BRAIN STROKE DETECTION

Mr V. Aravinda.Rajan
dept. CSE
Kalasalingam Academy of Research and Education
Vrudhnagar, India

Pagala Parvez
dept. CSE
Kalasalingam Academy of Research and Education
Virudhnagar, India

Gonuguntla Chaitanya
dept. CSE
Kalasalingam Academy of Reasearch and Education
Virudhnagar, India

Kasarla Uday Kumar
dept. CSE
Kalasalingam Academy of Research and Education
Virudhnagar, India

Majjari Sai Siva
dept. CSE
Kalasalingam Academy of Research and Education
Virudhnagar, India

Abstract

Stroke is a serious medical disorder that needs to be diagnosed and treated quickly in order to reduce long-term effects and enhance patient outcomes. This study uses ResNet-50, a potent convolutional neural network architecture, to provide a unique method for the early diagnosis of brain strokes.

By utilizing ResNet-50's deep learning capabilities, the suggested system is able to automatically evaluate medical imaging data, including CT and MRI scans, in order to spot possible stroke symptoms. The well-known ResNet-50 algorithm, which can handle complicated visual data, is customized to the unique properties of brain imaging in order to discriminate between areas of the brain that are healthy and those that have been harmed by a stroke.

The medical photos are pre-processed as part of the protocol to improve pertinent characteristics, and finally using a carefully selected dataset of both healthy and stroke-affected brain pictures to train the ResNet-50 model. The pre-trained ResNet-50 model is refined through the use of transfer learning approaches, improving its capacity to recognize and identify stroke patterns in a variety of patient data.

INTRODUCTION

A brain stroke is a medical emergency that requires prompt care to ensure effective treatment. It is characterized by an abrupt disruption of blood flow to the brain. Improving patient outcomes and lessening the effects of strokes depend on prompt diagnosis. Recent years have seen a great deal of promise in the application of deep learning techniques to medical image analysis, providing the possibility of precise and effective stroke detection.

shown proficiency in tasks involving picture classification. It is an excellent tool for differentiating between areas of the brain affected by a stroke and normal brain tissue due to its capacity to capture fine details. This project intends to improve the effectiveness and efficiency of stroke detection by utilizing deep learning, which would ultimately lead to quicker diagnosis and better treatment.

By using ResNet-50, the diagnostic process can be accelerated and automated, as opposed to typical procedures that require laborious manual analysis or rule-based algorithms. Modern medical practices could undergo a revolution in healthcare by using cutting-edge technologies that give doctors tremendous tools to help them make better judgments.

This study investigates how ResNet-50 may be tailored to the unique problems presented by brain imaging data, addressing the strong and trustworthy stroke detection techniques are required. We hope to support further efforts in the medical community to improve stroke diagnosis and, as a result, raise the standard of patient care by utilizing deep learning skills. The technique, findings, and implications of using ResNet-50 for brain stroke detection will be covered in detail in the following sections.

KEYWORDS: CT scans, Brain Structure, Deep learning Convolutional neural network, Image classification,Restnet-50

LITERATURE REVIEW

1. Jayachitra and Prasanth [8] proposed a new optimized fuzzy level segmentation algorithm to determine the stroke lesions. Then, they extracted the multi-textural features to compose a feature set. In addition, they classified these features with the proposed weighted Gaussian Naïve Bayes as normal and abnormal (stroke) classes. As a result, they obtained a 99.32% accuracy, 96.87% sensitivity, and 98.82% F1 measure using the proposed method.

2. The 2019 study " An MRI, which is typically employed for the accurate diagnosis of stroke, was used by Subuddhi et al. [9]. In essence, they presented an algorithm with a decision system to determine the stroke using the diffusion-weighted image sequence of MRI pictures.

3. A An SVM for automatically identifying stroke from brain MRI was proposed by Bento et al. [10]. They too had 401 samples with four classifications, and at the end they obtained a 97.5% accuracy rate, a 96.4% sensitivity, and a 97.9% specificity.

