Deep Learning Application in Medical Image

# RSNA Intracranial Hemorrhage Detection

# Project Proposal

*Version 1.0: Draft Version, 10-28-2019*
*Version 2.0: Add Dataset EDA, theoretical workflow and Baseline Model/Benchmarks, 10-30-2019*

## 1 Project Overview

Deep Learning techniques have recently been widely used for medical image analysis, which has shown encouraging results especially for large healthcare and medical image datasets. In the computer vision field, the deep learning model, such as Convolutional Neural Network(CNN) has shown better capabilities to segment and/or classify medical images like ultrasound and CT scan images in comparison to traditional machine learning techniques.

Recently, Deep Learning applications, in particular in applying the CNN model for analyzing Medical Images have achieved very promising results. The major application fields can be broadly separated into two categories: classification application and segmentation applications.

- Classification Applications

For a given set of labeled images, using the deep learning model to find the patterns between the input images and its corresponding class labels. The related applications, such as lung images detection from CT scanning to classify images patches into 7 classes[1]. This paper describes how to use the CNN model to classify the healthy tissue and six different interstitial lung disease patterns. The other example is to identify the thyroid nodules as malignant or benign from the chest X-ray and Ultrasound images[2].

- Segmentation Application

The other important application for Medical Image Analysis is to identify organs, lesions or substructures of organs from the Ultrasound, MRI or X-Ray images. Now, you can use deep learning models to segment the brain tumors from MRI images.[3]

With recent progress in Deep Learning field, this project will build a model and application to detect acute intracranial hemorrhage and its subtypes based on the rich medical image dataset which is provided by the Radiological Society of North America (RSNA®) in collaboration with members of the

[1] "Lung Pattern Classification for Interstitial Lung Diseases Using ...." https://ieeexplore.ieee.org/iel7/42/7463083/07422082.pdf.
[2] "Classification of thyroid nodules in ultrasound images using ...." https://ieeexplore.ieee.org/document/7952290.
[3] "Brain Tumor Segmentation - Papers With Code." https://paperswithcode.com/task/brain-tumor-segmentation.

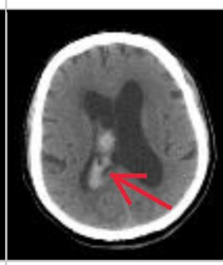
American Society of Neuroradiology and MD.ai. This is also a Kaggle Featured Prediction Competition launched months ago.[4]

## 1.1 Problem Statement

Intracranial hemorrhage, bleeding that occurs inside the cranium, is a serious health problem requiring rapid and often intensive medical treatment. For example, intracranial hemorrhages account 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 a critical step in treating the patient.

Diagnosis requires an urgent procedure. When a patient shows acute neurological symptoms such as severe headache or loss of consciousness, highly trained specialists review medical images of the patient's cranium to look for the presence, location and type of hemorrhage. The process is complicated and often time consuming.

This project is to develop a Classification/Segmentation model and build a web application to identify the five Hemorrhage sub-Types: Intraparenchymal, Intraventricular, Subarachnoid, Subdural and Epidural.

| | Intraparenchymal | Intraventricular | Subarachnoid | Subdural | Epidural |
|---|---|---|---|---|---|
| **Location** | Inside of the brain | Inside of the ventricle | Between the arachnoid and the pia mater | Between the Dura and the arachnoid | Between the dura and the skull |
| **Imaging** |  |  |  |  |  |
| **Mechanism** | High blood pressure, trauma, arteriovenous malformation, tumor, etc | Can be associated with both intraparenchymal and subarachnoid hemorrhages | Rupture of aneurysms or arteriovenous malformations or trauma | Trauma | Trauma or after surgery |
| **Source** | Arterial or venous | Arterial or venous | Predominantly arterial | Venous (bridging veins) | Arterial |
| **Shape** | Typically rounded | Conforms to ventricular shape | Tracks along the sulci and fissures | Crescent | Lentiform |
| **Presentation** | Acute (sudden onset of headache, nausea, vomiting) | Acute (sudden onset of headache, nausea, vomiting) | Acute (worst headache of life) | May be insidious (worsening headache) | Acute (skull fracture and altered mental status) |

[4] "RSNA Intracranial Hemorrhage Detection." https://www.kaggle.com/c/rsna-intracranial-hemorrhage-detection/overview/description.

