Brain Tumor Classification Through MRI Images Using DenseNet169 – Statistical Evaluation

[1] Anjali Jain, [2] Anjali Jain, [3] Rajesh Mehta
[1]Research Scholar, [2] Professor, [3] Assistant Professor
[1]USICT, GGSIPU, New Delhi
[2] USICT, GGSIPU, New Delhi
[3] CSED, Thapar Institute of Engineering and Technology, Punjab

[1] anj.1903.jain@gmail.com, [2] navin.rajpal@ipu.ac.in, [3] rajesh2010usit@gmail.com

ABSTRACT

According to world statistics, Cancer is the 10th leading cause of death in both men and women. Adding to this, the worldwide incidence rate of brain tumors was marked as 3.5 per 100,000 population(s) in 2020. One of the major reasons behind this alarming statistics is the high probability of tumors going undetected through the medical imaging practices in earlier stages of tumor development. This particular fact has drawn the attention of researchers working in the medical field to devise rapid and accurate diagnostic methods to reduce the mortality rate from brain tumors through identification at nascent stage and providing the requisite treatment. As digital image processing paves its way through Computer aided diagnostic in the medical field, exploring utility of deep learning techniques in medical image processing is a promising research avenue. This paper takes this cue to statistically evaluate DenseNet-169, a model based on DenseNet architecture as defined under Convolution Neural Network (CNN) for classifying Brain MRI images on the basis of presence or absence of brain tumor. Densenet-169 model along with a total of eight optimizers have been applied on publicly available brain tumor datasets to assess the performance of the said model in brain tumor classification. The performance of the model has been studied using a total of seven metrics and results obtained have been compared with the previous works in this field. The results obtained aid in concluding that DenseNet-169 with AdaDelta or AdaMax as the optimizer provide better results from the previous studies made. This in turn shall help in establishing DenseNet 169 as a model that can be effectively employed for tumor classification and extend the undertaken study further.

Keywords: Brain tumor, CAD, CNN, DenseNet, Deep learning, MRI, Medical image processing

INTRODUCTION

Brain tumor is a physical health condition which goes undetected for long and it can be diagnosed based on the symptoms which are shown by the patient. Symptoms of having a brain tumor may range from headaches, dizziness, and fainting tendency to hallucinations.

The first choice of diagnosis is never tumors due to which tumors have a prolonged existence in the human brain, which, in case, malignant, can prove life-threatening.

Radiological imaging is one of the most sought after way of diagnosing brain tumors. However, identifying tumors from imaging is time-consuming, challenging and tumors go undetected due to their small size or subjective evaluation of the radiologist in terms of classifying the detected matter as tumor or not. This calls for an automated diagnostic system which can work along with the radiologists in order to provide a ‘second opinion’ and thus leading to a more accurate and precise diagnosis [1].

These systems, being identified as, Computer Aided Diagnostic (CAD) systems are being derived from harnessing artificial intelligence techniques so as to minimize the human err due to subjective evaluation associated with huge amount of data that is collected for investigative purposes.

To generate an opinion in such a sensitive arena, the algorithmic approach being employed in computer aided diagnosis should be analogical similar to subjective evaluation done by human experts i.e. human mind shall be translated into systems. Researchers in the field of computer science have been working in this direction by improvising the existing automated diagnostic systems by employing the latest developments in the field of artificial intelligence.

Dramatic developments are being witnessed in the one of the subfield of Artificial Intelligence, namely, Machine learning due to which it has proven itself to be a major building block of analytical algorithms being developed for identification of tumors. However, as the data collected through the medical modalities namely MRI, CT, X-Ray, Ultrasound is enormous, the traditional methods of machine learning also fall short in analyzing the same [1][2].

Also, the medical images obtained for clinical purposes contain extremely valuable information which is required to be retained even after processing through the systems. For achieving this lossless output, efficient techniques for medical image processing are required [2].

This dire requirement of handling huge amount of sensitive data with extreme accuracy paved the way for deep learning techniques to be inculcated in the field of medical image processing and CAD systems. Unlike machine learning techniques, these Deep learning (DL) techniques found a mass popularity in the researcher community as all the feature extraction stages, feature selection and classification are automated in the model [3][4].

