CT Image Classification of Human Brain Using Deep Learning

Archana Kalidindi, Prasanna Lakshmi Kompalli (✉),
Sairam Bandi, Sri Raagh Rao Anugu
Gokaraju Rangaraju Institute of Engineering and Technology,
Hyderabad, India
prasannakompalli@gmail.com

Abstract—Computed Tomography (CT) images are cross-sectional images of any specific area of a human body which allows doctors to see inside of a patient. CT scan is almost always the first imaging modality used to assess patients with suspected hemorrhage. A CT scan provides image reports in the form of grey shades. It is sometimes difficult to distinguish between two areas because the shades of grey in a CT image are occasionally similar. CT scan (Particularly "Non-Contrast Head CT Scan") is the current guideline for primary imaging of patients with any head injuries or brain stroke like symptoms. To obtain any findings from the CT image, Radiologists or other doctors need to examine the images. Deep-learning is an important tool used in radiology and medical imaging which provides a better understanding of the image with more efficiency and quicker exam time. The main idea of this project is developing a model using classification algorithms which can be used to classify or detect hemorrhage in a CT image. The dataset consists of both normal CTs and CTs with hemorrhage. Deep learning is used to develop a model that can detect whether a CT image shows a hemorrhage or not.

Keywords—Computed Tomography, Deep Learning, CNN, Hemorrhage

1 Introduction

Traumatic head injuries from events such as accidents or falls might result in sudden hemorrhaging in the brain. Hemorrhage in a brain is a serious condition as this may result in paralysis or even death if not diagnosed quickly. To evaluate a subject with a head injury, CT (Computed Tomography) Scan is the primary diagnostic tool used as it can generate reports rapidly. Windowing method is generally used to display CT scans which transform the HU values (Hounsfield Unit) into grayscale values ([0, 255]). We have different window parameters and each parameter can be used to display different features of the brain tissues in a CT image. Generally, to interpret a CT image or to determine if a hemorrhage has occurred, we need an experienced radiologist. But this diagnosis process highly depends on the availability of a special-

ized radiologist. This is a time sensitive procedure and sometimes, delayed diagnosis of hemorrhage could lead to death.

In recent years, development of machine learning algorithms at image pattern recognition tasks have been growing quickly which has shown promising results in assisting the medical engineering field. With the sustained development of medical imaging techniques, there has been an improvement in accuracy and reliability of medical diagnosis. However, a computer aided diagnosis can reduce the diagnosis time and will be helpful in prioritizing the reports. Recent advances in machine learning have shown that the automatic image classification using algorithms has the capability to detect hemorrhage in a CT scan. Deep learning has shown promising results in automated classifications. It is so proficient at image tasks that we can also use neural networks to create diagnostic images, not just analyze them. Especially Convolutional Neural Network (CNN) is well-suited for analyzing images such as CT or MRI scans. CNNs are developed mainly to process images more efficiently and carry out image classifications. Therefore, CNNs are approximating the precision of radiologists when identifying important features in CT scans or any other diagnostic images.

We hypothesized that deep learning algorithms have the potential to automate the diagnosis procedure for classifying CT scan images. We developed a CNN model to classify CT scan images based on its features. This will help reduce the time in the hemorrhage diagnosis and could also be helpful in prioritizing radiologist worklist. And it could be helpful in regions where an expert radiologist is unavailable and assists trainees. There are less datasets that are publicly available and there is a need for more datasets to furthermore improve the work.

2 Literature Survey

To classify brain CT images, many models have been proposed in recent years. Many data scientists are trying to come up with an automated solution for detecting hemorrhage or cancer from a CT image. Before starting the project, we have gone through the previously followed approaches. In this section, we have listed few of those approaches that have been followed previously. This covers the automatic computer-based classification system for various brain related medical problems. Apart from the below mentioned models, we have also gone through various methodologies for image preprocessing, augmentations, and pattern recognition techniques. All of them have both advantages and disadvantages. As brain hemorrhage is a critical issue and there is no scope for even minute errors, we need to have a completely reliable model to classify them. Currently, there are no such reliable and adaptive approaches for automated classification of CT images.

