Array 26 (2025) 100398

Contents lists available at ScienceDirect

Array

journal homepage: www.sciencedirect.com/journal/array

Meta-learning based softmax average of convolutional neural networks using multi-layer perceptron for brain tumour classification

Irwan Budi Santoso *, Shoffin Nahwa Utama , Supriyono

Informatics Engineering, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim, Malang, Indonesia

ARTICLE INFO

Keywords:
Brain tumour
Magnetic resonance imaging
Convolutional neural network
Multi-layer perceptron
Softmax
Base-learner
Meta-learner

ABSTRACT

Brain tumour classification using Magnetic Resonance Imaging (MRI) is crucial for medical decision-making. The variability in tumour shape, size, and position poses challenges to classification methods. Convolutional Neural Networks (CNNs) are commonly used due to their proven performance, but their effectiveness diminishes with the high variability of tumour characteristics. This study proposes a meta-learning approach, leveraging the softmax average of multiple CNN models with a Multi-Layer Perceptron (MLP) as the meta-learner. The base-learner models include MobileNetV2, InceptionV3, Xception, DenseNet201, and ResNet50. This approach combines the softmax outputs of these CNN models, capturing their strengths to handle diverse tumour characteristics. The averaged outputs are fed into the MLP for increased classification performance. To evaluate the proposed method, we used several brain MRI image datasets, including Dataset 1 (Thomas Dubail Dataset), Dataset 2 (Mesoud Nickparcar Dataset), and Dataset 3 (Fernando Feltrin Dataset). The test results showed the proposed method's effectiveness in improving classification performance. For Dataset 1, the MLP with one hidden layer (128 neurons) achieved 97.47 % accuracy, improving the base learners' performance by 1.94 %–7.42 %. On Dataset 2, the MLP with 64 neurons reached 99.54 % accuracy, with a 0 %–2.44 % improvement. For Dataset 3, an MLP with two hidden layers (256 and 125 neurons) achieved 98.87 % accuracy, enhancing performance by 0.46 %–5.67 %.

1. Introduction

The presence of abnormal cells in the brain can cause the growth of tissue called brain tumours [1]. Glioma, Meningioma, and Pituitary are some types of brain tumours often found in brain tumour patients, and each requires different treatments ([53], [2]). Therefore, inaccuracy in classifying the type of brain tumour can lead to further medical errors and can cause patient death [3,4]. Medically, the classification of brain tumour types is carried out by radiologists by looking at the results of MRI brain scans [1]. MRI images can map the internal structure of the human brain and can provide the best spatial brain structure information[5,6]. However, the shape and size of the brain tumour often have similarities between one type of tumour and another. Hence, the classification of brain tumour types manually based on MRI images can potentially cause errors. The development of the current classification method is an alternative solution to help improve the performance of this brain tumours types classification.

Convolutional Neural Network (CNN) is a classifier often used to classify brain tumours because its performance is undoubted, as shown by Noreen et al. [7]. Khan et al. [3], and Musallam et al. [8]. On the other hand, several studies have used CNN for feature extraction of brain MRI images, as reported by Kang et al. [9], Ahmad & Choudhury [10], and Shanthi et al. [11]. There are data limitations and the convolution process ability in CNN, which has been proven to extract image features better. The feature extraction results are input for several conventional classifiers for tumour-type classification. This effort is to obtain or select a combination of classifier models with CNN extraction features that produce the best tumour classification performance among other combinations. This effort sometimes contributes to improving tumour classification performance. Still, it will be challenging to get the best results because of the feature extraction method, either with one or several separate CNN models with the classifier. The feature extraction results do not necessarily match the classifier method. What happens is that it is just a trial and error to combine several classifiers and choose the best results among them. Several researchers have also carried out pre-processing for brain MRI images to improve the performance of their proposed CNN, such as those by Rizwan et al. [12] and Musallam et al. [8]. Filters for denoising MRI images contribute to improving their

* Corresponding author.
E-mail address: irwan@ti.uin-malang.ac.id (I.B. Santoso).

https://doi.org/10.1016/j.array.2025.100398
Received 20 December 2024; Received in revised form 27 February 2025; Accepted 3 April 2025
Available online 4 April 2025

proposed CNN's performance. However, due to the high variation in tumour shape and size plus the complex human brain tissue, using only one CNN model will make it be challenging to get the best performance. To overcome the high variation in tumour shape and size, several previous researchers have combined models with different architectures [3, 13], combined convolution results from different models [14], and combined features generated from other blocks or modules in one model [7]. Khan et al. [3] combining the CNN architectures obtained better accuracy than before. The same study conducted by Younis et al. [13] also produced better performance than before when combined. Combining convolutions was carried out by Chatterjee et al. [14]. The combination of convolutions produced better performance than the previous model. Noreen et al. [7] combined features taken from different modules in one model as a solution to improve tumour classification performance. All of their proposed methods contributed to the performance of tumour classification. However, convolution, feature merging, or model architecture is the same principle used to obtain more representative features. On the other hand, the size of a classifier's feature representation or top layer classification can only be seen in real terms from the final classification results. Therefore, to get the best results, in addition to requiring the involvement of many CNN models, the classification output must also be seen, such as the resulting softmax value. An additional process, determining the average softmax value in the appropriate class, becomes input to the following learning process using other classifiers, such as Multi-Layer Perceptron (MLP), as the solution. This solution will overcome the high variation in the shape and size of the tumour by involving many CNN models and obtaining more representative features with further learning with specific classifiers.

Based on the potential and solutions obtained from previous studies, several CNN models must be involved to strengthen the performance of tumour classification with high variations in size, shape, and position [15,16]. These CNN models are, after this, referred to as base-learner models. Each CNN model in the base learner has an architecture with several convolution layers. The filter size of each convolution will overcome high variations in the size of the brain tumour shape. Each output layer (classification) of the base-learner CNN model, such as softmax, is the result of the learning process that measures how strong the representation of the convolution process results is. The average softmax value of each base-learner model in the class becomes the input of the meta-learner, such as MLP [17], to be retrained for the final classification of brain tumours. The MLP architecture can be specifically built to accommodate the input representation produced by the average softmax of each base-learner model. Therefore, in this study, we propose a softmax average meta-learning generated by CNN models (base--learner) using MLP to improve the performance of brain tumour classification. CNN models as base-learner models are CNN models whose performance is no longer in doubt, including MobileNetV2[18], InceptionV3[19], Xception[20], DensNet201[21], and ResNet50[22]. Furthermore, the main contributions in this study are as follows.

1. We formed a base-learner CNN model, including MobileNetV2, InceptionV3, Xception, DensNet201, and ResNet50, which have been adjusted on the top layer and transfer learning specifically for MRI image-based brain tumour classification.
2. We proposed a meta-learning scheme or framework based on the softmax average of the CNN base-learner model by involving MLP as a meta-learner.
3. We proposed an MLP architecture as a meta-learner to enhance the performance of the brain tumour classification based on the softmax average of the CNN base-learner model.

The organization of this paper then includes several Sections: Section 2 describes the methods and results of previous related studies, Section 3 describes the dataset used for testing, the steps of the proposed method, the experimental settings and hyperparameters of the model for testing, and the evaluation measures of the method used. Section 4 contains the results and discussion of the experiments. Finally, Section 5 contains the conclusions and suggestions for further research based on the results and discussion.

2. Related works

Previous studies have reported the focus of CNN model involvement in classifying brain tumours based on MRI images. The study focuses involving CNN models include, first, extracting MRI image features with conventional classifiers; second, the focus involving CNN models for feature extraction as well as classifiers, including architecture merging, convolution merging, and merging features from different modules, and third, the pre-processing focus to improve the performance of the proposed CNN model.

Kang et al. [9] proposed a scheme involving thirteen CNN models with transfer learning to extract MRI brain image features and nine conventional machine-learnings to evaluate features and perform the final classification. The evaluation results showed that the combination of feature extraction from ResNeXt101, DenseNet121, and MnasNet gave the best performance in classification. Ahmad & Choudhury [10] also conducted the same study using seven CNN models with transfer learning to extract MRI brain image features and using five conventional machine-learnings for classification. Their trials showed that the combination of VGG19 with SVM produced an accuracy of 99.39 % and was better than other combinations.

The involvement of CNN models in feature extraction of MRI brain images and classification has been developed in several previous studies. Chatterjee et al. [14] proposed ResNet(2 + 1)D, built by combining 2D and 1D convolution, and mixed ResNet, obtained by combining 2D and 3D convolution. Their experiments showed that both models performed better than ResNet3D with an accuracy of 96.98 %. Khan et al.[3] combined a 23-layer CNN model with VGG16 transfer learning to overcome overfitting on small datasets. Their evaluation showed that the combination they performed obtained an accuracy of up to 97.8 % on the big data and 100 % on the small data. Younis et al. [13] also combined CNN and VGG16 models and produced an accuracy of 98.14 %. Noreen et al. [7] performed two merges: merging features extracted from the Inceptionv3 module and features from the DensNet201 block. The combined features were then forwarded to softmax for tumour classification. The experiment results showed that the first merge produced an accuracy of 99.34 % and the second 99.51 %. A different study has been reported by Asif et al. [15], who used the transfer learning models Xception, DenseNet121, InceptionResNetV2, and NasNet Large, and at the classification stage, each model used a softmax layer. The experiment results showed that Xception yielded the best performance with an accuracy of 99.67 % on a large dataset and 91.94 % on a small dataset.

Several previous studies have also carried out that focused on pre-processing. Shanthi et al. [11] used a Gaussian filter to pre-process MRI brain images, which was continued using Long Short-Term Memory (LSTM) with the Adaptive Rider Optimizer (ARO) at the classification stage. The experiment results of their proposed method produced a maximum accuracy of 97.5 %. Rizwan et al. [12] proposed a Gaussian Convolutional Neural Network (GCNN) with a Gaussian filter to improve classification performance and obtained the best accuracy of 99.8 % on the first dataset and 97.14 % on the second dataset. Musallam et al. [8] proposed three pre-processing steps and a Deep Convolutional Neural Network (DCNN) to classify tumour types. The experimental results showed that their proposed method yielded an overall accuracy of 98.22 %.

3. Materials and methods

3.1. Dataset experiment

This study used three brain MRI image datasets to evaluate the

proposed method. The first dataset (THOMAS dataset) is a brain MRI image for brain tumour classification obtained from Kaggle [23]. The dataset has been refined from the SARTAJ dataset [24], which consists of brain MRI images of Meningioma, Glioma, and Pituitary tumours, and No Tumours. The total number of MRI images in the dataset was 3096, composed of 913 Meningioma images, 901 Glioma images, 844 Pituitary images, and 438 normal images (No Tumour). The second dataset (NICKPARVAR) is a brain MRI image from Kaggle [25]. The datasets, a combination of public datasets including Figshare [26], SARTAJ [24], and BR35H [27], were separated into datasets for training and testing. The dataset for training included 1321 MRI images of Glioma, 1339 Meningioma, 1457 Pituitary, and 1595 No Tumour. At the same time, the dataset for testing included MRI images of the brain of Meningioma, Glioma, Pituitary, and No Tumour with a total of 306, 300, 300, and 405, respectively. Examples of these brain MRI images can be seen in Fig. 1. The third dataset (FERNANDO dataset) is an MRI image of the brain obtained from Kaggle [28]. The dataset contained 4448 original images (jpeg = 3147, jpg = 1166, png = 102, other = 33) of magnetic resonance imaging (MRI) images of the skull, weighted in T1, T1 with contrast, and T2 sequences (T1, T1C+, and T2). The images included Meningioma (Low Grade, Anaplastic, Atypical, Transitional), Glioma (Astrocytoma, GanglioGlioma, OligodendroGlioma, Glioblastoma, Ependymoma), Neurocytoma (Central - Intraventricular, Extra-ventricular), NORMAL, other types of injuries (Cysts, Abscesses, Miscellaneous Encephalopathies), and Schwannoma (Acoustic, Vestibular - Trigeminal).