## 1.2 Metrics

- This project are are evaluated using a weighted multi-label logarithmic loss. Each hemorrhage sub-type is its own row for every image, and the model will predict a probability for that sub-type of hemorrhage. There is also an '`any`' label, which indicates that a hemorrhage of **ANY** kind exists in the image. The **'`any`'** label is weighted more highly than specific hemorrhage sub-types.
- For each image `Id`, prediction will have a set of predicted probabilities (a separate row for each sub-type). Then taking the log loss[5] for each predicted probability versus its true label. The ***loss*** is averaged across all samples.
- When calculating a weighted multi-label logarithmic loss, predicted input values of 0 and 1 are undefined. To avoid this problem, log loss functions typically adjust the predicted probabilities (p) by a small value (epsilon) and use the MinMax Rule: $max(min(p,\ 1-10^{-15}),\ 10^{-15})$

## 1.3 Development Framework

In Deep learning application field, there are two major deep learning frameworks: TensorFlow and PyTorch. The competitive strengths for each framework are:

- TensorFlow is mainly adopted by the industrial companies and PyTorch is mainly focused on research communities.
- TensorFlow has a large, well established user base, and industry is typically slower to pick up on new technologies. TensorFlow is much more efficient than PyTorch. Even modest savings in model run times can help a company's bottom line.
- PyTorch integrates neatly with Python, making the code simple to use and easy to debug.

Based on top of PyTorch, Fastai is the first deep learning library to provide a single consistent interface to all the most commonly used deep learning applications for vision, text,tabular data, time series and collaborative filtering. Fastai can provide more neat, more integrate APIs to build deep learning pipelines: from the data clean to model tuning and optimization.
So for the rapid proto-typing the model, this project will use Fastai/PyTorch to implement.

## 1.4 Dataset

The original Dataset is hosted on the Kaggle platform[6]. The download API :
kaggle competitions download -c rsna-intracranial-hemorrhage-detection

There are two-part data:

- Train.csv: include the ID and Label:
  ID is a combined string that includes the image filename and Hemorrhage type.

[5] "What is Log Loss? | Kaggle." https://www.kaggle.com/dansbecker/what-is-log-loss.
[6] "RSNA Intracranial Hemorrhage Detection | Kaggle." https://www.kaggle.com/c/rsna-intracranial-hemorrhage-detection/data.

Label is a target column, indicating the probability of whether that type of hemorrhage exists in the indicated image.
Format:
`[Image Id]_[Sub-type_Name]`, as follows:

```
Id,Label
1_epidural_hemorrhage,0
1_intraparenchymal_hemorrhage,0
1_intraventricular_hemorrhage,0
1_subarachnoid_hemorrhage,0.6
1_subdural_hemorrhage,0
1_any,0.9
```

- DICOM Images:
  **DICOM** is the standard for the communication and management of medical imaging information and related data. DICOM files can be exchanged between two entities that are capable of receiving image and patient data in DICOM format.
  DICOM images contain associated metadata. This will include `PatientID`, `StudyInstanceUID`, `SeriesInstanceUID`, and other features.

# 2. Analysis

RNSA Dataset provide the rich DICOM images (Total Train DICOM images: 674258; Total Test DICOM images: 78545) and training information data. Before to build the model, it is key to understand the images and data. The Data analysis Jupyter notebook can be found here(https://github.com/Pyligent/RSNA-Medical-Image-Detection/blob/master/RSNA%20Intracranial%20Hemorrhage%20-Data%20Exploration%20.ipynb)

## 2.1 Data Exploration

Stage_1_train.csv file includes all labels information, which include the subtype and id string.
Total shape of this csv is (4045572, 2). Please see figure 1 to see the Labels value counts.

