Application of Computer Vision in Intracranial Hemorrhage (ICH) Detection

Amit Bhatia
SCPD student – Artificial Intelligence Graduate Certificate
Department of Computer Science
Stanford University
amit0911@stanford.edu | amit9bhatia@yahoo.co.in

Abstract

Intracranial hemorrhage is a life-threatening emergency and requires immediate attention and treatment. Existing processes involve time-consuming manual review of CT scans and rely upon assured availability of trained radiologists. AI powered ICH detection tools that are capable of delivering equally strong performance – hold significant potential to improve treatment outcomes. With that objective, we have explored and leveraged recent advances in deep learning and especially computer vision and its application in medical image analysis to train a model that reviews a CT image and predicts presence / absence of ICH and its 5 sub-types. Our model has been built using Transfer Learning with features extracted from the DenseNet architecture pre-trained on ImageNet dataset and is able to achieve Test set AUC of 0.9329 for ICH detection. Model is able to predict with Precision of 51.4% and is able to capture 84.2% of ICH positives, while delivering an overall Accuracy of 86.4%. To make the model interpretable and offer additional inputs to radiologists, we have used Class Activation Maps approach to build a capability to highlight regions in the image that are influencing the network's decision and are potentially the sites of ICH.

1. Motivation

Intracranial hemorrhage (ICH), bleeding that occurs inside the cranium, is a serious health problem requiring rapid and often intensive medical treatment. It accounts for approximately 10% of strokes in the U.S., where stroke is the fifth-leading cause of death. Identifying the location and type of any hemorrhage present is hence a critical step in treating the patient.

Computed tomography (CT) is the most commonly used medical imaging technique to assess the severity of ICH in case of traumatic brain injury. According to the American Heart Association and American Stroke Association, the early and timely diagnosis of ICH is significant – as this condition can commonly deteriorate the affected patients within the first few hours after occurrence.

Traditional methods involve visual inspection by radiologists and quantitative estimation of the size of hematoma manually. The entire procedure is time-consuming and requires the availability of trained radiologists at every moment.

Limitations in the availability or experience of clinicians, especially in rural or resource-strapped health systems, to diagnose CT brains quickly can cause treatment delays. Therefore, automated hemorrhage detection tools - capable of providing fast inference that is also accurate to the level of radiologists; hold the potential to save thousands of patient lives.

2. Data description

We have worked on the dataset of CT scan (DICOM) images provided by Radiological Society of North America for the Kaggle competition - RSNA Intracranial Hemorrhage Detection.

Digital Imaging and Communications in Medicine (DICOM) is the standard for the communication and management of medical imaging information and related data. It incorporates standards for imaging modalities such as radiography, ultrasonography, computed tomography (CT), magnetic resonance imaging (MRI), and radiation therapy. DICOM includes protocols for image exchange (e.g., via portable media such as DVDs), image compression, 3-D visualization, image presentation, and results reporting.

The labelled training dataset has 753 K images. Each DICOM image has the raw pixel array (512, 512) of Hounsfield Unit values.

Hounsfield scale is a quantitative scale for describing radiodensity and used universally in CT scanning. HUs are obtained from a linear transformation of the original linear attenuation coefficient measurement into one in which the radiodensity of distilled water at standard pressure and temperature (STP) is defined as 0 HU, while the radiodensity of air at STP is defined as -1000 HU. Some approximate HU values for tissues commonly found on head CT scans are as follows: Bone: 1000 HU, ICH: 60 – 100 HU, Grey matter: 35 HU, White matter: 25 HU, Muscle / soft tissue: 20 – 40 HU, and Fat: -30 – -70 HU.

Because the human eye can perceive only a limited number of grey shades, the full range of density values is typically not displayed for a given image. Instead, the tissues of interest are highlighted by devoting the visible grey shades to a narrow portion of the full density range, a process called “windowing”. The same image data can be displayed in different window settings to allow evaluation of injury to different tissues. In general, head CT images are viewed on brain or bone windows to allow most emergency pathology to be assessed.