Each dataset was arranged into training, validation, and testing datasets in each evaluation of the CNN model and the proposed method. The first dataset was split with a composition of 10 % of the total dataset, namely 310 MRI images of the brain as the testing dataset. The remaining dataset was then split again with a composition of 10 %; 279 MRI images were the validation dataset, and the remaining 2507 MRI images of the brain are the dataset for training. In the second dataset, a testing dataset of 1311 brain MRI images and a training dataset of 5712 were allocated. Next, we split the training dataset with a composition of 10 % for validation. Thus, there were 572 MRI images of the brain for validation and 5140 MRI images for the model training process. At the same time, the third dataset included 3579 MRI images for training, 395 for validation, and 441 for testing. These datasets involved 17 classes/labels distinguished based on MRI sequences and tumour types. Splitting all of these datasets in detail can be seen in Table 1.

3.2. Pre-processing of brain MRI image

The input of the proposed method for brain tumour classification is MRI images of the brain. Fig. 1 is the examples of brain MRI images as input in the training process. In the training process, the MRI image data contain several types of brain tumour images and images of healthy people (No Tumour) (see Fig. 2). The next step is to pre-process the MRI images by resizing each image to have the same size. Resizing is done to meet the needs of the training and testing stages, which require the size of each image to be the same. In some scenarios, after this stage, scaling is performed on each pixel value of the MRI image to facilitate the training process of the base-learner CNN model. In the proposed framework or scheme, the size of the input MRI image (input shape) in each CNN model is 224 × 224 pixels, therefore, the size of the original MRI image must be changed to that size (see Fig. 3). In some of the test scenarios of this study, the MRI image pixel scaling process was carried out to change the pixel value from [0.255] to [0.1] by dividing each pixel by 255. This process was carried out to get stability and convergence in the network [29,30].

3.3. Base-learner of CNN model

This sub-chapter explains several CNN models selected as base-learner models, adjustment of CNN models on the top layer according to the case, learning mechanisms, and hyperparameters of each base-learner CNN model with output that will be forwarded to the meta-learning stage. This study's CNN models selected as base-learner models include MobileNetV2, InceptionV3, Xception, DensNet201, and ResNet50. Before the learning process, each CNN model on the top layer is adjusted to classify tumours according to the type of tumour or class in

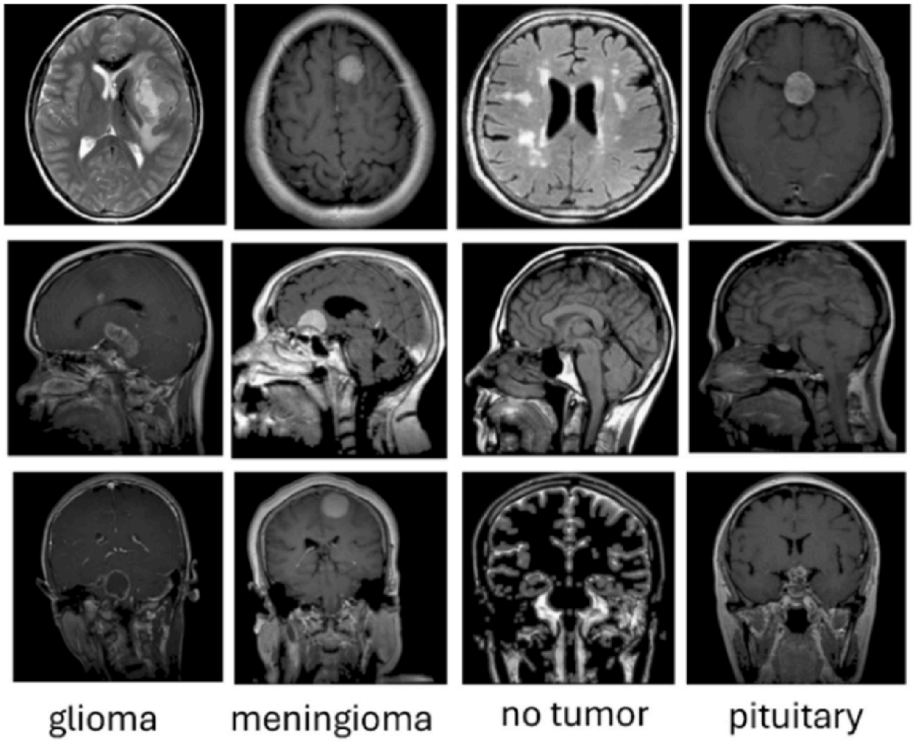

Fig. 1. Examples of brain MRI images include Glioma, Meningioma, No Tumour, and Pituitary.

Table 1
The composition of brain MRI images dataset for proposed methods evaluation.

Dataset	Labels		Training	Validation	Testing
Dataset 1 (THOMAS dataset)	0	Glioma	730	81	90
	1	Meningioma	740	82	91
	2	No Tumour	354	40	44
	3	Pituitary	683	76	85
		Total	2507	279	310
Dataset 2 (NICKPARVAR dataset)	0	Glioma	1189	132	300
	1	Meningioma	1205	134	306
	2	No Tumour	1435	160	405
	3	Pituitary	1311	146	300
		Total	5140	572	1311
Dataset 3 (FERNANDO dataset)	0	Glioma -T1	352	36	42
	1	Glioma -T1C+	412	45	51
	2	Glioma -T2	280	31	35
	3	Meningioma -T1	279	31	35
	4	Meningioma -T1C+	506	56	63
	5	Meningioma-T2	266	30	33
	6	Normal-T1	220	25	27
	7	Normal-T2	236	26	29
	8	Neurocytoma -T1	137	15	17
	9	Neurocytoma -T1C+	211	24	26
	10	Neurocytoma -T2	91	10	11
	11	Other Types -T1	123	14	15
	12	Other Types -T1C+	39	4	5
	13	Other Types -T2	46	5	6
	14	Schwannoma -T1	124	14	15
	15	Schwannoma -T1C+	157	18	19
	16	Schwannoma -T2	100	11	12
		Total	3579	395	441

the dataset used. The learning of each base-learner CNN model needs a hyperparameter to obtain good learning results. Table 2 shows an overview of the hyperparameter tuning of each base-learner CNN model, which includes customization on the top layer, input shape, epoch, learning rate, loss function, batch size, optimizer, and weight initialization. The input shape of each CNN model is $224 \times 224 \times 3$ and it is the recommendation of the CNN model in general. Top layer customization on each CNN base-learner model includes pooling, dense, and activation functions. Pooling on the top layer of MobileNetV2, InceptionV3, Xception, and DensNet201 uses Global Average Pooling (GAP), while ResNet50 uses Average Pooling (AP) and Flatten. The dense size is adjusted to the number of labels/classes in the dataset used namely 4 for Dataset 1 and 2 and 17 for Dataset 3. The activation function of all top layers uses softmax except for Xception, which uses sigmoid [31]. The use of GAP and activation functions for each base-learner model has proven robust and is recommended for the model [32–34]. Likewise, the use of the softmax is also recommended for the CNN model [35,36].

In this study, each base-learner model involves learning at epochs 50 and 100. The batch size and learning rate for each base-learner model learning are 16 and 0.001, and these sizes have shown the best performance results in previous studies [16,37,38]. The optimizers used in the learning process are Adam and Adamax, which have been proven to have more stable results [39,40]. The loss function in the learning process is categorical cross-entropy [2,32] because brain tumour classification involves multi-class. Meanwhile, the learning process for each base-learner model applies transfer-learning, namely the weight initialization is taken from the learning results of each base-learner CNN model on the ImageNet dataset [41]. As a result, it can help the learning of the base-learner model on small datasets [42].

MobileNetV2, as one of the base-learner models in this study, is one of the CNN architecture parts of MobileNet that uses depth-wise and pointwise convolution [18]. In MobileNetV2, two features are added: linear bottleneck and shortcut connections between bottlenecks. The total number of bottleneck layers in the model is 19 layers, with ReLU as the activation function used. The total parameters of the model in the training are 2,228,996 for the output layer with four-class denseness. MobileNetV2 in this study involves hyperparameters in Tables 2 and in its learning, the classification results are shown $\widetilde{h}_{1k}$, which states the softmax value in the kth class.

The next base-learner model is InceptionV3, a CNN architecture from the Inception family that has received several improvements [19]. In general, the InceptionV3 architecture has a steam block, inception block, reduction block, and auxiliary classifier block. The InceptionV3 model consists of 42 layers with smoothing labels. The total parameters of the model are 21,776,548, which are used for training. The number of parameters is generated in the output layer with four-class denseness. The InceptionV3 learning process in this study also involves hyperparameters in Table 2 with the sofmax value in the kth class shown by $\widetilde{h}_{2k}$.

Next is Xception, which has an architecture that continues the Inception architecture [20]. The Xception architecture has 36 convolutional layers consisting of 14 modules. Xception consists of 71 layers and has 20,815,148 training parameters for the output layer with a dense containing four classes. The Xception architecture has a convolution operation with a filter size of 1×1 as a 2D + 1D mapping. In this study, Xception involves a scenario with hyperparameters, as shown in Table 2. The classification results using Xception are shown by $\widetilde{h}_{3k}$, the sigmoid value in the kth class.

The following CNN architecture is DensNet201, which has layers to each subsequent layer connected using feedforward [21]. This architecture has as many as $(L \times (L + 1))/2$ connections, and each layer produces a feature map. The DenseNet201 architecture has four dense blocks, each containing 6, 12, 24, and 16 convolution blocks. For the output layer with a dense four classes, the model has 18,100,612 trained parameters. DenseNet201 in this study also applies a scenario with hyperparameters in Table 2 with the classification results shown by $\widetilde{h}_{4k}$, which is the softmax value in the kth class.

The next base-learner model is ResNet50, which has a CNN architecture that involves residual connections [22]. ResNet50 has a convolutional layer, an identity block, and a convolutional block. The model consists of 50 trained layers and includes five convolutional neural network blocks. The convolutional layer is used to extract features, so, the identity and convolutional blocks are used to transform features. The total number of model training parameters is 23,665,668, for the output layer with a dense containing four classes. In this study, ResNet50 also involves the hyperparameters in Table 2 in learning with the classification results shown in $\widetilde{h}_{5k}$.

Each CNN architecture has a convolution process with a different number of layers and filters. This condition provides an advantage in finding representative features of tumour types with variations, sizes, and uncertain positions [16]. On the other hand, the MobileNetV2 model is selected because it is the best among the MobileNet family with undoubted performance [18], as well as for InceptionV3, Xception, DensNet201, and ResNet50. These CNN models have been pre-trained with large data, namely 'ImageNet'. Through transfer learning, the results of previous learning can be a solution to overcome limited MRI image datasets and avoid overfitting [42]. Based on the total parameters for learning, the total parameters of these CNN models are still relatively small compared to VGG16 [37], and the computational complexity in learning must still be considered, despite the main goal being to improve classification performance.

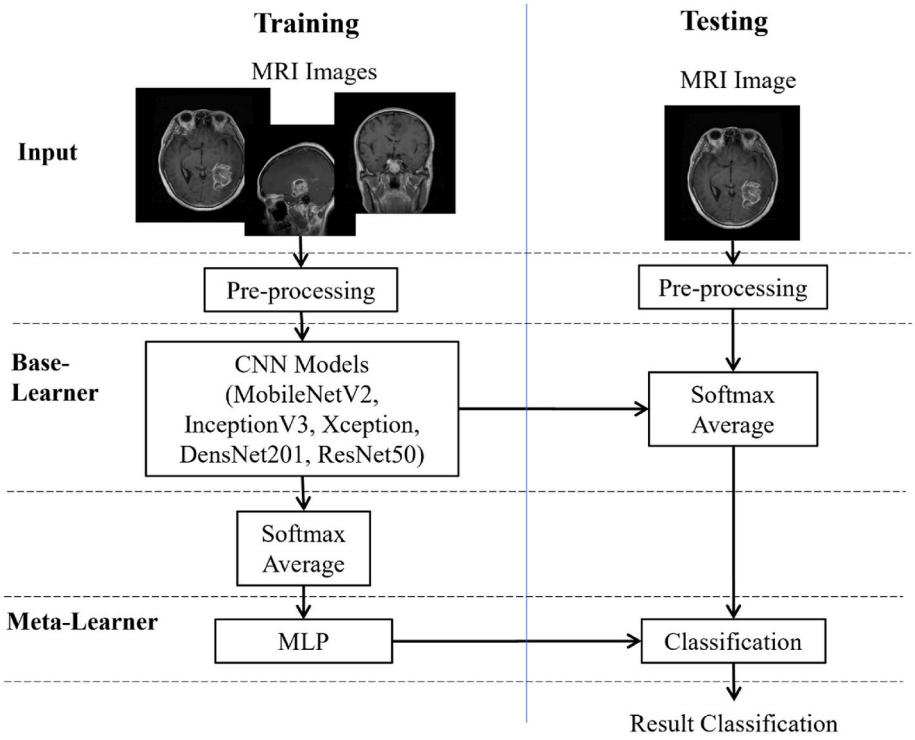

Fig. 2. Scheme of training and testing for tumour classification including base-learner CNN model and meta-learner of MLP.