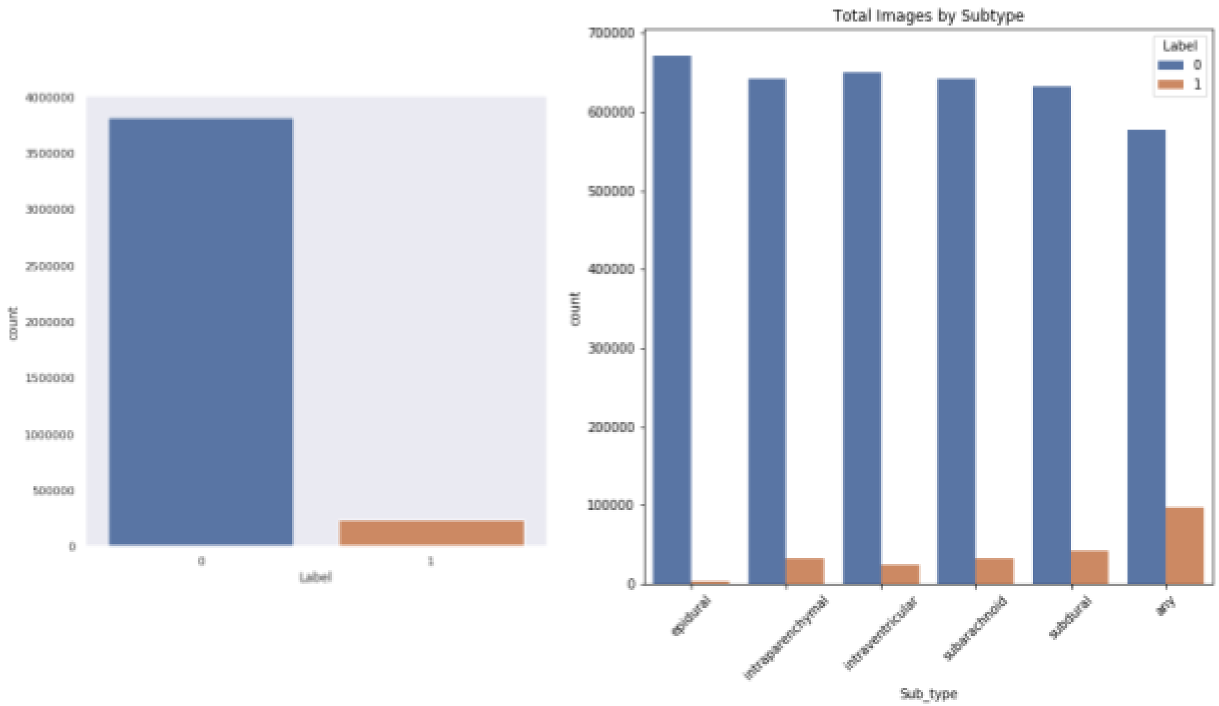



Figure 1: Label Information Plot

Obviously, the dataset is imbalance. (Label:0 ,3814760 v.s Label: 1 ,230812) , the balance rate is only around 6%.The Imbalance Problem is a common problem affecting machine learning due to having disproportionate number of class instances in practice. The easy way to deal with the imbalance dataset is to apply the sampling based approaches.