In this study, a pre-trained convolutional neural network based on DenseNet Architecture, namely, DenseNet -169, has been used to investigate MRI scans of brain tumors to analyze the effectiveness of the network model in classification of tumors. For evaluation of the network model, uniform datasets, data augmentation, hyper-parameter training and uniform optimizers have been used. The statistical effectiveness of the model has been derived through evaluation metrics namely Accuracy, Recall, Precision, F1 Score, Cohen-Kappa Score and Matthews correlation coefficient (MCC).

The reminder of this paper is organized as follows. Section 2 gives a brief of the materials and methods used in this study. Section 3 explains the results obtained out of this study and the inferences drawn from the same. The paper presents the conclusion and future work in section 5 and 6 respectively.

MATERIALS AND METHODS

A. Deep Learning

Deep Learning (DL) defined and designed as a subfield of Machine Learning which has a suite techniques inspired by neuroscience. However, it can be derived from [5] that primary source of inspiration for deep learning is not neuroscience. With the advent of different modalities of medical imaging that aid medical experts to draw diagnostic inferences, a huge amount of medical image data is being generated. And to achieve medical image analysis tasks such as detection and classification, deep learning algorithms have established themselves as a critical component of the whole modus operandi.

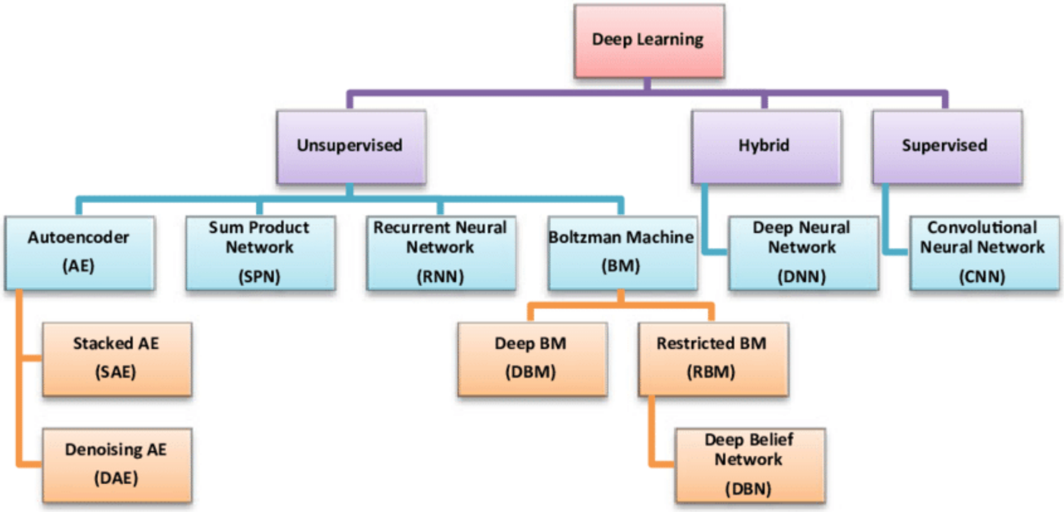


Figure-1. Classification of Deep learning techniques

As evident from Fig.1, there are a number of algorithmic approaches developed under the hood of deep learning with the base line capability of handling a large chunk of data to achieve the intended image processing operations. In terms of medical image processing operations related to brain tumors i.e. detection, classification, segmentation, CNNs have been recognized as the most utilized Deep learning algorithms.

B. CNN Architecture

Convolutional neural networks (CNN) are a powerful tool for learning the various representations of images and other structured data. A CNN is a type of artificial neural network which is able to preserve the spatial relationships in the data. It works on the concept of careful pruning of the connections between the nodes unlike the architecture of traditional neural network in which all nodes of one layer is connected to all the nodes in the next layer [6].

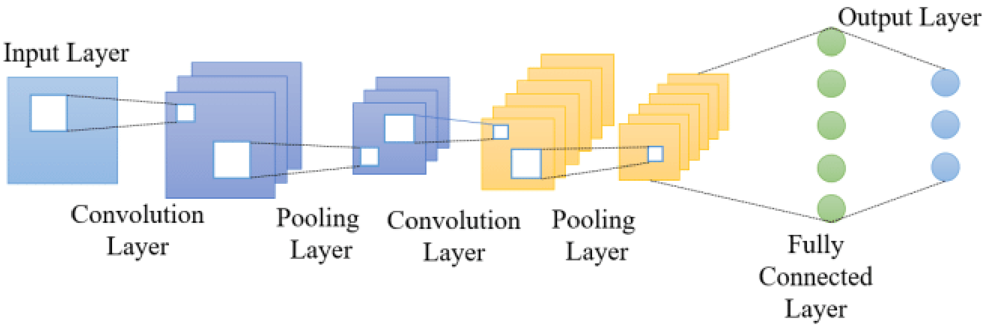


Figure-2. Basic architecture of CNN

These network models are capable of forming highly efficient representation of the data being fed as input in the form of grid structure. Each layer in the CNN works on a small region of the previous layer thus safeguarding the spatial relationships in the data. Following are the major building blocks of CNN architecture:

