International Journal of Scientific Research in Science, Engineering and Technology

Print ISSN - 2395-1990
Online ISSN : 2394-4099

OPEN ACCESS

Available Online at : www.ijsrset.com
doi : https://doi.org/10.32628/IJSRSET2411411

From Pixels to Predictions : Leveraging CNNs for Timely Ischemic Stroke Detection

Arnav Yadav, Hem Mehta, Mahen Shah
Eklavya School, Ahmedabad, Gujarat, India

ARTICLEINFO

Article History:

Accepted: 20 Aug 2024
Published: 05 Sep 2024

Publication Issue :
Volume 11, Issue 5
Sept-Oct-2024

Page Number :
01-04

ABSTRACT

Early detection of ischemic stroke is crucial for optimal patient outcomes. This research presents a Convolutional Neural Network (CNN) model developed using Python libraries such as NumPy, Pandas, Matplotlib, Seaborn, Keras, and TensorFlow for the accurate identification of isYPchemic stroke. The model was trained and evaluated on a publicly available dataset of medical images. Through meticulous data preprocessing, augmentation, and model optimization, the CNN achieved a remarkable success rate of over 90% in distinguishing ischemic stroke cases from healthy controls. This study demonstrates the potential of deep learning in developing a robust and efficient clinical decision support tool for the timely diagnosis of ischemic stroke.

Keywords : CNN, ANN, Tensorflow, Keras, Control, Ischemic Stroke, Activation function

I. INTRODUCTION

Ischemic stroke, a condition caused by the blockage of blood flow to the brain, accounts for the majority of stroke cases worldwide. Early detection and treatment are crucial for reducing mortality and long-term disability. Traditional diagnostic methods, such as CT scans and MRIs, rely heavily on the expertise of radiologists, which can lead to delays in diagnosis (Fiebach et al., 2010). In recent years, the integration of deep learning algorithms, particularly CNNs, into medical imaging has shown promise in automating the detection of ischemic stroke, offering faster and potentially more accurate results (Litjens et al., 2017).

This research paper aims to explore the application of CNN models and deep learning algorithms in detecting ischemic strokes, with a focus on the Python programming language and its associated libraries. The research will cover the theoretical background of CNNs, review previous research in this field, and provide a step-by-step methodology for building and evaluating a CNN model for stroke detection. Additionally, the results will be analyzed, and potential future directions for research will be discussed.

II. LITERATURE REVIEW

The application of CNNs in medical imaging has been a topic of growing interest, with numerous studies demonstrating their effectiveness in detecting various medical conditions, including ischemic stroke. One of the early studies by Krizhevsky, Sutskever, and Hinton (2012) showcased the power of CNNs in image classification, which laid the foundation for their use in medical diagnostics. Subsequent research by Litjens et al. (2017) provided a comprehensive review of deep learning techniques in medical imaging, highlighting the potential of CNNs in detecting abnormalities in brain scans.

Specifically, in ischemic stroke detection, Kamnitsas et al. (2017) developed a CNN-based approach that achieved high accuracy in segmenting stroke lesions in MRI images. Their model, known as DeepMedic, utilized a 3D CNN architecture to analyze brain scans in a fully automated manner. Similarly, McKinley, Meier, Wiest, and Reyes (2017) proposed a CNN-based method for stroke lesion segmentation that incorporated multi-scale feature extraction, leading to improved performance compared to traditional methods.

Despite these advancements, challenges remain in ensuring the generalizability of CNN models across different datasets and medical imaging modalities. This study seeks to build on previous research by implementing a CNN model for ischemic stroke detection using widely available Python libraries, thereby contributing to the growing body of knowledge in this field.

III. METHODOLOGY

The methodology section outlines the steps taken to build and evaluate the CNN model for ischemic stroke detection. The process is divided into several stages, including data collection, preprocessing, model design, training, and evaluation.

Data Collection

The foundation of this research was laid by constructing a comprehensive dataset of brain CT scans. These images were sourced from Kaggle and meticulously categorized into two distinct groups: a control group representing healthy individuals and an acute ischemic stroke group. This categorization formed the crux of our dataset, providing essential data for training and evaluating the CNN model.

Data Processing

The dataset, comprising CT scan images in PNG or JPG format, was sourced from Kaggle and organized into distinct folders for control and acute ischemic stroke cases. To facilitate subsequent analysis, these images were converted into NumPy arrays. A for loop iteratively transformed each image into a NumPy array, which was then appended to a single array. This array, along with corresponding image names, was subsequently structured into a Pandas DataFrame. The identical process was repeated for the control group.

