Addis Ababa University

School of Graduate Studies

College of Natural and Computational Sciences

Department of Computer Sciences

Cardiorespiratory Disease Detection Using Deep Transfer Learning

Kibrom Haftu Aleme

A Thesis Submitted to the Department of Computer Science in Partial Fulfillment for the Degree of Master of Science in Computer Science

Addis Ababa, Ethiopia

February, 2021

Addis Ababa University

School of Graduate Studies

College of Natural and Computational Sciences

Kibrom Haftu Aleme

Advisor: Yaregal Assabie (Ph.D.)

This is to certify that the thesis prepared by *Kibrom Haftu Aleme*, titled: *Cardiorespiratory Disease Detection using Deep Transfer Learning* and submitted in partial fulfilment of the requirements for the Degree of Master of Science in Computer Science complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the Examining Committee:

	Name	Signature	Date
Advisor:	_____	_____	_____
Examiner:	_____	_____	_____
Examiner:	_____	_____	_____

Abstract

Cardiorespiratory diseases are recognized as serious, worldwide public health concerns that have remained among the leading cause of death globally. These diseases range from common colds to life-threatening bacterial pneumonia that affects anyone, anywhere. Timely, accurate, and effective detection is critical, especially for infectious diseases to prevent the condition from becoming too severe. Deep learning a subfield of machine learning played a central role to detect cardiorespiratory diseases from X-ray images using transfer learning techniques. These approaches allow radiologists (or other health professionals) to extract relevant features that are undetectable to the naked eye or difficult to spot. However, there are still longstanding issues related to the application of deep learning for transfer learning. Difficulties arise when transferring knowledge from one domain into another task that could hurt model performance.

We, therefore, proposed a non-parametric Bayesian statistical model to investigate the effectiveness of transfer learning on X-ray images. The proposed model comprises two main components: The deep transfer learning and the detection component. The first component leverages unlabeled X-ray images using unsupervised variational autoencoder to EfficientNet architecture. Its purpose is to extract robust features for detection. The detection component further fine-tunes these features using an end-to-end supervised algorithm rather than current approaches that use a single feature space represented by the last fully connected layer of the convolutional neural network across all conditions. At this stage, we conducted a hyperparameter search for different parameters. Moreover, we developed an offline data augmentation algorithm to address the class imbalance problem that emerged from the dataset. In doing so, we expand the training data from 75,682 to 108,708 X-ray images

The proposed system was implemented on Google Cloud Platform using an open-source deep learning library called PyTorch. The National Institutes of Health dataset was used to train, validate, and test the proposed model. The dataset is comprising 112, 120 images from nearly 31,000 patients. We also evaluated the performance of our approaches using Tikur Anbessa Specialized Hospital X-ray images. We have achieved an 88.01% AUC score that outperforms the existing state-of-the-art approaches.

Keywords: Deep Learning, Cardiorespiratory Disease Detection, Deep Transfer Learning, CNN, Variational Autoencoder, X-ray Images, Unsupervised Algorithm, Latent Factor, AI

Dedication

The least I can do is dedicate this work to my parents.

Acknowledgments

I would like to acknowledge the following individuals and institutions for their contributions while carrying out this work.

In particular, I would like to thank my supervisor, Dr. Yaregal Assabie, who expertly monitors my progress and provide the right guidance towards cutting-edge researches. I am especially grateful to my supervisor's enthusiasm for my topic, which inspires me to redouble my effort.

I must express my sincere appreciation and gratitude to Dr. Fekade Getahun, who first introduced me to the wonderful world of Bayesian networks, which is the backbone of this thesis. Thank you for offering me a friendly shoulder and unceasing support for quenching my thirst for knowledge. I have been fortunate enough to know you and I will pay it forward.

My warmest gratitude also goes out to Dr. Mesfin Kifle, who significantly shaped my proposal at earlier stages. He eloquently stated sound reasoning deep-rooted and I find myself asking why I do what I do. It then revealed a scientific shakeup for answering difficult problems and marked a beginning for the whole another way of looking at current technologies.

I would like to extend my heartfelt thanks to Musie Meressa, Mahder Ayele, and Asmelash Teka for their unconditional grants and facilitate access to the Google cloud platform. It has been a continuous source of encouragement and optimism throughout. Thank you for being my support system to complete this thesis work at a reasonable time.

My acknowledgments would not be complete without thanking Tikur Anbesa Specialized Hospital for providing the data which plays a far-reaching impact on the science.

Needless to say, many people have contributed either directly or indirectly to this study. For example, I have benefited greatly from the comments and suggestions of Dr. Michael Mayhew. I would also like to acknowledge here my gratitude for the difficulty itself because it made me swallow my pride and be grounded to the earth. This thesis work was actually born out of curiosity and desperation.

Last but not least, I would like to thank families, colleagues, and teachers for the unconditional support and patience. Mom and Blen, I see nothing that is not worthy of thanksgiving. Every tear you cried, and every prayer you prayed has brought me here, to this moment. Thank you.

Table of Contents

List of Figures

List of Tables

List of Algorithms

Acronyms and Abbreviation

ANN	Artificial Neural Network
AI	Artificial Intelligence
AUC-ROC	Area Under the Receiver Operating Characteristic Curve
CAD	Computer-aided Detection
CT	Computed Tomography
CV	Computer Vision
CNN	Convolutional Neural Network
COPD	Chronic Obstructive Pulmonary Disease
DL	Deep Learning
DTL	Deep Transfer Learning
DCG	Dynamic Computation Graph
ESR	European Society of Radiology
FC	Fully Connected
GCP	Google Cloud Platform
IBP	Indian Buffet Process
ILD	Interstitial Lung Disease
ML	Machine Learning
MCMC	Markov chain Monte Carlo
MLP	Multi-layered Perceptron
MRI	Magnetic Resonance Imaging
NLP	Natural Language Processing
NCD	Non-communicable Disease
PACS	Picture Archiving Communications Systems
PNX	Pneumothorax
GPU	Graphical Processing Unit
KNN	K Nearest Neighbor
LBP	Local Binary Patterns
LSTM	Long Short-Term Memory
ML	Machine Learning
MRI	Magnetic Resonance Imaging
MSE	Mean Squared Error

NIH	National Institute of Health
NB	Naïve Bayes
PA	Posterior-Anterior
PP	Probabilistic Programming
PIL	Python Image Library
ReLU	Rectified Linear Unit
RGB	Red Green Blue
RNN	Recurrent Neural Network
SOTA	State-of-the-Art
SGD	Stochastic Gradient Descent
SVM	Support Vector Machine
TPU	Tensor Processing Unit
VAE	Variational Autoencoder
VM	Virtual Machine

Chapter 1: Introduction

1.1 Background

Cardiorespiratory diseases (cardiovascular and respiratory diseases) are the leading cause of death worldwide. More than 1 million people are hospitalized annually, with economic costs in the US alone amounting to tens of billions of dollars per year. Some of the common worldwide public health concerns are shown in Figure 1.1 [1, 2].

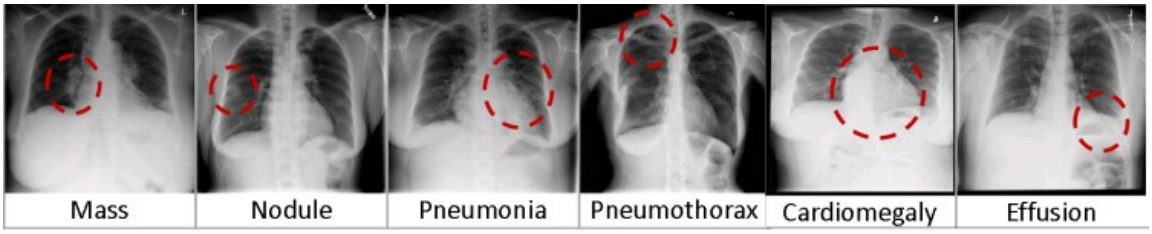

Figure 1.1: Common Cardiorespiratory Diseases

Seidu et al.[3] argued that 50% of cardiorespiratory deaths occur in Sub-Saharan African Countries. Similarly, Gouda et al.[4] found that tobacco smoke, air pollutions, and urbanization impacts have made these diseases a major burden. A study by Dalju [5] and Asgedom et al. [6] revealed a high prevalence of cardiorespiratory diseases in Ethiopia, accounting for the majority of non-communicable diseases(NCDs). NCDs are medical conditions that are non-infectious with an overwhelming burden in low and middle-income countries[7]. Lack of hospitals, experts, and insufficient medical devices made worse the diagnosis and treatment of these diseases in such countries[8].

