers

This selection was reinforced by past research on brain cancer MRIs [6]. In particular, a convolutional neural network (CNN) was deployed. This ML technique has been shown to be highly effective in image classification [7]. A review into CNNs describes the typical architecture and construction techniques [8]. As the task differs, the design of the CNN can change, although the general format remains essentially the same. This typical structure, shown in Fig. 1, is comprised of ordered convolutional and pooling layers that come together to make a feature extractor. The inputted data is converted into a feature representation. In combination with this, fully-connected neural layers are integrated with activation functions to perform the desired ML operation.

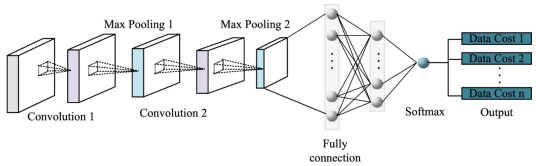


Fig. 1: Convolutional Neural Network Architecture [8]

Prior to data being used in ML, in order to increase suitability and feasibility of the model training, it should receive some sort of pre-processing. In order to effectively train a model, the data should be affected in a way that benefits the robustness and accuracy of the training process [9].

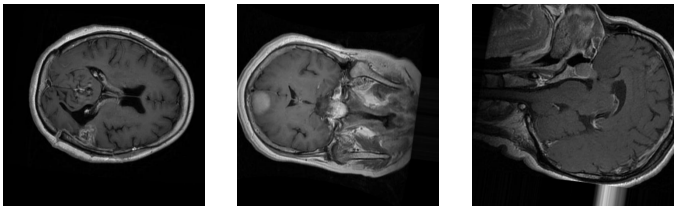
1) Pre-Processing: In image classification, effective pre-processing has been shown to increase the rate of identification [10]. For very large datasets, where manual review is not possible, automatic pre-processing is a must. Studies show that image cropping and reshaping, image resizing, and background noise removal are beneficial to the training process.

III. Data Analysis and Exploration

Based on the literature discussed previously, the data used in the following scenarios has undergone careful analysis in order to understand and prepare for upcoming ML. The MRI classifying dataset was sourced from a comprehensive collection of MRI images collected from a series of hospitals in Bangladesh [11]. The data holds great value and significant effort was made to collect and label high quality images.

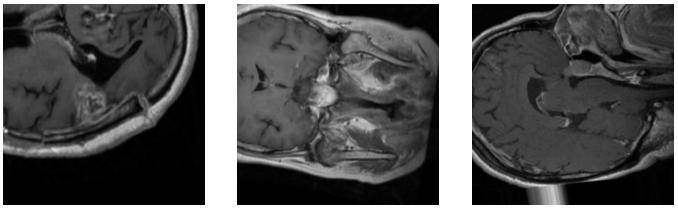
A. Brain Cancer MRI's

A sample of the three classes of data present is shown in fig. 2, pictures (a to b). The proprietor has already uniformly resized the images to an equal length and height. The ML for this set uses the hold-out method, 70% of the data is used for training, with the remaining 30% used equally between validaiton and training. To avoid bias or overfitting the data is shuffled using an scikit-learn cross-validation function that returns stratified randomised folds [12]. For the three sets some further ML specific pre-processing steps are executed, with training having more:



(a) Glioma (b) Meningioma (c) Tumor

Fig. 2: Sample images from MRI scans



(a) Glioma (b) Meningioma (c) Tumor

Fig. 3: Preprocessed sample images from MRI scans

- Random resized crop to 224×224 pixels
- Random horizontal flip with 50% probability
- Conversion to 3-channel greyscale
- Conversion to tensor format
- Normalisation to floating-point values in range [0,1]
- Standardisation using ImageNet mean and standard deviation

Additionally, the stroke dataset presented some inherent issues.

B. Stroke patient data

The stroke dataset used contains 5,110 patient records with demographic and clinical attributes such as age, heart disease, BMI, hypertension and average blood sugar level. The main variable (stroke) is binary. A small amount of missing BMI data was filled in by using the mean value.

A major challenge with this dataset is how unbalanced it is, with only about 3.4% of the cases actually having a stroke. The data exploration carried out indicated a higher chance of stroke among individuals within the age range of 50-80 years as shown in Fig. 4 and those with elevated glucose levels. When correlation analysis was carried out, it showed that age, hypertension and heart disease had subtle but meaningful relationship with stroke occurrences.