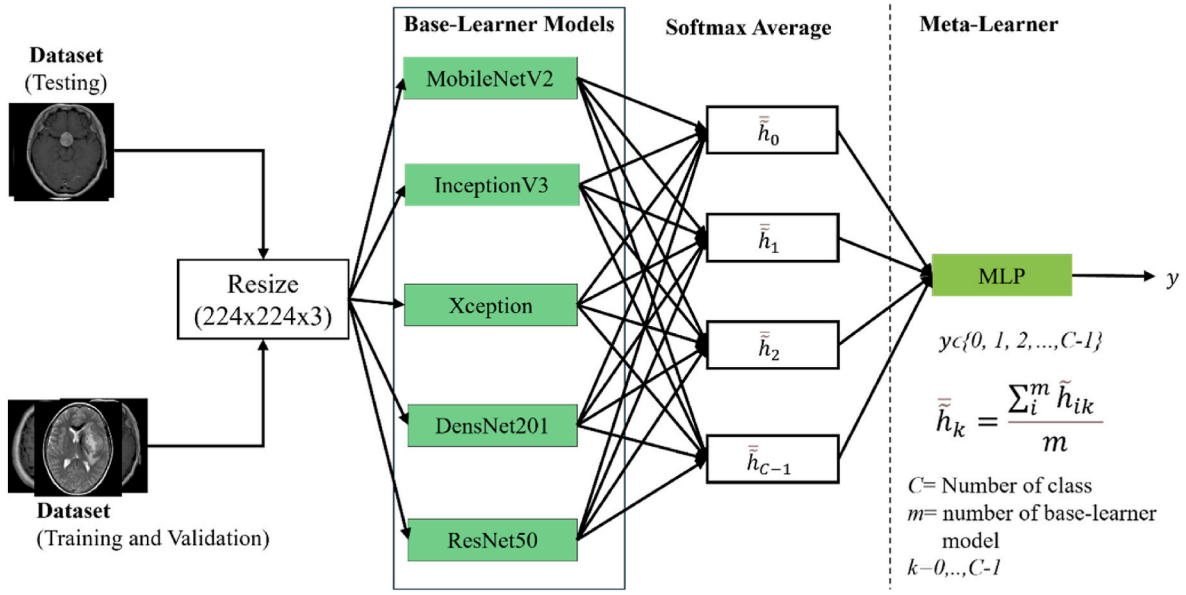

Fig. 3. Scheme (framework) of meta-learning based on softmax average.

3.4. Scheme of meta-learning

Before meta-learning is executed, brain MRI images are input in the pre-processing stage. The pre-processing results are then input for CNN base-learner models in the training or testing stage. Each base-learner model is then subjected to a learning process (training) to obtain model weights with hyperparameter scenarios in Table 2. The training process results using the CNN base-learner model are then used to determine the softmax value on the output layer of each CNN model. From the softmax value of each CNN base-learner model, the average softmax value involving all base-learner models in each corresponding class is then determined (see Fig. 3). For the evaluation of the proposed method on Datasets 1 and 2, the average softmax or sigmoid value on the output layer is as in Equation [1] – [4].

$$\bar{\bar{h}}_0 = \frac{\sum_i^5 \tilde{h}_{i0}}{5} \tag{1}$$

$$\bar{\bar{h}}_1 = \frac{\sum_i^5 \tilde{h}_{i1}}{5} \tag{2}$$

$$\bar{\bar{h}}_2 = \frac{\sum_{i}^{5} \tilde{h}_{i2}}{5} \tag{3}$$

$$\bar{\bar{h}}_3 = \frac{\sum_{i}^{5} \tilde{h}_{i3}}{5} \tag{4}$$

Where $\bar{\bar{h}}_0$, $\bar{\bar{h}}_1$, $\bar{\bar{h}}_2$, and $\bar{\bar{h}}_3$ are the softmax-average or sigmoid on a class or label 0 (Glioma), 1 (Meningioma), 2 (No Tumour), and 3(Pituitary). Meanwhile, $\tilde{h}_{i0}$, $\tilde{h}_{i1}$, $\tilde{h}_{i2}$, and $\tilde{h}_{i3}$ are the softmax or sigmoid values of the ith base-learner model at labels 0, 1, 2, and 3, respectively. For testing on the Dataset, the average softmax value at the output layer is as in Equation [5].

$$\bar{\bar{h}}_k = \frac{\sum_{i}^{5} \tilde{h}_{ik}}{17}, k = 0, 1, 2, \ldots, 16 \tag{5}$$

Where $\bar{\bar{h}}_k$ is the average softmax or sigmoid at class or label k with k = 0,1, 2, …, 16 are labels for Dataset 3 (see Table 1). At the same time, $\tilde{h}_{ik}$ is each softmax or sigmoid value in the ith base-learner model on the kth label or class.

The average of softmax or sigmoid generated by each base-learner model is then entered into the meta-learning process using MLP [17]. The training results using the MLP meta-learner at that stage are then used for the final classification. The MLPs tested refer to the proposed MLP architecture in Table 3. The proposed MLP architectures include MLP1, MLP2, MLP3, MLP4, and MLP5. The MLP1 architecture consists of an input layer with a size of 4 when implemented on Datasets 1 and 2 and a size of 17 when implemented on Dataset 3. This MLP architecture has one hidden layer with 32 neurons, and the activation function used is ReLU. Dropout 0.5 is applied to the number of neurons leading to the output layer, which has a dense size of 4 for Datasets 1 and 2 and 17 for Dataset 3 with the softmax activation function. The MLP2 and MLP3 architectures are almost the same as the MLP1 architecture, except that the number of neurons in the hidden layer is 64 and 128 respectively. For the MLP4 and MLP5 architectures, the input layer size, activation function used, and dropout size are the same as MLP1, MLP2, and MLP3, except that MLP4 and MLP5 have two hidden layers with the number of neurons in each hidden layer of 64 and 32, and 128 and 64 (see Table 3 and Fig. 4). All proposed MLPs are implemented at epoch 50 with the Adam optimizer.

Algorithm 1 contains detailed steps of the learning process of the CNN base-learner model and the MLP meta-learner model described from the proposed framework in Fig. 3. The algorithm shows how to obtain the softmax average based on the softmax value of each base-learner model. **Algorithm 2** provides detailed steps of the process to

Algorithm 1. Training of base-learner CNN and meta-learner MLP model

Input: D {Brain MRI image of training dataset}, V {Brain MRI image of validation dataset}, α {learning rate}, $\tilde{D}$ {label of training}, $\tilde{V}$ {label of validation}, e {epoch for base-learner}, $\tilde{e}$ {epoch for meta-learner}, b {bacth size for base-learber}, $\tilde{b}$ {bacth size for meta-learner}, Q {optimizer for base-learner}, $\tilde{Q}$ {optimizer for meta-learner}, C {number of labels/classes}

Output: M_1, M_2, M_3, M_4, M_5 {The training results of base-learner model: MobileNetV2, InceptionV3,Xception, DensNet201, and ResNet50}, $\tilde{M}$ { The training results of meta-learner model}

{pre-processing: image resizing}
$X \leftarrow \text{resize}(D, 224,224,3)$
$Z \leftarrow \text{resize}(V, 224,224,3)$
{training}
$M_1 \leftarrow model1.fit\ (X, Z, \tilde{D}, \tilde{V}, \alpha, e, b, Q)$ {training of MobileNetV2}
$M_2 \leftarrow model2.fit\ (X, Z, \tilde{D}, \tilde{V}, \alpha, e, b, Q)$ {training of InceptionV3}
$M_3 \leftarrow model3.fit\ (X, Z, \tilde{D}, \tilde{V}, \alpha, e, b, Q)$ {training of Xception}
$M_4 \leftarrow model4.fit\ (X, Z, \tilde{D}, \tilde{V}, \alpha, e, b, Q)$ {training of DensNet201}
$M_5 \leftarrow model5.fit\ (X, Z, \tilde{D}, \tilde{V}, \alpha, e, b, Q)$ {training of ResNet50}
$G \leftarrow X \cup Z$ {merge training and validation dataset}
$\tilde{G} \leftarrow \tilde{D} \cup \tilde{V}$ {merge training and validation label}
for $i \leftarrow 1$ to 5 {number of base-learner models}
    $\tilde{h}_{ik} \leftarrow \text{softmax}(M_i, G)$ {softmax value of each base-learner model}
    for $k \leftarrow 0$ to C-1 {number of class}
        $\bar{\bar{h}}_k \leftarrow \text{sum}(\tilde{h}_{ik})/5$ {softmax average of each class}
$\tilde{M} \leftarrow MLP.fit(\bar{\bar{h}}, \tilde{G}, \tilde{e}, \tilde{b}, \tilde{Q})$ {training of MLP,we can choose one of the models in **Fig.4** }
return $M_1, M_2, M_3, M_4, M_5, \tilde{M}$

Algorithm 2. Testing of meta-learner of MLP model

Input: M_1, M_2, M_3, M_4, M_5, $\tilde{M}$ {The training results of base-learner model: MobileNetV2, InceptionV3,Xception, DensNet201, ResNet50, and MLP}, $\tilde{X}$ {Brain MRI image of testing dataset}, C {number of labels/classes}

Output: y {classification result}

$\tilde{Z} \leftarrow \text{resize}(\tilde{X}, 224,224,3)$ {pre-processing}
for $i \leftarrow 1$ to 5 {number of base-learner models}
    $\tilde{h}_i \leftarrow \text{softmax}(M_i, \tilde{Z})$ {softmax value of each base-learner model}
    for $k \leftarrow 0$ to C-1 {number of class}
        $\bar{\bar{h}}_k \leftarrow \text{sum}(\tilde{h}_{ik})/5$ {softmax average of each class}
$\hat{s} \leftarrow \text{sigmoid}(\tilde{M}, \bar{\bar{h}})$ {sigmoid value}
$y \leftarrow \underset{0 \le k \le C-1}{\text{argmax}}(\hat{s})$ {class prediction}
return y

Table 2
Hyperparameter *tunning base-learner* of CNN model.

Hyperparameter	Base-Learner Models				
	MobileNetV2	InceptionV3	Xception	DensNet201	ResNet50
Input shape	224 × 224 × 3	224 × 224 × 3	224 × 224 × 3	224 × 224 × 3	224 × 224 × 3
Top layer					
Pooling	GAP	GAP	GAP	GAP	AP, Flatten
Dense (Dataset 1, Dataset 2, Dataset 3)	(4 , 4, 17)	(4 , 4, 17)	(4 , 4, 17)	(4 , 4, 17)	(4 , 4, 17)
Activation	softmax	softmax	sigmoid	softmax	softmax
Epoch	50,100	50,100	50,100	50,100	50,100
Batch size	16	16	16	16	16
Learning rate	0.001	0.001	0.001	0.001	0.001
Loss function	CCE	CCE	CCE	CCE	CCE
Optimizer	Adam, Adamax	Adam, Adamax	Adam, Adamax	Adam, Adamax	Adam, Adamax
Weight initialization	ImageNet	ImageNet	ImageNet	ImageNet	ImageNet

Global Average Pooling = GAP, Average Pooling = AP, Categorical Cross Entropy = CCE.

predict the class with brain MRI image input using the MLP meta-learner model from the learning results.