As argued by Ma et al.[9], chest X-ray is far more cost-effective and commonly performed radiologic examinations in various clinical-settings for the assessment of cardiorespiratory problems. It is used to examine abnormalities and visualize specific structures inside the airways, lungs, and hearts[10]. The interpretation of X-ray is a challenging task that needs technical expertise about human anatomy and the principles of pathophysiology otherwise the costs of medical errors adversely affect the diagnosis and treatment of diseases[11]. For this reason, computer-aided detection systems were then introduced and developed in the late 1950s[12]. A continuous and huge amount of innovative scientific researches was carried out in the field[13] especially with advanced computer vision (CV) algorithms[9]. It is therefore not surprising that artificial intelligence (AI) algorithms play a profound significance to facilitate faster detection of cardiorespiratory diseases from medical X-ray imaging in the service of healthcare[14].

Deep learning (DL) algorithms outperform the traditional machine learning (ML) methods such as support vector machine (SVM) and K-Nearest Neighbors (KNN) with sufficient data[15]. The major downside of such ML approaches is that they principally rely on handcrafted features (manual engineering) that involve time-consuming and tedious tasks (labor-intensive tasks)[16]. DL a subfield of ML appears to overcome this challenging issue. Difficulties arise, however, because of the data-hunger characteristics of the DL algorithms in which the availability of such huge datasets are quite limited and therefore transfer learning (TL) paradigm has become the de-facto standard approach [16, 17, 18].

One of the main difficulties with applying DL models for healthcare is that labeling and image acquisition requires medical experts to accurately annotate biological images[19]. TL algorithms are, therefore, impressive when a huge training dataset is scarce and has the potential to significantly reduce training overhead[20, 21]. TL is a ML approach to adapt knowledge from pre-trained networks from one domain into another related problem and conquer major issues of DL algorithms[20, 22]. There is a growing area of research using TL approaches for the diagnosis of cardiorespiratory diseases as exemplified in [23, 24, 25] and are increasingly becoming popular in medical domains.

Despite the breakthroughs in TL and DL algorithms for cardiorespiratory diseases, there remains a paucity of evidence where these approaches suffer from overfitting issues [26]. Since the vast majority of the existing studies are conducted on historical datasets to train algorithms, they showed performance drops when encountered in real-case scenarios [27]. Zhuang et al.[28] illustrates scenarios where knowledge transfer to the target task can show worsening performance (in a case known as a negative transfer) rather than improving. This means the representation is nowhere near reality. The real challenge of DL algorithms today however revolves around uncertainty estimations and remains opaque or black boxes. The lack of interpretability (failing to associate causes to effects from DL models) leads for example to medical errors and incorrect clinical recommendations[29, 30]. Our work brings together several important research strands to address the principal weaknesses of DL and TL approaches. We are inspired by recent advances in CV algorithms within the medical field with a series of breakthroughs, opening up the perspective of more accurate detection. Complementary efforts to accurately diagnose diseases will greatly be assessed from unlabeled data. The objective of this work is, therefore, to examine the effectiveness of applying DL-based frameworks to TL approaches (via unsupervised learning) for multiple cardiorespiratory disease detection on medical images (X-ray images).

1.2 Motivation

The discipline of radiology in medicine facilitates the prevention and diagnosis of diseases through the use of medical imaging techniques such as computerized tomography (CT), magnetic resonance imaging (MRI), and X-ray imaging. Chest X-rays are the most commonly performed radiological examinations preferred for their easy accessibility and relatively low cost to diagnose abnormal patterns at hospitals[31]. In the interpretation of X-rays images, one must consider different technical factors such as quality of the film, positioning, penetration, volumes, artifacts, and inspiratory efforts of the patient[32]. It is unusual for a patient to pass across a hospital without needing a radiologist along their journey[33]. Radiologists are physicians or medical doctors (MDs) who specialized in the analysis and interpretation of structures and organs[34]. However, a radiological error is not a recent phenomenon and there are no occasions radiologists get everything right[35].

There are several possible explanations for this. Fatigue is one inevitable condition in the workplace with a considerable contributing factor to medical diagnostic errors, resulting in financial constraints facing healthcare and a higher risk of unfavorable outcomes [36]. Diagnostic errors are situations that can cause harm to a patient. Errors and inconsistencies, are inextricably linked to radiologists' work and may result in a variety of unpredictable clinical incidents[37]. When it comes to the interpretation and detection of X-ray images, qualified radiologists have specialized knowledge and perceptual abilities compared to practicing radiologists[38]. What is now needed is regular training and knowledge transfer from experienced radiologists[39]. For example, the knowledge gained from learning to recognize pneumonia can be applied when trying to recognize pneumothorax through previous experiences.

The same applies to a popular area of research in AI known as transfer learning. Transfer learning techniques come adopted to overcome the lack of labeled data available for models in specific tasks and knowledge adaptation[40]. We are also driven and inspired by the principal AI strengths for instance accuracy, automation, and objectivity[41]. It is a whole another way to look in the world of health that appears as an exciting journey and wonderfully stimulating to observe the human body from the inside, to detect diseases from a computer screen.

1.3 Statement of the Problem

Annually, cardiorespiratory diseases are the underlying leading cause of death worldwide, accounting for millions of deaths[42]. The process of detecting cardiorespiratory diseases from medical X-ray images that are usually stored in the centralized Picture Archiving and Viewing Systems (PACS) can be time-consuming and error-prone. These medical errors can be expensive, adversely affect patient care. Accuracy and timely diagnosis are crucial especially for infectious diseases that need to be early and accurate to prevent the condition from becoming too severe. As radiologists might miss subtle signs in the X-ray due to exhaustion, automated DL and TL-based approaches have been proposed.

Different methods have been proposed to classify X-ray images. The authors in [43, 44] employed TL techniques based on the DL. ChestNet[45] vigorously challenged and further improved their works. While the performance of existing approaches has been improved significantly, certain conditions are still very difficult to predict. Pneumonia, infiltration, and nodule achieved less than 70% detection rate, precisely an overall AUC (short for the area under the curve) score of 0.6902. Despite the heroic DL-based TL research efforts, questions remain unanswered. The criticism here is the performance demonstrated across all the first major studies [43, 45, 46, 47, 48] and the most recent works of 2019[49, 50, 51] were found in between the AUC measures of 0.7451 and 0.808. These results indicate around 20% misclassification errors in the best-case. Therefore, the result is somewhat unconvincing. We base our work on four longstanding bottlenecks of cardiorespiratory disease detection. These are class imbalance problems, negative TL, optimization problems, and noisy labeled data (label error) or generally limited applicability.

Class imbalance problems: - Current detectors[43, 45, 47, 50, 51] have carried out their work based on an imbalanced NIH dataset. Each disease has extremely unbalanced sets ranging from 110 images of hernia to 9,547 infiltrations. However, Luque et al.[52] raised major issues from the nature of unbalanced classes which ultimately affect performance.