The DICOM images in the RSNA dataset have the associated meta-data related to the Windows used in saving the image: Window center, Window width, Rescale intercept, Rescale slope.

CT (computed tomography) is essentially a computerized x-ray imaging procedure in which a narrow beam of x-rays is aimed at a patient and quickly rotated around the body, producing signals that are processed by the machine’s computer to generate cross-sectional images—or “slices”—of the body. These slices are called tomographic images and contain more detailed information than conventional x-rays. Once a number of successive slices are collected by the machine’s computer, they can be digitally “stacked” together to form a three-dimensional image of the patient that allows for easier identification and location of basic structures as well as possible tumors or abnormalities.

Each DICOM image in our dataset also carries the meta-data related to its associated Study & CT Volume: Study ID, Sequence IDs, Coordinate positions and Orientations; which can be used to create the full stack of multiple slices that were captured in the individual study. In this dataset, each study stack has 20 – 60 slices.

Given the objective of identification of ICH & its 5 sub-types – each observation in the training dataset has a Y vector of dimension 6 corresponding to the following labels: 1. Epidural (ED), 2. Intraparenchymal (IP), 3. Intraventricular (IV), 4. Subarachnoid (SA), 5. Subdural (SD), and 6. Any (of the 5 sub-types). An image can have 0 – 5 ICH sub-type labels.

	ED	IP	IV	SA	SD	Any
#	3,145	36,118	26,205	35,675	47,166	107,933
%	0.42%	4.80%	3.48%	4.74%	6.27%	14.34%

Table 1: Details of ICH events for each of the labels in the training dataset

The neurologic consequences of ICH can vary extensively depending upon the size, type of hemorrhage and location and range from headache to death. The role of the Radiologist is to detect the hemorrhage, characterize the hemorrhage subtype, its size and to determine if the hemorrhage might be jeopardizing critical areas of the brain that might require immediate surgery.

While all acute (i.e. new) hemorrhages appear dense (i.e. white) on computed tomography (CT), the primary imaging features that help Radiologists determine the subtype of hemorrhage are the location, shape and proximity to other structures.

Intraparenchymal hemorrhage is blood that is located completely within the brain itself; intraventricular or subarachnoid hemorrhage is blood that has leaked into the spaces of the brain that normally contain cerebrospinal fluid (the ventricles or subarachnoid cisterns). Extra-axial hemorrhages are blood that collects in the tissue coverings that surround the brain (e.g. subdural or epidural subtypes). Patients may exhibit more than one type of cerebral hemorrhage, which may appear on the same image. While small hemorrhages are less morbid than large hemorrhages typically, even a small hemorrhage can lead to death because it is an indicator of another type of serious abnormality (e.g. cerebral aneurysm).

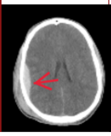
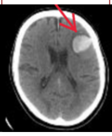
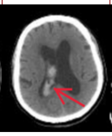
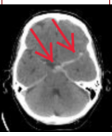
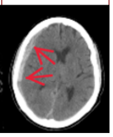
	ED	IP	IV	SA	SD
Location	Between dura & skull	Inside the brain	Inside the ventricle	Between arachnoid & pia matter	Between dura & arachnoid
Shape	Lentiform	Rounded	Conforms to ventricular shape	Tracks along sulci & fissures	Crescent
Image					

Table 2: Markers & CT images of various ICH sub-types

3. Approach

We began by studying the existing approaches for Image classification, object detection and Medical Image analysis associated with CT scans & DICOM images.

To enable rapid prototyping, we created a random sample of 100K images (Sample A) from the training dataset and split it into: Train (70%), Dev (15%) & Test (15%) datasets; ensuring similar distribution of ICH labels across the datasets.