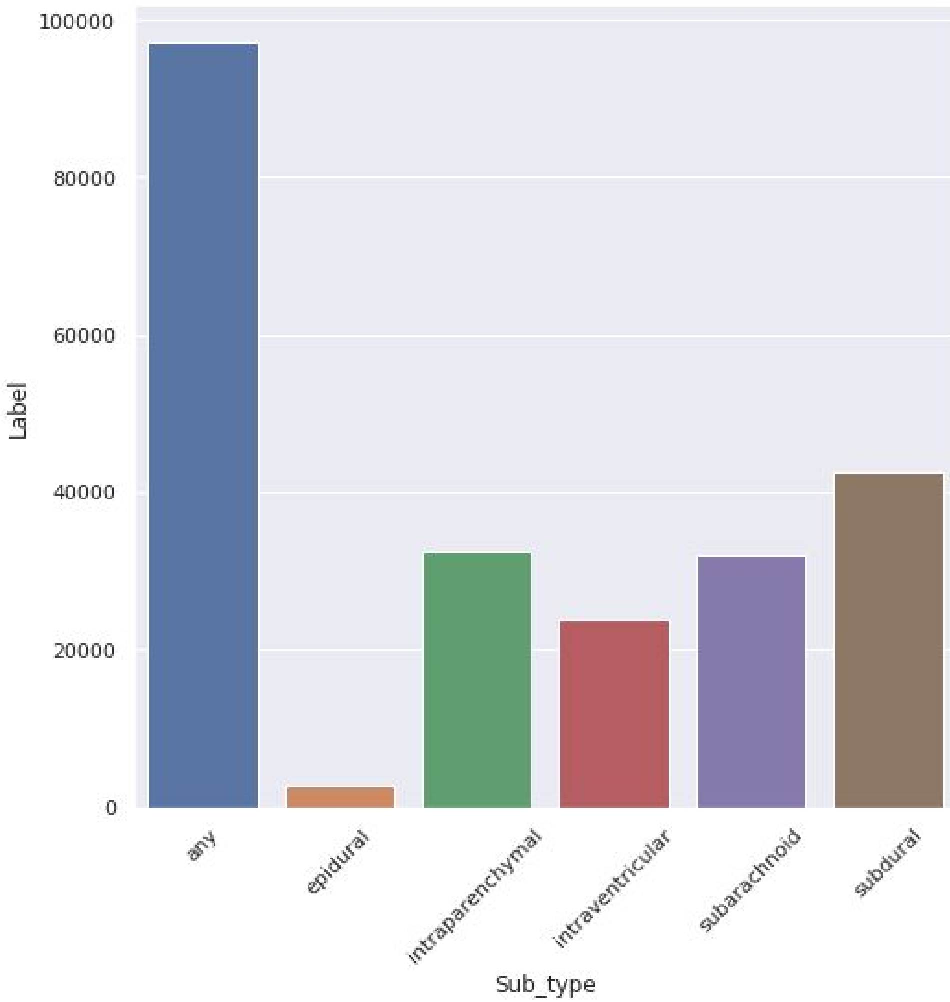
There are three methods in sampling:

- Oversampling: to add more of the minority class so it has more effect on the machine learning algorithm
- Undersampling: to remove some of the majority class so it has less effect on the machine learning algorithm
- Hybrid, a mix of oversampling and undersampling

In this project, will combine the undersampling and oversampling approaches and fine-tune the model to achieve better results.

Label Types

There are six sub-type in the label dataset: 'any', 'epidural', 'intraparenchymal', 'intraventricular', 'subarachnoid', 'subdural' . The following chart shows the label's distribution.




## 2.2 Data Visualization

Understanding the DICOM images is important in this project. For medical image, the windowing image will let model only focus on the useful information and also will let us to resize and reduce the image dataset, so it will be easier to train the model.
By using the PYDICOM library, I have built the the function the show the metadata and display different sub-type dicom images.
The detailed Jupyter notebook can be found at project's Github[7].

```
def show_dicom_metadata(filename):
    """
    show_dicom_metadata function is to get all important DICOM metadata, such as
windowing parameters and other information and also plot it
    input parameter:
```

[7]https://github.com/Pyligent/RSNA-Medical-Image-Detection/blob/master/RSNA%20Intracranial%20Hemorrhage%20-Data%20Exploration%20.ipynb

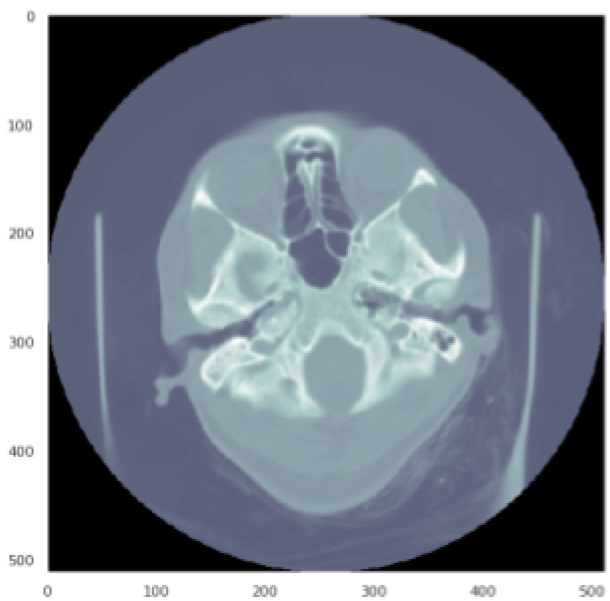
```
    filename: string, DICOM filename
    """

def display_dicom_image(df, sub_type, column_number,row_number):
    """
    display_dicom_image function shows the DICOM image from the training dataset
dataframe.
    df: data frame that includes the images and subtype information
    sub_type: string, what sub_type want to show
    column_number: int, how many images in a row
    row_number: int, how many rows want to show
   """
```

```
show_dicom_metadata(train_imgs[0])
```