Negative Transfer Learning (TL): - ChestNet[45] fine-tuning involved retraining only the last fully connected layer of a pre-trained DenseNet. However, knowledge transfer from a less related source may inversely cause drastic target performance degradation[53]. Moreover, learning the dimensionality of the layer (the number of factors) for detection in these fully-connected layers (or indeed other layers of the pre-trained network) remains an open question in the field to the best of our knowledge.

Optimization problems: - The authors of ChestNet[45] also note that no additional regularization was imposed on the fully connected layer during training and indicate that the fine-tuned network's generalizability could still be improved. Current detectors [43, 45, 47, 48, 49, 50, 51] have not closely examined the effect of numerous hyperparameters values.

Limited Applicability: - The existing works particularly on algorithms based on the NIH dataset [43] have conducted their work on 10% noisy data because the clinical radiology reports are extracted with a Natural Language Processing (NLP) tool with 10% label errors. Wang and Xia[45] note that a limitation of their work is its application in X-ray images from only one clinical setting and location. The performance of the model that was trained on the NIH dataset should, however, validated against an external dataset (preferably in real-world clinical settings).

Taken together, different diseases might need different numbers of features for detection. Some features might be important for the detection of one disease and irrelevant for the detection of another. A thorough analysis of the sources of misclassification errors in the different conditions or diseases has not been performed that determines which features in pre-trained networks (TL) are useful for which cardiorespiratory disease, for example, say by the analysis of backpropagation gradients. TL tends to reuse hidden-layers for a new task with the same or similar dimensionality as the hidden-layers of a pre-trained network and involves a limited search of appropriate architectures for the new task. The fine-tuned network may include spurious features that could lead to poor generalization and drops in performance. Indian buffet processes (IBPs) have been used to learn the full architecture of untrained neural networks, however, have not yet been applied to learn parts of network architectures for transfer learning[54] and Ni et al.[55] asserts that IBPs applications in healthcare are limited. Siddiqi [56] also recommends thoroughly investigating applications of TL and fine-tuning techniques.

Therefore, to our knowledge, this is the first study to investigate the effect of the structure learning of a fully-connected layer (or indeed other layers of a pre-trained model) through a deep transfer learning approach. We can also be the first group to demonstrate whether DL-based TL approaches generalize to patients from a different clinical setting. The goal is to leverage an unsupervised learning algorithm to extracts robust features and apply a supervised algorithm for detection to address each of the research gaps. Moreover, we'll conduct hyperparameter searches and tackle the class imbalance problem.

1.4 Objectives

General Objective

The general objective of this research work is to design and implement a cardiorespiratory disease detection model using deep transfer learning (DTL) from X-ray images.

Specific Objectives

To achieve the general objective of the study, the following specific objectives are defined:

- Review works related to the realm of the research
- Collect X-ray image dataset dedicated to cardiorespiratory diseases
- Extract and identify visual features characterizing cardiorespiratory diseases
- Design a DTL model for multiple cardiorespiratory disease detection
- Develop a prototype
- Test the performance of the algorithm
- Compare to benchmark current state-of-the-art (SOTA) approaches

1.5 Methods

The methodological approach employed in this study is design science research because the goal of our study is to design a new artifact[57] for the task of cardiorespiratory disease detection from X-ray images. Therefore, we will adapt standard methods as well as develop a new statistical approach for DTL and utilize it for X-ray image classifications. To achieve the general and specific objectives, the following methods are used.

Literature Review: Extensive systematic review related to DL, TL, IBP, cardiorespiratory disease detection, SOTA neural network approaches, Bayesian deep learning, SOTA computer vision architectures, supervised fine-tuning, and unsupervised feature learning, optimization techniques, and visual X-ray image analysis and associated technical factors will be carried out. Knowledge gaps in the field will be identified and what methods have been used to fill the earlier research gaps.

Data Collection: We will collect an X-ray image dataset to detect cardiorespiratory diseases from a large chest X-ray dataset (ChestX-ray14) comprising 112,120 images from nearly 31,000 patients. Our main source is drawn from a hospital known as the National Institute of Health (NIH)[43], a hospital dedicated to providing clinical researches. The NIH X-ray images are originally provided in PNG formats of 1024×1024 resolution. The whole size of the NIH dataset is close to 46GB of data.

Tools and Prototype Development: During prototype development, various open-source tools will be used. However, due to relatively large training sizes that require expensive computational resources, we will use commercial Google Cloud virtual machines (VMs). The choice of the programming language is Python because Python libraries are essential and well-suited for data science. The entire experiment will be carried out in an Anaconda PyTorch environment.

Evaluation and Testing: We will replicate the results of the ChestNet[45], fine-tuning a pre-trained convolutional network (DenseNet) on the ChestX-ray14 dataset. Furthermore, we will then evaluate our IBP-based transfer learning in the ChestX-ray14 image dataset and compare it to benchmark approaches or evaluate the performance of models. We will also retrain our models using the enriched subset of training examples and evaluate their performance in external validation (on new data that is unseen to the training algorithms). Therefore, the proposed model evaluation is based on the goal of the research work, and the performance scores measured using DL evaluation metrics.

1.6 Scope and Limitations

The proposed solution focuses on developing a DTL model from the ChestX-ray14 images for the detection of cardiorespiratory diseases. Its scope is currently limited to 14 diseases. Our priority here is to Pneumonia, Pneumothorax, Hernia, Edema, Fibrosis, Emphysema, Consolidation, Nodule, Mass, Cardiomegaly, Atelectasis, Infiltration, Pleural Thickening, and Effusion. The aim is to spot one or more dangerous anomalies (diseases) and uniquely distinguish them from normal radiographs (disease-free patients).

1.7 Application of Results

Our generalized cardiorespiratory disease detection can greatly help radiologists (or indeed other health professionals) extract undetectable to the naked eye or difficult to spot useful information. This opens up a perspective of precise and earlier diagnoses of fatal diseases. The detection approach will also make medical diagnosis more evident and tremendously minimize the risk of medical errors and reduce some of the financial constraints facing healthcare. Training computers to look at the security of X-rays and then spot dangerous items is an extremely crucial technological promise. Our approach will prove useful in expanding the field through its reliable and clinically applicable algorithm because part of our effort is to purposely assess its clinical relevance from another setting.

1.8 Organization of the Thesis

The rest of this thesis is structured as follows, linking all together to support the principal objective of the study. Chapter Two introduces strong theoretical backgrounds relevant to the research dimension including ML, DL, TL techniques, Bayesian DL, causal factors to visual X-ray image analysis, and the core technology of DL (neural network) approaches. Chapter Three briefly reviews the related work for cardiorespiratory disease detection from X-ray images. The fourth chapter presents the proposed model and its components. The model comprises a number of components designed to addresses each of the research gaps. Chapter Five deals with model implementation, evaluation, and experimental results. The research findings, programming tools and frameworks, data sources, experimental designs, and the effectiveness of hyperparameter searches will also be presented. Finally, we conclude our work and discuss future research directions in Chapter Six.

Chapter 2: Literature Review

This chapter seeks to explain the relevant technology, theories, and proposed model structures adopted in our field of study. It gives the reader an overview of the background of cardiorespiratory disease along with a brief review of the scientific literature on medical image processing and computing while identifying current state-of-the-art (SOTA) CV neural-network approaches for detecting abnormalities in X-ray images. The chapter then goes on to cover Bayesian DL paradigms and more specifically focus on the nonparametric Bayesian factor model to neural network structure learning in deep networks. The goal of this chapter is twofold. First, it highlights current research efforts and sets out what is already known about the particular topic and what methods have been used in researching the topic. The other is to indicate a knowledge gap in the research area.