- Convolutional Layer
- Activation Layer
- Pooling
- Dropout Regularization
- Batch normalization

A CNN model consists of multiple layers of convolution and activation layers, with pooling layers sandwiched between the two. Also, CNNs have fully connected layers at the end to aid in computing the final classification results as output. Some of the well known CNN architectures are namely, AlexNet, ResNet, InceptionNet, DenseNet and many more. In this paper, focus has been drawn on pre-trained DenseNet architecture, specifically, Dense169 and efficacy of the approach in achieving brain tumor detection has been studied.

C. DenseNet

To overcome the problem of ‘vanishing gradient’ which was encountered in CNN networks, ResNet introduced the idea of ‘skip connections’ or ‘connectivity patterns’. But, the number of parameters in ResNet is very high and also every layer has its own weights to learn.

These drawbacks were catered through a new network model namely DenseNet [7]. This model harnessed the potential of reusing the features, thus eliminating redundant learning of feature maps. Also, this network simplified the ‘connectivity pattern’ thus improving the performance of the CNN. The major structural blocks of DenseNet architecture are as follows [8]:

- **Dense Block:** This basic building block of DenseNet architecture(s) consists of densely connected dense units with nonlinear mapping functions of Batch Normalization, Regularization Unit and Convolution. For making the training of the network smooth and performance better, a pre-activation strategy is implemented with the said functions. When input is provided to a particular dense block, it is firstly spliced and then it is merged with the outputs of the previous dense blocks in a particular dense network. As a result, all the new features that are generated shall be passed to the subsequent dense units.

  Thus, all the shallow features of the dense block are reused and are effectively utilized which leads to gradient disappearance to a certain extent. The major advantage of this construct is that by utilizing a small number of convolution kernels, a large number of features can be generated.

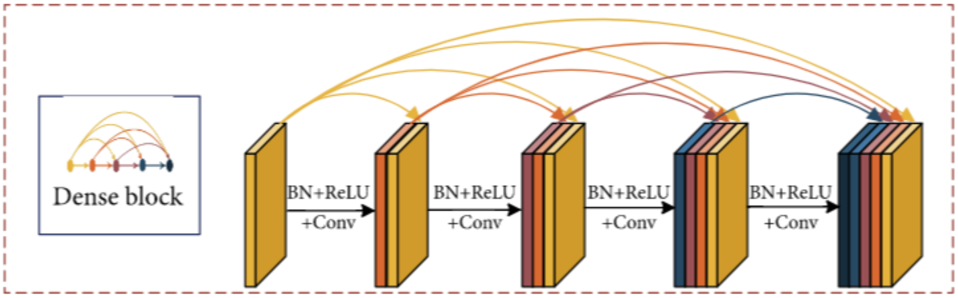


Figure-3. Dense Block in DenseNet

- **Transition Layer:** The structure residing between the adjacent dense blocks. It consists of 1×1 convolution and 2×2 average pooling layer. It leads to the compression of the input of the dense block and all the feature information that is extracted, reducing of the feature map size and dimensionality. This layer aids in mitigating over-fitting.