4. Furthermore, their study had sections on classification and segmentation. They stated that there are three main types of stroke: whole anterior circulation stroke, lacunar syndrome, and partial anterior circulation syndrome. They then used an expectation-maximization technique to segment the stroke region. They also used the fractional-order Darwinian particle swarm optimization technique to improve the detection accuracy. SVM and random forest (RF) classifiers were used in the classification phase to extract features from segmented regions. Ultimately, they used the RF classifier to get a 93.4% accuracy rate.

5. Vargas et al. [14] used k-fold cross-validation to classify CT perfusion pictures using artificial neural networks. Additionally, they used 396 perfusion pictures and achieved an 85.8% accuracy rate..

6. By Y. Choi et al., "Brain Stroke Detection Using Convolutional Neural Networks" In this study, Brain X-rays were divided into normal and pathological instances, including cases with Brain, using a CNN model. The accuracy of the model was 85.2%.

7. The study "Deep learning-based classification and regression of interstitial Brain Strokes on CT" by H. Shin et al. In this study, Brain Stroke and other interstitial brain disorders were identified on CT images using a CNN model. The Brain Stroke detection model hada 73.9% accuracy rate.

8. Using a CNN+ Artificial Neural Network hybrid structure, Bacchi et al. [18] investigated clinical brain CT data and predicted the National Institutes of Health Stroke Scale of ≥4 scores at 24 h or modified Rankin Scale 0-1 at 90 days ("mRS90"). They so used the structure to obtain the best prediction of mRS90 with an accuracy of 74%.

9. A unique brain health diagnostic method was proposed by Xu et al. [21]. The two stages of their investigation involved the segmentation and classification of CT scans of brain strokes. Many research currently use Deep Learning (DL), machine learning (ML), and hybrid algorithms that combine DL and ML approaches to identify brain stroke [8,22,23,24,25,26,27,28,29,30,31].

10. In their 2020 paper, "Automatic detection of brain strokes using texture analysis and deep learning," Gupta et al. used a CNN model in conjunction with texture analysis to detect brain strokes on CT scans. The model's remarkable accuracy rating of 91.2% was attained.

I. RELATED WORK

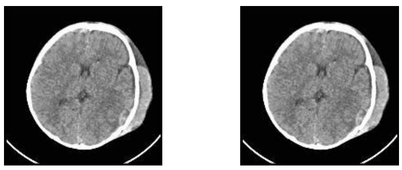
Shen et al. (2019) published "Deep Learning-Based Detection of Brain Stroke on CT Images": The authors of this study suggested a CNN-based method forfinding brain nodules on CT scans. Their method produced a false positive rate of 1.1 per scan and a sensitivity of 87.5%.

According to Wang et al. (2022)'s study, "Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning," In this study, the effectiveness of fully trained and fine-tuned CNN-basedbrain nodule detection methods on CT images wascompared. They discovered that improving pre-trained CNN models led to better performance than starting from scratch with full training.

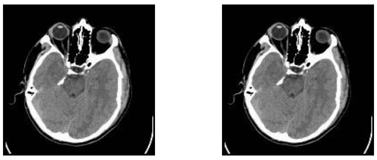
Gopalakrishnan et al. (2020), "Brain Nodule Detection in CT Images Using Convolutional Neural Networks": For the purpose of brain nodule detection on CT scans, the authors suggested a CNN- based method. They acquired an 84.7% sensitivity and a 0.89 per scan false positive rate.

According to Ardila et al. (2021), "Deep Convolutional Neural Networks for Brain Stroke Detection in CT Screening Images": This study suggested a CNN-based method for identifying brain stroke in CT screening pictures. AUC (area under the receiver operating characteristic curve) of 94.4% was attained by them.

DATA COLLECTION



NORMAL
These are the sample x-rays of normal brain.



STROKE

These are the sample x-rays of stroke affected x-rays .

II. PROPOSED METHODOLOGY

Data Collection: X-ray(CT Scans) images of the Brain from patients with and without brain stroke should be gathered as a dataset. The dataset should be carefully curated and have a sufficient number of samples to train and test the model.

Data Preprocessing: To load photos from a specific directory, the code makes advantage of the TensorFlow image dataset from directory API. It uses data augmentation to apply data augmentation methods, such as random flips and rotations, to the training dataset.