Specifically, Section 2.1 begins with a concise introduction to cardiorespiratory disease in line with the theoretical dimensions of this research. Section 2.2 covers a brief history of computer-aided detection and then extends over cardiorespiratory disease detection approaches to identify chest abnormalities and interpret X-ray images. Section 2.3 and 2.4, present relevant concepts of ML and neural network foundations respectively. It then progressed to present several factors found to be influencing the performance of DL models and therefore, the effectiveness of advanced techniques in contrast to conventional methods will be reviewed. Readers are then referred to Section 2.5, a CV for DL. Next is Section 2.6 that is entirely devoted to the applications of AI in healthcare fields. The focus of the seventh section is to provide background studies on Bayesian DL that bridges the gap between deep learning and probabilistic models.

2.1 Cardiorespiratory Diseases

Cardiorespiratory (cardiovascular and respiratory diseases) are among the leading cause of death worldwide. Economic transition, lifestyle changes, and increased pollution have increased the incidence of these diseases [58]. The prevalence of cardiorespiratory disease continues to rise over time and has a high incidence in the developing world [59]. In sub-Saharan Africa, the high prevalence of respiratory infections, household, and outdoor pollutions, increased tobacco epidemic, and lifestyle changes that come with urbanization have made cardiorespiratory diseases a major burden. Furthermore, the lack of hospitals, trained professionals, and medical diagnostic equipment are other challenges that make the diagnosis and treatment of these diseases difficult in these countries[8].

Cardiorespiratory diseases adversely affect cardiovascular and respiratory systems under certain conditions. Therefore, there is a definite need for a brief review of such systems. As humans breathe, air enters the lungs by flowing down the trachea, which then splits into two bronchi; it continues through the bronchioles and finishes in the alveoli. The respiratory bronchioles provide a transition to the respiratory part of the system, and the alveoli are tiny sacs of air that contain numerous blood capillaries. Alveoli is where most of the gas exchange takes place. Oxygen(O_2) leaves the air in the alveoli and flows into the bloodstream, while carbon dioxide (CO_2) is expelled out of the lung into the outside air as a waste product of cellular metabolism. The refreshed blood will be rich in oxygen before the oxygen is redistributed into tissues in the human body. Thus, the respiratory system is designed to bring oxygen into the body to fuel energy generation and remove carbon dioxide. This is where gas exchange takes place with the outside atmosphere. However, these organs are continuously exposed to varying levels of hazardous substances ranging from environmental toxins to side-effects of some drugs[60, 61].

In cardiorespiratory disease, breathlessness is a common sign and comes with multiple dimensions[62]. There could be several contributing factors. Regular exposure to air pollution, for example, can cause chronic cough and throat clearing. It increases the risk of many severe and life-threatening diseases such as asthma, respiratory infections, and chronic obstructive pulmonary disease (COPD)[63]. Disturbances of respiratory tract organs, including trachea, bronchioles, bronchioles, alveoli, pleura, and pleural cavity, and breathing muscles, are common in humans. These frequent storms develop respiratory diseases [64]. However, risk factors are not limited to air pollution that provokes disorders of cardiovascular and respiratory systems. According to Bhatta and Glantz [65], smoking is a major cause of cardiorespiratory disease in America. There are still emerging severe infectious cardiorespiratory diseases such as coronavirus which present a major threat and when, where, and how new diseases appear are still a source of considerable uncertainty.

To this end, it is hoped that the reader will have acquired a sufficient understanding of the cardiovascular and respiratory systems and the mechanics of breathing. Besides, it is necessary to consider the variations of diseases as multiple diseases occurred, ranging from the common cold to life-threatening bacterial pneumonia that affects anyone, anywhere. According to Pákó et al.[66], cardiorespiratory disease causes include inflammation of the airways, interstitial lung diseases (restrictive lung disease), diseases affecting the chest wall, and disease affecting the vascular system.

To support this line of reasoning, Rogliani et al. [67] also assess the strong association of airways diseases (asthma, chronic obstructive pulmonary disease (COPD)) and interstitial lung disease with several cardiovascular diseases and leading to increased risk of mortality. It is therefore essential to know whether the known condition has progressed or whether a new process has emerged that creates problems for patients before it reaches the point of no return or more complicated conditions. Then a careful examination is always warranted in the evaluation of patients.

2.2 Cardiorespiratory Disease Detection

The task of cardiorespiratory disease detection is to uncover physical signs when a patient poses challenging problems by combining medical knowledge, judgment, and experience usually with the help of computer-based systems. Such systems are generally known as computer-aided detection (CAD). The main goal of CAD is to extract a region of interest in medical images from a range of imaging techniques, such as radiography and magnetic resonance imaging (MRI) that can assist in the early diagnosis of disease[68]. CAD software programs can automatically facilitate the interpretation and visualization of X-ray images. According to Yanase and Triantaphyllou [12], the origins of CAD can be traced back to the late 1950s by biomedical researchers who started investigating the possibility of using a computer to solve problems in medicine and biology. Afterward, CAD systems began to serve as a foundation for accelerating research in the interface of computer science and medicine. Its extensive applications, appearing from head to toe in an objective search for abnormalities usually through the use of medical imaging techniques.

Medical imaging techniques can provide detailed images of human anatomy. Zhang et al. [69] emphasize the role of medical imaging techniques to create visual representations of the body's interior in order to make an accurate diagnosis. Creating visual representations of the interior body is good because what the clinician can see of the patient's exterior is insufficient to make accurate diagnoses. Moreover, one of the main reasons for developing CAD is to help physicians to avoid medical mistakes because manual interpretation and analysis are tedious and prone to error. It is often required in addition to a medical history and physical examinations. Medical imaging techniques are therefore useful for the production of images of structures in the human body. These images reveal details about structural, functional organs and tissues. The most common medical imaging techniques that have become increasingly important are radiography.

Radiography

Radiography (X-ray imaging) is the most widely used imaging technique due to its easy accessibility and relatively low cost for the assessment of cardiorespiratory problems. With rapid advances in X-ray imaging, research centers have accumulated a large repository of X-ray images that can have a significant role in the delivery of healthcare services to patients[70]. Such computer-based radiograph storage systems are known as Picture Archiving Communications Systems (PACS) so that physicians and other health professionals can then able to view the stored X-ray images anywhere on a computer screen within the hospital. Seibert [71] states 60–70% of medical imaging examinations performed at hospitals represent radiographic images. The purpose of medical x-ray imaging is to provide detailed information on specific aspects of the structure of the body. The parts of the human body that usually imaged are the breast, chest, and teeth. The chest is the body part in which cardiorespiratory diseases such as pneumonia, cardiomegaly, mass, nodule, consolidation, infiltration, pneumothorax, pleural effusion, and many other health complications are examined[70].

Chest x-rays allow the assessment of cardiopulmonary abnormalities and the visualization of various structures in the chest, including the lungs, heart, airways, and related structures [10]. The interpretation of chest x-rays is a complex task that requires deep technical radiological expertise to reduce medical errors. Expertise is typically found in radiologists. Radiologists are medical doctors who specialize in the diagnosis of MRI, computed tomography (CT), ultrasound, radiography, and other diagnostic imaging techniques [72]. In the interpretation of x-rays, one must consider different factors affecting the quality of X-ray images. One is a technical factor that includes exposure to film (patient radiation exposure), positioning, penetration, inspirational effort volumes, and the availability of artifacts. A quality X-ray is characterized by a posterior-anterior (PA) view, a full inspiratory effort where 10 posterior ribs are visible and the clavicle bones seen equidistant from the vertebral spinous processes. Other considerations include the anatomy of chest structures, the inspiratory efforts of the patient, and indeed the difference between male and female chest[73]. Castro-Gutierrez et al.[74] discussed several contributory factors to correct X-ray analysis such as illumination changes, patient actions, intestinal gases[75], and other external factors. Several factors could determine the quality of the X-ray images analysis and clinicians can play a paramount role to have high-quality X-ray images[76]

Factors [73, 77] are summarized as shown in Table 2.1.

