# Respiratory Disease Detection and Report Generation Using Deep Learning Techniques

**Submitted in partial fulfillment of the requirements**

**of**

**the degree of**

Bachelor of Technology

by

**Ankit Agrawal: 171210010**

**Keya Shukla: 171210033**

Supervisor:

**Dr. Rishav Singh**

Department of Computer Science & Engineering

**National Institute of Technology Delhi**

2021

# APPROVAL SHEET

This project work entitled “Respiratory Disease Detection and Report Generation Using Deep Learning Techniques” by Keya Shukla and Ankit Agrawal is approved for the degree of Bachelor in Technology

## Examiners:

Dr. Karan Verma

Dr. Sonia Sharma

## Supervisor:

Dr. Rishav Singh

## Director:

Dr. Satish Kumar

Date: 2nd May 2021

Place: Narela, New Delhi

# ACKNOWLEDGEMENT

The dissertation has been prepared with sincere effort that we have put in, but it would not have been possible without the kind support and help of many individuals of National Institute of Technology, Delhi. We would like to express our heartfelt gratitude to all of them for providing us such an opportunity to learn. We are highly thankful to the honorable Director for providing us with the best of facilities for research and development work. We are incredibly grateful to Dr. Rishav Singh, our supervisor and project in charge, for guiding us throughout the project development. His knowledge and experience in this field helped us throughout our project development. His guidance at every point of time was what helped us the most. His leadership and guidance helped us immensely whenever we had to go back to the drawing board. We would also like to thank all the faculty members who have taught us and built the base of knowledge on which we can take up any complex task and solve it.

Ankit Agrawal: 171210010

Keya Shukla: 171210033

# DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Ankit Agrawal: 171210010

Keya Shukla: 171210033

Date: 2nd May 2021

# ABSTRACT

In times where social distancing is a must but other ailments need to be tended to, Artificial Intelligence has found yet another ground-breaking application to provide aid to numerous communities, called Tele-testing. Tele-testing is the means of clinical impressions from a training set of suitable samples to form a working diagnosis which can support a doctor in clinically diagnosing a medical condition with the help of technical AI tools. It has had a huge impact in providing much-needed medical attention to rural areas in isolation from medical care as well as communities that are bound by lockdowns imposed due to pandemics. For this project, we have chosen to diagnose among fourteen respiratory ailments, namely, Atelectasis, Consolidation, Infiltration, Pneumothorax, Edema, Emphysema, Fibrosis, Hernia, Mass, Nodule, Cardiomegaly, Pleural thickening, Pneumonia, and Effusion, the data sets of which are available in ample amounts that would help us in training our model to achieve high accuracy in classifying the diseases. Our model intends to take a chest X-Ray image as an input, detect traits common to the following fourteen respiratory ailments, classify correctly and generate the corresponding diagnosis report. For this purpose, we will make use of the VGG-19 model, and using Graph Based Convolutional Network, we will generate the corresponding report.

# TABLE OF CONTENTS

**1.1 Introduction**

**1.2 Review of Literature**

# LIST OF FIGURES

# LIST OF TABLES

# CHAPTER 1.1: INTRODUCTION

## 1.1.1 General Introduction

Medical image diagnosis is a prevalent method to detect a wide spectrum of diseases in the human body. It involves a careful inspection of the medical images by well-trained physicians, often followed by elaborate report-writing and corresponding inference. However, such methods can be error-prone if done by inexperienced physicians, or even time-consuming and tedious for experienced physicians. Many regions in the world do not have access to world class health facilities, hindering the chances of people living there to have an accurate and early diagnosis of possible, medical diseases.

Artificial Intelligence may be used to complement physicians and doctors in diagnosing medical images of patients. With the abundance of patient data available, an AI model can learn to infer which conditions are symptoms of a particular disease. In places where healthcare facilities are poor, these AI machines can help doctors spot certain anomalies in the image. Such AI models are also very quick, being able to diagnose 100-150 images per second, with no dip in performance over time. Researchers have used and evaluated AI models on a variety of tasks, showing that they can diagnose medical images better than experienced doctors, and in much quicker time. Various other methods based on the application of neural networks have been tested with varying levels of accuracy on the task of medical image diagnosis.

Despite the advantages offered by such AI models, they have not been integrated in hospitals, or put to use for medical diagnosis. A critical field like healthcare leaves no room for mistakes, since even a small one could be fatal for the patient. A major hurdle in integrating the said AI models is the lack of trust that doctors have in them. The models must account for the predictions, making it difficult for doctors to accept such solutions in practice. Providing better feedback, including the uncertainty in the predictions and reasoning behind a certain diagnosis would help assure doctors of the reliability of such models.

A robust and comprehensive explanation module, if integrated properly into current deep learn ing networks designed for the task of automated medical diagnosis generation, would take a step towards building trust in AI systems to complement and supplement doctors in their analysis. An ideal system would be one which provides proper insights into the underlying factors behind a particular diagnosis, which a doctor could look at to improve his diagnosis. Note that we do not intend to replace doctors with AI systems, rather, we wish to use AI as a helping tool.

In this project, we seek to perform image recognition and classification to diagnose amongst fourteen diseases by analyzing chest x-ray images using deep learning techniques and generating the corresponding report based on the diagnosis.

We also aim to tackle the lack of explainability in current Artificial Intelligence (AI) medical image diagnosis methods. We work on integrating a strong explanation module into AI models, which can provide a comprehensive understanding of how an AI model is thinking when generating a diagnosis for a medical image, in turn supplementing doctors in faster and more accurate disease detection.

### 1.1.2 Problem Statement

We aim to develop an accurate model for the current Computer Aided Diagnosis (CAD) methods and diagnose amongst the following fourteen thorax diseases:

1. Atelectasis
2. Consolidation
3. Infiltration
4. Pneumothorax
5. Hernia
6. Mass
7. Nodule
8. Cardiomegaly
9. Edema
10. Emphysema
11. Fibrosis
12. Pleural thickening
13. Pneumonia
14. Effusion

by analyzing chest x-ray images using deep learning techniques and generating the corresponding diagnosis report, in turn supplementing doctors in faster and more accurate disease detection.

### 1.1.3 Empirical Study (Field Survey, Existing Tool Survey, Experimental Study)

For classification, we were fortunate enough to have several research papers to guide us as well as our own domain knowledge in the subject. However, this was not the case for the report generation module of the project. To explore the overall advancements in the topic and discover the most widely used evaluation metrics and state-of-the-art approaches, we studied the following existing survey:

- A Survey on Biomedical Image Captioning [7]

| Team | Year | Approach | BLEU |
|---|---|---|---|
| Liang et al. | 2017 | ED+IR | 26.00 |
| Zhang et al. | 2018 | IR | 25.01 |
| Abacha et al. | 2017 | CLS | 22.47 |
| Su et al. | 2018 | ED | 17.99 |
| Rahman | 2018 | ED | 17.25 |

*Table 1. Top-5 Participating Teams at ICLEF-CAPTION Competition Ranked by Average BLEU Metric, Approach and Year*

From this survey it is found that the following broad approaches and their respective variations were employed by researchers in the then burgeoning field of image captioning in an annual task called ICLEF-CAPTION:

- **Encoder-Decoder**
  - This approach was followed by Liang et al. (2017) as well as Karpathy and Fei-Fei (2015), who used a VGG-Net as the CNN encoder and LSTM as the RNN decoder. This was trained on three caption lengths and an SVM classifier was used to find the most optimum selection of decoder. To find the most similar result of caption and aggregated image, 1-Nearest-Neighbor approach was used. A simpler approach of encoder-decoder was also followed by Rahman (2018) and Su et al. (2018)

- **Image Retrieval**
    - Zhang et al. (2018) made use of this approach to retrieve images using the Lucene Image Retrieval Software (LIRE) and joined the captions of the three most similar retrieved images, which resulted in the new caption.

- **Classified UMLS Concepts**
    - This approach was carried out by Abacha et al. (2017) in their submission wherein GoogLeNet was used to identify UMLS concepts. The summed aggregation of UMLS semantic groups was returned as the caption.

ICLEF-CAPTION also made use of various different evaluation metrics, two of which are discussed below and will be used in our project:

- **BLEU**
    - One of the most popular evaluation metrics used in the task, it measures the overlap of the word n-gram amongst the generated caption and the ground truth caption. In order to punish short captions, a "brevity penalty" was included. Four variations of the BLEU metric, namely BLEU-1, BLEU-2, BLEU-3 and BLEU-4 (corresponding to unigrams, bigrams, trigrams and 4-grams respectively) were averaged to be used as ICLEF-CAPTION's official metric.

- **ROGUE**
    - Out of the many variations of the ROGUE metric, ROGUE-L is found to be the most popular in medical image captioning according to this survey. ROGUE-L pertains to the ratio of the longest common subsequence between machine generated and reference human descriptions, to either ROGUE-L precision, ROGUE-L recall or ROGUE-L F-score (which is a combination of the previous two).

## 1.1.4 Approach to the Problem

After conducting efficient literature survey and keeping the results of the survey in mind, we have identified the following steps, that are a hybrid of several different techniques used in classification and image captioning researches,  to help us in achieving our aim:

- Carrying out a comprehensive literature survey to find gaps in the existing research of deep learning models for medical images.
- Devising a novel module which can be inculcated into existing state-of-the art deep learning networks, taking cue from other success in the other disciplines.
- Implementing strong baselines to compare our novel ideas with the current state-of-the-art models, which we have coded and trained from scratch.

### 1.1.4.1 Technology & Platform Used

**Hardware**

We had access to the following AWS system:

- DGX-1 server
- Ubuntu 18.04
- 8 NVIDIA Tesla V100 GPUS (each of 32GB)
- 512GB RAM

**Software**

We used the following environment and libraries for coding and training the model:

- Python 3.5
- PyTorch 1.3.0
- TorchVision 0.4.1
- Matplotlib for visualization
- NLTK for primary BLEU score evaluation
- CoCoCaption for final evaluation

## 1.1.5 Support for Novelty

Medical image diagnosis is a prevalent method to detect a wide spectrum of diseases in the human body. The broad aim of Artificial Intelligence in the medical field is to improve the quality, efficiency and accuracy of certain predominantly manual medical procedures, surgeries and diagnoses. By incorporating data driven medicine and intelligent algorithms we can hope to eradicate frequent occurrences of misdiagnosis and also improve the course of treatment by treating the root cause of the illness rather than the existing symptoms. The advancement of AI in medicine marks an important milestone in the era of technology and with this project we hope to contribute to this cause.

This project would increase medical accuracy of the procedures used to diagnose respiratory conditions, improve time taken to cure or detect symptoms and also provide suitable access to remote locations that do not have proper medical facilities nearby.