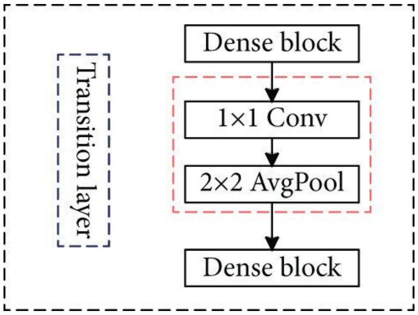
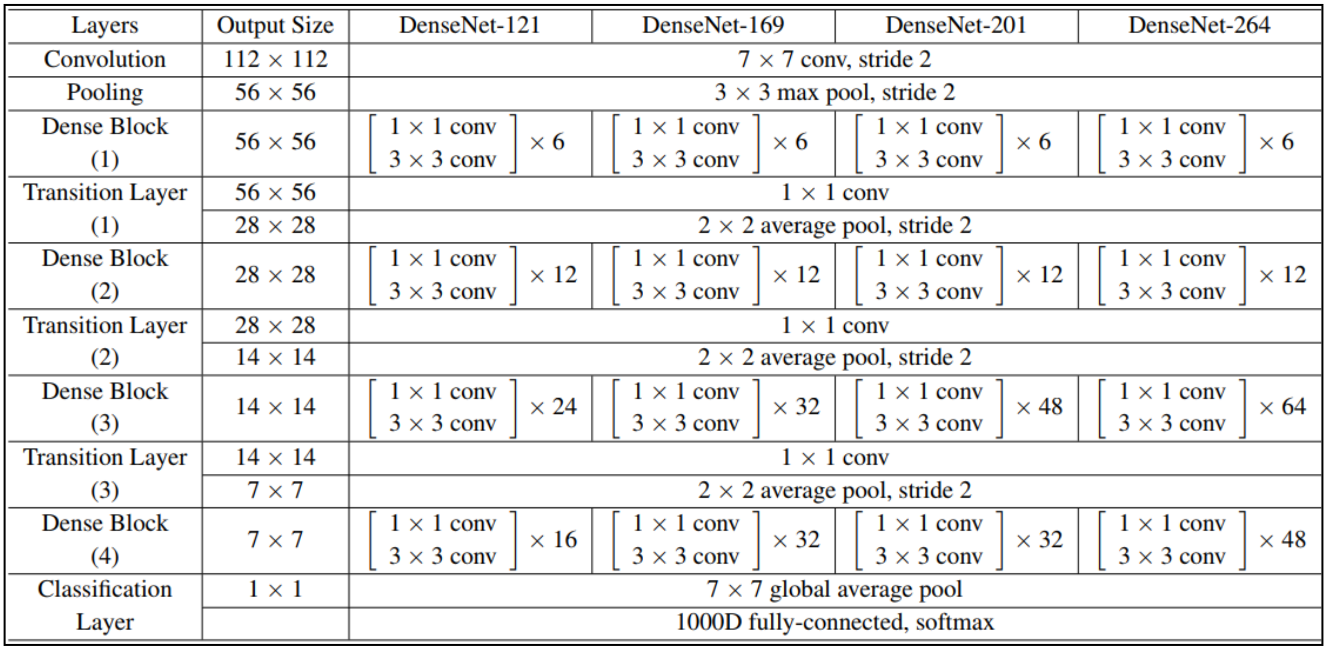


Figure-4. Transition Layer in DenseNet

- **Fully Connected Layer:** This layer is a classification prediction layer. It integrates the category feature information into network features. Then generated feature information is then classified after providing weights.

- **Convolution Layer:** Defined as the building block of a CNN architecture, this is the layer where the maximum computations take place. A convolution function can be defined as an application of a filter to an input that leads to activation.

  In case, the filter is applied repeatedly to an input, a map of activations i.e. feature map is created. It indicates the locations and strength of a detected feature in the input. This capability of convolution is harnessed in the convolution layer which enables CNN to learn a large number of filters in parallel, using which a highly specific features can be detected in the input and achieve predictive modeling problem, such as image classification.



Layers	Output Size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	112×112	7×7 conv, stride 2			
Pooling	56×56	3×3 max pool, stride 2			
Dense Block (1)	56×56	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	56×56	1×1 conv			
	28×28	2×2 average pool, stride 2			
Dense Block (2)	28×28	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	28×28	1×1 conv			
	14×14	2×2 average pool, stride 2			
Dense Block (3)	14×14	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 64$
Transition Layer (3)	14×14	1×1 conv			
	7×7	2×2 average pool, stride 2			
Dense Block (4)	7×7	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$
Classification Layer	1×1	7×7 global average pool			
		1000D fully-connected, softmax			

Figure-5. DenseNet Architectures

The evolution in the said network has led to the creation of improved architectures [9][10] as defined in Figure 5. In this paper, we are concentrating on the efficacy of DenseNet-169 to perform brain tumor classification and detection from the MRI images.