Table 2.1: Image quality factors on a radiograph

Factors		**Analysis**
Quality	High	Well penetrated, vertebral bodies visible behind the heart.
	Low	Under penetrated film-whitened and vertebral bodies not visible.
Positioning	PA	heart and mediastinum are closest to the film and therefore not magnified (Commonly used)
	AP	Used for patients who are on a chair or in bed. The clavicle bones cover the upper parts of the lung. The size of the heart and the mediastinum are also magnified
Rotation		Should be minimal and clavicle bones should be equidistant from the thoracic spinous processes
Inspiratory efforts		10 ribs should be visible, whereas in poor inspiratory efforts less than 6 ribs are visible.
Artifacts		Clearly seen on x-ray with well-defined opacities. Artifacts such as patients with tubes, pacemakers, and clothing (buttons)
Male vs Female chest		The amount of breast tissue creates a difference Breast tissue absorbs some of the x-rays may cause under or overexposure bilateral basilar lung infiltrates may cause asymmetrical lung density. Nipple shadow in the lower lung may misinterpret as a nodule.
Anatomical structures	Lungs	Normal air-filled lungs appear black, each zone must be compared in the right and the left
	Mediastinum	Heart- Cardiothoracic Ratio-The maximum transverse diameter of the heart should not exceed 50% of the maximum transverse diameter of the chest on a standard posteroanterior (PA) radiograph
	Bones	The clavicles, ribs, and the spine

Common Conditions and their radiological signs

Pneumothorax: - Pneumothorax is the finding on a chest radiograph with the most important immediate clinical impact. It can be spontaneous, traumatic, or secondary which is caused by diseases like tuberculosis, cystic fibrosis, a chronic obstructive pulmonary disease with tobacco smoking being a major risk factor[78]. Its radiological sign includes a thin, sharply defined opaque density representing the visceral pleura is the hallmark of pneumothorax[79].

Consolidation: - occurs when the alveoli of the lung are filled with dense material such as pus (infection), blood (pulmonary hemorrhage), fluid (pulmonary edema), and cells (cancer). Common patterns of consolidation on X-ray include irregular shape(wedge-shaped area), increased opacity, and loss of air[10].

Atelectasis: - complete or partial collapse of the lung accompanied by hypoventilation. It can affect a lobe, segment, or all of the lung, resulting in a decrease in the ventilation or perfusion ratio. Atelectasis may be caused by obstruction of the airway, compression of the lung, or decreased surface tension in the alveoli and most frequently observed on the upper right lobe of the lung [80].

Pleural Effusion: - The possibility of observing an effusion on radiographic images relies on the amount of fluid present and the position of the patient. It may be more difficult to identify the fluid on the posterior-anterior (PA) view[81].

Fibrosis: - Fibrosis is the development of excess connective fibrous tissue as a reactive process in a tissue or organ. The cardinal patterns of fibrosis in X-ray are reticulation and honeycombing (clustered airspaces with well-defined walls)[82].

Pneumonia: - Can be caused by bronchitis, tracheitis infections and its radiological patterns can be described as air-space consolidation and loss of volume[81].

Pleural Thickening: - It is characterized by scarring or fibrosis of the visceral pleura. Common radiological findings of pleural thickening include smooth pleural opacity, irregularities around the periphery of the lung. It can also be a manifestation of several diseases such as pneumonia, lung cancer, and bacterial infections[83, 84].

Mass and Nodule: -Mass is characterized as an opacity measuring a diameter of 3 cm or more. In contrast, a nodule is less than 3 cm in diameter (maximum for nodule is 3cm)[81].

2.3 Machine Learning

Thousands of years ago, humans invented writing, which contributed to an explosion of knowledge. Agriculture was also invented, and humans begin to settle down and indeed create civilizations over several centuries. Following that, electricity was then invented and brought continuous transformations in every major industry. Now, a similar revolution of this magnitude is on the horizon. We are in the midst of a cutting-edge technology revolution. The digital revolution has continued to shape the future. It opens the door to the whole array of new tremendous possibilities, and it makes us realize that we are still standing on the verge of a massive technological revolution. We, humans, have followed a pattern of a relationship with technology; first comes bright-eyed optimism followed by a fear of the unknown. There is a heated debate in the world whether we should feel threatened by it or that it will benefit humankind. In the pages that follow, incredibly exciting concepts will be discussed.

AI is an umbrella term for a branch of computer science aimed at machines to simulate human cognition in real-life situations. ML is a subset of AI focused on how to make computers learn on their own without the need for hand-coded instructions. It enables computers to learn from examples. The more it learns, the better it becomes. As the rapid improvement of technology continues, we begin to see machines that can learn any intellectual task that a human can. A more recent survey on ML by Tejpal and Kaur [85], witness widespread ML applications in different fields such as data mining, face recognition, the Internet of Things, and the medical field at an unprecedented rate. In this era, appearing persistently releasing its power in an extensive variety of applications. Generally, an enormously fruitful shift is taking place in the AI landscape from attempts to build computer programs that solve health problems to robotics and other advanced machinery.

The main tasks of ML include regression, classification, and clustering. Regression and classification tasks are supervised learning in which a model is learned from the labeled examples in the training dataset. There is a kind of supervision for both tasks. The term classification refers to the task where each instance/example is classified into one of the predefined categories. While regression is used to predict numerical data values for instances. However, both classification and regression tasks build a model that can predict the output of unseen instances by observing a set of class-labeled data. Whereas clustering

is unsupervised learning because the input examples (training data) are not class labeled. Such unsupervised learning algorithms are based on the intrinsic characters of the unlabeled data[86]. However, despite much of the ML task research efforts up to now, there are still challenging problems that need to be tackled. Some of these challenges include scalability, algorithm selection, and optimization techniques. The Bayesian optimization technique is one of the state-of-the-art black-box optimization techniques in tuning models [87]. Such techniques call for the Bayesian DL paradigms and which will be covered in Section 2.7.

2.4 Neural Networks and Deep Learning

A neural network is a technology within ML that attempts to mimic the activity of neurons in our brain using matrix mathematics. These remarkable arrangements and powerful ML systems are called artificial neural networks[88]. Those with more layers than traditional neural networks are known as deep neural networks, and therefore, the name deep learning.

2.4.1 Neural Networks

An artificial neural network (ANN) is a technology that is loosely modeled after the biological neural networks in human brains (imitating biological neurons in a simplified way). ANNs are now widely used for many artificial intelligence applications including image analysis, speech recognition, robotics, and many more that make a meaningful impact on society at large. Perhaps one of the common uses of AI today is facial recognition software. Research into the neural network has a long history and surprisingly appeared to show considerable scholarly attention in recent years. Recent examples of research into the neural network include research carried out to mimic the complex dynamics of neurons, synapses and to better emulate the overall brain's functions that are intended to bridge the gap between biological and artificial neural networks[89]. ANNs consist of connected layers of nodes called neurons built to simulate the activity of the human brain, precisely simulate the function of the synapse (also called connections) by transmitting information (signal) from one neuron to another. This gave rise to the computational theory of mind and ML[90]. An artificial neuron is a very simplified mathematical abstraction of a neuron in the human brain. These biological neurons give some inspirations and impressions around ANNs.

With that said, neural networks are implemented in such manners; a real number passes as a signal through various layers of simulated neural connections between artificial neurons.

The neurons are activated by some input and the connection between neurons is called edges, and each edge has weight. The output of each artificial neuron is computed by passing the sum of inputs of that neuron through a nonlinear function[91, 92]. Weights are adjustable parameters (real numbers) that define the input-output function of nonlinear modules. Simple neural networks consist of an input layer and an output layer. When more layers are stacked, the networks are called deep neural networks (DNN) that have recently emerged as powerful computer algorithms tackling real-world tasks as models of human brain responses and behavior[93]. Figure 2.1, illustrates a typical model for a single hidden layer with five inputs.