Another application where this project would prove to be useful is for pandemics and epidemics where tele-testing using deep learning techniques to diagnose diseases and generate digital reports would be the need of the hour to curb spread of viruses that cause global distress, such as the pandemic caused by the novel coronavirus.

## 1.1.6 Comparison of Other Existing Approaches

### 1.1.6.1 Deep Convolutional Neural Networks for Chest Disease Detection

- Comparison amongst three models BpNNs, CpNNs and CNN.
- Results achieved using a small batch of data.
- Fails to detect more than one disease.

| Network Model | Training Data (70%) | Validation Data (30%) |
|---|---|---|
| BpNN | 99.19% | 89.57% |
| CpNN | 85.23% | 84.71% |
| CNN | 100% | 92.4% |

*Table 2. Summary of Deep Convolutional Neural Networks For Chest Disease Detection*

### 1.1.6.2 Deep Learning Model for Thorax Disease Detection

- Pre-trained ResNet50 model
- X-ray images cropped to extract the rib cage part.
- Model re-trained on NIH-14 Dataset
- Total training time 29-30hrs (*approx*)

| Training data | Validation data |
|---|---|
| 93.03% | 97.49% |

*Table 3. Accuracies Achieved By Deep Learning Model For Thorax Disease Detection*

### 1.1.6.3 Using Convolution to Analyze X-Ray Radiographs for Multilabel Classifications of Thoracic Diseases

- Dense Convolution Neural Network
- X-ray images were downscaled to make memory efficient.
- Rectified Linear unit (ReLU) was used as the activation function
- Max pooling along with sigmoid function to perform independent binary classifications.
- Accuracy achieved on the validation was 92.94 percent.

### 1.1.6.4 An Automatic Detection of Major Lung Diseases Using Chest Radiographs and Classification by Feed Forward Artificial Neural Network

- Artificial Neural Network model to detect amongst three lung diseases.
- Segmentation was done through the use of intensity and discontinuity edge detection to find the boundaries of the lungs.
- Intensity converted a grayscale image into a binary image, and discontinuity detected edges in the image.
- Image classification occurs through the use of a feed-forward and back propagation neural network.
- Accuracy of 92% was achieved.

### 1.1.6.5 Deep Learning for Detection and Localization of Thoracic Diseases Using Chest X-Ray Imagery

- Transfer learning method with a ResNet 18 and its pre-trained weights.
- Modification in the last layer (14 neurons used instead of 1000)
- 10 crop technique used to increase the accuracy
- Accuracy of 84.94% was achieved.
- An AUC score of 0.8494 was achieved.

# CHAPTER 1.2: REVIEW OF LITERATURE

## 1.2.1 Summary of Papers Studied

We did a literature review of some successful deep learning methods used for detection and classification of diseases using the respective medical images. It has also helped us find gaps in the research done until the present time, which could be our potential direction of work.

Following is a brief summary of the research papers we reviewed:

Abiyev and Ma'aitah used Convolutional Neural Networks (CNNs), backpropagation neural networks (BpNNs), and competitive neural networks (CpNNs) to diagnose various chest diseases with single label classifications and achieved the following results: In the case of BpNN the accuracy for the training set was 99 percent (approx) and 90 percent (approx) for validation data. For CpNN, 85 percent (approx) on training and validation data. In the case of CNNs they achieved the highest accuracy of 100 percent on the training data and 93 percent on the validation data, However they achieved this results using a small batch of 620 images and their model lacked to detect more than one diseases in a X-ray radiograph. [1]

Shadeed, *et al*, in their research for thorax disease detection used a pretrained ResNet-50 for diagnosing thorax diseases. The X-ray images were cropped to extract the rib cage part and the model was re-trained on the NIH-14 dataset to detect one amongst fourteen chest diseases. The accuracy achieved for training data was 93.03 percent and for validation data was 97.49 percent. The total training time the proposed model took was 29-30 hrs. [2]

Zhan, *et al*, in their research for analyzing X-ray radiographs for Multi- Label Classification of Thoracic Diseases used a dense convolutional neural network (CNN) to process and analyse the large quantity of data. The images in the dataset were downscaled to make it more memory efficient. The accuracy achieved on the validation was 92.94 percent. Rectified Linear unit (ReLU) was used as the activation function, Max Pooling along with the sigmoid function at the very end to perform independent binary classifications. [3]

Khobragade, *et al*, made use of an artificial neural network (ANN) to process images of chest X-rays and identify lung diseases. The proposed method had the following steps: image processing, segmentation, feature extraction, and image classification. Image preprocessing removes irrelevant data on the radiograph, recovering useful information, strengthening the region of interest, and simplifying its features. The segmentation was done through the use of intensity and discontinuity edge detection to find the boundaries of the lungs. Intensity converted a grayscale image into a binary image, and discontinuity detected edges in the image. Image classification occurs through the use of a feed-forward and back propagation neural network. [4]

Wang, *et al,* in their research using ChestX-ray8 dataset used Deep Convolutional Neural Network (DCNN) to detect amongst eight common thoracic diseases and employed Natural Language Processing techniques to label the data from their archives based on radiological reports attached to the images. They also utilized a multi-label classification and localization framework to generate bounding boxes around the locations of thoracic ailments. However, variation in the accuracy for each disease from as low as 16 percent for Nodule to as high as 99 percent for cardiomegaly was observed making the detection unreliable. [5]

Rakshit, *et al,* in their research adopted Transfer learning method and a Res-Net 18 and its pre-trained weights based on the ImageNet challenge are used here as a feature extractor. Only the last layer of the classifier is modified. Instead of 1000 neurons in the case of ImageNet challenge, here 14 neurons have been used where each neuron corresponds to each of the classes. Also the ten-crop technique was used in order to increase the accuracy. The accuracy achieved using this technique was 84.94 percent. [6]

For the literature review of the Report Generation module, following were the sub-topics that we paid close attention to:

#### 1.2.1.1 Fully Supervised Image-Language Attention Models

Some proposed models in our literature review had a simple image model to process the given X- Ray or cancer images, giving a vector representation as output. This representation was then used by the language model to generate a medical diagnosis. To inculcate explainability, an attention mechanism was used. The higher the attention weights of a particular region in the given input, the higher the probability of that region having a significant role in the generation of a particular report. Also, these methods used fully supervised training, i.e., they had access to well-labeled datasets. The following were the specific papers we read:

- MDNet: A Semantically and Visually Interpretable Medical Image Diagnosis Network [8]
- TieNet: Text-Image Embedding Network for Common Thorax Disease Classification and Reporting in Chest X-rays [9]
- On the Automatic Generation of Medical Imaging Reports [10]

#### 1.2.1.2 Weakly Supervised Image-Language Attention Models

There are not many large datasets for automatic medical report generation. The very few that are available have a high probability of some errors in labeling, as well as bias introduced in the way the diagnosis has been written. To alleviate all these problems, some weakly supervised and semi supervised techniques have been proposed. The following were the specific papers we read:

- Towards Automatic Report Generation in Spine Radiology using Weakly Supervised Framework [11]
- CheXpert: A Large Chest Radiograph Dataset with Uncertainty Labels and Expert Comparison [12]

#### 1.2.1.3 Reinforcement Learning Based Image-Language Attention Models

Fully, weakly or semi supervised techniques need the loss function or optimization function to be continuous in nature. Some of the metrics that were used for evaluating the performance of medical diagnosis generators were not continuous. Hence, reinforcement learning was used to optimize deep learning models over such non-continuous metrics. The following were the specific papers we read:

- Hybrid Retrieval-Generation Reinforced Agent for Medical Image Report Generation [13]
- Clinically Accurate Chest X-Ray Report Generation [14]
- Show, Describe and Conclude: On Exploiting the Structure Information of Chest X-Ray Reports [15]

## 1.2.2 Integrated Summary of Literature Studied

After an extensive literature survey on medical image recognition, classification and diagnosis generation using Deep Learning, we had brainstormed some potential ideas and directions to pursue which would ensure that there is a potential to achieve a higher accuracy than the current state of art of medical diagnosis for a particular X-Ray. The following are the possible directions we thought of pursuing:

- Cropping and Downscaling the images in the dataset to make it more memory efficient.
- Using Deep Convolutional Network models as they have proved to achieve better classification of images with high accuracy.
- Using VGG-19 and possibly use ensemble methods to achieve higher accuracy.
- Using SVM as a classifier instead of softmax as it trains faster and produces stable results.
- Using loss functions derived from semantic metrics for training the models. This helps the models learn to produce clinically accurate and semantically more meaningful reports.
- Using stacked attention modules and dual attention modules to help the model provide better attention maps that would produce more accurate reports as well as provide better feedback on the predictions.
- Using transformers [16] instead of traditional RNN architectures, as they have provided better results on a variety of language tasks.
- Learning a joint space to learn aligned visual and semantic features for better correlation between the X-Ray and generated medical diagnosis.

To this end, we read more papers to develop a concrete idea. A review of some of the prominent ones are mentioned subsequently.

### 1.2.2.1 Classification

In the world of Image Recognition, the VGG model is one of the most popular and preferred models. It started out as one of the competitors and subsequent winners of the ImageNet Large Scale Visual Recognition Challenge, where it gained popularity for its remarkable top-5 test accuracy of 92.7%. Trained on the same dataset as the AlexNet, VGG proved to be superior to it owing to its architecture consisting of sixteen layers in VGG16 and nineteen layers in VGG19, much deeper as compared to AlexNet’s eight-layer architecture, in coordination with its smaller filter sizes. To elaborate, VGG utilized 3x3 convolution layers, which is the smallest possible combination to cover the corner values, with a padding and stride value of one. The intuition behind this was that a stack of smaller filters has the same receptive field as the traditional 7x7 convolution layer resulting in lesser parameters and increased depth. Therefore, its combination of small filters and deep layers made it a reliable, faster and more robust model.

The architecture of VGG-Net comprises a total of sixteen layers, with the input being an image of dimensions 227x227. This input is then sent to the stack of 3x3 convolutional layers. There are a total of five 2x2 max pooling layers with a stride value of 2 that succeed some convolution layers. The architecture also shows three fully connected layers out of which one is an output layer. The first two layers have 4096 channels and the output layer has a value of a 1000, corresponding to the 1000 ImageNet classes.

Similarly, a variation of the VGG model was created that had nineteen layers, instead of the traditional sixteen-layer architecture, and was called the VGG19 model. Its architecture consisted of sixteen convolutional layers, a provision for spatial padding, five max pooling layers and an output layer with the softmax activation function. This model also made use of the ReLU activation function for the rest of its layers. This model was prominently used in problems of Transfer Learning.

