Undergraduate Conference on Intelligent Computing and Systems (UCICS 2025)
26-27 February, 2025; Varendra University, Rajshahi, Bangladesh

Brain Stroke Detection with Machine Learning on CT-Scan Images

MD. Nahid Hasan
Computer Science and Engineering
Varendra University
Rajshahi, Bangladesh
securenahid@gmail.com

Salma Anam Jasy
Computer Science and Engineering
Varendra University
Rajshahi, Bangladesh
salma0anaam@gmail.com

Shrabana Rani Chakrabartty
Computer Science and Engineering
Varendra University
Rajshahi, Bangladesh
shrabanaranichakrabartty28may@gmail.com

Kazi Mahmudul Hasan
Computer Science and Engineering
Varendra University
Rajshahi, Bangladesh
shafi16221@gmail.com

Md. Fatin Ilham
Computer Science and Engineering
Varendra University
Rajshahi, Bangladesh
mdfatin.ilham@gmail.com

Md. Musfiqur Rahman Mridha
Computer Science and Engineering
Varendra University
Rajshahi, Bangladesh
mdmridha100730@gmail.com

***Abstract*—Day by day brain diseases are increasing because people are now focusing on their work too much not focus on health. Stroke is also a serious brain disease which commonly occurs at any age, people. The severity can decrease by detecting early and starting treatment early. This model with ML that will detect Stroke from a CT-Scan image that a brain stroke or not. Use Laplacian, CLAHE as image preprocessing and use Feature extractors DenseNet169, DenseNet121, DenseNet201, VGG16, Xception with several classifier SVM, CustomClassification, XGBoost, Logistic Regression. After that got a truly best result 99.67% which was rare on this field and previous work. At present, hope this result is the highest in the basis of accuracy.**

Index of Terms- Introduction, Related Work, Methodology, Result and Discussion, Conclusion.

I. Introduction

People are under pressure about their job, study and work. But they don't know about that this pressure will lead them to severe health issues like stroke. A stroke is a situation because of hypertension and diabetics interrupt blood flow to the brain, leading to cell death and neurological impairment. According to cdc.gov Only in USA 795000 have a stroke in a single year [1].

Fig. 1 shows how many people had strokes in selected countries around the world in 2016 [2]. Worldwide 1099.310 person have stroke per 100,000 persons in 2021 [3]. Day by day it's become a burden for human beings. If it is to diagnose this serious disease as soon as possible, it can be efficiently treated and reduce long-term disability. Machine Language (ML) can be one of the great parts of this process. There are several types of processes that can be used to identify the stroke or not. Basically, it can be determined by the behavior of the patient but the actual identification of stroke or not is very important. To identify this bring up with a solution with deep learning and machine learning technology with an accuracy is 99% on CT-Scan Image datasets.

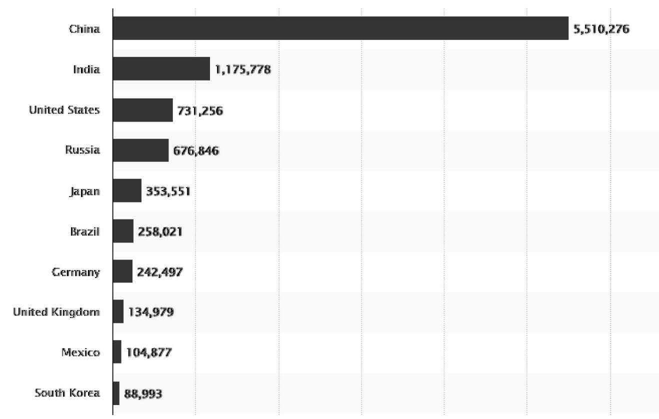

Fig. 1. Strokes in selected countries on 2016

II. Related Work

Brain Stroke detection is a very significant work for patient but there is some few work about it with low accuracy.

Uppal et al. have Enhancing accuracy in brain stroke detection using RMSProp and Adadelta. They used Stroke Prediction Dataset, 2021 from Kaggle. They achieve 94.9% accuracy of testing [4].

Chaki et al. implement Artificial Intelligence Brain Stroke Detection using CNN model by used 1320 EEG data samples. Obtain 54 times out of 88 times accuracy, Recall: 52.27%, Precision: 35.23%, F1-Score: 53.41% [5].