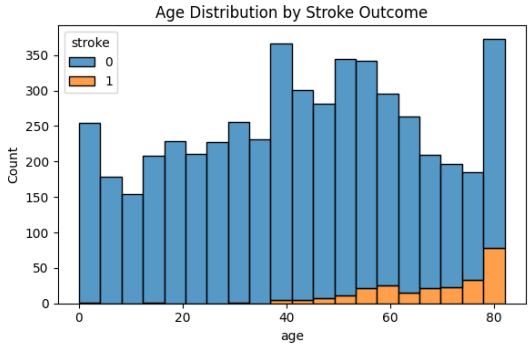


Fig. 4: Age Distribution by Stroke Outcome

To get the data ready for training, word-based features were converted to 0/1 columns so the models could process them. All numerical values were also rescaled to a similar range to ensure no feature dominated the other while the model was trained. To deal with the class imbalance problem, SMOTE was used to generate synthetic stroke cases, which helped in detecting rare instances of stroke. The preprocessing and exploration guided the modelling choices in later stages.

After the initial data-processing step has been completed for all the datasets, the ML algorithms can now be established.

IV. Experimental Setup

As the content of the datasets varies in format and shape, the experimental setup for each varies too.

A. Image classification training and evaluation

The MRI dataset was trained using a convolutional neural network (CNN) implemented as a customisable PyTorch model called `MRIClassifier`, based on the architecture shown in Fig.1. The architecture consists of the following components:

1) **Initialisation and Configuration:**
   - The model accepts two configurable parameters: `num_classes` (default 3) to specify the number of output classes, and `act_func` (default `nn.ReLU`)

to define the activation function used throughout the network.
- An internal activation factory function is implemented to instantiate the specified activation function, automatically handling the `inplace` parameter when supported by the activation function.

2) **Feature Extraction Blocks:**
The feature extraction component comprises four convolutional blocks arranged sequentially, each following the pattern: convolution → batch normalisation → activation → max pooling.
- *Block 1:* Processes the input 3-channel image (RGB or similar) using a `Conv2d` layer with 32 output channels, 3×3 kernel size, and padding of 1 to preserve spatial dimensions. This is followed by `BatchNorm2d` for normalisation, the configured activation function, and `MaxPool2d` with stride 2 to downsample by half.
- *Block 2:* Increases feature depth from 32 to 64 channels using a `Conv2d` layer with 3×3 kernel and padding 1, followed by batch normalisation, activation, and max pooling to further reduce spatial dimensions.
- *Block 3:* Expands feature maps from 64 to 128 channels through convolution with 3×3 kernel and padding 1, applying batch normalisation, activation, and max pooling.
- *Block 4:* The final convolutional block increases depth from 128 to 256 channels using a `Conv2d` layer with 3×3 kernel and padding 1, followed by batch normalisation, activation, and max pooling.

3) **Adaptive Global Pooling:**
- An `AdaptiveAvgPool2d` layer reduces the spatial dimensions of the feature maps to 1×1, regardless of input size. This ensures a fixed-size feature vector of 256 elements enters the classifier, making the network adaptable to varying input dimensions.

4) **Classification Head:**
- A dropout layer with probability 0.5 is applied for regularisation to prevent overfitting during training.
- A fully connected linear layer (`nn.Linear`) maps the 256-dimensional feature vector to `num_classes` output logits (3 by default), producing raw class scores.

5) **Forward Pass Pipeline:**
The forward propagation through the network follows this sequence:
a) Input images pass through the four convolutional feature extraction blocks.
b) The resulting feature maps are globally pooled using adaptive average pooling to 1×1 spatial dimensions.
c) The pooled features are flattened into a 1D vector of 256 elements.
d) The flattened vector passes through the classifier (dropout followed by linear layer) to produce class logits.
e) The output logits are returned for subsequent loss calculation or prediction.

This architecture progressively increases feature depth (3→32→64→128→256 channels) while reducing spatial dimensions through max pooling, allowing the network to learn hierarchical representations from low-level edges to high-level semantic features for MRI classification.