We have considered weighted binary cross entropy loss with 2x weightage for label = “Any” as the loss metric (on similar lines as the Kaggle competition) to evaluate the candidate models.

$$Loss = \sum_{c=1}^{5} y_c log(p_c) + (1 - y_c)\log(1 - p_c) \; + \; 2 * \{y_6 log(p_6) + (1 - y_6)\log(1 - p_6)\}$$

Equation 1: Weighted binary cross entropy loss

We tried to explore multiple approaches in a systematic manner and organized the design & sequence of our experiments based on the following dimensions: Data pre-processing, Network architecture, Transfer Learning and Optimization loss function.

Data pre-processing: We considered different pre-processing strategies ranging from feeding the raw HU values to the network to multiple Window based settings –

the approach that is used universally by the radiologists. These settings map the visual range of the displays to a specified window, and assign all HU values outside this window range to 0 or U (taken U = 255). Windowing functions are defined based on linear or sigmoidal conversion as follows [1]:

$$F_{lin}(x) = \min(max(Wx+b, U), 0)$$
$$where\ W = \frac{U}{WW}, \quad b = -\frac{U}{WW}\left(WL - \frac{WW}{2}\right)$$

Equation 2: Linear window

$$F_{sig}(x) = \frac{U}{1 + e^{-(Wx+b)}}$$
$$where\ W = \frac{2}{WW}\log\left(\frac{U}{\epsilon} - 1\right), \quad b = \frac{-2WL}{WW}\log\left(\frac{U}{\epsilon} - 1\right)$$

Equation 3: Sigmoid window

where WW is Window Width, WL is Window Level or Center and epsilon is the margin between the upper / lower limits and window end/start grey levels which determine the slope at the center (taken epsilon = 1).

We have evaluated three window settings for the image: Brain window (WW = 80, WL = 40), Subdural window (WW = 200, WL = 80) and Soft tissue window (WW = 380, WL = 40).

Network architecture: We have tried multiple network architectures starting with a simple 2-D CNN with 7 *(Conv-BN-Relu-MaxPool) + 2*Dense layers wherein we fed the 512 x 512 x 1 image to the network.

Given that CT studies generate a stack of slices, we also explored a simple 3-D CNN in our experiments [2]. We used the z-coordinate of the position attribute in the meta-data to create the right sequence of image slices in the stack. Researchers in [3] have observed that a 3D-convnet informed by 3 consecutive images (image under evaluation and “flanking” images immediately superior and inferior) was as accurate as a network that employed 5 or more consecutive images, sparing the need for learning even more context. Basis this, we restricted the 3-D volume to 3 consecutive slices and fed to the network the primary image slice – flanked by its prior & subsequent image slices.

We also explored the idea of a CT study being made up of a sequence of 2D slices and evaluated combinations of CNN + Bidirectional LSTM w/o and w/ Attention mechanism [4]. We created Sample B for this experiment and considered all the slices which came from the studies which had 32 slices each (mode value). This specific selection was done just to simplify the implementation by ensuring a fixed length sequence. Sample B has 144 K, 18 K, 16 K images in train, dev and test sets respectively.

Transfer learning: is a cornerstone of computer vision and is quite effective in getting the first version of a solution implemented relatively quickly. We tried the following classic CNN architectures pre-trained on ImageNet for feature extraction and then trained a fully connected classifier using those features: VGG-19, Xception, NASNet, InceptionResNetV2, EfficientNetB7 and DenseNet201. Feature extraction was done after applying Global Average Pooling on the maps of the last convolutional layer of the network.

In one of the experiments, we intended to fine-tune the last couple of layers of the classic networks but could not complete it due to computational challenges as the raw data had to be streamed during training due to memory constraints. This is in contrast to the approach of feature extraction wherein the extracted features were persisted and were significantly compacter as compared to raw data; allowing them to be loaded into memory for training.

Optimization loss function: Given that it’s a multi-label exercise, we went ahead with Binary cross entropy as the loss metric for the optimizer. Since, correct detection of ICH is relatively more important, the model evaluation metric has been kept as a weighted Binary cross entropy loss with a weight of 2x to the label “Any ICH”; and we hence explored that too as the optimization objective.