Srinivas et al. have stroke detection model with Random Forest model. They used UCI machine learning repository as dataset. Obtain accuracy of 96.88% [6].

Ahmed et al. have Brain Stroke Detection model with 3D CNN. Accuracy 92.5%. Used Computed Tomography Scan images signal [7].

Kaya et,al, had Brain Stroke Detection from CT Images with ResNetl01 with the 97.93% accuracy used computerized tomography images [8].

III. Methodology

In this section, we describe the methodology of our study. In Fig. 2, the methodology is demonstrated with a diagram.

A. Data Collection

For this, collected CT scan images from Kaggle, Brain Stroke CT Image Dataset [9], where publicly available ensuring a balanced dataset of stroke-positive and stroke-negative cases. That shows on TABLE I. Additionally, a data sample is provided in Fig. 3.

TABLE I
DATASET DESCRIPTION

Dataset Name	Normal	Stroke	Total
Brain Stroke CT Image	1550	950	2500

B. Prepossessing

After that, we modified the data set for us and started the prepossessing technique. The prepossessing technique we used CLAHE (Contrast Limited Adaptive Histogram Equalization) and Laplacian. Laplacian is a second order derivative operator in image processing that highlights regions of rapid intensity change, commonly used for edge detection. CLAHE is an image enhancement technique that improves local contrast while preventing over-amplification of noise.

C. Data Splitting

The data set has a total of 2500 images we have split that with the ratio in 80:20. 80% of our data are used for training and 20% data are used for testing.

D. Feature Extraction

With that, we have used Feature Extraction. It Identifies and extracts meaningful patterns or features from an image (like edges, textures, or shapes) that represent its important characteristics for further analysis. Mostly Use DenseNet169, DenseNet121, DenseNet201, VGG16, Xception which are very effective on my dataset images.

E. Classification

After that classify the images With SVM (Support Vector Machine), XGBoost, Logistic Regression and a custom classifier which builds a binary classification model using TensorFlow/Keras with a pre-trained DenseNet backbone. Custom layers like global average pooling, dense layers, batch normalization, and dropout are added, ending with a sigmoid output. The model is compiled with the Adam optimizer, binary cross-entropy loss, and a learning rate scheduler. Class weights handle imbalanced data, while callbacks like early stopping, model checkpointing, and a custom metrics callback ensure better performance monitoring. The model is trained and evaluated using metrics like precision, recall, F1 score, and accuracy to ensure reliable and robust results. Precision, recall, F1 score, and accuracy are evaluation metrics used to measure a model's performance. We ensure reliable results by assessing correctness (precision), sensitivity (recall), balance (F1 score), and overall accuracy in predictions.

IV. Result and Discussion

After the completion of Fig.-3 full process, we got a effective results on detection. We produced Confusion Matrix Plot, ROC Curve, Precision-Recall Curve and F1 scores from our model. With custom clasifier, XGBoost, Logistic Regression, SVM and combination of feature extrator like DenseNet169, DenseNet121, DenseNet201, VGG16, Xception produced all the result. DenseNet121 with the combination of custom classifier we got 99.19% Accuracy, Precision 99.20%, Recall 99.19%, F1 Score 99.19%, AUC 99.98% This value don't satisfy us so we go for next, DenseNet201 and custom classifier produced 98.87% Accuracy, Precision 98.90%, Recall 98.87%, F1 Score 98.87%, AUC 99.97% and VGG16 with custom classifier satisfy us with 99.67% Accuracy, Precision 99.68%, Recall 99.68%, F1 Score 99.68%, AUC 99.93% this preaty much better from other result. The accuracy is 99.67% That is pretty much rare in this field. On our dataset we got a perfect result with VGG16 and Custom Classifier, which will make separate this research paper from other. The Confusion Matrix measure the effectiveness of ML on our separate data set like class 1 and class 0. Our Overall Output from all Combination Are shown in TABLE II. TABLE III shows that this paper providing better result then other works. The confusion matrix is provided in Fig. 4. Additionally, A set of curves is provided in Fig. 4 (receiver operating characteristic (ROC) curve, precision vs recall curve) and Fig. 5 (loss vs learning rate curve, precision, recall, and f1-score vs epochs curve, training and validation accuracy and loss curve).