### 1.2.2.2 Report Generation

**Transformers as Language Models**

[17] recognized and tries to alleviate three major problems found in previous works:

- The visual encoders found in previous methods rely on top-down approaches to find suitable representations of the input images.
- Almost every previous work relies on recurrent architectures to predict the sentences from visual features. Recurrent architectures are unable to fully utilize the computation resources available due to the inherent time dependence of predictions in them.
- Continuous Maximum Likelihood Expectation is not able to account for the discrete task of language generation.

This work tries to incorporate bottom-up features from the input images to get better predictions, at the same time utilizing transformers proposed by Vaswani et. al. [17] . At the same time, the model proposed uses reinforcement learning to eliminate the exposure bias and the discrepancy between MLE and metrics used to evaluate the predictions. As shown in Fig. 4, the model first uses a Region detector (based on DenseNet [18]) to extract the relevant regions of the image. The DenseNet has been pretrained on the Chest X-Ray 14 dataset as for domain adaptation. These regions are then fed into a visual encoder (enclosed in the yellow box) to get the attended features of each region. This visual encoder acts as the top-down attention module, and is composed of 3 stacked multi-head self-attention layers and a fully connected layer.

These attended regions are then fed into a Captioning Decoder. This module consists of three parts: a self-attention module, a cross modal attention module and a simple feed-forward module. The decoder generates a separate sentence for each attended region from the visual encoder. The whole model is trained using a semantic loss using the CIDEr [19] score as a reward for word generated. The model uses the IU Chest X-Ray dataset for training and evaluation.

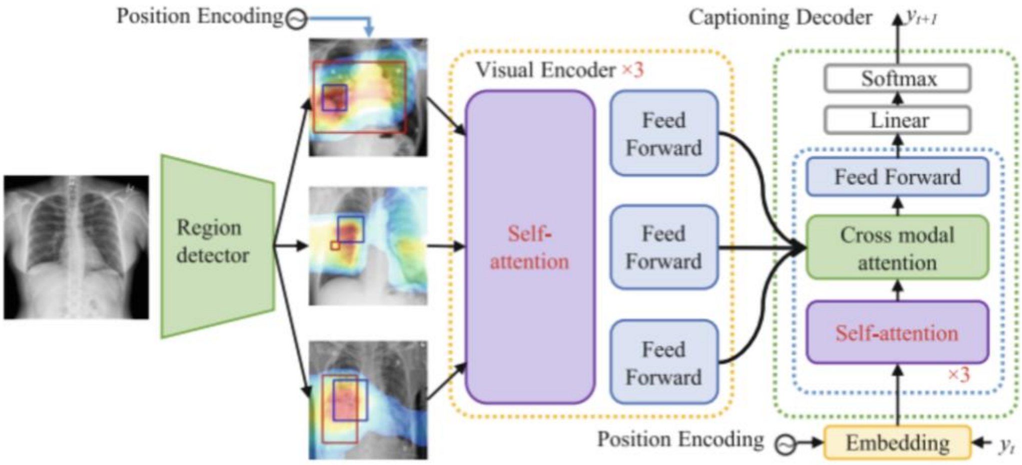


*Fig 1. Model Proposed in [17]*

### Unsupervised Multimodal Representation Learning

[20] exploits joint embeddings space of the visual and semantic features to allow for cross-domain retrieval and conditional report generation to provide accurate results. The main contributions of this work are as follows:

- Establish evaluation metrics and baselines for embedding-based report generation via retrieval and distance metrics in the embedding space.
- Document the relation between supervision level and representation quality in joint embedding spaces.
- Document the effect of using different sections of the report on the learned representations.

The text is reduced to embeddings by four methods: TF-IDF over bi-grams, Glove word embeddings, sentence embeddings and paragraph embeddings. The visual features are obtained from the pre-final layer of a DenseNet-121 model and are further reduced to obtain 64 dimensional vectors. The paper reports the results for five different types of alignments: linear transformations, adversarial domain adaptation, procrustus refinement, semi-supervised alignment and orthogonal regularization. Comparison between different methods are provided on the MIMIC-CXR dataset [21]

### Multi-view Image Fusion and Medical Concept Enrichment

The authors of [22] propose to solve the task of automatic medical diagnosis generation with the help of a generative encoder-decoder model, focusing on a subset of diseases in the chest. The encoder used is a deep CNN (ResNet-152) whereas the decoder used is a hierarchical LSTM. There are primarily two stages in their framework:

- Multi-view visual representation
- Medical concepts

### Multi-view Visual Representation

Generally, datasets contain both the frontal and lateral X-Rays of a patient, but still treat it as two different cases, whereas ideally, the lateral picture should be complementing the frontal one. The encoder extracts features from the frontal and lateral parts, providing two separate feature vectors. The combination of these two features is not as simple as just concatenating or adding them, since each feature may contribute a different amount to the detection of a disease, with the frontal usually having more weight. Hence, an attention mechanism is used, along with a sentence-level LSTM, to combine the two X-Ray features into a multi-view representation. To make sure that the sentence-level LSTM learns consistent features, the authors use a cross-view consistency (CVC) loss.

### Medical Concepts

Many a times, the words generated by the deep learning network are repeated according to the statistics of the dataset it was trained on. To ensure that the medical-related contents such as diagnosis and organs are correct, the encoder is fine-tuned to extract the most frequent medical concepts given an X-Ray image. This ensures a rich descriptive semantics are learned by the decoder, since the learned concepts are fused with each decoding step by the word-level attention model.

# CHAPTER 1.3: REPORT ON PRESENT INVESTIGATION

## 1.3.1 OVERALL DESCRIPTION OF PROJECT

For this project, we have devised a custom approach to the problem of disease detection, its subsequent classification and generation of a comprehensive report. Our approach will be divided into 4 blocks, namely: ‘Literature Survey’, ‘Data Preprocessing’, ‘Classification’ and ‘Report Generation’. An in-depth analysis of these blocks is given in the subsequent subsections.

Since we decided to work with an extremely large amount of data, it was necessary to check for any major discrepancies, contradictory formats, missing values, as well as keep the workability of the dataset in mind. As a solution to all these issues, it was decided that the dataset will have to be balanced, by keeping an equal number of chest x-ray images for each disease. As for the quality standards of each image, the CLAHE algorithm was applied to each chest x-ray to enhance resolution and improve contrast (Section 1.6). Ultimately, the entire dataset was converted into an HDF5 format to make it more tractable. The aforementioned steps were carried out in the Data Preprocessing Module (Section 1.3.1.1).

The next major aspect of the project was the Classification Module. One of the major challenges was to access the data from the HDF5 created earlier. This is discussed as supplementary material in Section 1.6. Before classification can be performed, meaningful relationships between the different features of the dataset are determined by performing exploratory data analysis. This gives us a clearer insight into what we should look for while tuning the model to aid the classification process. Finally, classification is performed using VGG-19 as the model. This approach is discussed in more detail in the Classification Module subsection (Section 1.3.1.2).

The final module of this project is the Report Generation Module, wherein we discuss the customised approach of using a graph based convolutional neural network to achieve the task of creating a comprehensive report from the chest x-rays after classification. (Section 1.3.1.3)

### 1.3.1.1 Data Preprocessing Module

The goal of this block is to clean, balance and transform the data into a workable format. The original dataset has 1,12,120 chest x-ray images which, given the current computational resources, is too bulky for the scope of this project. For this reason, we have decided to reduce the size of the dataset.

Another issue arose with this decision: the dataset cannot be split randomly based on the required number of images as the resulting mini-dataset would become skewed.

To tackle this problem, we devised an approach wherein we split the dataset into ‘buckets’ and selected an equal number of workable images from those ‘buckets’. In implementation, these buckets were referred to as subsets of the dataset that consisted of 18000 images under each disease category that were randomly selected. In this way, we managed to avoid wrongly labeled images as well as null valued images. This approach made it possible for us to parse this dataset with our current resources and would also help in obtaining a better accuracy due to the removal of null and incorrect values. After this, we sought out to devise another strategy to make this dataset more accessible.

The next goal of this block is to convert the dataset into a workable format. For this project, we have converted the mini-dataset into an HDF5 format. HDF5 (Hierarchical Data Format version 5) is a format that emulates the organization of files in a computer. Mainly used to organize large data into a file directory. The data in an HDF5 can be structured in various ways into files as per the user. An HDF5 eases accessibility and manipulation of the data.

At this point, all the images in the Chest X-Ray dataset are very different from each other and a standard is required to make them all similar in contrast as well as size. For contrast, the CLAHE algorithm is used. And for size, the images are downscaled from 256x256 to 128x128, which is the new standard size of the images in the dataset. The final HDF5 format is given below:

| | |
|---|---|
| Atelectasis | 18000 , float64 |
| Cardiomegaly | 18000 , float64 |
| Consolidation | 18000 , float64 |
| Edema | 18000 , float64 |
| Effusion | 18000 , float64 |
| Emphysema | 18000 , float64 |
| Fibrosis | 18000 , float64 |
| Finding Labels | 18000 , S100 |
| Follow-up | 18000 , int64 |
| Hernia | 18000 , float64 |
| Image Index | 18000 , S16 |
| Infiltration | 18000 , float64 |
| Mass | 18000 , float64 |
| Nodule | 18000 , float64 |
| Original Image Height | 18000 , int64 |
| Original Image Pixel Spacing x | 18000 , float64 |
| Original Image Pixel Spacing y | 18000 , float64 |
| Original Image Width | 18000 , int64 |
| Patient Age | 18000 , int64 |
| Patient Gender | 18000 , S1 |
| Patient ID | 18000 , int64 |
| Pleural Thickening | 18000 , float64 |
| Pneumonia | 18000 , float64 |
| Pneumothorax | 18000 , float64 |
| View Position | 18000 , S2 |
| images | (1800, 128 ,18,1) uint8 |
| Path | 18000 , S43 |

*Table 4. Parameters of the dataset after resizing the images*

## 1.3.1.2 Classification Module

The goal of this block is to classify the images based on the disease(s) present in them. For this purpose, the HDF5 file had to be accessed, a dataframe was to be created out of the labels and images, the model had to be trained and evaluation metrics were used to test the model and its working.

### Exploratory Data Analysis

Exploratory data analysis is a key step before any sort of computation or operations can be performed on the data. EDA is usually utilized by data scientists in the industry to investigate the data sets and perceive cause-effect relationships within the sample considered. Detecting outliers and discrepancies in the data also becomes very easy when carried out in a pictorial manner, through graphs plotted using the various libraries available in Python. In this project, EDA helps us to recognize some additional insights, reasonings and patterns in the data that solidify the claims made by the model after training. We made some key analyses using the ‘Patient Age’, ‘Patient Gender’, and ‘Finding Labels’ columns, using Python’s ‘Seaborn’ and ‘Matplotlib’ libraries.

For administrative purposes, the plot below makes it clear that constituting the majority, at around 3100 patients sampled, were males, and around 2500 were females. This shows that the sample of patients considered is not too skewed and is capable of giving a practical insight that can be useful for further prognosis of other patients not included in this sample.