D. Image Acquisition

A: An open source dataset of Brain MRI scans for tumors containing a total of 3264 files. The dataset is classified into training and testing data. The data has been classified into respective tumor classes namely Glioma, Meningioma, Pituitary and No tumor. The dataset is available for download at https://github.com/sartajbhuvaji/brain-tumor-classification-dataset. The statistics of the dataset are as follows:

Class	Training data	Testing data	Total
Normal	395	105	500
Glioma	826	100	926
Meningioma	822	115	937
Pituitary	827	74	901

Table (1) - Dataset split

B: An open source dataset of Brain MRI scans for tumors containing a total of 7022 files. The dataset is classified into training and testing data. The data has been classified into respective tumor classes namely Glioma, Meningioma, Pituitary and No tumor. The dataset is available for download at https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset. The statistics of the dataset are as follows:

Class	Training data	Testing data	Total
Normal	1595	405	2000
Glioma	1321	300	1621
Meningioma	1339	306	1645
Pituitary	1457	300	1757

Table (2) - Dataset split

E. MRI Image Analysis

A. Image pre-processing

The images of the datasets mentioned under section 2.2 were resized to as 224 x 224 as the model input is a blob that consists of a single image of 1x3x224x224. Some sample images from the datasets are provided below.

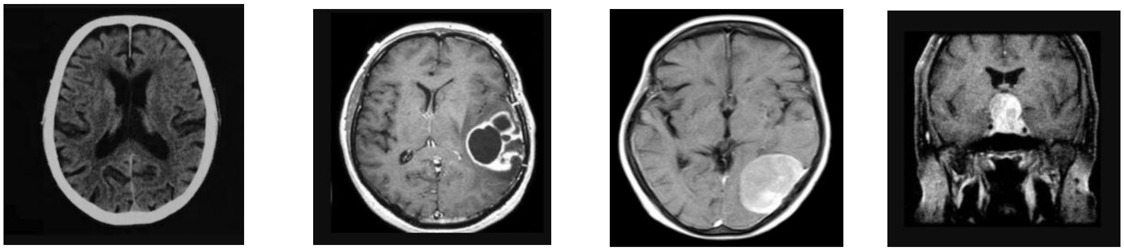


Figure-6. Sample images of Dataset A (from left to right) a) No tumor b) Glioma c) Meniongioma d) Pituitary

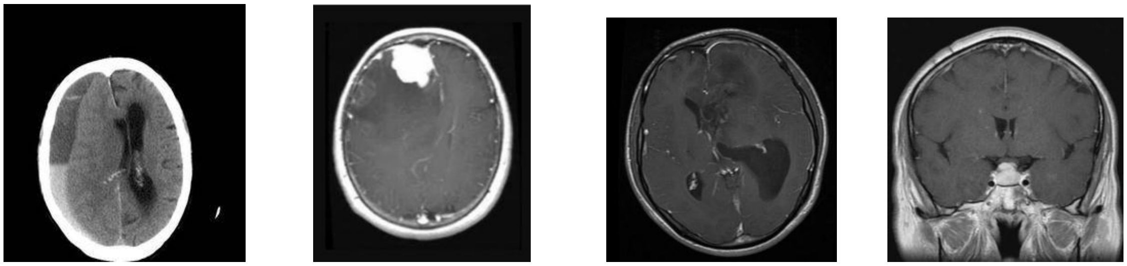


Figure-7. Sample images of Dataset A (from left to right) a) No tumor b) Glioma c) Meniongioma d) Pituitary

B. Data Augmentation

In order to avoid the over-fitting problem of the models, a variety of methods like dropout, adding regularization term are used. One such technique is data augmentation [11] [12] which is used to generate additional data for the models to train, test and validate the data. In this analysis, rotation, zoom, fill, shear, flipping, width and height shift have been utilized in order to achieve data augmentation for the datasets provided under section 2.2.

F. Parameter numbers and Hyper-parameters

Hyper-parameters are an essential part of deep neural network training along with image pre-processing. Table 3 provides the values of these parameters for this analysis.

Hyperparameter	**Options**
Parameters (trainable)	12,491,140
Cost function	Categorical Cross entropy
Epochs	50

Batch size	20
Optimizer	• **Adam** (learning_rate=0.01) • **SGD** (learning_rate=0.01,momentum=0.9) • **RMSprop** (learning_rate=0.01,momentum=0.9) • **Adadelta** (learning_rate=0.01,rho=0.95) • **Adagrad** (learning_rate=0.01) • **Adamax** (learning_rate=0.001, beta_1=0.9, beta_2=0.999) • **Nadam** (learning_rate=0.001, beta_1=0.9, beta_2=0.999) • **Ftrl** (learning_rate=0.001,learning_rate_power= 0.5,initial_accumulator_value=0.1, l1_regularization_strength=0.0, l2_regularization_strength=0.0, l2_shrinkage_regularization_strength=0.0)

Table (3) –Network Hyperparameterst

G. Network Training

The set of training images is split into training and validation with the split ratio being 4:1. The model calculates the loss function using categorical cross entropy and updates the weights by using different optimizers as mentioned in Table 3 in different iterations.