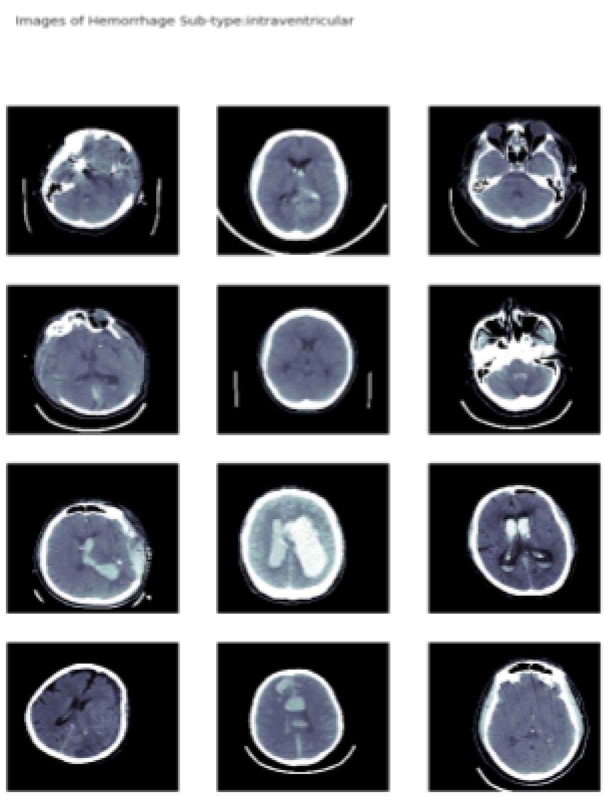
```
Filename.........: ../input/rsna-intracranial-hemorrhage-detection/stage_1_train_images/ID_000039fa0.dcm
Storage type.....: (0008, 0018) SOP Instance UID                    UI: ID_000039fa0
Patient id.......: ID_eeaf99e7
Modality.........: CT

Window Center.........: 30
Window Width.........: 80
Rescale Intercept.........: -1024
Rescale Slope.........: 1
Image size.......: 512 x 512, 524288 bytes
Pixel spacing....: ['0.488281', '0.488281']
Slice location...: (missing)
```