Our proposed meta-learning scheme is similar to Stacking, but different from Bagging and Boosting. The main difference between our proposed meta-learning scheme and Stacking is in the approach of combining predictions of several models towards a meta-learner model [43,44]. In our proposed scheme, there is an additional process between the base-learner and meta-learner predictions. Calculating the average softmax value of each base-learner model is an additional process in our proposed scheme. Thus, the input feature of the meta-learner model is not the final prediction result but the average softmax for each class or label (see Fig. 3). This additional process will make it easier to obtain better features by considering the average softmax value between the base-learner models. Meanwhile, the difference between Bagging and Boosting lies in the approach of combining predictions, the nature of the base-learner model, and the combination of models used. Bagging uses an approach with parallel weak model training, while Boosting base-learner model training is done sequentially. In terms of nature, the base-learner model in our proposed scheme is heterogeneous, while, in Bagging and Boosting it is homogeneous. In terms of the base-learner models combination, Bagging and Boosting use Majority Voting or Everaging, while our proposed scheme uses ML, namely MLP.

3.5. Performance evaluation of methods

To evaluate the performance of the base-learner CNN model and the proposed MLP meta-learner, accuracy (Acc), precision (Pres), sensitivity (Sens), specificity (Spec), and F-score (Fscr) measures obtained are used based on the false negative (fn), true positive (tp), false positive (fp), and true negative (tn) values [45]. In this study, labels, namely Glioma, Meningioma, Pituitary, and No Tumour are tested. Therefore, performance indicators are defined for each label. For Glioma, tp is the number of times the Glioma image is labeled as Glioma based on the classification result, fn is the number of times the Glioma image is labeled as something other than Glioma based on the classification result, tn is the number of times the Glioma image is labeled as something other than Glioma, and fp is the number of times the Glioma image is labeled as Glioma based on the classification result. The calculations of Acc, Pres, Sens, Spec, and Fscr are specified in Equation [6]-[10] [46]; [47]. For a comprehensive overview of classification performance, this study also

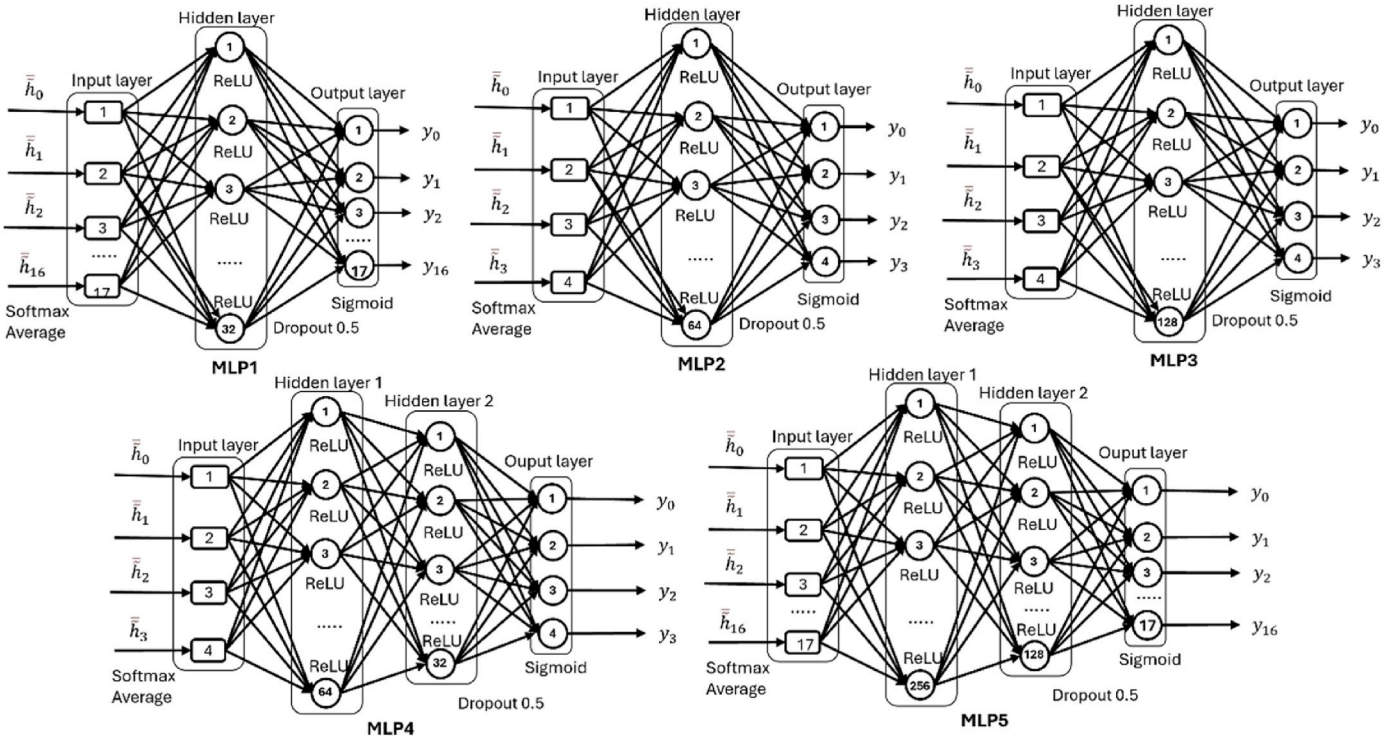

Fig. 4. Architecture of MLP meta-learner.

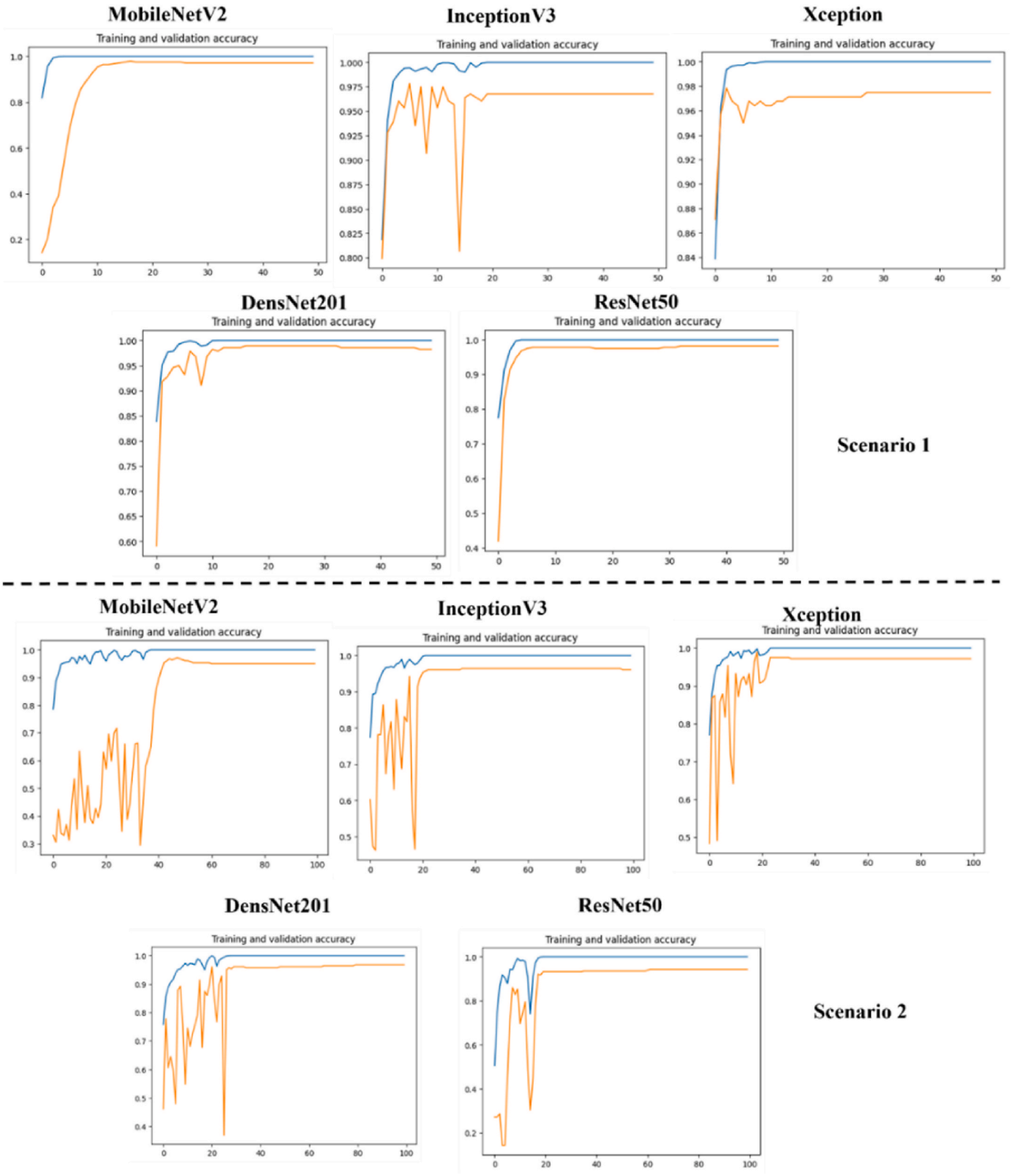

Fig. 5. Accuracy training and validation of each base-learner CNN model on Dataset 1.

used the ROC (Receiver Operating Characteristic) and AUC (Area Under the Curve) curves [48].

$$Acc = (tp + tn)/(tp + fp + tn + fn) \quad (6)$$

$$Pr\ es = tp/(fp + tp) \quad (7)$$

$$Sens = tp/(fn + tp) \quad (8)$$

$$Spec = tn/(fp + tn) \quad (9)$$

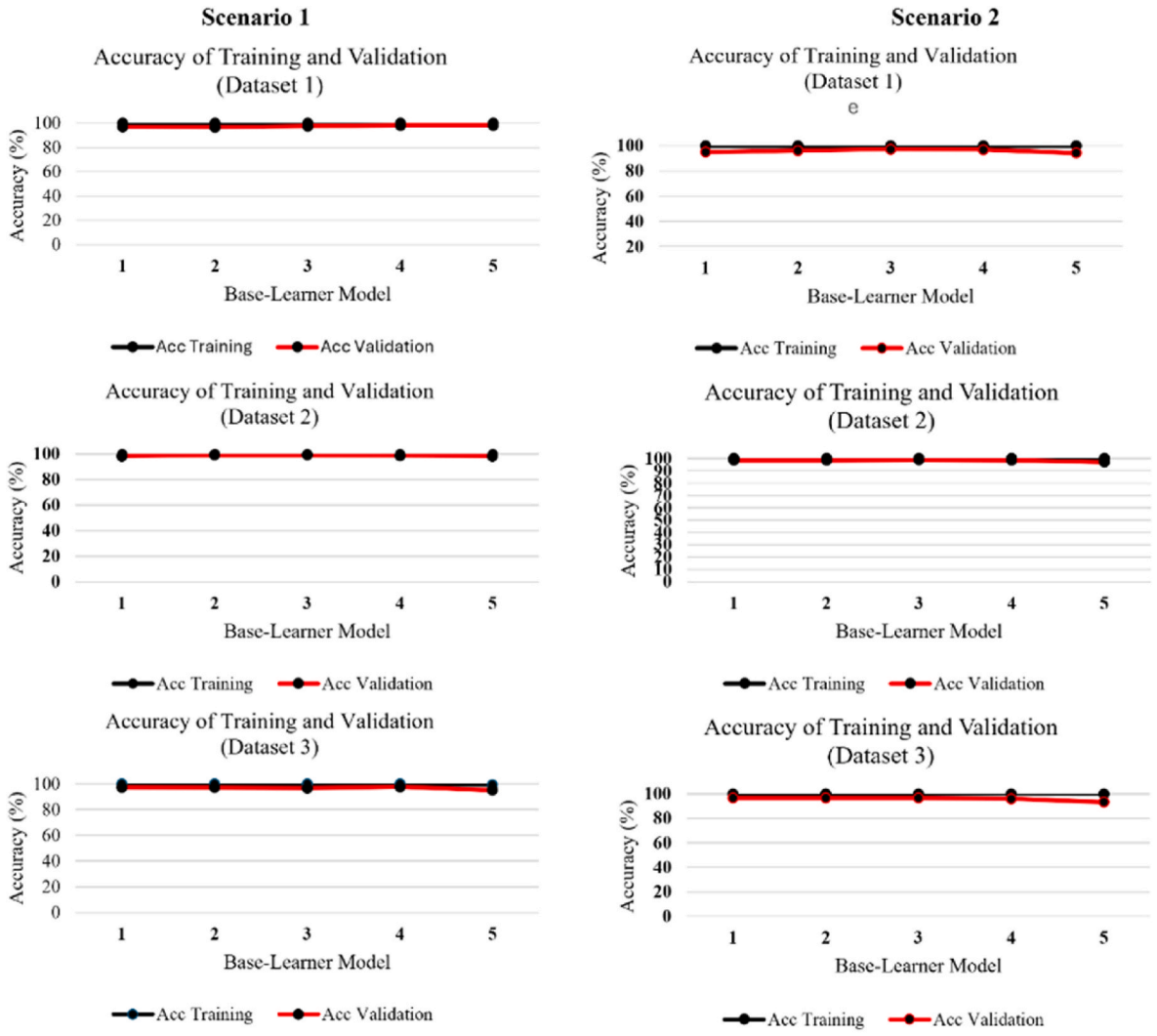

Fig. 6. Accuracy training and validation of each base-learner CNN model on the last epoch.

$$\text{Fscr} = \frac{2(\text{Press})(\text{Sens})}{(\text{Pres} + \text{Sems})} \tag{10}$$

4. Results and discussion

4.1. Training and validation performance of base-learner CNN model

Before the meta-learning process, the training process was carried out on each base-learner CNN model, namely MobileNetV2, InceptionV3, Xception, DensNet201, and ResNet50. There are two scenarios for the base-learner model learning process based on the hyperparameter tuning in Table 2 and the pre-processing of brain MRI images. The first scenario is the learning process through transfer-learning with Adamax optimizer and epoch = 50, while the second scenario, starting with scaling [0,1] against brain MRI images, transfer-learnings of each CNN base-learner model use Adam optimizer and epoch = 100. The scenario is then tested on all datasets. In this study, all of these processes were implemented using Google Colab.