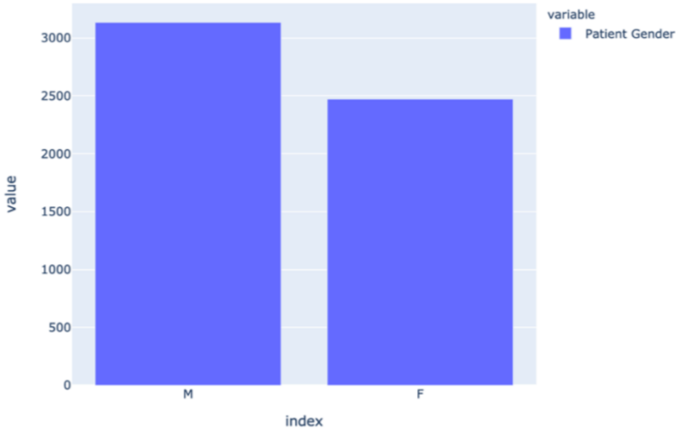
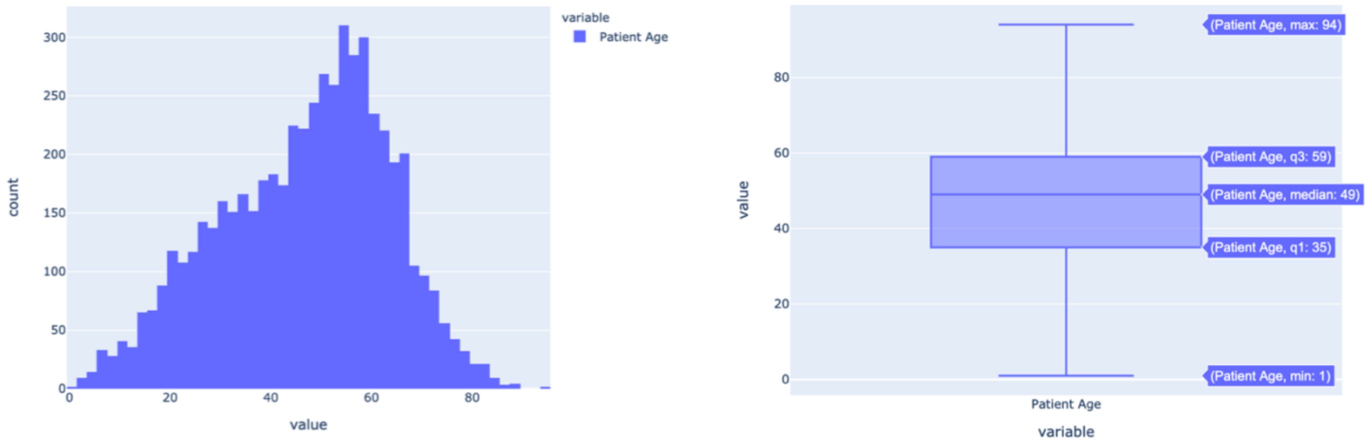


*Fig 2. Count Plot of Patient Gender*

The plots below show that chest x-rays spanning over almost all age groups were considered. This doesn't limit the analyses, diagnoses or conclusions made in this project to any specific age range, thus adding to the practicality of the solution. The corresponding box plot further shows some more key insights pertaining to the patients' age.



*(a)*

*(b)*

*Fig 3. (a). Histogram of Patient Age and (b). Box Plot of Patient Age*

Lastly, the frequency of each disease can be seen in the plot below. Since all chest x-rays with the disease label 'No findings', corresponding to either label-less or healthy patients, were removed, it is safe to assume from this plot that 'Infiltration' is the most common ailment present in the sampled chest x-rays. It is also safe to assume that not many cases of Hernia were identified from these chest x-rays manually. The varied frequencies of all these diseases would impact the corresponding accuracy of each disease as well, once the model is trained and evaluated.

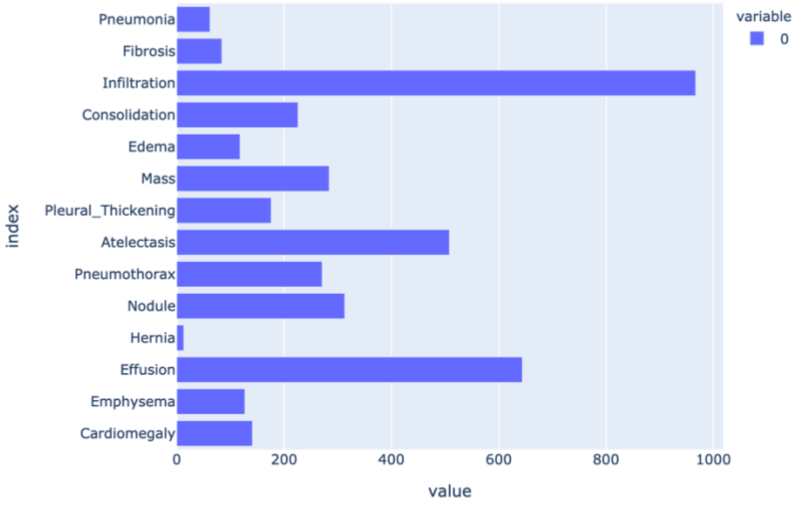


*Fig 4. Bar Plot of Disease Frequency*

We have also fixed the issue of data imbalance, as can be seen in Figure 9. (a) within the smaller dataset that we created. Now the data, as seen in Figure 9. (b), is not skewed even in terms of the absence or presence of each disease, as there are equal occurrences of both instances before training.

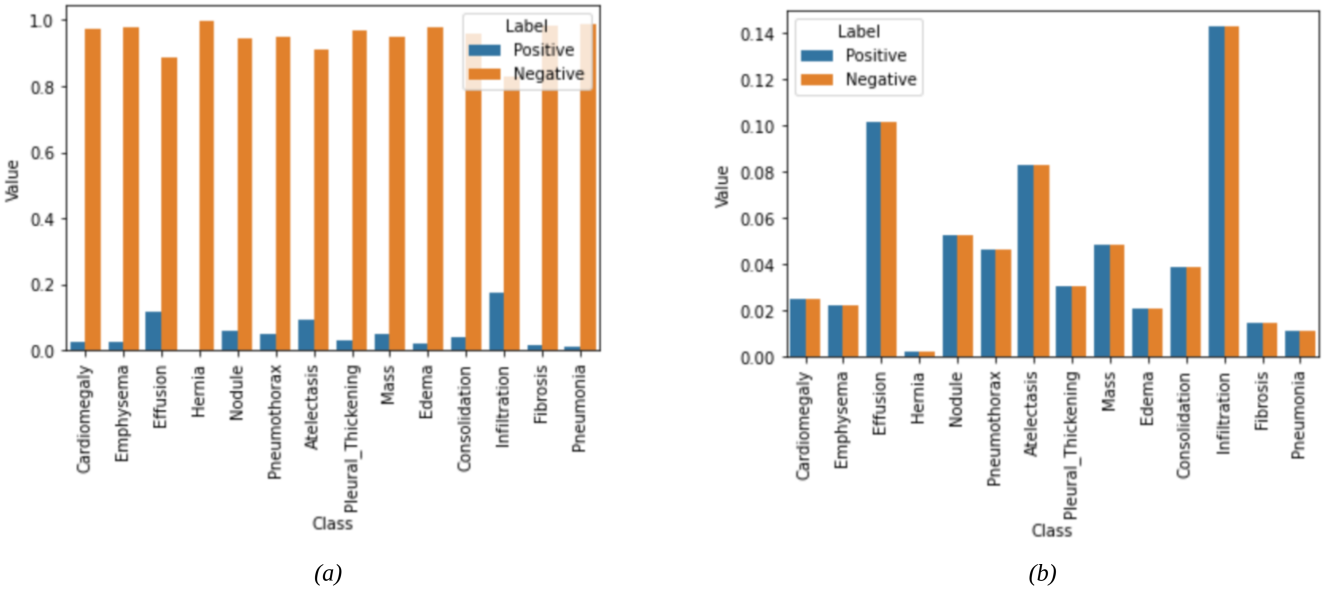


*Fig 5. (a). Bar Plot of Positive and Negative Disease Labels Before Equalization and (b). Box Plot of Positive and Negative Disease Labels After Equalization*

## Classification Using VGG-19

The last step of this block is to perform the actual classification. For this purpose, we made use of a pre-trained VGG-19 model using PyTorch. The dataset was split into training, testing, and validation dataset in the following ratio: 5000:303:303, respectively.

We made use of max pooling as our pooling operation since the chest x-ray images in our dataset had a black background.

With a batch size of 32, the dataloaders for training, testing and validation datasets were created, and the model was defined. We set the last fully connected output layers and add them to the pre-trained Sequential model. The model was trained on the Adam optimizer for 15 epochs. The training process took approximately 8-9 hours to complete, but had it been for better computational resources, we could've trained on more epochs in a relatively shorter amount of time.

We have calculated the class-wise accuracy of each disease for the training, testing and validation datasets. The results and accuracies pertaining to each dataset are discussed in the next chapter of the report.

As for the final output, due to shortcomings in our dataset and computational resources, we obtained some positive and some negative examples of classifications which are visible in the figure below. Based on the corresponding plot of the chest x-ray, the most likely disease is obtained and displayed along the image as the ground truth of classification. The positive example (Figure 10 (a)), clearly shows that the levels of Fibrosis and Nodule are the highest and cross the defined threshold, and so the ground truth displays these diseases as its diagnosis. On the other hand, Figure 10 (b), shows the negative example in which the model is unable to figure out which disease is present in its corresponding chest x-ray, as all are found to be equally likely. Therefore, its ground truth is given as 'No Findings', and our model fails here.