After the completion of all the epochs, performance metrics namely Accuracy, Loss, F1-score, Recall, Precision, Cohen-Kappa Score and Matthews Correlation Coefficient are calculated for training, validation and testing set of images.

Evaluation Metrics

The following metrics have been selected for statistically analyzing the performance of this paper.

- **Precision:** Ratio of the correctly predicted ‘Number of positives’ to the total number of positives predicted is called Precision. This metric provides a reflection of the extent to which the model is able to learn the positive characteristics of the sample. Higher the precision, more accurate is the prediction of the positive sample.

$$P=\frac{TP}{TP+FP}$$

where, TP = True positives

FP = False positive

- **Accuracy:** Ratio of the number of samples that have been correctly sampled to the total number of samples is termed as Accuracy. This metric reflects the performance of the model across all the classes.

$$A=\frac{TP+TN}{TP+FP+FN+TN}$$

where, TP = True positives

FP = False positives

TN = True negatives

FN = False Negatives

- **Recall:** This metric is the percentage of the number of positives that been predicted correctly to the actual number of positives. High value of this metric show that the higher amount of accuracy with which the target sample will be predicted. Also, it reflects that missing a bad sample is less likely.

$$R=\frac{TP}{TP+FN}$$

where, TP = True positives

FN = False Negatives

- **F1- Score:** This metric is defined as the harmonic mean of precision and recall.

$$F1=\left(2*\right)\frac{\mathrm{Pr}ecision*\mathrm{Re}call}{\mathrm{Pr}ecision+\mathrm{Re}call}$$

As a thumb rule, the weighted average of F1 should be used to compare classifier models instead of global accuracy.

$$WeightedF1=\sum_{i=1}^{N} w_i * F1_i$$

- **Cohen – Kappa Score:** This metric is defined in terms of 'reliability'. It is a quantitative measure of reliability for two evaluating entity or raters that are engaged in subjective interpretation of the same component which is corrected for the chances that the evaluating entities may agree by chance. k can take a negative value i.e. less than 0. If K is closer to 0, then it can be inferred that there is random agreement among raters, whereas, if K is closer to 1 than it can be inferred that there is a complete agreement between the raters.

$$k=\frac{p_o-p_e}{1-p_e}$$

where, k is Cohen's Kappa

P_o is probability of agreement [P_o = Number of Agreement/Total]

P_e is probability of random agreement [$P_o = P_{(correct)} + P_{(incorrect)}$]

- **Matthews Correlation Coefficient(MCC):** This metric is a single-value classification metric which summarizes the confusion matrix or an error matrix.

$$MCC=\frac{TN*TP-FN*FP}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

where, TP = True positives

FP = False positives

TN = True negatives

FN = False Negatives

MCC ranges between -1 and 1. If the value is closer to -1, then predictions are marked as random with respect to actual. And if the value is closer to 1 then it signifies agreement between actual and predictions.

RESULTS

For statistical evaluation of DenseNet-169 for classifying Brain tumor MRI images, the implementation and the execution were done in Kaggle Notebook using Tensorflow and Keras libraries.

It is understood that Epoch size is an important hyper-parameter; however, there is no clear or established criterion for training the networks. In this study, 50 have been chosen as the epoch size. GPU runtime of the framework was used to train, validate and test the models.

A total of eight optimizers have been investigated for each of the network by executing the model for each of the optimizers for the datasets. Cost function, Batch size and Epoch size has been taken as a constant for this study. Loss was calculated for the model and the model was evaluated using Accuracy, Loss, F1-score, Recall, Precision, Cohen-Kappa Score and Matthews Correlation Coefficient. Also, classification report was generated after complete run of the model to get a summarized insight in the performance as a whole.

Performance evaluation on Dataset A

Results of the pre-defined metrics for inferring the overall performance of DenseNet – 169 in brain tumor classification using datasets defined in section 2.2 have been recorded in Table 4.