2.1 Ischemic stroke

We found that there were many automated classification systems for human brain images. Among those systems, there were many approaches for detection of strokes in

a CT image. In 2006, Lee and others put forward a technique to detect initial symptoms of "acute stroke" [1] on brain CT scan images. They have used an "adaptive partial smoothing filter" in their technique. Detection of required regions in an image is sometimes difficult as images will have noise in it. To focus on the stroke region and enhance the detectability, they smoothened the images by preserving the edges remain and reducing noise around the stroke region. This image processing has improved the efficiency of their model. Mald and Usin have also proposed few image processing approaches in the year 2001 and 2004 respectively. Usin have used segmentation on images using mathematical concepts like standard deviation and mean. Later in 2007, Preslowaski and others proposed a different approach to detect "Ischemic Stroke" using wavelet. He added to Lee's concepts of filtering images by reducing the noise and contrast. In 2008, Fauzi came up with a completely different concept of two-phase segmentation. To get the results easily, he used both clustering and Fcmto process the images and converted them into binary images which in turn will make the model generate results quickly. He used the concept of machine learning and built a decision tree and extracted results. Next year using the previous idea of wavelet-based processing, Chawla and others also proposed a model for detecting and classifying the strokes and this was a two-level classification.

2.2 Brain hemorrhage and hematoma

In 1993, Atam proposed a "Histogram Based K-means initial clustering" to detect hemorrhage regions in a CT image. Later in 1998, two scientists used "Possibilist Hopfield Neural Network" (PHNN) and "Fuzzy Hopfield Neural Network" (FHNN) for segmentation to classify the hemorrhage in a CT images. Then, many scientists have started using new concepts for creating more reliable models for this problem. Few of them have further studied and enhanced the previously used clustering-based and region-based segmentation approaches. Later, in 2004, a model was proposed by Hodgson which showed 97% sensitivity. Although it had good results, this model was not so reliable as it ignored few portions of the image. After that, many data scientists have come forward to develop models to classify a CT image. To classify an image and to detect hematoma, Liaohas developed an approach where he used a threshold value for identifying the hematoma with the help of decision rules by setting a binary value. Later in 2009, a semi-automated technique was proposed by 'Bardera' where he combined a region growing approach with his method and the results obtained were close to the actual real time results. Chan presented a method where he used 'top-hat transformation' and 'left-right asymmetry' for detecting hemorrhage (particularly AIH).

3 Methodology

The below steps are followed for implementing the project:

3.1 Dataset acquire

The dataset we used is obtained from Kaggle datasets which contains CT images in PNG format and csv file. Labels of the images are stored in the csv file.
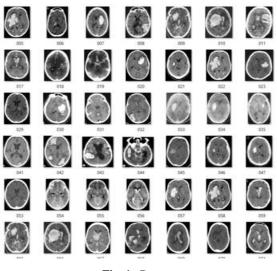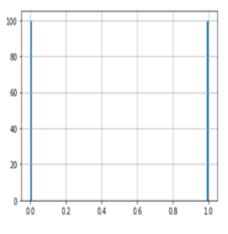

Fig. 1. Dataset

3.2 Data exploration and pre-processing

Figure 2 is a line plot which indicates that the number of images used in this dataset are equally distributed i.e., both the classes has same number of images.

Fig. 2. Data Exploration-1

Then, we explored each image in the dataset and we found that there is a variation in dimensions of the images used. Figure 3 shows the cumulative information regarding height and width of images in the dataset.

	height	width
count	200.000000	200.000000
mean	355.505000	433.720000
std	116.785247	142.059481
min	134.000000	135.000000
25%	252.000000	298.750000
50%	345.000000	435.500000
75%	447.250000	554.000000
max	821.000000	957.000000

Fig. 3. Data Exploration-2

Figure 4 shows the variations in the height and width of the images. In this project, we resized all the images to an optimal size. The trade-off here is to decrease the overfitting, but there can be a loss of information.
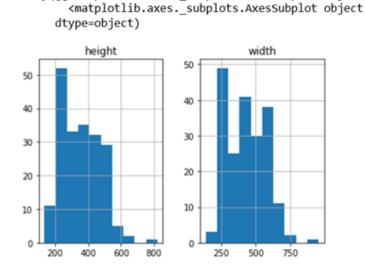

Fig. 4. Dimensions of the images

3.3   Image augmentation

The initial dataset that we had was smaller and to increase the efficiency of the model, there is a clear need for image augmentation. Image augmentation can simply be done by flipping and rotating the images in our dataset. Instead of that, we made use of "ImageDataGenerator" module for image augmentation. This was developed as part of "keras. preprocessing. image" module. "ImageDataGenerator" provides us with many parameters to augment the data. As this is a part of keras module, this can be used to directly generate different batch of images in each epoch while training the model. In this way, the model can see a slightly variant image in each epoch and this could improve the efficiency of the model.
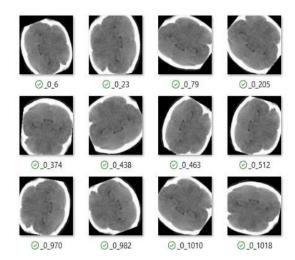

Fig. 5. Image Augmentation Example

3.4   CNN classifier

The word 'deep learning' itself refers to have more than 2 layers in it. Being a deep learning algorithm, CNN encompasses of several layers (input, hidden and output layers) which are connected, and it can automatically learn the features from a dataset or an image without any manual feature extractions.