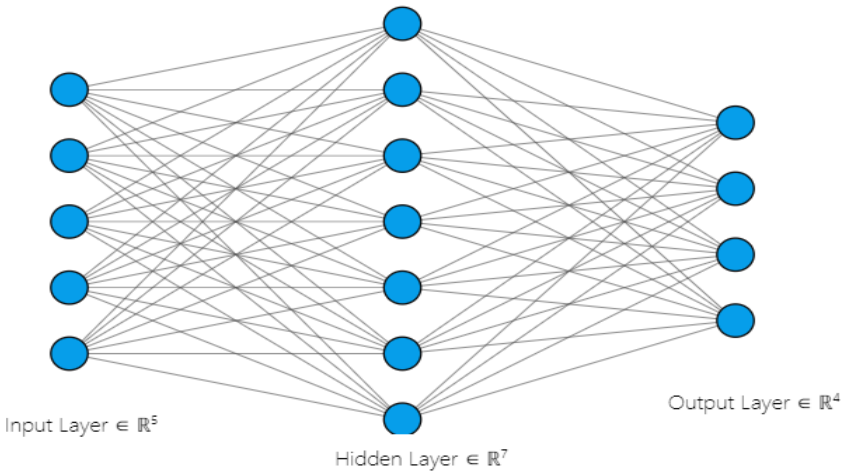

Figure 2.1: Examples of a Simple Neural Network

The first layer that represents that input value is called an input layer. The layers preceding the output layers are called hidden layers that is where all the computations happen. The last one is an output layer that contains the computed values from the hidden layer. The neurons are connected by weighted inputs as shown in Figure 2.2 [94].

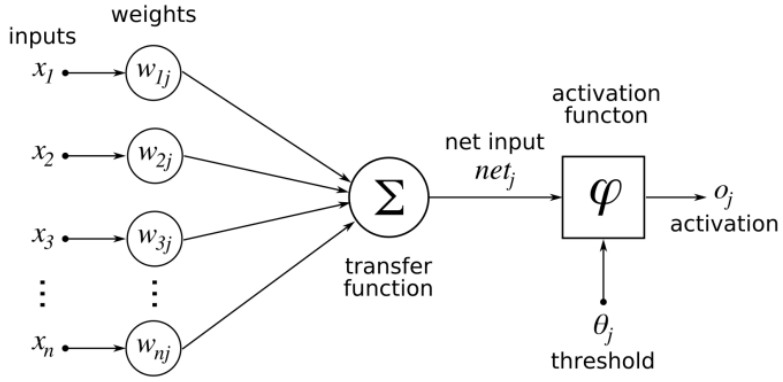

Figure 2.2: The Structure of an Artificial Neuron, the basic component of ANNs [94]

The function of a single neuron is intuitive; it receives input data, which may come in several forms such as images or text from either the outside world or other neurons, multiplied it with the assigned weights, then passes through an activation function. These artificial neurons use the activation function to calculate output, and the neurons can be connected to more extensive multilayer networks. It is therefore indeed a mathematical abstraction to a biological neuron which, in a very simplified way, receives electrical signals through its dendrites, transforms them in its synapses, and activates or not depending on the signals received. If a biological neuron activates, it means that it transmits the electrical signal received to other neurons. The artificial neural network generally consists of layers stacked with artificial neurons, and layers may differ from each other concerning how they transform the inputs. Looking at Figure 2.2, it is apparent that the neural network receives input values from the left-hand side and propagates the input values to the next layer repeatedly until it reached the output layer in a layer-by-layer process. This continued process is called forward propagation and this type of neural network is named accordingly a feed-forward neural network. It is necessary here to clarify exactly what is meant by an epoch because the term is used to refer to the processing of all examples/instances through the network in a training set one by one [94, 95]. The epoch-by-epoch process continues until an acceptable level of performance measure is reached.

The idea of the artificial neuron was applied to a binary classification problem: the algorithm obtained is called the perceptron. The activation of a neuron in the context of a perceptron thus corresponds to one of the two classes. Learning a perceptron corresponds to finding the weights of the neuron, which makes it possible to return the desired value. Significant analysis and discussion on the subject were presented by Singh and Banerjee[95] in which perceptron as the most basic form of neural network architecture with zero hidden layers is most prominently used for the classification of linearly separable patterns and can be referred to as a binary classifier. At the time, perceptron's made it possible to solve simple binary classification problems but were limited by their inability to solve certain types of problems. That's when a multi-layered perceptron comes in handy.

To solve this problem, it is necessary to pass from a perceptron having a layer of neurons, to a neural network with several layers (the multi-layered perceptron). The multi-layered perceptron (MLP) is an extension of the concepts derived for a feed-forward neural

network (that consists of only one hidden layer); however, a multilayer perceptron consists of multiple hidden layers. The hidden layers are used for computation, and the output layer represents the modeling purpose. Each node of the hidden layer must be connected with all nodes of the input layer, and then each node of the output layer must be connected to all nodes in the hidden layer, as shown in figure 2.3 [96]. As the neural network begins to contain more layers of interconnected nodes, it enables to learn more complex representation of the mathematical equation to map from input to output [95, 96].

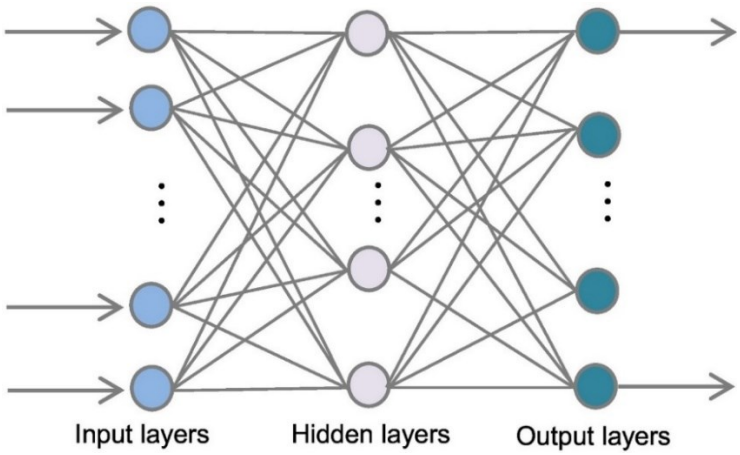

Figure 2.3: A common MLP Structure [96]

Through these connections, the training implementation of MLP can be divided into two stages: forward and backward propagations using the backpropagation algorithm[96]. The main goal of backpropagation is to incrementally adjust the weights for the network to produce values as close as possible to the expected values from the training data. In other words, backpropagation is used to compute gradients and these gradients are then used in gradient descent to train the neural network. Learning in a neural network is really about adjusting the weights either up or down, therefore, the error is reduced[97]. It is necessary here to clarify exactly what is meant by gradient descent because it is the way to decrease a loss function. Gradient descent is a very popular optimization technique in deep learning that attempts to find a global or local minimum point of a function. It is an optimization algorithm used to find function parameters (weights or coefficients) that minimize the differences between actual output and predicted output values. This enables a model to learn the direction that a model should take so that the network error will be reduced. Gradient descent has made it possible by iteratively moving in the direction of steepest descent and has many applications such as mini-batch, full-batch and stochastic gradient descent(SGD)[98].

However, there is no one way to improve the performance of DL models. In the pages that follow, a more detailed illustration of advanced neural network concepts will be given because the neural network has continued to advance quite impressively and it is time to take an interest in its future.

2.4.2 Deep Learning

Neural networks have been around since the earliest time of computing (the 1980s and 1990s) and are inspired by how human brains respond to the world. Deep networks have become feasible because of a significant increase in computing power and a vast amount of data. Fortunately, we reached a point in computing where neural networks are practical on real-world problems. To put this into perspective, Dean [99] reveals the core reasons why there is a growing revolution in the field. One is the paradigm of computing power that pushes the boundaries of DL researches. Another important aspect that adds to the current wave of neural network excitement is the data. It is an essence that allows machines to learn. Thus, all this gives birth to the modern AI movement.