Optimizer	Labels - Class	Precision (%)	Recall (%)	F1 – Score (weighted average)	Average Accuracy (%)	Cohen Kappa Score	Matthews Correlation Coefficient
Adam	0 – Glioma	52	69	62	62	0.4854	0.4902
	1 – Meningioma	66	50				
	2 – No tumor	65	61				
	3 – Pituitary	68	76				
SGD	0 – Glioma	72	68	69	69	0.5786	0.5789
	1 – Meningioma	58	60				

	2 – No tumor	61	69				
	3 – Pituitary	81	79				
RMSprop	0 – Glioma	63	55	63	55	0.407	0.4316
	1 – Meningioma	11	45				
	2 – No tumor	85	36				
	3 – Pituitary	76	78				
Adadelta	0 – Glioma	87	87	90	90	0.8646	0.8655
	1 – Meningioma	89	83				
	2 – No tumor	88	88				
	3 – Pituitary	95	93				
Adagrad	0 – Glioma	87	97	90	90	0.8646	0.8655

	1 – Meningioma	89	83				
	2 – No tumor	88	88				
	3 – Pituitary	95	93				
Adamax	0 – Glioma	84	97	87	87	0.8174	0.8205
	1 – Meningioma	77	85				
	2 – No tumor	90	83				
	3 – Pituitary	98	82				
Nadam	0 – Glioma	77	91	78	79	0.7076	0.721
	1 – Meningioma	94	63				
	2 – No tumor	60	73				
	3 – Pituitary	74	98				

Ftrl	0 – Glioma	~0	29	45	29	0	0
	1 – Meningioma	~0	~0				
	2 – No tumor	~0	~0				
	3 – Pituitary	~0	~0				

Table (4) - Results for Dataset A

Performance evaluation on Dataset B

Results of the pre-defined metrics for inferring the overall performance of DenseNet – 169 in brain tumor classification using datasets defined in section 2.2 have been recorded in Table 5.

Optimizer	**Labels - Class**	**Precision (%)**	**Recall (%)**	**F1 – Score (weighted average)**	**Average Accuracy (%)**	**Cohen Kappa Score**	**Matthews Correlation Coefficient**
Adam	0 – Glioma	76	95	87	87	0.8257	0.8275
	1 – Meningioma	80	76				
	2 – No tumor	98	88				
	3 – Pituitary	91	91				

SGD	0 – Glioma	68	69	74	76	0.6831	0.6879
	1 – Meningioma	78	59				
	2 – No tumor	92	93				
	3 – Pituitary	65	90				
RMSprop	0 – Glioma	32	91	59	60	0.4614	0.497
	1 – Meningioma	79	47				
	2 – No tumor	96	62				
	3 – Pituitary	28	83				
Adadelta	0 – Glioma	87	95	92	92	0.8881	0.8893
	1 – Meningioma	80	89				
	2 – No tumor	98	96				

	3 – Pituitary	99	97				
Adagrad	0 – Glioma	91	92	92	92	0.8888	0.8894
	1 – Meningioma	79	87				
	2 – No tumor	98	95				
	3 – Pituitary	97	91				
Adamax	0 – Glioma	97	97	97	97	0.9573	0.9575
	1 – Meningioma	92	96				
	2 – No tumor	98	99				
	3 – Pituitary	99	94				
Nadam	0 – Glioma	81	98	88	88	0.84	0.8437
	1 – Meningioma	92	75				

	2 – No tumor	82	95				
	3 – Pituitary	97	89				
Ftrl	0 – Glioma	~0	29	45	29	0	0
	1 – Meningioma	~0	~0				
	2 – No tumor	~0	~0				
	3 – Pituitary	~0	~0				

Table (5) - Results for Dataset B

DISCUSSION

The results for the proposed study have been provided under section 3. As summarized under section 2.6, Accuracy, Loss, F1-score, Recall, Precision, Cohen-Kappa Score and Matthews Correlation Coefficient have been used for evaluating the CNN architecture for a comprehensive inference.

For Dataset A, the highest average accuracy of 90% was obtained when optimizers were set as AdaGrad or AdaDelta. For Dataset B, For Dataset B, highest average accuracy of 97% was obtained when optimizer was set as AdaMax.

For validating the obtained results further, comparison between accuracy of proposed classifier and previous works on same datasets was carried out. Four models as mentioned in Table 6 were evaluated for their performance in [13] on Dataset B as taken in this particular study. As it can be inferred from the results, DenseNet-169 outperforms the other models for the same dataset.