To establish a clear distinction between the two groups, a new column titled 'stroke' was introduced. Images from the ischemic stroke group were assigned a value of 1, indicating the presence of stroke, while those from the control group were assigned a value of 0. The two DataFrames were subsequently concatenated and shuffled, followed by a reset of the index. Image names were discarded to prevent data leakage. The final DataFrame was partitioned into training and testing sets, denoted as X_train, X_test, y_train, and y_test respectively. To optimize computational efficiency, these datasets were converted back to NumPy arrays. Finally, to standardize pixel values within a suitable range for CNN processing, the image arrays were normalized to values between 0 and 1.

Model Designing

The model consists of a CNN model followed by a dense ANN model. The model consists of 4 Conv2D layers followed by flattening it and then consists of a ANN layer of 128 neurons followed by the output node. The hidden layer consists of a ReLu function followed by a sigmoid function which is good for binary classification.

The training data consists of 1608 images which are trained in 45 epochs with 51 images in each batch. The test loss was found out to be 0.991 and the accuracy was 90%.

IV. DISCUSSIONS

The discussion section delves into the implications of the results and compares them with previous studies. The high accuracy of the CNN model in this study aligns with findings from other research, such as the works by Kamnitsas et al. (2017) and McKinley et al. (2017). However, this study also highlights the need for more extensive validation on diverse datasets to ensure the robustness of the model in real-world clinical settings.

Moreover, the integration of additional data, such as clinical features and patient history, could further improve the model's predictive capabilities. Future research could explore the use of more advanced architectures, such as 3D CNNs or attention mechanisms, to enhance the performance of stroke detection models (Vaswani et al., 2017).

V. CONCLUSIONS

This essay has explored the utilization of CNN models and deep learning algorithms for detecting ischemic stroke through medical imaging. By leveraging Python libraries such as NumPy, Pandas, Keras, and TensorFlow, a CNN model was developed and trained to achieve high accuracy in stroke detection. The results of this study underscore the potential of deep learning in revolutionizing medical diagnostics, particularly in the field of stroke detection.

As the field of deep learning continues to evolve, further research is needed to refine these models and ensure their applicability in clinical practice. The findings of this study contribute to the growing body of knowledge on the use of CNNs in medical imaging and highlight the importance of interdisciplinary collaboration between data scientists and healthcare professionals.

VI. REFERENCES

[1]. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., & Zheng, X. (2016). TensorFlow: A system for large-scale machine learning. In the 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16) (pp. 265-283).

[2]. Fiebach, J. B., Schellinger, P. D., Jansen, O., Meyer, M., Wilde, P., Bender, J., & Sartor, K. (2010). CT and diffusion-weighted MR imaging in randomized order: diffusion-weighted imaging results in higher accuracy and lower interrater variability in the diagnosis of hyperacute ischemic stroke. Stroke, 33(9), 2206-2210.

[3]. Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in Science & Engineering, 9(3), 90-95.

[4]. Kamnitsas, K., Ledig, C., Newcombe, V. F., Simpson, J. P., Kane, A. D., Menon, D. K., & Rueckert, D. (2017). Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. Medical Image Analysis, 36, 61-78.

[5]. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems (pp. 1097-1105).

[6]. Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., & van Ginneken, B. (2017). A survey on deep learning in medical image analysis. Medical Image Analysis, 42, 60-88.

[7]. Maier, O., Menze, B. H., von der Gablentz, J., Häni, L., Heinrich, M. P., Liebrand, M., & Reisert, M. (2017). ISLES 2015-A public evaluation benchmark for ischemic stroke lesion

segmentation from multispectral MRI. Medical Image Analysis, 35, 250-269.
[8]. McKinley, R., Meier, R., Wiest, R., & Reyes, M. (2017). Loss-aware training of a segmentation forest for ischemic stroke lesion segmentation. In International Workshop on Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries (pp. 51-62). Springer, Cham.
[9]. Powers, D. M. (2011). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. Journal of Machine Learning Technologies, 2(1), 37-63.
[10]. Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. Nature, 323(6088), 533-536.
[11]. Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of Big Data, 6(1), 60.
[12]. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., & Polosukhin, I. (2017). Attention is all you need. In Advances in Neural Information Processing Systems (pp. 5998-6008).
[13]. Walt, S. V. D., Colbert, S. C., & Varoquaux, G. (2011). The NumPy array: A structure for efficient numerical computation. Computing in Science & Engineering, 13(2), 22-30.
[14]. Waskom, M. L. (2021). seaborn: statistical data visualization. Journal of Open Source Software, 6(60), 3021.

APPENDICES

Appendix A: Python code snippets for preprocessing, model design, and evaluation.
StrokePredicterEngine.ipynb

Appendix B: Dataset description and access details
Acute Ischemic Stroke MRI