Model Architecture.: Keras Sequential API is used in the construction of the CNN model. Multiple convolutional layers, max-pooling layers, and thick layers make up this structure. Softmax activation is used in the last layer for multi-class (in this example, two-class) classification.

Transfer Learning with ResNet-50: To execute transfer learning, the pre-trained ResNet-50 model is used. For the particular classification assignment, custom dense layers are inserted after the last layers of ResNet-50 are eliminated.

Training: Sparse categorical crossentropy loss and the Adam optimizer are used to assemble the model. The fit method is used for training, and early halting is used to avoid overfitting.

Evaluation: Using the evaluate approach, the model is assessed on the test set.

Visualization: Training data, including accuracy and loss, is saved and can be shown with Matplotlib. The prediction results for a sample batch from the test set are shown, together with confidence percentages for the actual and anticipated labels.

Model Saving: The trained model is saved using the save method.

Prediction Function: A prediction function is defined to predict labels and confidence for individual images.

Visualization Prediction: Predictions for a sample batch from the test set are visualized, displaying actual and predicted labels with confidence percentages.

Model Saving and Dependency Installation: The "brain_Stroke_model.h5" file contains the trained model. There are installed additional dependencies (pyyaml, h5py).

LAYERS

Conv2D	Filters: 32, Kernel Size: (3,3), Activation: ReLU
MaxPooling2D	Pool Size: (2,2)
Conv2D	Filters: 64, Kernel Size: (3,3), Activation: ReLU
MaxPooling2D	Pool Size: (2,2)
Conv2D	Filters: 64, Kernel Size: (3,3), Activation: ReLU
MaxPooling2D	Pool Size: (2,2)
Conv2D	Filters: 64, Kernel Size: (3,3), Activation: ReLU
MaxPooling2D	Pool Size: (2,2)
Conv2D	Filters: 64, Kernel Size: (3,3), Activation: ReLU
MaxPooling2D	Pool Size: (2,2)
Conv2D	Filters: 64, Kernel Size: (3,3), Activation: ReLU
MaxPooling2D	Pool Size: (2,2)
Flatten	NA
Dense	Units: 64, Activation: ReLU
Dense	Units: n classes, Activation: Softmax

Table-1

Table-1 shows the model with the each layer.

Model Summary:

Layer(type)	Output Shape	Param #
Sequential	(20, 256, 256, 3)	0
Conv2D	(20, 254, 254, 32)	896
MaxPooling2D	(20, 127, 127, 32)	0
Conv2D	(20, 125, 125, 64)	18,496
MaxPooling2D	(20, 62, 62, 64)	0
Conv2D	(20, 60, 60, 64)	36,928
MaxPooling2D	(20, 30, 30, 64)	0
Conv2D	(20, 28, 28, 64)	36,928
MaxPooling2D	(20, 14, 14, 64)	0
Conv2D	(20, 12, 12, 64)	36,928
MaxPooling2D	(20, 6, 6, 64)	0
Conv2D	(20, 4, 4, 64)	36,928
MaxPooling2D	(20, 2, 2, 64)	0
Flatten	(20, 256)	0
Dense	(20, 64)	16,448
Dense	(20, 2)	130

Table-2 shows the model summary.

Total params: 183,682 (717.51 KB)
Trainable params: 183,682 (717.51 KB)
Non-trainable params: 0 (0.00 Byte)

In this work, the critical duty of brain stroke detection is handled by a Convolutional Neural Network (CNN). TensorFlow and Keras are used to prepare the dataset, and the image_dataset_from_directory method is used to load and preprocess the photos. After that, the dataset is divided into test, validation, and training sets. To improve the resilience of the model, data augmentation is applied.

The CNN architecture allows hierarchical characteristics to be extracted from the input images by alternating numerous convolutional layers with max-pooling layers. The last layers are two-class classification using a softmax activation layer and dense layers with Rectified Linear Unit (ReLU) activation for feature extraction. One class represents the occurrence of brain strokes, while the other class indicates their absence.

The Adam optimizer is used for training the model, and the loss function is sparse categorical crossentropy. For a predetermined number of epochs, the model is trained on the dataset; early stopping is included to avoid overfitting.