Model	Accuracy (in%)
EfficientNetB7	88.18
EfficientNetV2B1	89.17
EfficientNetB1	89.55
ResNet50	79.32
DenseNet169	**97**

Table (6) - Comparison with related work based on Dataset B

Further, the pre-trained model studied in this paper was applied to the dataset used in [14] to establish the results obtained. A classification accuracy of 93% was obtained by the CNN model proposed in [14]. When DenseNet-169 with AdaDelta as the optimizer was applied to the same dataset, an accuracy of 97% was obtained.

Thus, it can be inferred from the independent and comparative results that the DenseNet-169 with either AdaDelta or AdaMax as its optimizer has sheer potential of achieving an average classification accuracy of 92% - 97%.

CONCLUSION

DenseNet-169 as a part of deep learning architectures was studied with respect to its performance in classifying the brain MRI scans on the basis of presence or absence of tumor. A statistical evaluation and analysis was done for the said network model with respect to different optimizers and two different datasets.

The results obtained can aid the researchers working in this field to understand the metrics and conditions under which this network give its best performance which can then be used to design better computer aided diagnostic system using this deep learning architecture for detection of tumors from MRI images. The study taken herewith has explored relatively small data regime with random weights from which it can be concluded that DenseNet 169 can prove to be an efficient and effective approach for achieving data-intensive analytical results if deployed with an appropriate set of parameters (e.g. optimizers, batch size).

FUTURE WORK

It is proposed to extend and validate the current study by evaluating the model on other publicly available datasets. Also, to gather and establish more concrete results, the said model shall be compared with other CNN architectures available in the literature by performing the same evaluation methodology with same datasets and hyper-parameters. This will give a better insight in the overall efficiency and supremacy of the architectural marvel of the studied model.

Additionally, as in the current study, default values of the parameters have been considered, the future work will consist of producing results by providing variations in the parametric values to identify the best configuration(s) for the network.

Funding and/or Conflicts of interests/Competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

[1] H.-C. Shin et al. “Deep convolutional neural networks for computer aided detection: CNN architectures, dataset characteristics and transfer learning”, *IEEE transactions on medical imaging*, Vol.35, pp. 1285–1298, 2016.

[2] D. Ravi et al. “Deep learning for health informatics”, *IEEE journal of biomedical and health informatics*, Vol. 21, pp. 4–21, 2017.

[3] D. S. Kermany et.al, “Identifying medical diagnoses and treatable diseases by image-based deep learning”, *Cell*, vol. 172, pp.1122–1131, 2018.

[4] G. Litjens, et. al “A survey on deep learning in medical image analysis”, *Medical image analysis*, Vol.42, pp. 60–88, 2017.

[5] I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016.

[6] A. Krizhevsky, I. Sutskever, Geoffrey E. Hinton "Imagenet classification with deep convolutional neural networks.", *Communications of the ACM*, Vol. 60, pp.84-90, 2017.

[7] MZ Alom et.al , *The history began from alexnet: A comprehensive survey on deep learning approachs*, 2018.

[8] Tao Zhou et. al, “Dense Convolutional Network and Its Application in Medical Image Analysis”, *BioMed Research International*, vol.2022, 2022.

[9] G Huang et.al, "Densely connected convolutional networks." *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700-4708. 2017.

[10] C. Szegedy, et. al “Going deeper with convolutions”, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1-9, 2015.

[11] K. He et al., “Deep residual learning for image recognition”, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778, 2016.

[12] C. Shorten, T.M. Khoshgoftaar, “A survey on image data augmentation for deep learning,” *Journal of Big Data*, Vol. 6, pp.60, 2019.

[13] D. Filatov, Ahmad Hassan Yar, G. N, “Brain Tumor Diagnosis and Classification via Pre-Trained Convolutional Neural Networks”, *medRxiv*, Vol. 7, 2022.

[14] D. C. Febrianto, I. Soesanti, H. A. Nugroho, "Convolutional neural network for brain tumor detection," *IOP Conference Series: Materials Science and Engineering*, Vol. 771, No. 1, 2020.

[15] E. Irmak, "Multi-classification of brain tumor MRI images using deep convolutional neural network with fully optimized framework." *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, Vol. 45, No.3, pp 1015-1036, 2021.

[16] S. Deepak, P.M. Ameer, “Brain tumor classification using deep CNN features via transfer learning”, *Computers in Biology and Medicine*, 2019**.**