Substantial research has been conducted that explains the important role of DL algorithms. Recently, Alom et al. [100] draw attention to the growing body of literature that recognizes the merits of DL in the fields of CV, image processing, speech recognition, cybersecurity, NLP, biomedical imaging, robotics, and much more. This discipline, a sub-domain of ML, has recently brought AI back to the fore. It has long been a question of great interest within the DL scientific community at inventing new algorithms and making non-stop progress on that front. Deep networks (the word "Deep" referring to the depth of networks or a large number of layers) therefore have their origins in relatively old approaches, however, it has become a powerful tool of machine learning and AI. The deeper it dives, the more complex information it extracts. It greatly benefited from its hierarchical structures to effectively capture complex functions [92].

Since its inception, DL has been showing outstanding data-driven applications in several domains and demonstrate state-of-the-art performance over traditional machine learning approaches. Perhaps the most interesting characteristic of DL is its capability to extract features on its own from a dataset rather than other methods that use hand-crafted features. Whereas traditional ML approaches use handcrafted engineering features[100].

DL continued to gain upward momentum because of its capability on feature learning than the traditional handcrafting techniques. Handcrafting refers to a set of rules and algorithms

to extract features from raw data. This is common in conventional ML techniques that heavily rely on human knowledge intervention or domain knowledge. This data processing (commonly called feature extraction) is often very time-consuming and may require human expertise. The idea of DL is to automatically build this relevant representation of data through the learning phase, thus avoiding human intervention[101]. We, therefore, speak of learning by representation. A DL algorithm will learn increasingly complex hierarchical representations of data. Moreover, Wani et al.[101] provide some explanation as to why DL in the choice of features has a profound impact on the remarkable success of ML systems. According to the rationale behind the authors, DL algorithms seek to exploit the unknown structure in the input distribution to discover useful representations, often at multiple levels, with higher-level learned features defined in terms of lower-level features. It removes the dependency on hand-designed features and create algorithms to develop its interpretation of data, in turn build its own set of features.

Johnson and Khoshgoftaar [102] define representation learning as the process of using ML to map the features of raw input data to that new feature space (new representation) to improve classification and detection tasks. Non-linear activations functions would be performed to map the input raw data to the new representation. Such multiple non-linear activations produce hierarchical representations of the raw input data and increasing the level of abstraction in each transformation. As sufficient data passes through the hidden layers of deep networks, the network can learn high-level feature representations of inputs through multiple compositions of hidden layers. Zhao et al.[103] , in their study of DL and its applications to health monitoring systems, shed light on the underlying representation learning to the extent where no extensive human labor and expert knowledge is needed. Zhao et al. [103] also noticed that the popularity of DL today can undoubtedly contribute to the increasing computing power, increasing data size, and advanced DL research.

Current research in the field is exceptionally active and new architectures of neural networks, new types of layers, and new learning techniques appear very regularly. And yet, there are still many areas for improvement of deep neural networks. Here are a few areas related to training dynamics of deep neural networks fall under three headings: 1) Transfer Learning 2) Hyperparameter Tuning 3) Overfitting and Regularization

2.4.3 Transfer Learning

This section attempts to provide a summary of the literature relating to transfer learning (TL) to adapt pre-trained networks to a different task domain from the one it was trained on. An assumption characterized in traditional ML to obtain a high accuracy classification model is to train data and test data having the same input feature space and the same data distribution and availability of enough training dataset[104]. Especially, training neural networks with many layers requires large-scale training sets. Unfortunately, the underlying assumption suffers from serious limitations in practical applications due to many reasons. The challenge now is to obtain training data that have the same feature space and the same data distribution as the test data. Another potential challenge is that a sufficiently large dataset size is rare to find. Previous DL scientific works have shown significant promise but require large amounts of data for training[105].

To deal with such kind of domain adaption scenarios, the notion of TL comes to the rescue; it considers overcoming the unique complexities faced in training deep neural networks. In general, therefore, leading to a substantial reduction in training overhead. For instance, Lu et al. [106] introduced TL to train a deep neural network of magnetic resonance images (MRI) of a brain because the quantity of brain datasets is often too small for training entire DL structures that can be easily trapped into overfitting problem.

TL appeared as a popular technique to improve a learner from one domain by transferring knowledge from a related domain. However, the difference in data distributions among related, but different domains is a long-standing problem for knowledge adaptation. The knowledge of the source domain should fit the target domain [107]. When transferring knowledge from a less related source, it may inversely hurt the target performance, a phenomenon known as a negative transfer which occurs when initial learning makes learning difficult in other contexts. This indicates a task that would harm the learning efficiency without any practical benefit[108].

Human beings transfer knowledge based on previous experiences to learn or solve a problem across domains. True learning, therefore, goes beyond a mere reminder or recall of what has been learned, but rather implies the generalization of new concepts or skills acquired so that it can be applied in other contexts. In the same vein, Zhuang et al.[28] state the fact that TL is aimed to leverage the transfer of knowledge in different tasks however related domains to break down barriers between them and improve the learning

performance. TL takes advantage of the knowledge (characteristics, weights) of previously trained models to train newer models and even tackle problems such as having less data for the newer task. In other words, it utilizes previously learned knowledge to tackle the bottleneck of low accuracy prediction models that occurred on small datasets (target domains with insufficient training data). Taken together, TL opens up radical new forms of knowledge transfer across domains instead of training a deep neural network from scratch. There are, however, other possible explanations regarding the transfer of knowledge. Deep down, an important question should be answered during the learning transfer process. A question of what part of the knowledge can be transferred from the source to the target to improve the performance of the target task. This identifies what part of the knowledge is specific to the source and what is common between the source and the target tasks. There are possible problematic situations that affect the success of transfer learning. Zhuang et al.[28] explain scenarios where knowledge transfer can make things worse rather than better (also known as negative transfer).

TL has been widely used in a variety of applications, including medical imaging [105, 106, 109], credit risk analysis (domains) [110], tuberculosis detection[111], forensic applications[112], software user experience(UX)[113], information extraction[114], drug discovery[115] and many more diverse set of domains. The realms in transfer learning are endless. The application of DL to TL which is known as deep transfer learning (DTL) is also popular. Vu et al. [116] proposed a transfer learning approach based on the autoencoder for the Internet-of-Things (IoT) network attack detection. The concept of the autoencoder is used for unsupervised feature representation which will be covered in later sections of this paper. Therefore, DTL is the paradigm using deep learning to transfer learning. According to Wen et al.[117], DTL extracts far more robust high-level features transferred from the source to the target domain compared to the conventional shallow structures. In a research study, Sufian et al.[118] revealed how complex and expensive a high-quality large-scale medical dataset is to find. Another important practical implication is that DL models demand large computing power to train deep networks from scratch. Two distinct reasons emerged from this. First, concerning a dataset, and more specifically given the fact high-quality annotated medical dataset is rare to find, and second, DL systems consume huge data and are fueled by the data. There is, therefore, a definite need for DTL that significantly minimizes the demand for training time and training data for a target domain-specific task.

In the TL approach, the neural network is trained in two stages a pre-trained stage followed by a fine-tuning process. In the first stage, the deep network was pre-trained on a large-scale benchmark dataset with sufficient size (e.g., ImageNet) but not necessarily similar to the target dataset (that represents a limited size). In the fine-tuning stage, the pre-trained network is further trained on the particular target task of interest, which may have insufficient labeled examples than the pretraining dataset [17]. In that way, the use of a methodology that follows the principles of TL marked a difference in training ML models.