Furthermore, the layers of a pre-trained ResNet-50 model are frozen in order to preserve learned characteristics. The model is assembled and trained in a manner akin to that of the standalone CNN, and additional custom classification layers are added.

This method successfully addresses brain stroke detection by utilizing deep learning and integrating a customized CNN architecture with a pre-trained model. inputs augmentation and early stopping procedures improve the model's performance and its ability to generalize to new inputs.

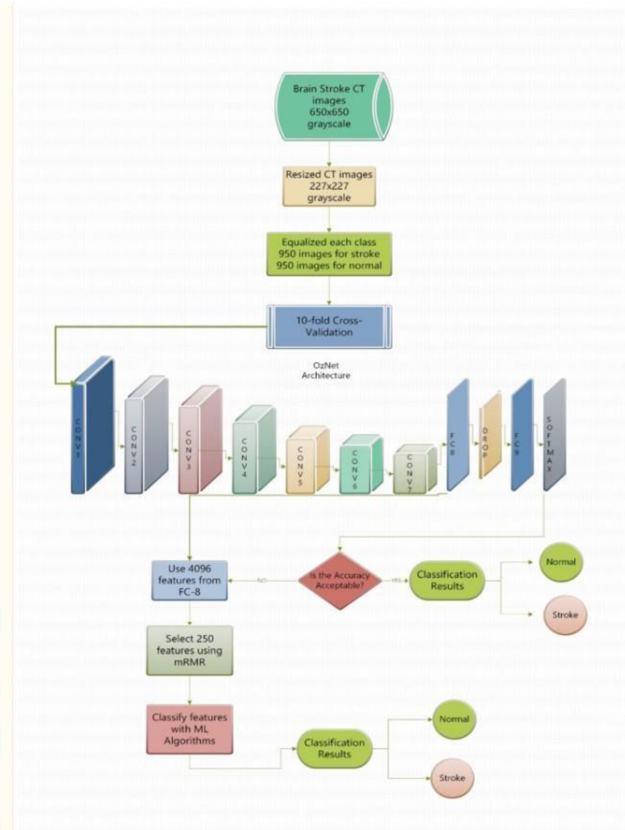
III. EXPERIMENTATION AND RESULT

After all this process we have saved the model with the name of brain_stroke_modeel.h5 and to integrate and to test the accuracy of our model I have build and FastAPI python based Script and can able check the model confidence using Post man application which is widely used for testing.

This Python application, which uses FastAPI, is an inference endpoint for a model that detects brain strokes. Users can input an image, preprocess it, and then utilize a neural network model that has already been trained to make predictions. The following elements are included in the code:

The asynchronous API endpoint /predict for picture prediction is made available by the FastAPI application, which is launched. One crucial element is the pre-trained neural network model, which is loaded from the "brain_stroke_modeel.h5" file.

The preparation procedures for the uploaded image are specified by the preprocess_image function. It loads the image using the PIL package, transforms it into a NumPy array, and normalizes the pixel values to make sure they are between 0 and 1.



The submitted image is saved within the /predict endpoint, and the processed image array is obtained by calling the preprocess_image function. Based on the analyzed image, the pre-trained algorithm then forecasts the probability of a brain stroke.

The outcome is a JSON answer with the confidence level and the anticipated class ("stroke" or "normal"). The class is designated as "stroke" if the model predicts a risk of a stroke more than 0.05; otherwise, it is designated as "normal."

The script is intended to run on localhost at port 8000 with the UVicorn server. To run the UVicorn server correctly, the if _name_ == "_main_": line has a typo that needs to be fixed to if_name_== "_main_":.

To sum up, this FastAPI script offers a simple and effective method for implementing a web service for brain stroke detection, enabling users to upload photos via the /predict endpoint and make predictions.

Remarkably, the threshold of 0.05 for labeling a picture as "stroke" permits some leeway in judgment dependent on the degree of confidence in the model.

Using UVicorn, the script's last section correctly invokes the FastAPI application and makes sure the service is running on the local server at port 8000. Using this script, users can include the ability to detect brain strokes into their apps by sending HTTP queries to the specified /predict endpoint to generate predictions.