V. Conclusion and Future Work

Our future work will be more interesting. We will add an AI feature, that can predict the level of stroke, and how serious condition the patient is. This paper presents an efficient brain stroke detection model using machine learning on CT-scan images. By using CLAHE, Laplacian preprocessing, and deep learning feature extractors (DenseNet, VGG16, Xception) with various classifiers (SVM, XGBoost, Logistic Regression, and a

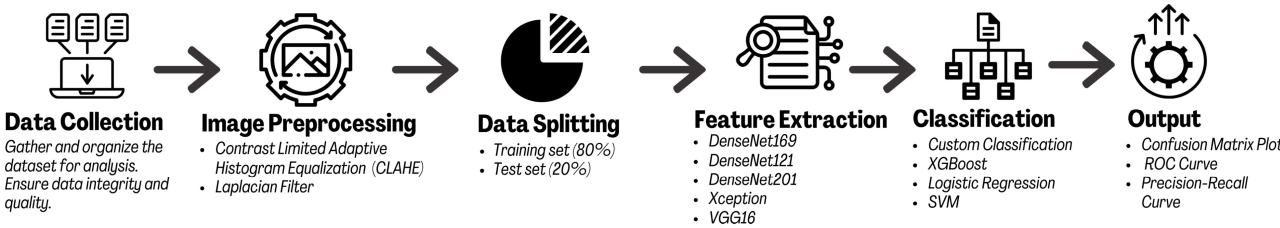

Fig. 2. Process Workflow

TABLE II
PERFORMANCE COMPARISON OF DIFFERENT FEATURE EXTRACTORS AND CLASSIFIERS

Feature Extractor	Classifier	Accuracy	Precision				Recall				F1 Score			AUC
			Class 0	Class 1	Mac. AVG	Wtd AVG	Class 0	Class 1	Mac. AVG	Wtd AVG	Class 0	Class 1	Mac. AVG	
DenseNet169	Custom Classifier	0.9903	0.9872	0.9935	0.9904	0.9904	0.9936	0.9871	0.9903	0.9903	0.9904	0.9903	0.9903	0.9997
	SVM	0.9662	0.9618	0.9707	0.9662	0.9662	0.9711	0.9613	0.9662	0.9662	0.9664	0.9660	0.9662	0.9935
	XG Boost	0.9195	0.9223	0.9167	0.9195	0.9195	0.9164	0.9226	0.9195	0.9195	0.9194	0.9196	0.9195	0.9818
	LR	0.9275	0.9433	0.9128	0.9281	0.9281	0.9100	0.9452	0.9276	0.9275	0.9264	0.9287	0.9275	0.9747
DenseNet121	Custom Classifier	0.9919	0.9873	0.9967	0.9920	0.9920	0.9968	0.9871	0.9919	0.9919	0.9920	0.9919	0.9919	0.9998
	SVM	0.9662	0.9801	0.9530	0.9666	0.9666	0.9518	0.9806	0.9662	0.9662	0.9657	0.9666	0.9662	0.9945
	XG Boost	0.9259	0.9206	0.9314	0.9260	0.9260	0.9325	0.9194	0.9259	0.9259	0.9265	0.9253	0.9259	0.9820
	LR	0.8873	0.9199	0.8593	0.8896	0.8896	0.8489	0.9258	0.8873	0.8873	0.8829	0.8913	0.8871	0.9544
DenseNet201	Custom Classifier	0.9887	1.0000	0.9779	0.9890	0.9890	0.9775	1.0000	0.9887	0.9887	0.9886	0.9888	0.9887	0.9997
	SVM	0.9581	0.9582	0.9581	0.9581	0.9581	0.9582	0.9581	0.9581	0.9581	0.9582	0.9581	0.9581	0.9897
	XG Boost	0.9549	0.9609	0.9490	0.9550	0.9550	0.9486	0.9613	0.9549	0.9549	0.9547	0.9551	0.9549	0.9901
	LR	0.9388	0.9535	0.9250	0.9392	0.9393	0.9228	0.9548	0.9388	0.9388	0.9379	0.9397	0.9388	0.9857
VGG16	**Custom Classifier**	**0.9967**	**0.9968**	**0.9968**	**0.9968**	**0.9968**	**0.9968**	**0.9968**	**0.9968**	**0.9968**	**0.9968**	**0.9968**	**0.9968**	**0.9993**
	SVM	0.9050	0.9200	0.8910	0.9055	0.9055	0.8875	0.9226	0.9050	0.9050	0.9034	0.9065	0.9050	0.9668
	XG Boost	0.9452	0.9453	0.9452	0.9452	0.9452	0.9453	0.9452	0.9452	0.9452	0.9453	0.9452	0.9452	0.9870
	LR	0.8406	0.8732	0.8131	0.8431	0.8432	0.7974	0.8839	0.8406	0.8406	0.8336	0.8470	0.8403	0.9158
Xception	Custom Classifier	0.9887	0.9967	0.9810	0.9888	0.9889	0.9807	0.9968	0.9887	0.9887	0.9887	0.9888	0.9887	0.9994
	SVM	0.9469	0.9399	0.9541	0.9470	0.9470	0.9550	0.9387	0.9468	0.9469	0.9474	0.9463	0.9469	0.9887
	XG Boost	0.8647	0.8650	0.8645	0.8647	0.8647	0.8650	0.8645	0.8647	0.8647	0.8650	0.8645	0.8647	0.9404
	LR	0.8744	0.8976	0.8537	0.8756	0.8757	0.8457	0.9032	0.8744	0.8744	0.8709	0.8777	0.8743	0.9270