Given that the overall ICH incidence rate in the dataset was 14.3% and 4 of the 5 ICH sub-types had incidence rates < 5%, we also experimented with Focal loss as the loss measure for the optimizer (5).

We did the entire implementation using Keras within a Kaggle kernel running on single Tesla P100 GPU. We had also tried multi-GPU training on AWS Sagemaker notebook instance provisioned with p3.16xlarge but couldn’t execute it successfully. It appears that there are a couple of open issues with Keras implementation of multi-GPU training (e.g. AttributeError: '_TfDeviceCaptureOp' object has no attribute '_set_device_from_string').

4. Experiment results & key observations

Experiment	#1
Data	Sample A
Pre-processing	Raw HU
Network	7 *(Conv-BN-Relu-MaxPool) + 2*Dense
Optimization function	Weighted Binary Cross Entropy Loss
Train loss	1.744
Test loss	1.722

Experiment	#2
Data	Sample A
Pre-processing	Raw HU
Network	7 *(Conv-BN-Relu-MaxPool) + 2*Dense
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	1.147
Test loss	1.131

Experiment	#3
Data	Sample A
Pre-processing	Raw HU
Network	Pre-trained InceptionResNetV2 w/o Top + 2*Dense
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	1.198
Test loss	1.200

Experiment	#4
Data	Sample A
Pre-processing	Raw HU
Network	Pre-trained EfficientNetB7 w/o Top + 2*Dense
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	1.213
Test loss	1.305

Experiment	#5
Data	Sample A
Pre-processing	Raw HU
Network	Pre-trained DenseNet w/o Top + 2*Dense
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	1.058
Test loss	**1.082**

Experiment	#6
Data	Sample A
Pre-processing	Raw HU
Network	Pre-trained DenseNet w/o Top + 2*Dense
Optimization function	Focal Loss
Train loss	1.250
Test loss	1.270

Experiment	#7
Data	Sample A
Pre-processing	Linear Windows: Brain, Subdural & Soft tissue
Network	Pre-trained DenseNet w/o Top + 2*Dense
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	1.264
Test loss	1.240

Experiment	#8
Data	Sample A
Pre-processing	Sigmoid Windows: Brain, Subdural & Soft tissue
Network	Pre-trained DenseNet w/o Top + 2*Dense
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	0.942
Test loss	1.127

Experiment	#9
Data	Sample A
Pre-processing	Raw HU. Primary image with adjacent flanking slices
Network	Pre-trained DenseNet w/o Top + 2*Dense
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	0.995
Test loss	1.088

Experiment	#10
Data	Sample B (slices from studies which had 32 slices)
Pre-processing	Raw HU
Network	Pre-trained DenseNet w/o Top + 4*Dense
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	1.175
Test loss	1.476

Experiment	#11
Data	Sample B (slices from studies which had 32 slices)
Pre-processing	Raw HU. Sequence of 32 slices
Network	Pre-trained DenseNet w/o Top + Bi-LSTM
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	1.084
Test loss	1.534

Experiment	#12
Data	Sample B (slices from studies which had 32 slices)
Pre-processing	Raw HU. Sequence of 32 slices
Network	Pre-trained DenseNet w/o Top + Bi-LSTM + Attn.
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	1.308
Test loss	1.571

Experiment	#13
Data	Full Dataset
Pre-processing	Raw HU
Network	Pre-trained DenseNet w/o Top + 4*Dense
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	0.743
Test loss	**0.874**

Experiment	#14
Data	Full Dataset
Pre-processing	Raw HU. Primary image with adjacent flanking slices
Network	Pre-trained DenseNet w/o Top + 4*Dense
Optimization function	Un-weighted Binary Cross Entropy Loss
Train loss	0.553
Test loss	0.886

Table 3: Experiment results

Results consistently demonstrate the effectiveness of Transfer learning. This is even more striking as we have used the pre-trained networks only as feature extractors and not fine-tuned their convolution layers.