In recent years, the neural networks are showing good results in classification problems. CNNs are especially considered to yield better results in image classification than any other ML classification algorithms.

To classify an image using CNN, we pass the image as the input through a series of layers and at the end it'll generate the output. The first layer in a CNN is always a "convolutional layer". In this layer, there will be a smaller matrix called as 'filter' or 'kernel' and this smaller matrix will be passed over the input image matrix. This is known as 'convolution' where the filter moves along the input and performs some computations. At the end, we'll obtain another matrix as output whose size will be

smaller than the input size. Subsequent values are computed according to the following formula:

$$G[m,n] = (im * f)[m,n] = \sum_{j*} \sum_{j} f[i,j] im[m-i, n-i] \qquad (1)$$

Here, im – Input Image
f – Filter or Kernel
m&n – Shape of Resultant Matrix

In general, the first layer of a CNN model learns the basic features from an image such as edge, corners etc. The output of this layer will be given to the next layer as input. Figure 6 shows how the dimensions of the image decreases from 1st layer to the next layer.
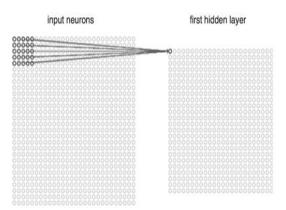

Fig. 6. Convolution Layer

After the convolution layer, there will be a "non-linear" layer which specifies an activation function. This activation function helps in determining the output coming out of a node in the network, based on its input. In this project, we used the "Relu" activation function as it produced better results for us.

Then comes "Pooling Layer" which is mainly used for reducing the dimensions of the image. Since the model learns some features in the previous layers itself, it compresses the image and all those features that are no longer required will not be available.

The last layer of any CNN should be a "fully connected dense layer" [7]. This layer must not be missed as this layer takes the resultant information from the convolutional layers and generates the actual output.[9]

4   Results

4.1   Confusion matrix

- **True Positive (TP)** - Person has a hemorrhage and the model predicted it correctly.
- **True Negative (TN)** - Model predicted correctly that the person does not have hemorrhage.
- **False Positive (FP)** - Person has no hemorrhage, but the model prediction is wrong.
- **False Negative (FN)** –Person has hemorrhage, but the model predicted it wrongly.

Table 1. Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	10	0
Actual Negative	3	7

4.2   Accuracy

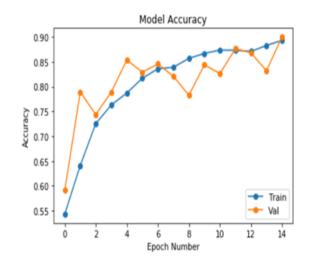Figure 7 shows the variations in accuracy at each epoch for both training and testing dataset.

Fig. 7. Accuracy

4.3 Loss

Figure 8 shows the variations in model loss at each epoch for both training and testing dataset.
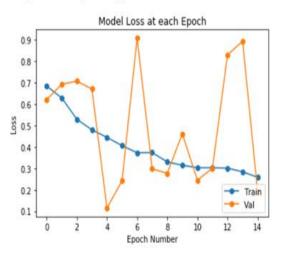

Fig. 8. Loss

4.4 CNN architecture

The following Figure 9 represents the CNN layers used in this model.
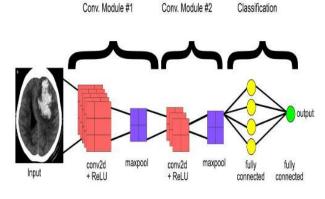

Fig. 9. Architecture

4.5 Prediction

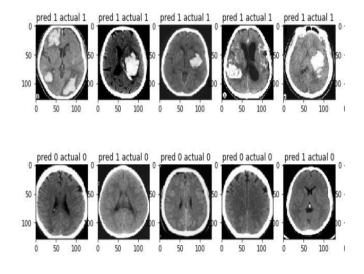Figure 10 is a snippet of the results obtained after the image prediction done by the model.