Table 3
Proposed architecture of MLP for meta-learning.

Layer	MLP1	MLP2	MLP3	MLP4	MLP5
(1) Input layer	$\bar{\bar{h}}_k$	$\bar{\bar{h}}_k$	$\bar{\bar{h}}_k$	$\bar{\bar{h}}_k$	$\bar{\bar{h}}_k$
Dataset 1 dan 2	4	4	4	4	4
Dataset 3	17	17	17	17	17
(2) Hiden layer 1, activation	32, ReLU	64, ReLU	128, ReLU	64, ReLU	256, ReLU
(3) Hidden layer 2, activation	–	–	–	32, ReLU	128, ReLU
(4) Dropout	0.5	0.5	0.5	0.5	0.5
(5) Ouput layer, activation	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid

The results of the base-learner CNN model learning at the last epoch in each test scenario can be seen in Fig. 6, with an example of the learning process shown in Fig. 5. The results of the first scenario test on Dataset 1, Dataset 2, and Dataset 3, generally showed that there was no overfitting in the learning process. The difference between training accuracy indicates that this condition and validation accuracy is relatively tiny. Learning all base-learner models on Dataset 1 produced training accuracy ranging from 99.92 % to 100 % with validation accuracy of 96.77 %–98.21 %; on Dataset 2, training accuracy ranged from 99.98 % to 100 % and validation accuracy ranged from 98.25 % to 99.13 %, while on Dataset 3, training accuracy ranged from 99.61 % to 99.97 % and validation accuracy ranged between 95.19 % and 97.72 %. In the second scenario, transfer learning of each CNN base-learner model on Dataset 1 produced a training accuracy of 99.96 %–100 % and a

Table 4
Performance of brain tumour classification by base-learner CNN model and meta-learner MLP on testing Dataset 1.

Experimental Scenario	Model		Acc (%)	Average (%)				Δ (%)[a]
				Pres	Sens	Spec	Fscr	
Scenario 1 (Optimizer: Adamax Epoch = 50)	Base-Learner	MobileNetV2	93.55	93.18	94.48	97.83	93.70	3.87
		InceptionV3	94.52	93.88	95.31	98.19	94.45	2.9
		Xception	94.52	94.65	95.29	98.11	94.94	2.9
		DensNet201	95.48	95.09	96.12	98.49	95.54	1.94
		ResNet50	93.23	92.76	93.91	97.75	93.14	4.19
	Meta-Learner	MLP2	96.45	96.39	96.96	98.79	96.64	0.97
		MLP3	**97.42**	**97.79**	**97.79**	**99.09**	**97.77**	–
		MLP5	96.45	97.04	96.94	98.75	96.94	0.97
Scenario 2 (Scaling[0,1] Optimizer: Adam Epoch = 100)	Base-Learner	MobileNetV2	93.55	93.60	94.46	97.80	93.94	3.87
		InceptionV3	93.87	93.66	94.74	97.93	94.09	3.55
		Xception	93.55	94.11	94.15	97.77	94.08	3.87
		DensNet201	94.19	93.99	94.97	98.03	94.41	3.23
		ResNet50	90.00	89.72	90.51	96.60	90.08	7.42
	Meta-Learner	MLP1	95.48	95.87	96.09	98.43	95.94	1.94
		MLP2	95.48	95.87	96.09	98.43	95.94	1.94
		MLP4	**95.81**	**96.18**	**96.37**	**98.54**	**96.22**	1.61

[a]) Δ =(max. accuracy of meta-learner) – (all model in all scenario).

Table 5
Performance of brain tumour classification by base-learner CNN model and meta-learner MLP on testing Dataset 2.

Experimental Scenario	Model		Acc (%)	Average (%)				Δ (%)[a]
				Pres	Sens	Spec	Fscr	
Scenario 1 (Optimizer: Adamax Epoch = 50)	Base-Learner	MobileNetV2	98.93	98.91	98.84	99.64	98.87	0.61
		InceptionV3	99.16	99.10	99.09	99.73	99.09	0.38
		Xception	99.08	99.06	99.01	99.70	99.03	0.46
		DensNet201	**99.54**	**99.51**	**99.50**	**99.85**	**99.51**	0
		ResNet50	98.78	98.70	98.68	99.60	98.69	0.76
	Meta-Learner	MLP2	99.47	99.42	99.42	99.83	99.42	0.07
		MLP3	99.47	99.42	99.42	99.83	99.42	0.07
		MLP5	99.31	99.28	99.25	99.78	99.26	0.23
Scenario 2 (Scaling[0,1] Optimizer: Adam Epoch = 100)	Base-Learner	MobileNetV2	98.47	98.47	98.35	99.49	98.41	1.07
		InceptionV3	99.16	99.10	99.09	99.73	99.09	0.38
		Xception	99.24	99.20	99.17	99.75	99.19	0.3
		DensNet201	99.39	99.35	99.34	99.80	99.34	0.15
		ResNet50	97.10	97.10	96.86	99.03	96.94	2.44
	Meta-Learner	MLP1	99.39	99.34	99.34	99.80	99.34	0.15
		MLP2	**99.54**	**99.50**	**99.51**	**99.85**	**99.50**	–
		MLP4	99.47	99.42	99.42	99.83	99.42	0.07

[a]) Δ =(max. accuracy of meta-learner) – (all model in all scenario).

validation accuracy of 94.27 %–97.13 %; in Dataset 2, the training accuracy ranged from 99.98 % to 100 % and a validation accuracy of 97.1 %–98.95 %, while in Dataset 3, the training accuracy was 99.94 %–99.97 % and a validation accuracy of 93.16 %–96.46 %. Fig. 5 shows the learning process of all base-learner models on Dataset 1. In the first scenario, training and validation accuracy performances were more stable at epochs of more than 20. In contrast, in the second scenario, training and validation accuracy performances were more stable at epochs of more than 40. The optimizers difference in the two scenarios affected of the performance stability of each epoch. Adamax stabilized faster without image scaling than the Adam optimizer with MRI image scaling at the beginning of the process.

4.2. Performance of base-learner CNN model on the testing dataset

The test results of each base-learner model for each scenario and dataset are shown in Table 4, Table 5, and Table 6. In the first scenario and Dataset 1, MobileNetV2 produced a brain tumour classification accuracy of 93.55 %, better than ResNet50, which obtained an accuracy of 93.23 %. InceptionV3 and Xception obtained the same accuracy of 94.52 % and better than MobileNetV2. Meanwhile, DensNet201 yielded the best accuracy compared to all other base-learner models, which was 95.48 %, as well as the precision (Pres), sensitivity (Sens), specificity (Spec), and F-score (Fscr) were also the best, with values of 95.09 %, 96.12 %, 98.49 %, and 95.54 % respectively. For the second scenario, DesnNet201 obtained a classification accuracy of 94.19 %, better than MobileNetV2, InceptionV3, Xception, and ResNet50, which yielded tumour classification accuracy of 93.55 %, 93.87 %, 93.55 %, and 90.00 %, respectively. DensNet201 also achieved the best classification sensitivity, specificity, and F-score, as shown in Table 4. While Xception, this scenario yielded the best precision compared to the others.

For evaluation using Dataset 2, in the first scenario, DensNet201 obtained the best brain tumour classification accuracy of 99.54 % compared to all other base-learner models, as well as the precision, sensitivity, specificity, and F-score values. Meanwhile, each MobileNetV2, InceptionV3, Xception, and ResNet50 each yielded classification accuracies of 98.93 %, 99.16 %, 99.08 %, and 98.78 %. In the second scenario, DensNet201 also obtained the best classification accuracy of 99.39 %, the precision, sensitivity, specificity, and F-score values (see Table 5). The evaluation of the base-learner model using Dataset 3 also

Table 6
Performance of brain tumour classification by base-learner CNN model and meta-learner MLP on testing Dataset 3.

Experimental Scenario	Model		Acc (%)	Average (%)				Δ (%)[a]
				Pres	Sens	Spec	Fscr	
Scenario 1 (Optimizer: Adamax Epoch = 50)	Base-Learner	MobileNetV2	97.28	97.27	95.88	99.83	96.45	1.59
		InceptionV3	97.73	97.90	96.95	99.86	97.35	1.14
		Xception	97.51	97.33	96.96	99.84	97.11	1.36
		DensNet201	98.41	98.81	97.26	99.90	97.90	0.46
		ResNet50	93.20	93.01	92.31	99.57	92.31	5.67
	Meta-Learner	MLP2	98.19	98.66	96.78	99.88	97.48	0.68
		MLP3	98.41	98.81	96.94	99.90	97.65	0.46
		MLP5	**98.87**	**99.18**	**97.60**	**99.93**	**98.26**	–
Scenario 2 (Scaling[0,1] Optimizer: Adam Epoch = 100)	Base-Learner	MobileNetV2	97.73	97.38	97.02	99.86	97.15	1.14
		InceptionV3	95.46	96.37	95.18	99.71	95.70	3.41
		Xception	98.19	98.77	97.65	99.88	98.08	0.68
		DensNet201	98.19	98.67	**97.94**	99.88	**98.22**	0.68
		ResNet50	95.46	95.18	94.97	99.71	95.02	3.41
	Meta-Learner	MLP1	**98.41**	**98.87**	97.62	**99.90**	98.16	0.46
		MLP2	97.96	98.63	97.28	99.87	97.86	0.91
		MLP4	97.96	98.70	97.36	99.87	97.93	0.91

[a]) Δ =(max. accuracy of meta-learner) – (all model in all scenario).

Table 7
Paired t-tests of base-learner model performance against MLP meta-learner.

Dataset	Model		Mean		t Stat[a]	P(T ≤ t) one-tail
	Base-Learner	Meta-Learner	Base-Learner	Meta-Learner		
Dataset 1 (Scenario 1)	MobileNetV2	MLP3	94.548	97.972	−5.90752	0.002055
	InceptionV3	MLP3	95.27	97.972	−5.31088	0.003021
	Xception	MLP3	95.502	97.972	−6.39416	0.001536
	DensNet201	MLP3	96.144	97.972	−5.20432	0.003248
	ResNet50	MLP3	94.158	97.972	−5.88131	0.002089
Dataset 2 (Scenario 2)	MobileNetV2	MLP2	98.638	99.58	−6.40742	0.001524
	InceptionV3	MLP2	99.234	99.58	−6.08236	0.001847
	Xception	MLP2	99.31	99.58	−6.26048	0.00166
	DensNet201	MLP2	99.444	99.58	−6.23355	0.001687
	ResNet50	MLP2	97.406	99.58	−6.36839	0.001559
Dataset 3 (Scenario 1)	MobileNetV2	MLP5	97.342	98.768	−4.24838	0.006588
	InceptionV3	MLP5	97.958	98.768	−3.79109	0.009625
	Xception	MLP5	97.75	98.768	−3.36478	0.01409
	DensNet201	MLP5	98.456	98.768	−4.24814	0.006589
	ResNet50	MLP5	94.08	98.768	−4.29322	0.006357

[a] Level of significant on the hyphotesis testing is 0.05 (5 %), t Critical one-tail = 2.131847.

yielded almost the same performance as the evaluation results on Dataset 1 and Dataset 3. In the first scenario, DensNet201 obtained classification accuracy, precision, sensitivity, specificity, and F-score of 98.41 %, 98.81 %, 97.26 %, 99.90 %, and 97.90 %, respectively. With these results, DensNet201 had better classification performance than other base-learner models. In the second scenario, DensNet201 and Xception achieved the same classification accuracy of 98.19 %, better than MobileNetV2, InceptionV3, and ResNet50 (see Table 6).