The design of the training loop is configured to utilise the defined model. The path of the data is visualised and displayed in simple flow charts for training and testing, shown in Appendix Fig. 8 and Fig. 9. Additionally, during training the performance of the model over each epoch was recorded. To provide extra insight into the training and broaden the scope, the model was run with varying hidden layer activation functions.

For the stroke dataset, after the data was cleaned, three primary machine learning models were used.

B. Stroke analysis training and evaluation

The selected models were:
- K-Nearest Neighbours (KNN)
- Logistic Regression
- Random Forest Classifier

The data was split into two parts for training and testing, 70% was used for training the models and the remaining 30% was set aside to test how well they performed. The performance of the models were evaluated using four main measurements; accuracy, precision, recall, and F1-score. After applying SMOTE, the models were retrained. To ensure a balanced evaluation of the stroke data and to compare with the classical models, a multi-layer perceptron (MLP) was used to train the balanced data.

V. Results

For the MRI set, the final results of the performance of the model over the testing set can be visualised in the form a confusion matrix, shown in Fig. 5. The classification report summarises the model's quantitative performance (see Table I).

TABLE I: Classification report for CNN brain tumor prediction (ReLU)

Class	Precision	Recall	F1-Score	Support
Brain Glioma	0.93	0.94	0.94	301
Brain Meningioma	0.77	0.91	0.84	301
Brain Tumor	0.94	0.76	0.84	307
Accuracy	0.87	0.87	0.87	0.87
Macro Avg	0.88	0.87	0.87	909
Weighted Avg	0.88	0.87	0.87	909

The result of the differing acitvation function performances are displayed in a line chart showing the running accuracy and loss of the training set and a table of final epoch performance. Shown in, Fig. 6 (a) and (b), and table II.

For the stroke prediction set, the following confusion matrix and classification report summarise the model performance on the test data.

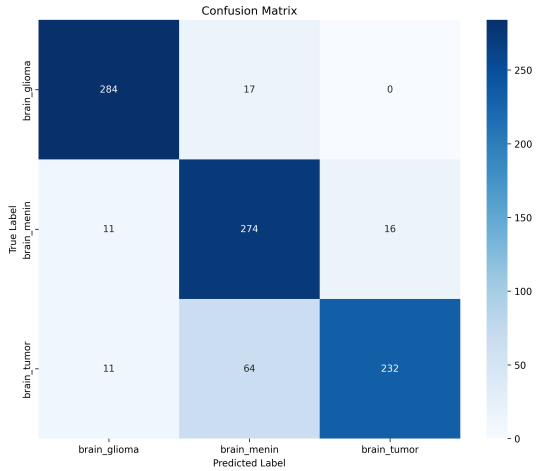


Fig. 5: Confusion matrix for brain cancer set (ReLU)

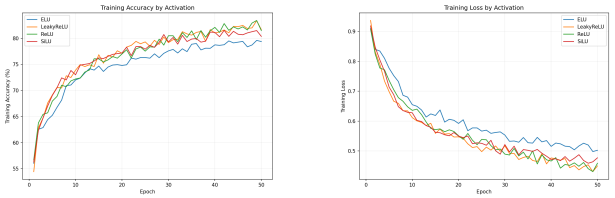
TABLE II: Final epoch comparison of activation functions for CNN MRI model

Activation	Train Loss	Val Loss	Train Acc%	Val Acc%
LReLU	0.4499	0.3561	81.67	86.47
SiLU	0.4770	0.3747	80.37	85.48
ReLU	0.4588	0.4156	81.48	84.71
ELU	0.5023	0.7485	79.38	63.70

VI. Discussion

The stroke prediction set revealed important insights into model performance. When the models were applied to the unbalanced data, the results showed high accuracy, but barely predicted any actual strokes (low recall).Which means they usually just guessed "no stroke" every time. Logistic Regression performed best overall as shown in Fig. 7 and table III, it has a strong recall score of 0.79 for the stroke cases while maintaining a good accuracy. This improvement shows how important it is to deal with unbalanced data when making predictions. The neural network achieved a high accuracy of 92% but struggled to identify actual stroke cases (low recall), which suggests it had difficulty in learning the patterns of the minority class.