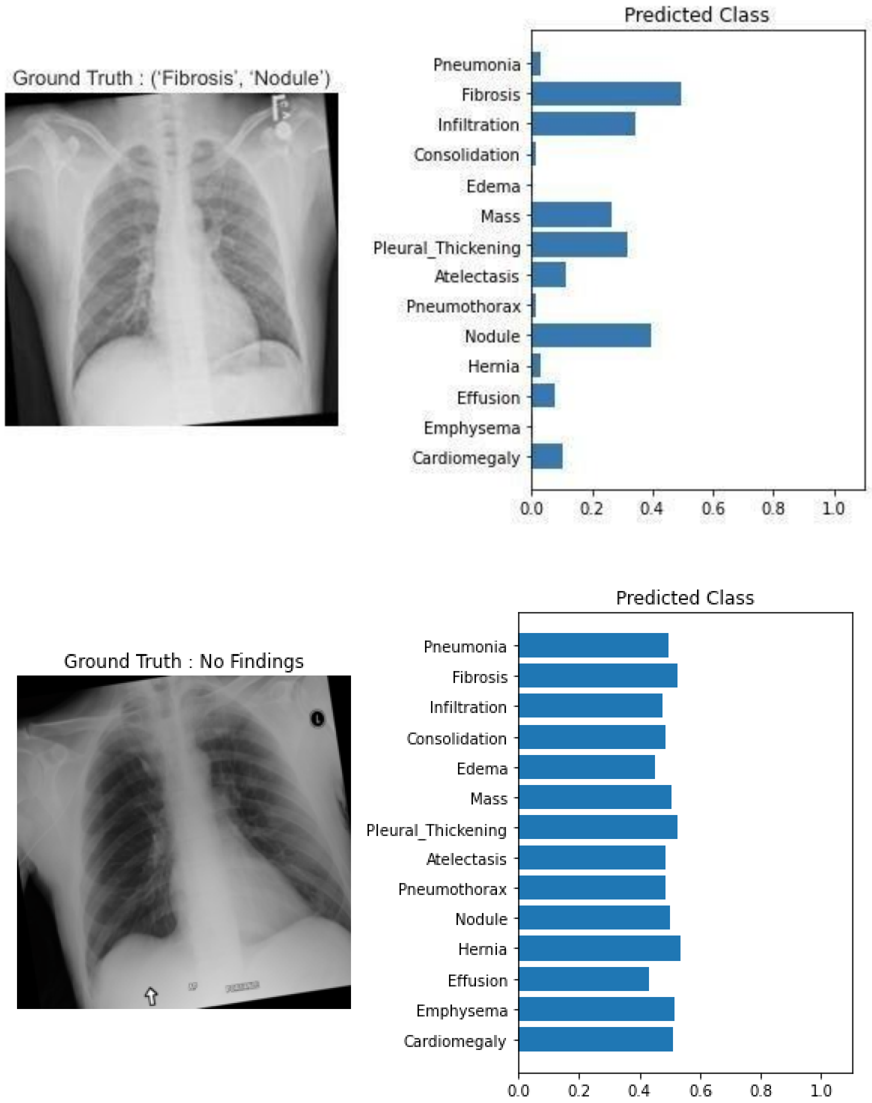


*Fig 6. (a). Positive Example and (b). Negative Example*

### 1.3.1.3 Report Generation Module

#### Basics of Graph Based Deep Learning

All the approaches mentioned in Sec. 1.2.2.2 were good in their own ways. However, they offered minimal performance improvements in SOTA or were really good in just a single metric. Seeing the saturated research in these approaches, we dropped our initial ideas of potential directions to pursue, as they would not have a significant contribution to the research community.

Graph Neural Networks, introduced recently, have strong representation and explanation power. To our surprise, we found very little work done in integrating this type of neural network in automatic medical diagnosis generation models, despite evidence of profound success in other disciplines in terms of rich information encoding and tractability. The subsequent sections give an overview of the progress of graph neural networks.

## Graph Convolution Overview

The basic building blocks of all graph neural networks are summarized below:

### Graph Neural Network (GNN)

A graph is type of non-euclidean data structure that consists of Edges (E) and Vertices (V), G(V, E). It consists of directed or undirected edges. Graph neural networks are used nowadays in a variety of tasks like classification and scene understanding. However the main challenge with GNN is to find a way to encode the input data into the graphical structure. GNNs are explicitly used in social network related applications, where the data is already present in form of connecting nodes, where neighbouring nodes are more correlated that distant nodes.

### Graph Convolutional Network (GCN)

Similar to convolution operation in traditional CNNs where convolution operation is performed on image grids, in GCN, convolution is performed over the graph nodes. Graph convolutions can be divided broadly into 2 categories namely:

- Spatial Convolution
- Spectral Convolution

For our purpose we will be focusing on Spatial Convolutions in GCNs. It uses the spatial relation among the graph nodes to define convolution, where the node is updated based on the representation of neighbouring nodes. Graph Convolutional Neural Networks use the following propagation rule [23] :

$$H (l+1) = \sigma ( \hat{D}-1/ 2 \hat{A} \hat{D}-1/ 2 H (l) W (l) ) \qquad (2.1)$$

Where H (l) is the activation layer of the l th layer of Graph neural network. H (0) is the input to the initial layer of the network X, i.e., the input feature vector. X ∈ R NXD 0 where N is the number of nodes in graph and D 0 is the number of feature representation of each node and H (l) ∈ R NXD . A is the adjacency matrix for the graph which showns the association between different nodes. It is a square matrix, A ∈ R NXX . The adjacency matrix A lacks self loops, and hence on update the node only depends on the representation of its neighbours and not itself. To propagate self information, a new adjacency matrix Â = A + I N where I N is the identity sqaure matrix of size NXN. D̂ is the diagonal matrix specifying the degree of each node in the graph. Nodes in the graph having large degrees may be accounted for by large values and vice versa. This can cause problems of exploding gradients or vanishing gradients. Also since Stochastic Gradient Descent, one the the general ways in which such networks are trained, are sensitive to range of values, this might lead to problems. Hence, for normalization, we multiply adjacency matrix A with inverse of degree matrix D̂. W (l) is the trainable weight matrix corresponding to the layer l. σ is the activation function to introduce non-linearity, similar to traditional CNNs. Here, σ (·) is RELU(·) activation function.

### Benefit of Graphs in Multi-label Disease Detection

As we have seen in the previous two sections that in graph convolutional networks (GCNs), the node features in the next layer of the network is essentially a weighted aggregation of its neighboring nodes (that are directly connected). These types of networks can be used to model human diseases since ailments (or damage) in specific parts of the human body can be strongly related to abnormalities in same or nearby organs. Hence the diseases inherently assume a graphical structure, with abnormalities spatially related within the human body. Below we see two examples to support these facts.

- Accidents can lead to bone breakage and injuries. Take an example of broken ribs. The rib bone, in unfortunate circumstances, can penetrate one of the nearby organs like the liver or stomach and cause excessive swelling.
- Another example can be taken of a recent disease that plagues humanity: COVID-19. The COVID-19 virus mainly resides in the lungs of humans, and in the process, damages the lung cells and eventually lung tissues. This may cause an excess accumulation of fluid in the lungs which leads to pneumonia and other lung diseases.

From these examples, we have shown that there is a high degree of correlation between diseases that affect same/nearby organs or body parts.

## 1.3.2 Design Diagrams

### 1.3.2.1 Architecture Diagrams

Given below is our proposed architecture of the whole project, explaining in detail the various modules, what has been achieved under them, as well as the sub modules implemented.

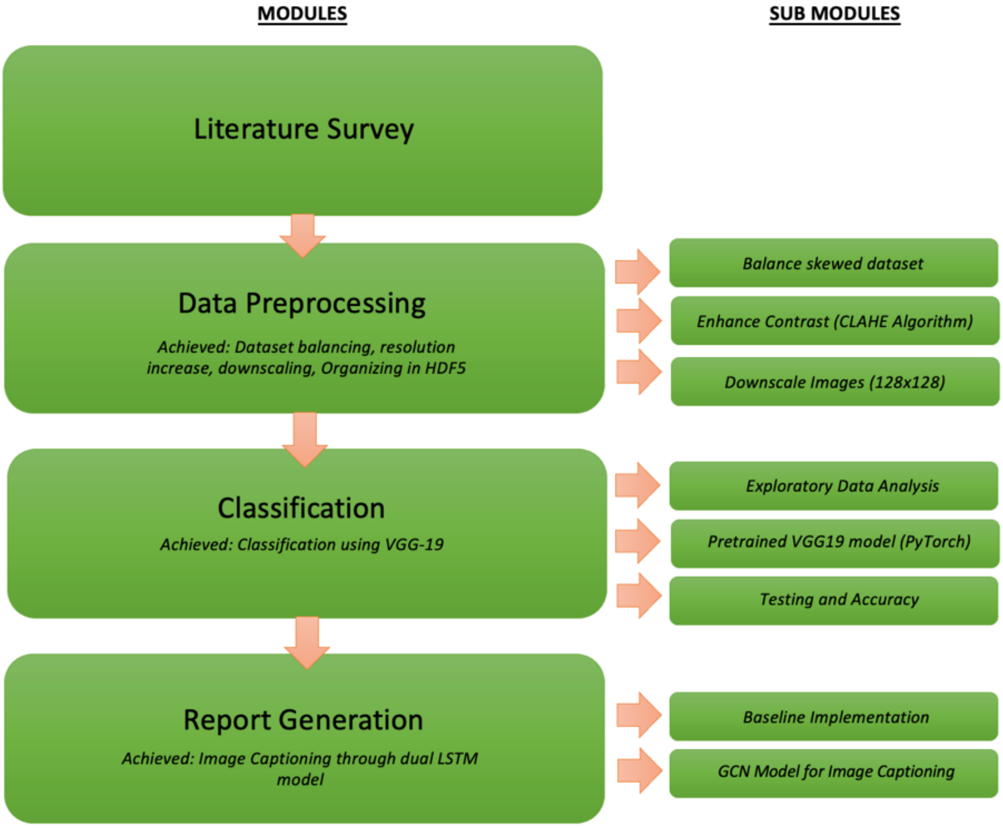

MODULES
SUB MODULES
Literature Survey
Data Preprocessing
Achieved: Dataset balancing, resolution increase, downscaling, Organizing in HDF5
Balance skewed dataset
Enhance Contrast (CLAHE Algorithm)
Downscale Images (128x128)
Classification
Achieved: Classification using VGG-19
Exploratory Data Analysis
Pretrained VGG19 model (PyTorch)
Testing and Accuracy
Report Generation
Achieved: Image Captioning through dual LSTM model
Baseline Implementation
GCN Model for Image Captioning

*Fig 7. Model Architecture*

We take a deeper look into the diagrams that support the different modules of this project below. Fig 12 shows the architecture of VGG-19 that was used in the classification module, complete with all the layers that were used and their specifications.