The evaluation results of all base-learner models on all datasets showed that the brain tumour classification performance in the first scenario was generally better than in the second scenario. The Adamax optimizer used in the first scenario was more stable. It converged faster than the Adam optimizer in the second scenario, even though MRI image scaling was performed before learning. This reason can be proven by the training process of each base-learner model in each epoch, as shown in Fig. 5. Meanwhile, Densnet201 is one of the base-learner models that generally, in each scenario and dataset, provided better classification performance than other base-learner models. The most substantial reason is that DensNet201 has a layer always connected to the next layer with feedforward, therefore, it does not reduce the features produced by each convolution layer. Apart from its advantages, all CNN base-learner models have different architectures from one another, thus, the results of all these models will produce output (softmax), which will be forwarded to the meta-learning process and can improve tumour classification performance.

4.3. Performance of meta-learner on the testing dataset

The results of the meta-learning trial using MLP on all datasets and scenarios are shown in Table 4, Table 5, Table 6, and Table 8. For the evaluation on Dataset 1 and the first scenario, using the proposed scheme or framework, the MLP3 meta-learner yielded the best brain tumour classification performance among the meta-learners. MLP3 obtained an accuracy of 97.42 % with precision, sensitivity, specificity, and F-score values of 97.79 %, 97.79 %, 99.09 %, and 97.77 %, respectively. With these results, the MLP3 meta-learner was better than the performance of MLP2 and MLP5, which produced a classification accuracy of 96.45 %. MLP3 increased the classification accuracy by 1.94 %–4.19 % against the base-learner model in the first scenario and 3.32 %–7.42 % against the base-learner model in the second scenario. In the second scenario, MLP4 obtained the highest classification accuracy compared to MLP1 and MLP2 meta-learners. MLP4 yielded a tumour classification accuracy of 95.81 %. Although the classification accuracy of MLP4 was not better than MLP3 in the first scenario, it could increase the accuracy of the base-learner model in both the first and second scenarios by 0.33 %–5.81 % (see Table 4). In evaluating the proposed

Table 8
Best performances of base-learner CNN model and meta-learner of MLP.

Model	Class/Label Tumour	Confusion Matrix (Cm)				Pres (%)	Sens (%)	Spec (%)	Fscr (%)
Testing Dataset 1									
Base-leaner	Glioma	84	5	1	0	97.67	93.33	99.09	95.45
DensNet201 (95.48 %)	Meningioma	2	84	3	2	93.33	92.31	97.26	92.82
	No Tumour	0	0	44	0	91.67	100	98.50	95.65
	Pituitary	0	1	0	84	97.67	98.82	99.11	98.25
Meta-learner	Glioma	85	5	0	0	**98.84**	**94.44**	**99.55**	**96.59**
MLP3(97.42 %)	Meningioma	1	88	0	2	**94.62**	**96.70**	**97.72**	**95.65**
	No Tumour	0	0	44	0	**100**	100	**100**	**100**
	Pituitary	0	0	0	85	**97.70**	**100**	99.11	**98.84**
Testing Dataset 2									
Base-leaner	Glioma	297	3	0	0	**99.66**	99.00	**99.90**	99.33
DensNet201 (99.54 %)	Meningioma	1	304	0	1	**98.70**	**99.35**	99.60	99.02
	No Tumour	0	0	405	0	100	100	100	100
	Pituitary	0	1	0	299	**99.67**	99.67	**99.90**	**99.67**
Meta-learner	Glioma	299	1	0	0	99.34	**99.67**	99.80	**99.50**
MLP2 (99.54 %)	Meningioma	2	302	0	2	99.34	98.69	**99.80**	99.02
	No Tumour	0	0	405	0	100	100	100	100
	Pituitary	0	1	0	299	99.34	99.67	99.80	99.50
Testing Dataset 3									
Base-leaner	[illegible]					100	100	100	100
DensNet201(98.41 %)	[illegible]					100	100	100	100
	[illegible]					94.44	97.14	99.51	95.77
	[illegible]					97.22	100	99.75	98.59
	[illegible]					100	100	100	100
	[illegible]					91.43	96.97	99.26	94.12
	[illegible]					100	100	100	100
	[illegible]					96.67	100	99.76	98.31
	[illegible]					100	100	100	100
	[illegible]					100	100	100	100
	[illegible]					100	90.91	100	95.24
	[illegible]					100	100	100	100
	[illegible]					100	100	100	100
	[illegible]					100	93.33	100	96.55
	[illegible]					100	100	100	100
	[illegible]					100	75.00	100	85.71
Meta-learner	[illegible]					100	100	100	100
MLP5 (98.87 %)	[illegible]					100	100	99.51	**97.22**
	[illegible]					**94.59**	**100**	99.75	98.59
	[illegible]					97.22	100	100	100
	[illegible]					100	100	**99.51**	**97.06**
	[illegible]					**94.29**	**100**	100	100
	[illegible]					100	100	**100**	**100**
	[illegible]					**100**	100	100	100
	[illegible]					100	100	100	100
	[illegible]					100	100	100	95.24
	[illegible]					100	90.91	100	100
	[illegible]					100	100	100	100
	[illegible]					100	100	100	96.55
	[illegible]					100	93.33	100	100
	[illegible]					100	100	100	
	[illegible]					100	75.00	100	85.71

method on Dataset 2, using the proposed scheme or framework, the MLP2 meta-learner in the second scenario yielded the best brain tumour classification performance among the other meta-learners. MLP2 obtained an accuracy of 99.54 % with precision, sensitivity, specificity, and F-score values of 99.50 %, 99.51 %, 99.85 %, and 99.50 %, respectively. With these results, MLP2 could increase the classification accuracy by 0 %–2.44 % against the base-learner model in the first and second scenarios. The base-learner model with the same accuracy as MLP2 was DensNet201 in the first scenario. However, all meta-learner models produce tumour classification accuracy that is more consistent than the base-learner model. The accuracy of the meta-learner models in the first and second scenarios for Dataset 2 ranged from 99.31 % to 99.54 % (see Table 5). Testing on Dataset 3, the framework with the MLP5 meta-learner in the first scenario produced the best brain tumour classification performance among all models and all scenarios. MLP5 obtained an accuracy of 98.87 %, and the precision, sensitivity, specificity, and F-score values were also the best, with values of 99.18 %, 97.60 %, 99.93 %, and 98.26 %, respectively. MLP2 increased classification accuracy by 0.46 %–5.67 % over the base-learner model in the first scenario and 0.68 %–3.41 % over the base-learner model in the second scenario.

From the test results on each scenario and dataset, the MLP meta-learner produced a significant improvement in classification performance over the base-learner model. The statistical test results using a paired *t*-test, at a significance level of 0.05, MLP3 in the first scenario (Dataset 1) significantly yielded an average classification performance more than all base-learner models. These results are shown by the value of $|t\ Stat| > (t\ Critical\ one\text{-}tail)$ or $(P\ (T \leq t)\ one\text{-}tail) < 0.05$ (see Table 7). MLP2 in the second scenario (Dataset 2) and MLP5 in the first scenario (Dataset 3) also significantly obtained an average classification performance greater than the CNN base-learner model.

In general, based on the test results using the paired *t*-test, the MLP

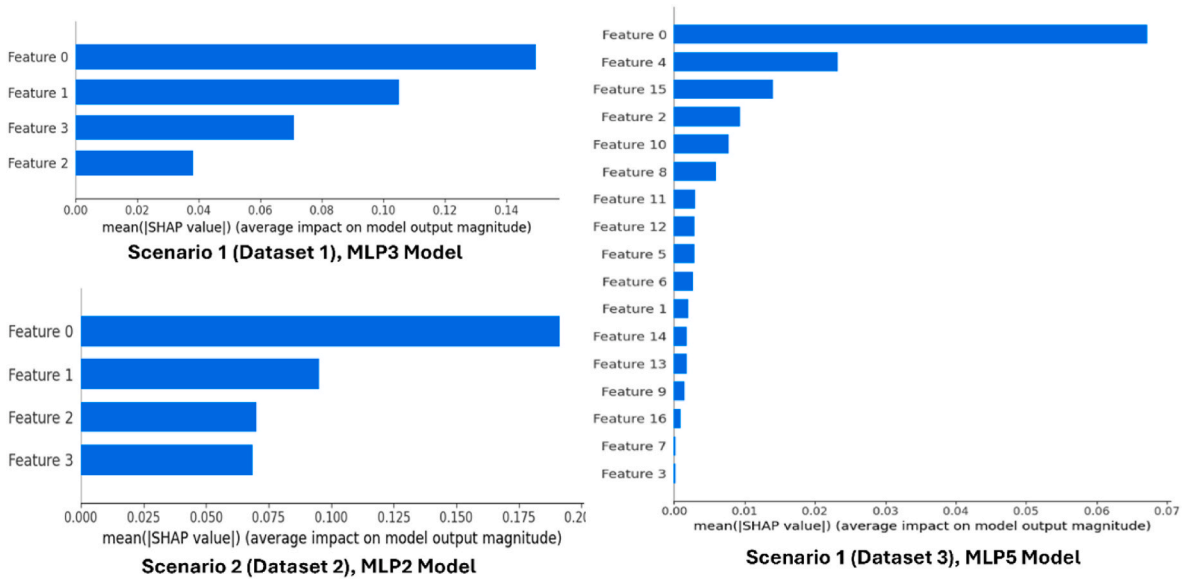

Fig. 7. Average impact of features (softmax average) on output meta-learner of MLP using mean |SHAP value|.

meta-learner in each scenario and dataset can work better to improve the performance of brain tumour classification. The average softmax produced by each base-learner model can be considered as a representation of the learning of the base-learner model. By involving validation dataset in determining the softmax value and its average, it can complement the features that the base-learner model does not capture. The average softmax is then learned more deeply by the MLP through deeper learning to direct the classification error in the base-learner model toward the correct target. Therefore, the average sofmax involving validation dataset will reduce the increase stagnation in classification performance by the meta-learner (MLP). Fig. 7 shows an example of the mean |SHAP value | of the softmax average feature in each class on the output of the meta-learning model (MLP2, MLP3, and MLP5). Feature 0 is the softmax average for the Glioma class and has the highest average impact on the output of the meta-learning model for each dataset scenario. The feature with the next highest impact is feature 1 (softmax average of Meningioma). This feature had the highest impact after Glioma on the meta-learner model in the first scenario (Dataset 1) and the second scenario (Dataset 2). For the first scenario (Dataset 3), feature 0 (softmax average of Glioma) and feature 4 (softmax average of Meningioma) also had the greatest impact on the output of the meta-learner. From the physical characteristics, Glioma and Meningioma have almost similar shapes and sizes, therefore, it is difficult to classify. For these reasons, both features (softmax average of Glioma and Meningioma) have a significant impact on the meta-learner in improving tumour classification performance.

The softmax average feature based on the mean |SHAP value generally has a significant impact on the performance of the MLP meta-learner. Meanwhile, the overall classification performance of the MLP meta-learner is shown in Fig. 8. From the ROC curve, MLP3 in the first scenario (Dataset 1) obtained the best overall classification performance. Therefore, the increase in accuracy produced by the base-learner model is undoubted. The AUC value above 0.99 for each class (0 = Glioma, 1 = Meningioma, 2 = No Tumour, and 3 = Pituitary) shows that the model's ability to classify tumours is very high. The overall classification performance with the same results was also shown by the MLP2 meta-learner in the second scenario (Dataset 2) with an AUC value for each class above 0.99. The MLP5 meta-learner involved data with a large number of classes (Dataset 3). In addition to producing high classification accuracy, the overall classification performance was also undoubted. These results were shown by the ROC curve and AUC values, which, in macro and micro terms, were more than 0.99.