The CNN model demonstrated strong performance in classifying brain tumor types. As shown in the confusion matrix (Fig. 5), the model correctly identified the majority of cases for each class, with particularly high accuracy for Brain Glioma (284/301) and Brain Meningioma (274/301). Misclassifications were relatively low, though the model tended to confuse Brain Tumor cases with Brain Meningioma (64 instances). Consistent with the classification report, the overall accuracy was 87%. An activation ablation further indicated that LeakyReLU offered the best generalisation (final validation accuracy approximately 86.5%), followed by SiLU (approximately 85.5%) and ReLU (approximately 84.7%), whereas ELU underperformed (approximately 63.7% validation accuracy) with a notably higher validation loss. We therefore adopted LeakyReLU in the final CNN, likely benefiting from its small negative slope which mitigates dying-ReLU effects without the instability observed with ELU.



(a) Combined training accuracies for MRI set (b) Combined training losses for MRI set

Fig. 6: MRI set training performance

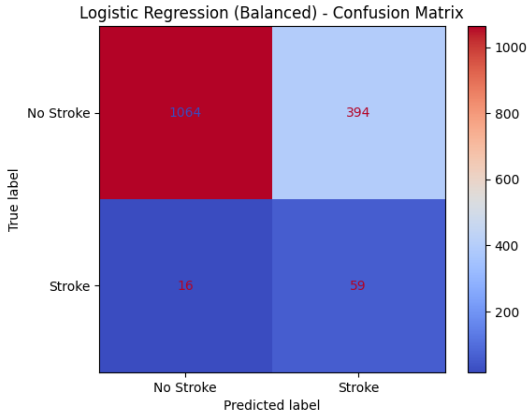


Fig. 7: Confusion matrix for logistic regression (SMOTE-balanced dataset)

References

[1] V. L. Feigin, B. A. Stark, C. O. Johnson, G. A. Roth, C. Bisignano, G. G. Abady, M. Abbasifard, M. Abbasi-Kangevari, F. Abd-Allah, V. Abedi, A. Abualhasan, N. M. Abu-Rmeileh, A. I. Abushouk, O. M. Adebayo, G. Agarwal, P. Agasthi, B. O. Ahinkorah, S. Ahmad, S. Ahmadi, and Y. Ahmed Salih, "Global, regional, and national burden of stroke and its risk factors, 1990–2019: a systematic analysis for the global burden of disease study 2019," *The Lancet Neurology*, vol. 20, no. 10, pp. 795–820, Sep 2021. [Online]. Available: https://www.thelancet.com/journals/laneur/article/PIIS1474-4422(21)00252-0/fulltext

[2] J. Heo, J. G. Yoon, H. Park, Y. D. Kim, H. S. Nam, and J. H. Heo, "Machine learning–based model for prediction of outcomes in acute stroke," *Stroke*, vol. 50, no. 5, pp. 1263–1265, May 2019. [Online]. Available: https://www.ahajournals.org/doi/10.1161/STROKEAHA.118.024293

[3] R. Abdulkadirov, P. Lyakhov, and N. Nagornov, "Survey of optimization algorithms in modern neural networks," *Mathematics*, vol. 11, no. 11, 2023. [Online]. Available: https://www.mdpi.com/2227-7390/11/11/2466

[4] M. Aboonq, "Potential role of logistic regression analysis to identify significant risk factors associated with stroke," *Bulletin of Egyptian Society for Physiological Sciences*, vol. 44, no. 1, pp. 17–28, Jan 2024.

[5] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, no. 16, pp. 321–357, Jun 2002.

[6] J. Kang, Z. Ullah, and J. Gwak, "Mri-based brain tumor classification using ensemble of deep features and machine learning classifiers," *Sensors*, vol. 21, no. 6, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/6/2222

[7] P. Wang, E. Fan, and P. Wang, "Comparative analysis of image classification algorithms based on traditional machine learning and deep learning," *Pattern Recognition Letters*, vol. 141, pp. 61–67, jan 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167865520302981

TABLE III: Classification report for stroke prediction

Class	Precision	Recall	F1-Score	Support
No Stroke	0.99	0.73	0.84	1458
Stroke	0.13	0.79	0.22	75
Accuracy	0.73	–	–	1533
Macro Avg	0.56	0.76	0.53	1533
Weighted Avg	0.94	0.73	0.81	1533