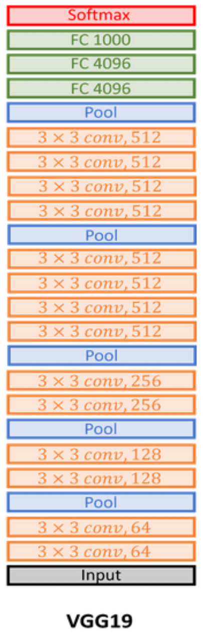


*Fig 8. Classification Module Architecture*

Fig 13 shows the grouping of different ailments in graphical structure. The green (solid) boxes represent the different diseases that are the different graph nodes. The dotted boxes denote the organs where the diseases under then occur, they are not part of target output. Diseases linked to same organ are connected together. The circle (hollow) represents the global node. This diagram is instrumental in the prior knowledge construction discussed in Section 1.4.1.2. As for the detailed architecture of the report generation module, Fig 14 gives more insight to the steps followed all throughout the discussion of our approach in Section 1.4.1.2

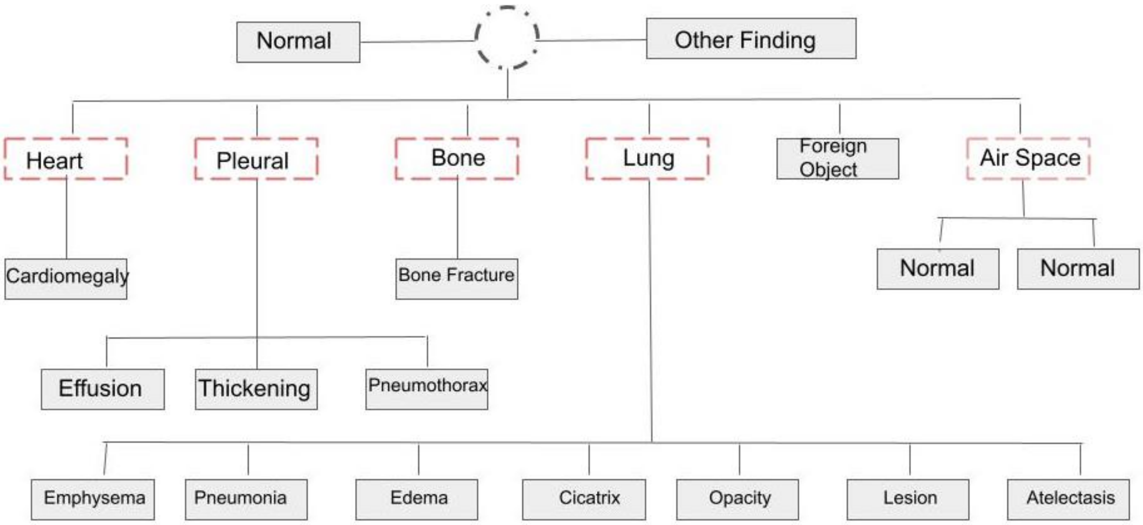


*Fig 9. Prior Knowledge Graph*

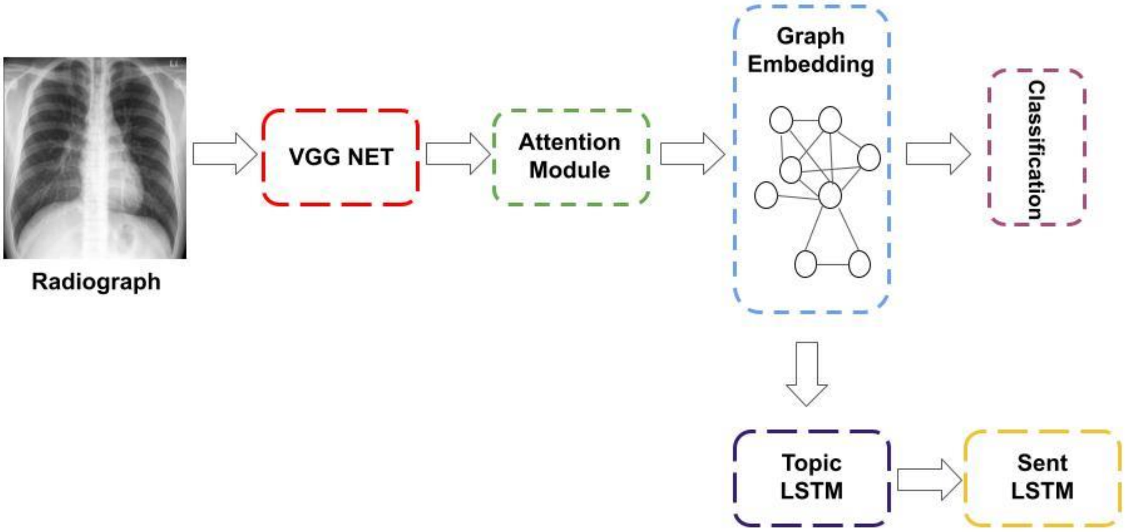


*Fig 10. Report Generation Module Architecture*

# CHAPTER 1.4: RESULTS AND DISCUSSIONS

## 1.4.1 Implementation Details and Issues

### 1.4.1.1 Baselines

In this section, we talk about the deep learning models that we have implemented. We will be using these models as baselines to evaluate the performance boost given by our novel idea.

**Show and Tell: A Neural Image Caption Generator [24]**

This research paper introduced a very simple deep learning model to tackle the task of image captioning. It has served as baselines for most of the following research done in image captioning, and even medical report generation. It consists of a visual deep CNN to encode the image into characteristic feature vectors, which are then used by the language generating RNN to generate captions. The simplicity of this model means that there are many scopes for improvement, which have been explored by succeeding works. We implemented the exact same architecture to use as a baseline.

### Show, Attend and Tell: Neural Image Caption Generation with Visual Attention [25]

This research paper built upon the technique proposed in [24] They took cues from how humans describe a picture. Humans generally look at certain areas of the picture, rather than the whole picture, while describing the picture. This was termed as ”attention” by the authors, which they introduced into the deep learning model proposed earlier in [24]
The attention model works by taking a weighted sum of the image encoding vectors generated by the visual deep CNN. These weights are generated by taking a dot product of the current hidden state vector of the language generating RNN with each of the image encoding vectors. The dot product denotes similarity, with image encoding vectors more similar to the current hidden state vector having higher weights. We implemented the exact same architecture to use as a baseline.

### On the Automatic Generation of Medical Imaging Reports [26]

This paper introduced a strong baseline for medical image captioning. It still used a visual deep CNN to generate the image feature encoding vectors, but instead of a simple language generating RNN, a hierarchical LSTM, introduced by Krause *et. al.* [27] , was used. A hierarchical LSTM had two separate LSTMs, a sentence LSTM to generate the overall idea or topic of the sentence, and a word LSTM to generate the actual words for the sentence, based on the topic generated by the sentence LSTM. This structure helped in remembering information for longer times. Since a sentence contains around 10-12 words, the time steps for which information is needed to be remembered by the word LSTM is much shorter.

Apart from the above, the authors integrated two novel ideas into this pipeline:

- An auxiliary multi-label classifier loss to generate the tags (these tags denote diseases)
- A co-attention mechanism. This mechanism worked by taking the current hidden state vector of the sentence LSTM to calculate weights with both the visual feature encodings as well as the semantic feature encodings according to [25] The weighted visual feature vector and semantic feature vector were then concatenated and passed through a linear layer to generate a co-attention vector, which is then used as input to the sentence LSTM.

For our implementation of the deep learning model proposed by [26], we did not have access to the tags. Hence, we have not used the auxiliary multi-label classification loss, nor does the co attention model use the semantic feature encoding vectors of the tags, due to their unavailability.

We use the weighted sum of the following losses to train the model:

- Cross-entropy loss of stop vectors (denotes to stop producing more topic vectors)
- Cross-entropy loss of words predicted by the word LSTM

In the subsequent sections, we refer to the Sentence LSTM as Topic LSTM (generating the topic or essence of the sentence) and Word LSTM as Sentence LSTM (generating the words of the sentence according to the topic).

### 1.4.1.2 Our Approach

In subsequent sections, we describe the various components of our model, their working, and prerequisite constructions required for defining the model architecture.

**Prior Knowledge Graph Construction**

To help the model learn about various abnormalities and subsequently generate reports, we find it important for the model to have a strong prior. As mentioned before, graphs can be used to model relationships between various entities, and hence we employ a graph based approach to enforce prior knowledge using knowledge graphs and graph convolutional neural networks in the manner described below.

We first mine the dataset for labels to capture the broad class of abnormalities that have been recorded in the dataset. Specifically, we filter out common words from sentences in the reports and record unique words depicting abnormalities, and label the radiographs accordingly. In total, we come up with 17 labels which we further use to construct a prior knowledge graph.

We now construct a knowledge graph that covers common findings and abnormalities (collectively termed as keywords) in the radiographs. The design of this graph is inspired and based on the clustering done in previous works such as CheXpert [12] Specifically, we represent each keyword by a node in the graph. We group each keyword (and therefore the corresponding node) on the basis of which organ it is related to. We leave out ‘Normal‘, ‘Other Finding‘ and ‘Foreign Object‘ labels from this grouping. Keywords or nodes grouped together are organized under a dotted virtual node representing the corresponding organ. We also include a root node to connect to all other nodes and for it to represent the overall (or global) information.

### Model Architecture

We first give a brief overview of the model architecture and then discuss the particulars of each part. Our model is built on top of the hierarchical dual-LSTM model described in our strongest baseline ”On the Automatic Generation of Medical Imaging Reports” [26]. A VGG 19 model pre-trained on the Chest X ray 14 dataset [12] extracts a feature vector for each radiograph which is fed into a spatial attention module. The spatial attention module, based on the VGG Net features, assigns a feature vector to each node of the graph in the graph embedding module. The output of the graph embedding module is used to both drive the classification branch, which consists of a multi-label classifier, and the report generation branch, which consists of the hierarchical LSTM  module. We train the classification branch first, and freeze its weights afterwards to train the report generator branch.

### Spatial Attention Module

For each node (excluding the global and virtual nodes) in the knowledge graph, a 1x1 convolutional filter is applied on the features extracted from the DenseNet model to calculate an attention map. This attention map is applied on the DenseNet features, the resulting values are aggregated together and serve as the initial vectors of the nodes in the graph embedding module. The initial vector of the global node is obtained by applying global average pooling on the DenseNet features. This helps each node to attend over the required regions to detect the corresponding keyword.

### Graph Embedding Module

The graph embedding module is essentially a group of consecutive graph convolution operations applied on a graph structure. The graph structure in this module has a node corresponding to each non-virtual node in the knowledge graph. All nodes under a particular virtual parent node are connected through bi-directional edges, and these nodes are also connected to the global node. All convolutional operations in this module are applied to this graph structure. A typical graph convolution (GC) layer works in the following way: each node of a graph structure receives features from its adjacent nodes aggregated together in the form of a message. The message received and the features of the node are combined together and processed to give new features for the node. In this particular implementation, the GC layers are also followed by a BatchNorm layer [28] and ReLU activation.

### Classification Module

The features obtained from the graph embedding module are passed through a global pooling layer to obtain a single value for each node. This values are concatenated together to obtain a graph level vector corresponding to the radiograph. These are then passed through a fully connected layer with sigmoid activation to give the class probabilities. A weighted binary-cross entropy loss is used to train the classification module and the graph embedding module. We also add an auxiliary classifier (a fully connected layer and sigmoid activation) on the initial features of each node to enforce each node to represent the label it was originally assigned.