Features extracted from DenseNet [6] have delivered best performance. DenseNets connect each layer to every other layer in a feed-forward fashion.

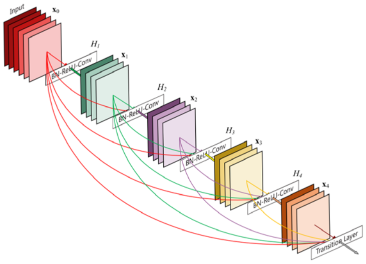


Figure 1: DenseNet

They have several compelling design advantages: they alleviate the vanishing-gradient problem, strengthen feature propagation, encourage feature reuse, and substantially reduce the number of parameters.

We also observe that a simple solution based on Raw HUs, Transfer learning based on 2-D ConvNet with Un-weighted Binary cross entropy loss was able to deliver the best result amongst the host of experiments that we had carried out. We did not see any incremental lift coming via alternate approaches involving Windowing techniques, Hybrid 3-D ConvNet, CNN + Bidirectional LSTM and Focal Loss.

	Any ICH		ED		IP	
	Train	Test	Train	Test	Train	Test
AUC	0.9843	**0.9329**	0.9581	0.9004	0.9678	0.9237
P@85pR	61.87%	**51.37%**	0.42%	0.40%	23.33%	19.99%
ACC@85pR	90.93%	**86.41%**	0.42%	0.40%	84.66%	82.85%
R@85pR	95.66%	**84.16%**	100.00%	100.00%	95.88%	86.20%
	IV		SA		SD	
	Train	Test	Train	Test	Train	Test
AUC	0.9882	0.9567	0.9625	0.9053	0.9735	0.9238
P@85pR	37.32%	31.96%	21.34%	17.61%	26.67%	23.09%
ACC@85pR	94.28%	93.44%	82.92%	80.71%	83.22%	81.05%
R@85pR	95.27%	81.12%	96.35%	84.69%	96.62%	86.86%

Table 4: Summary statistics of current solution

We are able to achieve Test AUC of 0.9329 for ICH detection. Model is able to predict ICH presence with Precision of 51.4% and is able to capture 84.2% of ICH positives while delivering an overall Accuracy of 86.4%.

There is scope for improvement in ICH sub-type detection where Precision ranges from 18-32% while delivering Recall of 80-85%. Results are significantly sub-optimal for epidural ICH which is a rare class with event rate of 0.42%.

5. Visualization

We have used Class Activation Maps [7] based on Global Average Pooling to highlight regions in the image that are influencing the network's decision.

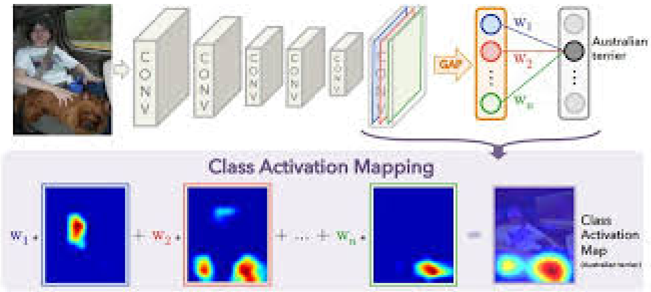


Figure 2: Class Activation Maps

This approach is extremely helpful in improving the interpretability of CNNs and also giving the radiologists and doctors additional inputs to assist them in their review.