[8] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A review of convolutional neural networks in computer vision," *Artificial Intelligence Review*, vol. 57, no. 4, Mar 2024. [Online]. Available: https://link.springer.com/article/10.1007/s10462-024-10721-6

[9] M. Salvi, U. R. Acharya, F. Molinari, and K. M. Meiburger, "The impact of pre- and post-image processing techniques on deep learning frameworks: A comprehensive review for digital pathology image analysis," *Computers in Biology and Medicine*, vol. 128, p. 104129, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010482520304601

[10] P. M. Diop, J. Takamoto, Y. Nakamura, and M. Nakamura, "A machine learning approach to classification of okra," in *2020 35th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, 2020, pp. 254–257. [Online]. Available: https://ieeexplore-ieee-org.hwu-ezproxy.idm.oclc.org/document/9183312

[11] M. M. Rahman, "Brain cancer - mri dataset," *Mendeley Data*, vol. 1, Aug 2024. [Online]. Available: https://data.mendeley.com/datasets/mk56jw9rns/1

[12] scikit learn, "sklearn.model_selection.stratifiedshufflesplit." [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.StratifiedShuffleSplit.html

APPENDIX A
TRAINING AND TESTING LOOP DIAGRAMS

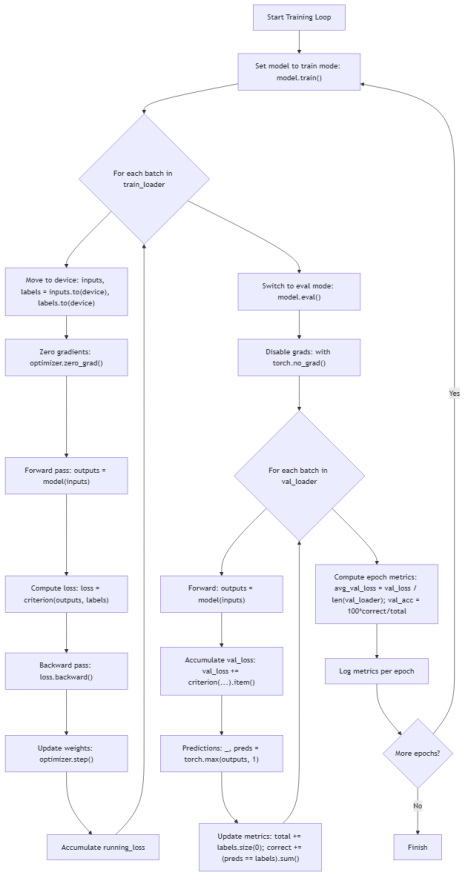


Fig. 8: Training Loop for MRI Classifier

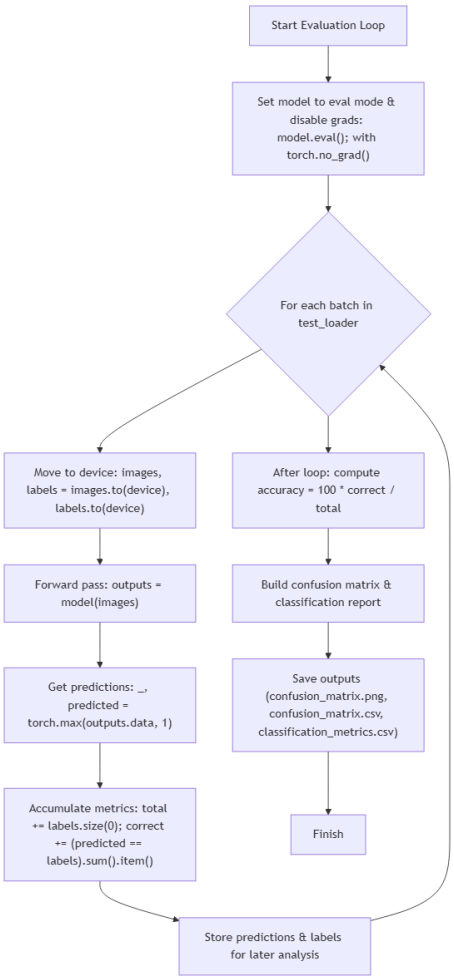


Fig. 9: Testing Loop for MRI Classifier