### Report Prediction Module

Once the graph embedding and classification module are trained, their weights are frozen to train the report prediction module. We follow a hierarchical LSTM approach as mentioned before: a topic LSTM generates a topic for each sentence in the report, and a sentence LSTM generates the sentence conditioned on the topic vector it is given.

We initialize the hidden state of the topic LSTM as the feature vector stored in the global node of the graph embedding module. The next hidden states are calculated using a another attention mechanism consisting of a couple of fully connected layers. All the node features (except the global node features) obtained from the graph embedding module and the hidden state of the topic LSTM are concatenated together and fed into the attention mechanism. The attention mechanism predicts the weights for each of the input node features. A weighted sum of these node features (called the context vector) is used as the next hidden state.

The sentence LSTM network generates a sentence corresponding to each topic vector obtained from the topic LSTM. This LSTM takes as input the topic vector and the corresponding context vector as input and generates the sentence word-by-word. Note that the context vector and topic vector both are used at each step of the sentence LSTM to update the hidden state and the gates of the LSTM network. We use a cross-entropy loss to drive the training of the hierarchical LSTM module.

### 1.4.1.3 Implementation Issues

Due to changes in technologies and their dependencies, we ran into multiple issues in classification and report generation implementation, specifically pertaining to handling the HDF5 data format with the updated version of tensorflow and keras. Since HDF5 is supported by a downgraded version of keras and the rest of the code only runs on an updated version of tensorflow, we ran into a lot of unexpected issues.

Apart from this, our major issue was in the classification module with the lack of proper computational resources. On 18000 images, our model took 1 hour per epoch, which would've been a huge problem when we ultimately increased our epochs. This led us to further decrease the size of our dataset down to 5,600 images, which was more of a reduction than we had anticipated.

We believe that had it been for better GPU resources and storage capacity, our model would've been very efficient in completing the classification and report generation task, possibly suitable for a more practical use. For instance, instead of the current availability of using the free 12GB NVIDIA Tesla K80 GPU allocated by Google Colab used in our project, a higher GPU would be better to process the rest of the images comprising the original chest x-ray dataset.

## 1.4.2 Testing Plan

### 1.4.2.1 Evaluation Metrics

There are standard metrics for evaluating text generated by deep learning models, viz., BLEU [29] and ROGUE [30].

**BLEU**

The Bilingual Evaluation Understudy (BLEU) [29] metric is the most popular evaluation metric for natural language generation. It is calculated using the following formula:

$$BLEU = BP. \exp\left(\sum_{n=1}^{N} (w_n \log p_n)\right)$$

It is basically a modified n-gram precision, which is the fraction of n-grams in the candidate text which are present in any of the reference texts, penalized by the brevity in candidate texts (BP).

**ROGUE**

The Recall Oriented Understudy for Gisting Evaluation (ROGUE) [30] metric is based on recall, as is evident from the name. There are various ROGUE metrics introduced by the authors. We use the ROGUE-L metric, which is based on longest common subsequence (LCS). Suppose A and B are candidate and reference summaries of lengths m and n respectively. Then, we have

$$\mathrm{P} = \frac{LCS(A,B)}{m} and\ R = \frac{LCS(A,B)}{n}$$

F is then calculated as the weighted harmonic mean of P and R, as

$$\mathrm{F} = \frac{(1+b^2)RP}{R+b^2P}$$

### 1.4.3 Limitations of the Solution

For enhancing and increasing the dataset size, concise diagnostic reports (because text mining must be done on the medical reports to retrieve labels), good camera quality for clear images to act as input to the model, GPU resources of an industrial scale to train the model and process the images, a moderate sized team of data scientists to perform analyses, tune the model hyperparameters, etc.

On a practical level, this solution would require the model to be deployed over an app for admin, doctor and patient. This would further require patients and doctors to have smartphones and IT support in the hospital for development and assistance. This could be a problem in rural areas, as only 25% (as of 2018) of people in rural areas are known to possess smartphones and 0.4% of rural households have access to the internet.

# CHAPTER 1.5: SUMMARY AND CONCLUSIONS

## 1.5.1 Findings

### 1.5.1.1 Classification

According to the tables given below, our VGG-19 model has obtained the following accuracies and a final model accuracy 73.4%:

| S.No | Labels | Accuracy |
|---|---|---|
| 1 | Cardiomegaly | 45.24 |
| 2 | Emphysema | 60.62 |
| 3 | Effusion | 84.40 |
| 4 | Hernia | 57.08 |
| 5 | Nodule | 87.30 |
| 6 | Pneumothorax | 89.82 |
| 7 | Atelectasis | 77.70 |
| 8 | Pleural_thickening | 52.46 |
| 9 | Mass | 58.58 |
| 10 | Edema | 87.4 |
| 11 | Consolidation | 77.68 |
| 12 | Infiltration | 76.24 |
| 13 | Fibrosis | 27.46 |
| 14 | Pneumonia | 84.52 |

*Table 5 . Train Dataset Accuracy Report*

| S.No | Labels | Accuracy |
|---|---|---|
| 1 | Cardiomegaly | 44.884488 |
| 2 | Emphysema | 59.405941 |
| 3 | Effusion | 81.848185 |
| 4 | Hernia | 55.775578 |
| 5 | Nodule | 84.488449 |
| 6 | Pneumothorax | 87.788779 |
| 7 | Atelectasis | 72.937294 |
| 8 | Pleural_thickening | 50.825083 |
| 9 | Mass | 58.085809 |
| 10 | Edema | 86.138614 |
| 11 | Consolidation | 75.247525 |
| 12 | Infiltration | 72.607261 |
| 13 | Fibrosis | 25.412541 |
| 14 | Pneumonia | 83.828383 |

*Table 6 . Test Dataset Accuracy Report*

| S.No | Labels | Accuracy |
|---|---|---|
| 1 | Cardiomegaly | 42.574257 |
| 2 | Emphysema | 60.726073 |
| 3 | Effusion | 82.38284 |
| 4 | Hernia | 58.415842 |
| 5 | Nodule | 87.458746 |
| 6 | Pneumothorax | 89.768977 |
| 7 | Atelectasis | 76.897690 |
| 8 | Pleural_thickening | 51.815182 |
| 9 | Mass | 59.7375974 |
| 10 | Edema | 86.132614 |
| 11 | Consolidation | 76.567657 |
| 12 | Infiltration | 70.627063 |
| 13 | Fibrosis | 27.7272772 |
| 14 | Pneumonia | 82.838284 |

*Table 7 . Validation Dataset Accuracy Report*

As it can be seen from the three tables above, the model did very well on the training dataset for most diseases but for some, it was unable to perform to a satisfactory level. We attribute this shortcoming to the fact that we had to severely cut down on the dataset, as we could not train the original one on our current computational resources.

### 1.5.1.2 Report Generation

The baselines that we have implemented in Section 1.3.1 serve as strong baselines which any new idea we propose should beat in performance. All the 3 deep learning baselines were evaluated on the IU X-Ray dataset, whose results are given below. We also list the values obtained by our model.

We observed that due to the very small size of the IU X-Ray dataset, the models easily overfit. This is another challenge that evaluating on two datasets of different sizes puts forward, that of finding a model which achieves a satisfiable level of performance even with few instances of data. The results of the third baseline implemented by us were comparable in BLEU scores with the respective papers, but did not match in ROGUE score, with ours being much better. This led us to find huge inconsistencies in the results mentioned in the papers. We have decided to show our novel idea's performance against the scores we are achieving, rather than the ones mentioned in the papers, since there may be a difference in the environmental setup, or the train-test split.

| Model | BLEU-1 | BLEU-2 | BLEU-3 | BLEU-4 | ROGUE |
|---|---|---|---|---|---|
| Simple | 0.213 | 0.039 | 0.010 | 0.003 | - |
| Attention | 0.241 | 0.071 | 0.0202 | 0.00 | - |
| Heir | 0.492 | 0.364 | 0.279 | 0.219 | 0.534 |
| Graph | 0.505 | 0.373 | 0.287 | 0.226 | 0.547 |

*Table 8. Report Generation Result Table*

Our model's performance is superior to the SOTA model on the Chest X-Ray 14 dataset. Due to the coronavirus pandemic, we haven't been able to work further on our model or even fine-tune the hyper-parameters to achieve the optimal performance. But, the given results and our simplistic approach strongly suggests that our model is capable of performing better than this if improved further.

During inference, the attention maps obtained from the spatial attention module help to visualize which areas the model is focusing on to predict each label, hence providing visual explanation for its results. Furthermore, this would be extremely helpful to doctors for gauging the model’s confidence in its predictions.

We were unable to evaluate the performance of the baselines on the MIMIC-CXR dataset due to the extremely large size (nearly 5TB). We hence dropped the dataset from our considerations for training and evaluation.

### 1.5.2 Conclusion

We took up a B.Tech. project that would help us apply the skills we learned in the field of deep learning for a bigger cause. To work in that direction, we are working towards improving reliability and explainability of medical diagnosis provided by deep learning models. We carried out an extensive literature survey to understand and get to speed with the research work that had been done. We implemented and tested some strong baselines on datasets to get an idea of the work that will be needed to be done. Upon completion of this survey, we devised a reliable model for classification and a graph-based explanation module for report generation which could be integrated into any deep learning model pipeline. As hypothesized, the model performed better than our implemented baselines, proving how a simple graph-based module could improve performance as well as interpretability of deep learning models, so that they could be used to supplement doctors in diagnosing.

### 1.5.3 Future Work

Our work opens up an avenue for research of more complex classification and graph-based explanation modules to aid in the task of diagnosing and making explainable medical image reports. There have been stronger graph neural networks introduced in recent years which have shown tremendous results in the fields of biology and chemistry. A simple future direction would be to integrate those networks for this particular task. While research can be an ongoing process, there needs to also be development of a software which can leverage this research and summarize it in a succinct, clear and transparent way for doctors to use. Hence, the future directions of this work entail both research and software development, providing immense potential in both sectors.