Table 8 shows in more depth how the proposed MLP improves the performance of tumour-type classification without reducing the already good performance. In Dataset 1, MLP3 increased the precision, sensitivity, specificity, and F-score of Glioma and Meningioma classification. MLP3 can increase the precision, specificity, and F-score of No Tumour classification without decreasing the good classification sensitivity. Likewise, for the type of Pituitary tumour, MLP3 can increase the precision, sensitivity, or F-score without reducing the classification specificity performance (see Table 8). Likewise, the evaluation on Dataset 3, MLP5, increased precision, sensitivity, or F-score without decreasing the specificity of the Glioma-T2 classification. MLP5 can increase the performance of precision, sensitivity, specificity, and F-score of Meningioma-T2 classification without reducing the classification performance of other types of tumours. Likewise, MLP5 can increase the precision, specificity, and F-score performance of Normal-T2 classification without decreasing the classification sensitivity and other performances. In contrast to the test on Dataset 2, MLP2 obtained the same accuracy as the best base-learner model (DensNet201). MLP2 increases sensitivity and F-score classification performance. Still, it slightly decreases precision and specificity, and Meningioma MLP2 can improve the specificity of Glioma classification but cannot maintain precision, sensitivity, or F-score performance. Consequently, for evaluation with Dataset 2, the purpose of the classification depends on whether to increase the classification's precision, sensitivity, or specificity. If we emphasise increasing the sensitivity and F-score of Glioma classification, then MLP2 is the best.

Several previous studies have evaluated their proposed methods using Dataset 2, as reported in Ref. [49] that proposed the HOG-XG Boost method [50], proposed SSBTCNet, and [1] proposed image enhancement and CNN (see Table 9). HOG-XG Boost used a histogram of oriented gradients (HOG) to extract MRI image features and passed to the XG Boost classifier. With the tumour's significantly various shape, size, and position, HOG will have difficulty obtaining representative features, especially for Glioma and Meningioma tumour types with similar shapes, sizes, and pixel intensities. Therefore, the resulting features are complex for XG Boost to classify. Meanwhile, SSBTCNet in Ref. [50], unlabeled data in the training process of the method could produce errors in the supervised classification network due to the similar shape, size, and intensity of tumour pixels. The same thing was done by Ref. [1]. With the similar shape, size, and intensity of tumour pixels, image enhancement will find it difficult to produce images that a CNN model easily classifies. Meanwhile, our proposed method involves

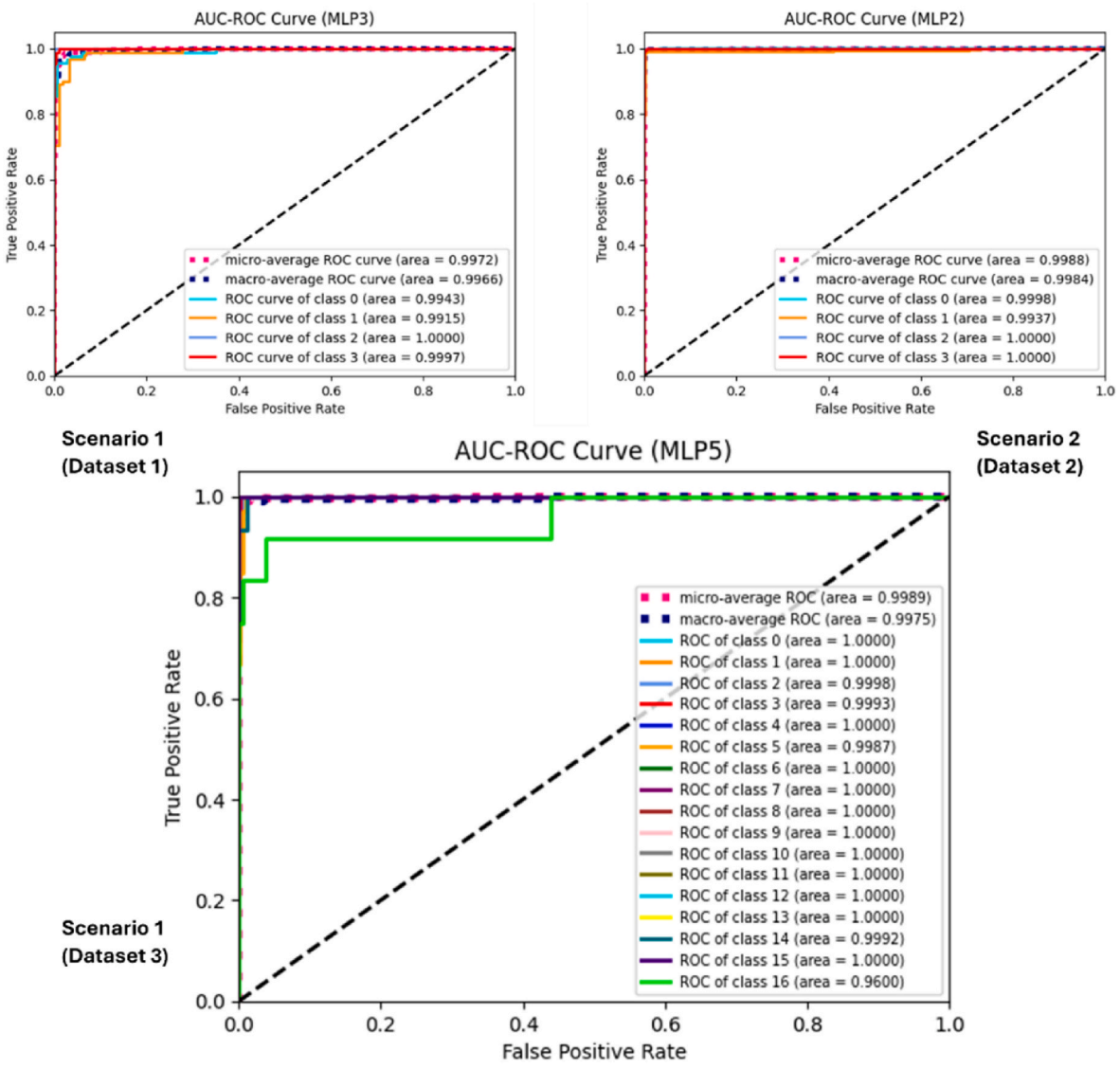

Fig. 8. AUC-ROC on meta-learner of MLP models.

Table 9
Comparison of the existing methods with the proposed method.

No.	Authors	Pre-processing	Classification Methods	Acc (%)		
				Testing Dataset 1	Testing Dataset 2	Testing Dataset 3
1	This Work, Original Model in [18]	Image resizing: 224 × 224 × 3	MobileNetV2 (transfer learning)	93.55	98.93	97.28
2	This Work, Original Model in [19]	Image resizing: 224 × 224 × 3	InceptionV3 (transfer learning)	94.52	99.16	97.73
3	This Work, Original Model in [20]	Image resizing 224 × 224 × 3	Xception (transfer learning)	94.52	99.08	97.51
2	This Work, Original Model in [21]	Image resizing: 224 × 224 × 3	DensNet-201(transfer learning)	95.48	**99.54**	98.41
3	This Work, Original Model in [22]	Image resizing: 224 × 224 × 3	ResNet50(transfer learning)	93.23	98.78	93.20
4	[49]	Data noise elimination	HOG-XG Boost	–	92,02	–
5	[50]	fuzzy contrast enhancement	SSBTCNet model	–	96,5	–
6	[1]	Gaussian-blur + CLAHE Image resizing : 224 × 224 × 3	CNN	–	97,84	–
7	Proposed Method	Image resizing: 224 × 224 × 3 *+ scaling [0,1]	Base-learner CNNs + softmax average +(MLP)	**97.42**	**99.54***	**98.87**

several CNN models as base learners to overcome the shape, size, or position of tumours with high variations. Moreover, the existence of meta-learning using MLP with average softmax input produced by the base-learner model will improve classification performance. The softmax or sigmoid value obtained from each CNN model (base-learner) is considered a feature representation in the weight values for each class in each model. The softmax average obtained by averaging these features from several CNN models is the best feature for the MLP meta-learner because it considers the feature representation of several CNN models involved. Analysis using the average of the SHAP values shows that the softmax average feature has a high impact on the MLP output. In this study, the softmax average feature of the Glioma and Meningioma classes had the highest impact because physically the shape and size of these types of tumours are most similar. In terms of complexity, the methods reported by Shilaskar et al. [49], Rasheed et al. [1], and Atha & Chaki [50] have lower complexity or computation than our proposed method. It must be considered in the future although the focus of the study is on improving classification performance.

4.4. Limitations and potential future research

Involving many deep learning models as base-learners and meta-learner models for the final classification stage of MRI-based brain tumours will impact computational complexity. The existence of this computational complexity causes a large consumption of learning time. In addition, the impact of unbalanced data and the potential for overfitting in meta-learning needs to be considered. Computational complexity occurs during learning, therefore, it will indirectly affect clinical application. However, in clinical implementations, learning the model can be separated from the implemented system. The system applied clinically may only contain learning results, consequently, time consumption can be faster, however, complexity remains an essential consideration in addition to improving classification performance. Therefore, the involvement of several CNN models as base learners and MLP as meta-learner models in this study also impacts computational complexity. The application of graph-based Autoencoders and clustering methods [51,52], can be an alternative to obtain tumour features that represent MRI images while considering computational complexity.

Another essential issue that needs consideration is the problem of generalization and the robustness level of the proposed method. In addition to paying attention to the complexity of computation, overfitting, and data imbalance, the potential implementation of the system practically in the future requires external data validation using independent datasets from various institutions. The external data validation can be obtained well if there are sufficient resources. Therefore, the involvement of external data can be a potential for future research. The use of several different datasets and scenarios in the study is to reduce generalization bias while maintaining the robustness of the proposed method. Low-weight models with high capabilities supported by external data validation are still a challenge in the future. These challenges are reducing computational complexity and feasibility for practical or clinical implementation. From a medical perspective, a more in-depth analysis of the radiomic features of protein biomarkers will bridge imaging and molecular diagnostics and provide a more comprehensive understanding of tumour biology. Thus, these efforts can have a strong relationship to improving tumour classification performance, especially for clinical and practical applications.

5. Conclusion

Integrating several CNN models for brain tumour classification based on MRI images of brain tumours followed by meta-learning is one solution to improve the performance of brain tumour classification. The CNN models include MobileNetV2, InceptionV3, Xception, DesnNet201, and ResNet201 as base-learner models that can overcome the presence of tumour shapes, sizes, and sizes with high variations. With the framework proposed in this study, the softmax output of the base-learner model was processed again by calculating its average and forwarded to the meta-learning stage with the proposed MLP (meta-learner). The evaluation results on Dataset 1. The proposed framework with MLP meta-learner produced the best classification accuracy, precision, sensitivity, specificity, and F-score of 97.42 %, 97.79 %, 97.79 %, and 99.09 %, respectively. In Dataset 2, it produced the best classification accuracy, precision, sensitivity, specificity, and F-score of 99.54 %, 99.50 %, 99.51 %, 99.85 %, and 99.50 % respectively, and 98.87 %, 99.18 %, 97.60 %, 99.93 %, and 98.26 % in Dataset 3. With these results, our proposed method could be an alternative to support diagnostics for medical personnel in classifying brain tumour types with high accuracy. For clinical application, it is necessary to add training and testing data and direct trials on patients.

CRediT authorship contribution statement

Irwan Budi Santoso: Writing – review & editing, Supervision, Methodology, Formal analysis, Conceptualization. **Shoffin Nahwa Utama:** Software, Project administration, Data curation. **Supriyono:** Writing – original draft, Visualization, Validation, Supervision, Software, Investigation.

Declaration of competing interest

The authors declare that there is no conflict of interest regarding the publication of this article. The author confirms that this article is free from plagiarism.

Acknowledgment

This research was financially supported by Universitas Islam Negeri Maulana Malik Ibrahim Malang through the research scheme of the National Development Applied Research (Penelitian Terapan Pembangunan Nasional) with contract number 1143A/LP2M/TL.00/02/2024.