Following are few examples where we have used CAMs to create the heatmap of activations and then overlaid that on the original image:

Image 1: Ground truth = SA ICH + SD ICH

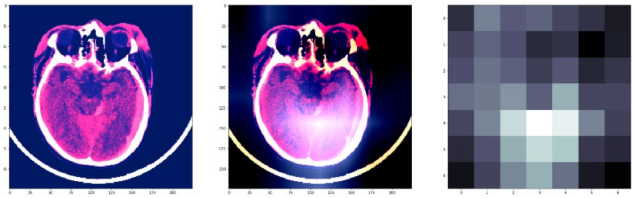


Image 2: Ground truth: ICH = 0

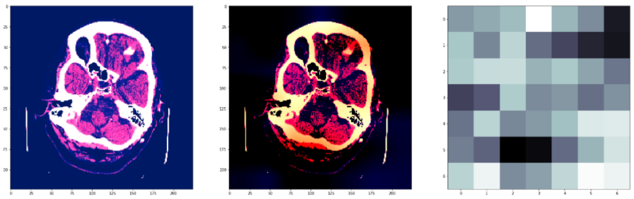


Image 3: Ground truth: IV ICH + SA ICH + SD ICH

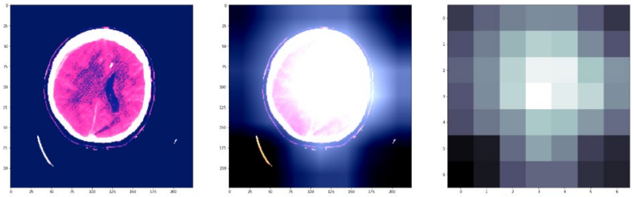
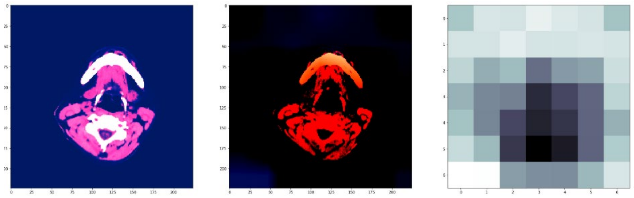


Image 4: Ground truth: ICH = 0



6. Future work

We would like to conduct further experiments to enhance performance by training on Multiple GPUs via TensorFlow, PyTorch, MXNet. This would provide us the necessary compute to build a deeper version of transfer learning wherein we can fine tune some / all of the layers in the convolutional base.

We would also like to build and test a Hierarchical decision system: Stage 1: 2-class Detector – ICH present = 0/1, Stage 2: 5 class Labeler – presence / absence of each sub-type. Stage 2 model would be trained only on ICH positive images and is expected to boost system's performance on the task of ICH sub-type identification. It would be used to score only those images that get triggered by the Stage 1: 2-class Detector.

Efforts would be put in to improve performance on Epidural hemorrhage (rare class) by data augmentations and considering an appropriate loss measure like Focal loss.

References

[1] H. Lee, M. Kim, S. Do. Practical window setting optimization for medical image deep learning. arXiv preprint 2018:arXiv:1812.00572.

[2] K. Jnawali, M. R. Arbabshirani, N. Rao, A. A. Patel, "Deep 3D convolution neural network for CT brain hemorrhage

classification" in Medical Imaging 2018: Computer-Aided Diagnosis, International Society for Optics and Photonics, vol. 10575, pp. 105751C, 2018.

[3] W. Kuo, C. Häne, P. Mukherjee, J. Malik, and E. L. Yuh. Expert-level detection of acute intracranial hemorrhage on head computed tomography using deep learning. PNAS November 5, 2019 116 (45) 22737-22745.

[4] M. Grewal, M. M. Srivastava. RADnet: Radiologist level accuracy using deep learning for hemorrhage detection in CT scans. In: 15th IEEE international symposium on biomedical imaging. Piscataway, NJ: IEEE, 2018:281–284 Read More: https://www.ajronline.org/doi/abs/10.2214/AJR.18.20328

[5] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar. Focal loss for dense object detection. arXiv preprint arXiv:1708.02002, 2017. 1, 3, 4

[6] G. Huang, Z. Liu, K. Q. Weinberger, and L. Maaten. Densely connected convolutional networks. In CVPR, 2017. 2, 6

[7] B. Zhou, A. Khosla, L. A., A. Oliva, and A. Torralba. Learning Deep Features for Discriminative Localization. In CVPR, 2016. 2, 3, 4, 5