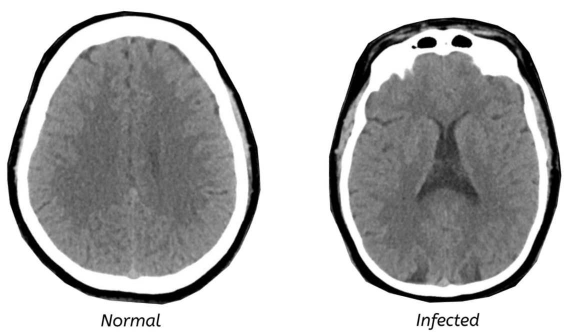

Fig. 3. Data Set Sample

TABLE III
PERFORMANCE METRICS COMPARISON

	Accuracy	Recall	Precision	F1-Score
Chaki et al. [5]	61.36%	52.27%	35.23%	53.41%
Uppal et al. [4]	95.8%	Not Given	95%	Not Given
My Work	**99.67%**	**99.68%**	**99.68%**	**99.68%**

Custom Classifier), our model achieved an impressive 99.67% accuracy. The results demonstrate state-of-the-art performance in stroke detection. Working on the dataset that can detect the stroke from MRI Images. Hope our model will get perfect results with the best accuracy. It can be a great tool for personal diagnosis. It can be worked with a large number of data at a time. For any campaign or where we need the result faster then we can freely use it there. It can make the medical system modern and faster.

REFERENCES

[1] CDC, "Stroke facts," Stroke, 24-Oct-2024. [Online]. Available: https://www.cdc.gov/stroke/data-research/facts-stats/index.html. [Accessed: 12-Jan-2025].

[2] "Number of stroke incident cases in select countries worldwide in 2016," Statista. [Online]. Available: https://www.statista.com/statistics/1105695/stroke-incident-cases-for-select-countries-worldwide/. [Accessed: 12-Jan-2025].

[3] Q. He et al., "Global, regional, and national burden of stroke, 1990–2021: A systematic analysis for Global Burden of Disease 2021," Stroke, vol. 55, no. 12, pp. 2815–2824, 2024.

[4] M. Uppal et al., "Enhancing accuracy in brain stroke detection: Multi-layer perceptron with Adadelta, RMSProp and AdaMax optimizers," Front. Bioeng. Biotechnol., vol. 11, p. 1257591, 2023.

[5] J. Chaki and M. Woźniak, "Deep learning and artificial intelligence in action (2019–2023): A review on brain stroke detection, diagnosis, and intelligent post-stroke rehabilitation management," IEEE Access, vol. 12, pp. 52161–52181, 2024.

[6] A. Srinivas and J. P. Mosiganti, "A brain stroke detection model using soft voting based ensemble machine learning classifier," Measur. Sens., no. 100871, p. 100871, 2023.

[7] Y. Ahmed, V. Haldar, T. Singh, and A. Saini, "Brain stroke detection using 3D CNN," in 2024 2nd International Conference on Disruptive Technologies (ICDT), 2024, pp. 784–788.

[8] N. Çınar, B. Kaya, and M. Kaya, "Brain stroke detection from CT images using transfer learning method," in 2023 13th International Conference on Advanced Computer Information Technologies (ACIT), 2023.