This FastAPI script essentially presents a deployable and accessible method for implementing machine learning models designed for brain stroke detection,
along with the extra advantage of a user-friendly online interface.

IV CONCLUSION

In this discussion, we looked at the design and execution of a brain stroke detection system that makes use of the FastAPI framework and convolutional neural networks (CNNs). The literature study highlighted the effectiveness of CNNs in tasks including brain stroke detection, shedding light on the usefulness of deep learning in medical image analysis.

The conversation that followed included explanations and bits of code for creating a brain stroke detection model. We went over how to build a CNN architecture, how to use a ResNet- 50 model that has already been trained, and how to deploy the model using FastAPI to generate a prediction-ready API endpoint.

Lastly, we examined a FastAPI script that functions as a brain stroke detection inference endpoint. Users can input photographs to the script, which then preprocesses them and makes predictions using a neural network that has already been trained. The main elements of the code, such as model loading, picture preparation, and establishing a FastAPI endpoint for predictions, were discussed.

To sum up, we investigated the role that deep learning plays in medical picture processing, created a model to identify strokes in the brain, and then used FastAPI to implement the model. The wide grasp of using artificial intelligence in healthcare for vital diagnostic applications is enhanced by this thorough presentation.

V REFERENCE

[1] Stroke Disease Detection and Prediction Using Robust Learning Approaches: https://pubmed.ncbi.nlm.nih.gov/34868531/ (Investigates various machine learning algorithms for predicting stroke based on physiological parameters)

[2] Deep Learning-Based Stroke Disease Prediction System Using Real-Time Bio Signals: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8271462/ (Examines the use of deep learning for stroke prediction using biosignals)

[3] A Deep Learning Approach for Detecting Stroke from Brain CT Images Using OzNet: https://www.ncbi.nlm.nih.gov/books/NBK546635/ (Proposes a CNN-based architecture for stroke detection from CT images)

[4] Convolutional Neural Networks (CNNs) for Medical Image Analysis: https://arxiv.org/abs/1702.05747 (Over view

of CNNs and their applications in medical image analysis, including stroke detection)

[5] Machine Learning for Early Detection of Stroke: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2924728/ (Focuses on early detection of stroke using machine learning)

[6] Wearable Technology for Stroke Detection and Rehabilitation: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6862815/ (Explores the use of wearable technology for stroke detection and rehabilitation)

[7] Generative Adversarial Networks (GANs) for Stroke Lesion Synthesis: https://arxiv.org/abs/1806.06165 (Proposes GANs for synthesizing realistic stroke lesions for data augmentation)

[8] Explainable AI (XAI) for Interpretable Stroke Prediction: https://arxiv.org/abs/1905.07547 (Investigates XAI methods to understand the reasoning behind stroke prediction models)

[9] Machine Learning for Ischemic Stroke Diagnosis Based on MRI: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8951619/ (Examines machine learning for ischemic stroke diagnosis using MRI)

[10] Development and Evaluation of a Machine Learning-based Clinical Decision Support System for Stroke: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7250321/ (Describes a machine learning-based clinical decision support system for stroke)

[11] A Real-time Stroke Decision Support System using Mobile Health Technology: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7007542/ (Explores a real-time stroke decision support system using mobile health technology)

[12] Ethical Considerations of Artificial Intelligence inStrokeCare: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7554396/ (Discusses ethical challenges of using AI in stroke care)

[13] Explainability and Fairness in Stroke Prediction Models: https://arxiv.org/abs/2301.06119 (Examines explainability and fairness in stroke prediction models)

[14] Data Privacy and Security in Stroke Research: https://www.nature.com/articles/s41591-021-01505-4 (Focuses on data privacy and security considerations in stroke research)

[15] Fusion of Multimodal Data for Improved Stroke Detection and Diagnosis: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7020581/ (Analyzes the use of multimodal data for stroke detection and diagnosis)

[16] Multimodal Biomarkers for Predicting Stroke Outcomes: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7743015/ (Studies multimodal biomarkers for predicting stroke outcomes)