Data availability

Data will be made available on request.

References

[1] Rasheed Z, Ma YK, Ullah I, Ghadi YY, Khan MZ, Khan MA, Abdusalomov A, Alqahtani F, Shehata AM. Brain tumor classification from MRI using image enhancement and convolutional neural network techniques. Brain Sci 2023;13(9). https://doi.org/10.3390/brainsci13091320.

[2] Reyes D, Sánchez J. Performance of convolutional neural networks for the classification of brain tumors using magnetic resonance imaging. Heliyon 2024;10 (3):e25468. https://doi.org/10.1016/j.heliyon.2024.e25468.

[3] Khan MSI, Rahman A, Debnath T, Karim MR, Nasir MK, Band SS, Mosavi A, Dehzangi I. Accurate brain tumor detection using deep convolutional neural network. Comput Struct Biotechnol J 2022;20:4733–45. https://doi.org/10.1016/j.csbj.2022.08.039.

[4] Loughan AR, Aslanzadeh FJ, Brechbiel J, Rodin G, Husain M, Braun SE, Willis KD, Lanoye A. Death-related distress in adult primary brain tumor patients. Neuro-Oncology Practice 2020;7(5):498–506. https://doi.org/10.1093/nop/npaa015.

[5] El-Dahshan EAS, Mohsen HM, Revett K, Salem ABM. Computer-aided diagnosis of human brain tumor through MRI: a survey and a new algorithm. Expert Syst Appl 2014;41(11):5526–45. https://doi.org/10.1016/j.eswa.2014.01.021.

[6] Varuna Shree N, Kumar TNR. Identification and classification of brain tumor MRI images with feature extraction using DWT and probabilistic neural network. Brain Informatics 2018;5(1):23–30. https://doi.org/10.1007/s40708-017-0075-5.

[7] Noreen N, Palaniappan S, Qayyum A, Ahmad I, Imran M, Shoaib M. A deep learning model based on concatenation approach for the diagnosis of brain tumor. IEEE Access 2020;8:55135–44. https://doi.org/10.1109/ACCESS.2020.2978629.

[8] Musallam AS, Sherif AS, Hussein MK. A new convolutional neural network architecture for automatic detection of brain tumors in magnetic resonance imaging images. IEEE Access 2022;10:2775–82. https://doi.org/10.1109/ACCESS.2022.3140289.

[9] Kang J, Ullah Z, Gwak J. Mri-based brain tumor classification using ensemble of deep features and machine learning classifiers. Sensors 2021;21(6):1–21. https://doi.org/10.3390/s21062222.

[10] Ahmad S, Choudhury PK. On the performance of deep transfer learning networks for brain tumor detection using MR images. IEEE Access 2022;10:59099–114. https://doi.org/10.1109/ACCESS.2022.3179376.
[11] Shanthi S, Saradha S, Smitha JA, Prasath N, Anandakumar H. An efficient automatic brain tumor classification using optimized hybrid deep neural network. International Journal of Intelligent Networks 2022;3:188–96. https://doi.org/10.1016/j.ijin.2022.11.003.
[12] Rizwan M, Shabbir A, Javed AR, Shabbir M, Baker T, Al-Jumeily Obe D. Brain tumor and glioma grade classification using Gaussian convolutional neural network. IEEE Access 2022;10:29731–40. https://doi.org/10.1109/ACCESS.2022.3153108.
[13] Younis A, Qiang L, Nyatega CO, Adamu MJ, Kawuwa HB. Brain tumor analysis using deep learning and VGG-16 ensembling learning approaches. Appl Sci 2022;12(14). https://doi.org/10.3390/app12147282.
[14] Chatterjee S, Nizamani FA, Nürnberger A, Speck O. Classification of brain tumours in MR images using deep spatiospatial models. Sci Rep 2022;12(1):1–11. https://doi.org/10.1038/s41598-022-05572-6.
[15] Asif S, Yi W, Ain QU, Hou J, Yi T, Si J. Improving effectiveness of different deep transfer learning-based models for detecting brain tumors from MR images. IEEE Access 2022;10:34716–30. https://doi.org/10.1109/ACCESS.2022.3153306.
[16] Santoso IB, Utama SN, Supriyono. A new voting of convolutional neural networks for brain tumor detection based on MRI images. International Journal of Intelligent Engineering and Systems 2024;17(1):212–27. https://doi.org/10.22266/ijies2024.0229.21.
[17] Yang P, Wang J, Zhao H, Li R. MLP with riemannian covariance for motor imagery based EEG analysis. IEEE Access 2020;8:139974–82. https://doi.org/10.1109/ACCESS.2020.3011969.
[18] Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC. MobileNetV2: inverted residuals and linear bottlenecks. Proceedings of the IEEE computer society conference on computer vision and pattern recognition. 2018. p. 4510–20. https://doi.org/10.1109/CVPR.2018.00474.
[19] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. Proceedings of the IEEE computer society conference on computer vision and pattern recognition, 2016-decem. 2016. p. 2818–26. https://doi.org/10.1109/CVPR.2016.308.
[20] Chollet F. Xception: deep learning with depthwise separable convolutions. In: 30th IEEE conference on computer vision and pattern recognition. CVPR; 2017. p. 1800–7. https://doi.org/10.1109/CVPR.2017.195. 2017.
[21] Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: 30th IEEE conference on computer vision and pattern recognition. CVPR; 2017. p. 2261–9. https://doi.org/10.1109/CVPR.2017.243. 2017.
[22] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. Proceedings of the IEEE computer society conference on computer vision and pattern recognition, 2016-decem. 2016. p. 770–8. https://doi.org/10.1109/CVPR.2016.90.
[23] Thomasdubail. Brain tumors 256x256. https://www.kaggle.com/datasets/thomasdubail/brain-tumors-256x256; 2023.
[24] Bhuvaji S, Kadam A, Bhumkar P, Dedge S, Kanchan S. Brain tumor classification (MRI). https://doi.org/10.34740/kaggle/dsv/1183165; 2020.
[25] Nickparvar M. Brain tumor MRI dataset. https://doi.org/10.34740/kaggle/dsv/2645886; 2021.
[26] Cheng J. Brain tumor dataset Figshare. https://figshare.com/articles/dataset/brain_tumor_dataset/1512427; 2017.
[27] Hamada A. Br35H brain tumor detection 2020. https://www.kaggle.com/datasets/ahmedhamada0/brain-tumor-detection; 2020.
[28] Feltrin F. Brain tumor MRI images 17 classes. https://www.kaggle.com/datasets/fernando2rad/brain-tumor-mri-images-17-classes; 2019.
[29] Ba JL, Kiros JR, Hinton GE. Layer normalization. http://arxiv.org/abs/1607.06450; 2016.
[30] Swiderska-Chadaj Z, de Bel T, Blanchet L, Baidoshvili A, Vossen D, van der Laak J, Litjens G. Impact of rescanning and normalization on convolutional neural network performance in multi-center, whole-slide classification of prostate cancer. Sci Rep 2020;10(1):1–14. https://doi.org/10.1038/s41598-020-71420-0.
[31] Lakshmi Mp. Image classification using machine learning techniques. Int J Adv Res 2019;7(5):1238–45. https://doi.org/10.21474/ijar01/9157.
[32] Alhichri H, Alswayed AS, Bazi Y, Ammour N, Alajlan NA. Classification of remote sensing images using EfficientNet-B3 CNN model with attention. IEEE Access 2021;9:14078–94. https://doi.org/10.1109/ACCESS.2021.3051085.
[33] Lin M, Chen Q, Yan S. Network in network. 2nd International Conference on Learning Representations, ICLR 2014;2014:1–10.
[34] Malla PP, Sahu S, Alutaibi AI. Classification of tumor in brain MR images using deep convolutional neural network and global average pooling. Processes 2023;11(3):1–17. https://doi.org/10.3390/pr11030679.
[35] Goodfellow I, Bengio Y, Courville A. Deep learning. MIT Press; 2016.
[36] Santoso IB, Adrianto Y, Sensusiati A, Wulandari D, Purnama I. Epileptic EEG signal classification using convolutional neural network based on multi-segment of EEG signal. International Journal of Intelligent Engineering and Systems 2021;14(3):160–76. https://doi.org/10.22266/ijies2021.0630.15.
[37] Santoso IB, Adrianto Y, Sensusiati AD, Wulandari DP, Purnama IKE. Ensemble convolutional neural networks with support vector machine for epilepsy classification based on multi-sequence of magnetic resonance images. IEEE Access 2022;10:32034–48. https://doi.org/10.1109/ACCESS.2022.3159923.
[38] Taheri S, Golrizkhatami Z, Basabrain AA, Hazzazi MS. A comprehensive study on classification of breast cancer histopathological images: binary versus multi-category and magnification-specific versus magnification-independent. IEEE Access 2024;12:50431–43. https://doi.org/10.1109/ACCESS.2024.3386355.
[39] Kingma DP, Ba JL. Adam: a method for stochastic optimization. 3rd International Conference on Learning Representations, ICLR 2015;2015:1–15.
[40] Xiao B, Liu Y, Xiao B. Accurate state-of-charge estimation approach for lithium-ion batteries by gated recurrent unit with ensemble optimizer. IEEE Access 2019;7:54192–202. https://doi.org/10.1109/ACCESS.2019.2913078.
[41] Deng J, Dong W, Socher R, Li L-J, Li Kai, Fei-Fei Li. ImageNet: a large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition; 2010. p. 248–55. https://doi.org/10.1109/cvpr.2009.5206848.
[42] Patil A, Rane M. Convolutional neural networks: an overview and its applications in pattern recognition. Smart Innovation, Systems and Technologies 2021;195:21–30. https://doi.org/10.1007/978-981-15-7078-0_3.
[43] Rokach L. Ensemble-based classifiers. Artif Intell Rev 2010;33(1–2):1–39. https://doi.org/10.1007/s10462-009-9124-7.
[44] Naimi AI, Balzer LB. Stacked generalization: an introduction to super learning. Eur J Epidemiol 2018;33(5):459–64. https://doi.org/10.1007/s10654-018-0390-z.
[45] Jiao Y, Du P. Performance measures in evaluating machine learning based bioinformatics predictors for classifications. Quantitative Biology 2016;4(4):320–30. https://doi.org/10.1007/s40484-016-0081-2.
[46] Dalianis H. Clinical text mining: secondary use of electronic patient records. Springer Cham; 2018. https://doi.org/10.1007/978-3-319-78503-5.
[47] Hicks SA, Strümke I, Thambawita V, Hammou M, Riegler MA, Halvorsen P, Parasa S. On evaluation metrics for medical applications of artificial intelligence. Sci Rep 2022;12(1):1–9. https://doi.org/10.1038/s41598-022-09954-8.
[48] Sokolova M, Japkowicz N, Szpakowicz S. Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation BT - AI 2006: Advances in artificial intelligence. Journal of the America Pomological Society 2006;59(1):1015–21. https://doi.org/10.1007/11941439_114.
[49] Shilaskar S, Mahajan T, Bhatlawande S, Chaudhari S, Mahajan R, Junnare K. Machine learning based brain tumor detection and classification using HOG feature descriptor. In: 2023 international conference on sustainable computing and smart systems (ICSCSS); 2023. p. 67–75. https://doi.org/10.1109/ICSCSS57650.2023.10169700.
[50] Atha Z, Chaki J. SSBTCNet: semi-supervised brain tumor classification network. IEEE Access 2023;11:141485–99. https://doi.org/10.1109/ACCESS.2023.3343126.
[51] Berahmand K, Daneshfar F, Salehi ES, Li Y, Xu Y. Autoencoders and their applications in machine learning: a survey. Artif Intell Rev 2024;57(2). https://doi.org/10.1007/s10462-023-10662-6. Springer Netherlands.
[52] Berahmand K, Saberi-Movahed F, Sheikhpour R, Li Y, Jalili M. A comprehensive survey on spectral clustering with graph structure learning. 2025. p. 1–21. http://arxiv.org/abs/2501.13597.
[53] Kementerian Kesehatan RI. Tumor otak. Pedoman Nasional Pelayanan Kedokteran 2017;92. http://kanker.kemkes.go.id/guidelines/PNPKOtak.pdf.