[9] Kaggle.com. [Online]. Available: https://www.kaggle.com/datasets/afridirahman/brain-stroke-ct-image-dataset. [Accessed: 14-Jan-2025].

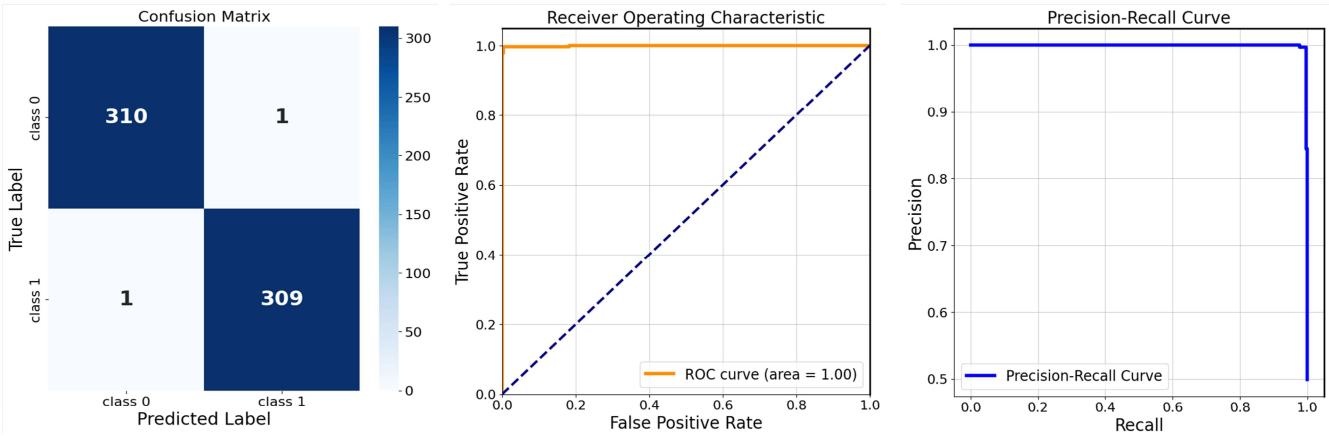

Fig. 4. Confusion Matrix, ROC, Precision-Recall Curve

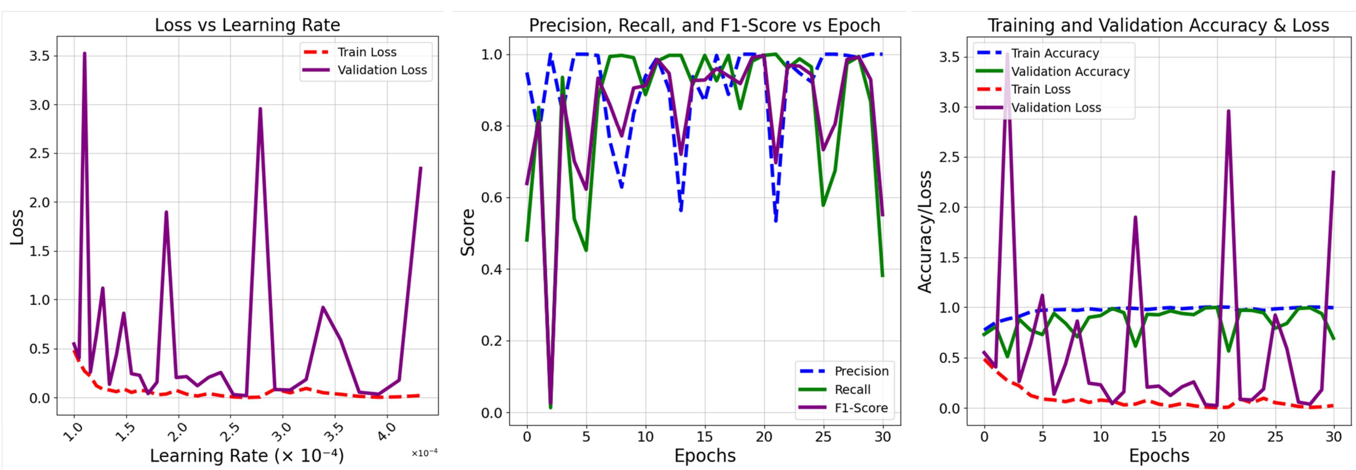

Fig. 5. Training and Validation, Loos vs Learning, Precision, Recall and F1-Score vs Epoch Curve