**Sub_Type: intraventricular**

```
display_dicom_image(train_df, 'intraventricular', 4,4)
```



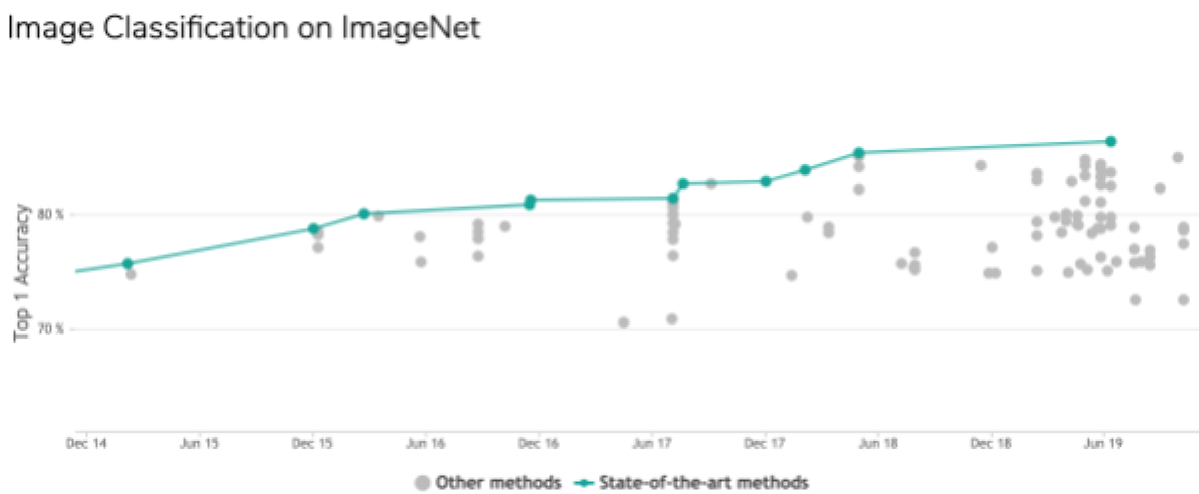

# 3 **Model and Benchmarks**

## 3.1 **State of the Art Models**

Medical Image Classification is a core problem in computer vision and deep learning R&D area. How to choose the right model is the key for the projects. Usually, the model choosing will be based on the benchmark on the ImageNet database. The models pre-trained for image classification, and then transfer to a variety of computer vision applications. This project will follow this methodology and use the transfer learning method to apply the pre-trained models to build the hemorrhage subtype detector.

Recent research in image classification has demonstrated improved performance by using the larger neural networks and higher images. For example, the State of the Art(SoTA) model in

benchmark is currently held by ResNeXt-101 32*48d architecture[8] with 829M parameters by using 224*224 images for training. More efficiently SoTA Model is EfficientNet-b7[9] with just 66M parameters by using 600*600 images for training.More recently, revised ResNeXt-101 32*48d models has got 86.4% accuracy[10].
More detailed information of SoTA models is as follows[11]:




## 3.2 **Model and benchmarks**

**Model and Network Architecture:**
This project will use `ResNeXt` model to build the hemorrhage subtype classification. Due to GPU limitation and efficiency consideration, this model will only use the ResNeXt 50, a smaller architecture to train the data. Also will try to use other models to tune the results.

**Benchmarks for hemorrhage subtype classification:**
So far in this Kaggle competition, the benchmarks are as the following chart.

| Models | Training Image Size | Metrics |
|---|---|---|
| ResNeXt-101 32x16d | 256x256 | 0.086 |
| EfficientNet B0 | 256x256 | 0.077 |
| EfficientNet B1 | 224x224 | 0.073 |
| VGG19 | 224x224 | 0.073 |

[8] "Exploring the Limits of Weakly Supervised Pretraining." 2 May. 2018, https://arxiv.org/abs/1805.00932.
[9] "EfficientNet: Rethinking Model Scaling for Convolutional ...." 28 May. 2019, https://arxiv.org/abs/1905.11946.
[10] "Fixing the train-test resolution discrepancy." 14 Jun. 2019, https://arxiv.org/abs/1906.06423.
[11] "Image Classification on ImageNet - Papers With Code." https://paperswithcode.com/sota/image-classification-on-imagenet.

| | | |
|---|---|---|
| Resnet50 | 256x256 | 0.089 |
| inceptionV3 | 224x224 | 0.079 |

## 4. Theoretical workflow

Workflow is very important in machine learning project. this project's will follow the following workflows and pipelines:

- **Data Exploration and Analysis**
  Understanding the dataset, Create a meta dataset from DICOM image Dataset
  Read DICOM image and analysis image data, Windowing Scaling, Normalizing Dataset

- **Pre-processing Data(training images)**
  Remove useless images information, resample the dataset to a small dataset to quick prototyping, crop images that only include important information, e.g. only have brain tissues, rescaling into 256*256 px image and reduce the huge data size for training

- **Data Refinement and Data Augmentation**
  Apply the data augmentation methods to improve the model generalization and reduce the overfitting.

- **Model Evaluation, Validation**
  Train and Valid dataset split, prepare DataBunch(fastai api) for model, Use the pretrained xResNet50 Model to train and fine tune,optimization

- **Model Justification and Benchmark Comparison**
  This project will choose the `Resnet50` as the baseline model. Will compare each model's performance with the above benchmarks.