Fig. 10. Prediction Example

5 Conclusion

As part of our project, we prepared a classifier model using CNN deep learning algorithm. This classifier model can classify the images of human brain CT scans into either hemorrhage or not. One of the problems that we faced in this project is the dataset. As we mentioned, our dataset size is low, and we couldn't get many images. To improve the efficiency of the model despite having less images, we used image augmentation concept. This was done with the help of ImageDataGenerator which was present in keras module. The accuracy of the model is fine, and it can be increased if we can get more images. Another important point to be noted here is the false negative counts. False Negative is a test result which wrongly indicates that a person has a hemorrhage. As this is a medical problem, this is something which needs to be taken care of. In our project, we didn't get any false negatives, but a bigger dataset might create this problem. The aim of this project is to assist the existing diagnosis procedure by prioritizing the radiologist's or other specialized doctor's worklist. This will save a lot of time especially when the doctor or the radiologist has too many patients to be checked after.

6 Future Enhancements

This classifier model basically has a smaller number of images in the dataset. So, even though the accuracy is good, the efficiency of the model needs to be still improved. Collecting a bigger dataset can largely increase the efficiency of this model. Quality of the images in the dataset can also have an influence on the model. The complication with a medical problem is the false negative and false positive counts. False negative, mainly in our problem, is a major aspect because this could lead to the death of the patient. To handle this, we can imbalance the dataset such that there are more positive classes in it. This might cause an increase in false positive count, but false positive is just an inconvenience to the patient whereas false negatives could result in death.

7 References

[1] Adams HP, Brott TG, et al. Guidelines for thrombolytic therapy for acute stroke

[2] Allen CH, Ward JD. An evidence-based approach to management of increased intracranial pressure. Crit Care Clin-1998

[3] Al-Shahi R, Warlow C. A systematic review of the frequency and prognosis of the arteriovenous malformation of the brain in adults. Brain-2001

[4] Auer RN, Sutherland GR. Primary intracerebral hemorrhage: pathophysiology. Can J Neurol Sci – 2005

[5] Ott KH, Kase CS, et al. Cerebellar hemorrhage: diagnosis and treatment a review of 56 cases. Arch Neurol. 1974

[6] Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Adv Neural Inf Process Syst. 2012 https://doi.org/10.1145/3065386

[7] Gao H, Zhuang L, Laurens M, Kilian QW. Densely connected convolutional networks.

[8] K Prasanna Lakshmi, et al. "Video Genre Classification using Convolutional Recurrent Neural Networks" International Journal of Advanced Computer Science and Applications

[9] Jan K, Vladimir G, Daniel C. Regularization for deep learning: a taxonomy.

[10] Nitish S, Geoffrey H, Alex K, Ilya S, Ruslan S. Dropout: a simple way to prevent neural networks from overfitting.

[11] "Karen S, Andrew Z. Very deep convolutional networks for large-scale image recognition. arXiv e-prints. 2014"

[12] "Kaiming H, Xiangyu Z, Shaoqing R, Jian S. Deep residual learning for image recognition."

8 Authors

Archana Kalindi is the Assistant Professor in Information Technology Department at Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, India. She received the Bachelor's Degree in Information Technology (2007), Master's Degree in Software Engineering (2011) and currently pursuing Ph.D in Computer Science in the area of IoT and Deep Learning. Her research interest includes Deep

Learning, Data Science, Machine Learning. Published many research articles in renowned journals and conferences. Email: archana.buddaraju@gmail.com

K Prasanna Lakshmi is working as a Professor and Head of Department in Information Technology Department in Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad. She received the Bachelor's Degree in Mechanical Engineering (2000), Master's Degree in Computer Science (2002) and Ph.D in Computer Science (2016) in the area of Data Stream Mining. She has total 18 years of teaching and 10 years of research experience. She is a reviewer for various conferences, journals and books, Editorial Board Member for IGI Global Publications. Her research interests include Data Science, Artificial Intelligence, Data Stream Mining, Web Mining, Big Data Analysts, Social Networking. Published many research articles in renowned national and international journals and conferences. Email: prasanna-kompalli@gmail.com

Bandi Sairam is an UG student finishing his last year at Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, India. Currently he is doing an internship at NCR corporation as a Software Engineer. Sairam's research concentrate on Data Science and Automation. Email:sanram1999@gmail.com

Sri Raagh Rao Anugu is an UG student finishing his last year at Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, India. His research concentrates on Machine Learning. Email: sriraaghraoanugu333@gmail.com