# CHAPTER 1.6: APPENDIX

## 1.6.1 CLAHE Algorithm

Solutions to the problems of image recognition in computer vision have been found to be more accurate when the images in question are high in resolution and clear in contrast. The intuition behind this comes from the very essence of contrast which relays the fact that the differences in color and radiance can strongly affect the clarity of objects amongst its environment, as depicted below:

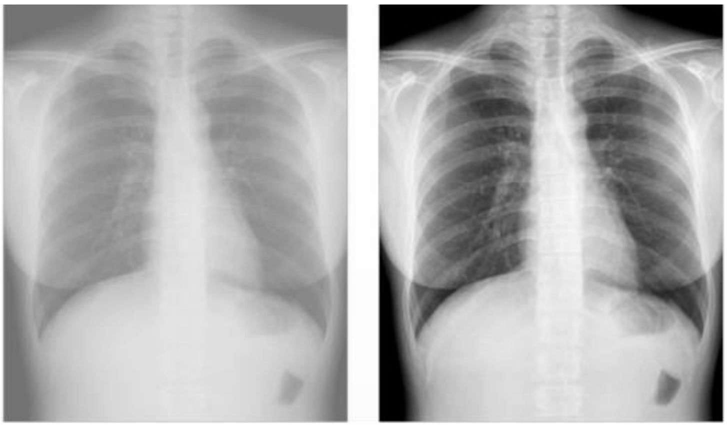


*Fig 11 . Difference between high and low contrast images*

While the significance of contrast has been discussed, it is important to note that a good contrast is not achieved by the highest value of contrast, but rather an optimal one. Too much contrast ends up in lost information and increase in noise, while a low contrast results in indistinguishable objects in the image.

To improve contrast, a technique called Histogram Equalization is used. The main idea behind this technique is that a more 'spread out' or equalized histogram depicts that all gray levels have been used in proper proportion in the image. This is similar to the intuition behind an equalized histogram in statistics, wherein intensities are distributed, as shown in the image below. However, Histogram Equalization does not work well when pixels are not evenly distributed across an image.

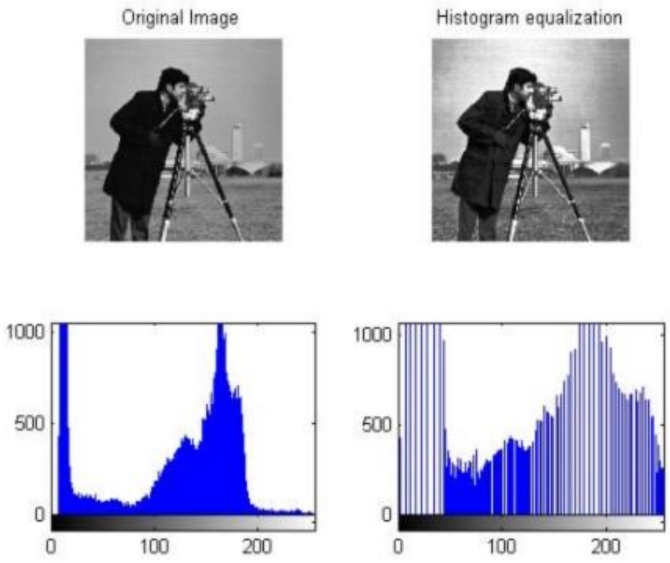


*Fig 12 . Histogram Equalization*

As a solution to this problem, it was thought that multiple histograms can be developed corresponding to different regions of the image, where pixels are not uniformly distributed. This technique was now called Adaptive Histogram Equalization (AHE). But it was found that AHE resulted in extremely high contrasts in the image, which in turn included a lot of noise.

A new technique called Contrast Limited AHE, or CLAHE, was developed. In this technique, clip limits are added to the histogram so that the added noise which exceeds these limits can easily be identified and cut off. This significantly reduced the extremely high amplification and controlled noise in the image.

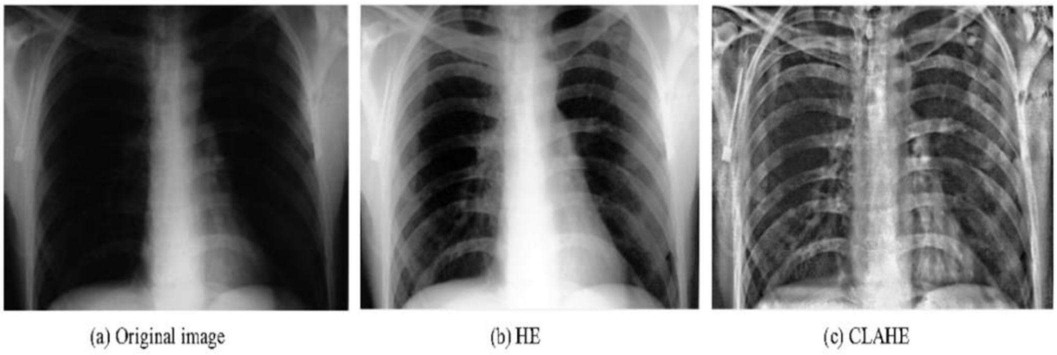


*Fig 13. Difference in images after using different algorithms*

## 1.6.2 Accessing Data from HDF5

The HDF5 format, despite being an underused data storage format due to its complexities, has many advantages when it comes to handling exceptionally large datasets.

For this project and our current computational resources, 18000 images proved to be very bulky, therefore using an HDF5 was the go-to option. The aforementioned complexities came into the picture when train/test split had to be done. The data in the HDF5 is organized into files and data can only be accessed through those files. Oftentimes the data is converted into another data type and so, necessary conversions have to be made to feed it to any model.

To handle any sort of data stored in the HDF5, we had to make use of a Python library that acted as an interface to the HDF5 data format, called h5py. It acts as a wrapper around the HDF5. The data had to be mounted to Google Drive and the corresponding path was referenced. The get() function was used to retrieve the contents of each folder.

# CHAPTER 1.7: LITERATURE CITED

## REFERENCES

[1] Rahib H. Abiyev, Mohammad Khaleel Sallam Ma'aitah, "Deep Convolutional Neural Networks for Chest Diseases Detection", *Journal of Healthcare Engineering*, vol. 2018, Article ID 4168538, 2018. https://doi.org/10.1155/2018/4168538

[2] Shadeed, G.A., Tawfeeq, M.A. and Mahmoud, S.M., 2020. Deep learning model for thorax diseases detection. *Telkomnika*, *18*(1), pp.441-449.

[3] Zhan, T., Deniz, S., Gonzalez, P., Whaley, I., Garcia, D., Vinh, S., Eddy, J., Zhan, F., Choi, V. and Zhan, J., 2020, January. Using Convolutional Neural Networks to Analyze X-Ray Radiographs for Multi-Label Classifications of Thoracic Diseases. In *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 1041-1048). IEEE.

[4] Khobragade, S., Tiwari, A., Patil, C.Y. and Narke, V., 2016, July. Automatic detection of major lung diseases using Chest Radiographs and classification by feed-forward artificial neural network. In *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)* (pp. 1-5). IEEE.

[5] Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M. and Summers, R.M., 2017. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2097-2106).

[6].Rakshit, S., Saha, I., Wlasnowolski, M., Maulik, U. and Plewczynski, D., 2019, June. Deep Learning for Detection and Localization of Thoracic Diseases Using Chest X-Ray Imagery. In *International Conference on Artificial Intelligence and Soft Computing* (pp. 271-282). Springer, Cham.

[7] V. Kougia, J. Pavlopoulos, and I. Androutsopoulos, "A survey on biomedical image captioning," CoRR, vol. abs/1905.13302, 2019.

[8] Z. Zhang, Y. Xie, F. Xing, M. McGough, and L. Yang, “Mdnet: A semantically and visually interpretable medical image diagnosis network,” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3549–3557, 2017.

[9] X. Wang, Y. Peng, L. Lu, Z. Lu, and R. M. Summers, “Tienet: Text-image embedding network for common thorax disease classification and reporting in chest x-rays,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 9049–9058, 2018.

[10] B. Jing, P. Xie, and E. Xing, “On the automatic generation of medical imaging reports,” in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), (Melbourne, Australia), pp. 2577–2586, Association for Computational Linguistics, July 2018.

[11] Z. Han, B. Wei, S. Leung, J. Chung, and S. Li, “Towards automatic report generation in spine radiology using weakly supervised framework,” in MICCAI, 2018.

[12] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya, and et al., “Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison,” Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, p. 590–597, Jul 2019.

[13] Y. Li, X. Liang, Z. Hu, and E. P. Xing, “Hybrid retrieval-generation reinforced agent for medical image report generation,” in Advances in neural information processing systems, pp. 1530–1540, 2018.

[14] G. Liu, T. H. Hsu, M. B. A. McDermott, W. Boag, W. Weng, P. Szolovits, and M. Ghassemi, “Clinically accurate chest x-ray report generation,” CoRR, vol. abs/1904.02633, 2019.

[15] B. Jing, Z. Wang, and E. Xing, “Show, describe and conclude: On exploiting the structure information of chest x-ray reports,” in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, (Florence, Italy), pp. 6570–6580, Association for Computational Linguistics, July 2019.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in Advances in neural information processing systems, pp. 5998–6008, 2017.

[17] Y. Xiong, B. Du, and P. Yan, “Reinforced transformer for medical image captioning,” in Machine Learning in Medical Imaging (H.-I. Suk, M. Liu, P. Yan, and C. Lian, eds.), (Cham), pp. 673–680, Springer International Publishing, 2019.

[18] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4700–4708, 2017.

[19] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, “Cider: Consensus-based image description evaluation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4566–4575, 2015.

[20] T.-M. H. Hsu, W.-H. Weng, W. Boag, M. McDermott, and P. Szolovits, “Unsupervised multimodal representation learning across medical images and reports,” 2018

[21] A. E. W. Johnson, T. J. Pollard, S. J. Berkowitz, N. R. Greenbaum, M. P. Lungren, C. Deng, R. G. Mark, and S. Horng, “MIMIC-CXR: A large publicly available database of labeled chest radiographs,” CoRR, vol. abs/1901.07042, 2019.

[22] J. Yuan, H. Liao, R. Luo, and J. Luo, “Automatic radiology report generation based on multiview image fusion and medical concept enrichment,” Medical Image Computing and Computer Assisted Intervention – MICCAI 2019, p. 721–729, 2019.

[23] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” CoRR, vol. abs/1609.02907, 2016.

[24] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, “Show and tell: A neural image caption generator,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3156–3164, 2015.

[25] K. Xu, J. L. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio, “Show, attend and tell: Neural image caption generation with visual attention,” in Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37, ICML’15, p. 2048–2057, JMLR.org, 2015.

[26] B. Jing, P. Xie, and E. Xing, “On the automatic generation of medical imaging reports,” in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), (Melbourne, Australia), pp. 2577–2586, Association for Computational Linguistics, July 2018.

[27] J. Krause, J. Johnson, R. Krishna, and L. Fei-Fei, “A hierarchical approach for generating descriptive image paragraphs,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3337–3345, 2017.

[28] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” CoRR, vol. abs/1502.03167, 2015.

[29] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “Bleu: a method for automatic evaluation of machine translation,” in Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, (Philadelphia, Pennsylvania, USA), pp. 311–318, Association for Computational Linguistics, July 2002.

[30] C.-Y. Lin, “ROUGE: A package for automatic evaluation of summaries,” in Text Summarization Branches Out, (Barcelona, Spain), pp. 74–81, Association for Computational Linguistics, July 2004.