2.4.4 Hyperparameter Tuning

It is quite difficult to train a network successfully because it requires us to minimize a function, which we know very little about. However, there is ample room for further improvement to the standard training algorithms to push the performance of network models towards the state of art. One area that can make a significant difference in the performance of DL models is the choice of hyperparameters (also called hyperparameter tuning). When training a neural network, one must make several choices, and picking the most optimal hyperparameter value is often difficult and time-consuming[119]. In general, therefore, finding an optimal value involves an iterative process in which it often starts with an idea such as training a deep neural network of a certain number of hidden layers and units to be used between the input and the output layers. Then, code it up, and based on how it is performing and keep iterating upon unsatisfactory outcomes by adjusting configuration settings and refining ideas to achieve an acceptable level of performance. Yang and Shami [120] defined hyperparameter tuning as configuration variables that govern the behavior of the learning algorithm or more broadly, the training process itself to design the ideal model architecture.

Having defined what hyperparameter tuning is supposed to mean, it is now necessary to distinguish hyperparameters from parameters. There is a profound distinction between the parameters and the hyperparameter. The term hyperparameter is quite different from these learnable parameters that are adjusted with gradient descent. This is exemplified in the work undertaken by Dong et al.[121], who define hyperparameters as parameters whose values are set fixed before starting a learning process. Whereas the values of parameters are determined and derived from the training process (for example, biases and weight coefficients of a neural network). For an in-depth review of hyperparameters, Yoo[122] provides numerous examples of hyperparameters such as regularization parameters,

learning rate, mini-batch size, cost function, and the number of hidden layers. As an illustration, the learning rate is taken into account which determines the step size of the gradient descent iterative parameter updates. It determines the rate at which the network weights are updated. In its most general sense, the term learning rate has come to be used to refer to the notion that decides the movement speed in the direction of the gradient[122]. If the learning rate is so high, then the model is difficult to train because the gradient descent may go unstable by just oscillating around the local minimum value and may not even converge. In contrast to this, a considerably small learning rate converges to a global minimum smoothly, but the learning process would take more training time in more epochs and more likely overfit. The goal seeks to find an optimal learning rate that converges smoothly in a reasonable time[120]. To converge the learning rate at a reasonable time, Park et al.[123] adopted a strategy to adjust the learning rate during training rather than applying a fixed value.

To evaluate the performance of various hyperparameter settings, the data is usually split into training, testing, and validation sets. Multiple hyperparameter settings are applied to the training set and evaluated on the validation set. The best performance settings are chosen for the evaluation of the test set to achieve the final performance. According to Bakambekova and James[124], training of a neural network can be expressed simply as the process of finding the right set of weights and biases or trainable parameters, so that if fed with input data the network will give the right output.

Taken together, the choice of hyperparameters can have the potential to make a significant improvement to the standard use of training algorithms. Shrestha and Mahmood [125] highlighted several shortcomings and suggest hyperparameter tuning or optimization (commonly learning rates and regularization parameters) for enhancing the accuracy of the neural networks. The purpose of regularization is the prevent overfitting and the regularization parameter affects the degree of influence on the loss function. A more detailed explanation of regularization will be covered in the following section (Section 2.4.5)

2.4.5 Overfitting and Regularization

Neural networks open up a feature-rich framework with practically unlimited scope to improve the performance of the given models by increasing the complexity of the network. Complexity can be increased by manipulating various factors such as a growing number of hidden layers, increasing the nodes in hidden layers, using complex activation functions, and increasing the training epochs. Such an arbitrary increase in complexity typically leads to overfitting. What is appealing about neural networks is that it is capable of learning complex patterns in data; however, it naturally comes with overfitting problems. Overfitting is a phenomenon to model the training data too well by memorizing the training data rather than identifying the features and structures. Such memorization brings about significantly worse performance on unseen data [126]. Overfitting happens because of the presence of limited dataset size, noise, and complex model classifiers [127]. Overfitted data affects the generalization capability of neural networks. Generalizability is defined as the deviation of model performance when evaluated on previously seen examples and to data that has never been seen before. There are rigorous theoretical justifications to reduce the generalization error. However, one of the main difficulties with this line of reasoning is that improving model generalizability emerged with serious challenges[128].

Generalizing beyond the training set examples is the fundamental goal of machine learning systems. Continued outstanding research efforts have been conducted to develop effective strategies and techniques to have machine learning systems that work well on training examples and the new data[129]. There are quite several techniques that help to prevent overfitting. Among these techniques, Maharjan et al.[130] mentioned regularization as the robust technique for reducing the risk of overfitting of data, and if the regularization is not carried out, then the network weights rise over time, and the learning rate decreases. This makes the neural network complex and the network cannot accurately classify the unseen data. Dropout and weight decay (also called L2 regularization) are the most common form of stochastic regularization techniques[131]. Another related regularization approach is batch normalization that enhances the generalization of a network by normalizing feature representations through internal covariance shift reduction[132]. However, regularization is not the only strategy that has been introduced to minimize overfitting.

Ying [127] illustrates why overfitting happens and raised important theoretical issues that have a bearing on the effect of poor generalization. It offers some important insights from

the perspective of cause and various strategies to address these multifaceted issues. To put it simply, the proposed strategies include early-stopping, regularization, training data expansion, and network reduction to reduce classification complexity. Training data expansion is proposed to deal with the nature of limited dataset size (insufficient training data) that leads to overfitting. To acquire more training data, Shorten and Khoshgoftaar [128] suggest an approach called data augmentation that brought solutions to overcome the problem of overfitting due to limited data. It modifies the small dataset to generate artificially inflated more training data. Additional training examples are generated using various geometric transformations such as scaling, zooming, cropping of images, rotation, translations, horizontal and vertical flipping [133]. Moreover, massive labeled data might not even be possible to collect in certain real-world issues such as medical diagnosis. This has led to leveraging data-driven methods based on large-scale unlabeled data (also known as unsupervised learning)[134]. Generally speaking, overfitting remains a major challenge for computer vision (CV) tasks such as denoising and classification[135]. A combination of breadth of coverage and depth of CV backgrounds will be covered in Section 2.5.

2.5 Deep Learning in Computer Vision

Computer vision has become a separate field of computer science in the last two decades. It is an interdisciplinary area that comprises all aspects of image and video processing that can be used in artificial visual systems for automatic scene analysis, interpretation, and understanding. The history of computer vision can be traced back to the 1960s unlike traditional image processing. The enormous growth of cutting-edge methods, algorithms, papers, and the widely discussed appearance of autonomous vehicles brought computer vision into public awareness[136]. For example, computer vision is assisting autonomous vehicles to figure out where the other cars and pedestrians are, so as to avoid them. Today, advances in many fields of knowledge benefit from advances in computer vision. Within the paradigm shifts of artificial intelligence across the different fields of science, Ibrahim et al. [137] indicate that there is a very important milestone in computer vision in tackling a variety of complex visual tasks with machine eyes. A huge enthusiasm in the field has allowed researchers to develop algorithms that simulate the human vision to learn the visual world and draw conclusions. Object recognition, which identifies whether the input image contains a particular object, is one such classic example of a visual task[138]. The underlying computer vision technologies have vast areas of applications such as remote sensing, medical image processing, precision agriculture, satellite image, defense, and

many more[139]. More specifically, Chouhan et al.[139] surveys computer vision applications in classification and diagnosis of disease from plant leaf images to enhance crop productivity.

Computer vision is therefore quite appealing in a wide range of fields. The application of computer science to medical imaging represents one of the greatest promises of computer vision. As an example, imagine a radiologist who would be assisted by software allowing to detect anomalies on a radiograph that managed to extract relevant information about certain bone structures present in an x-ray image. The time saved for the radiologist and the gain in safety for the patient can be enormous. As a research area, there is a growing body of literature that recognizes the importance of computer vision in the field of healthcare.

The specific question here would be what does deep learning allows for computer vision? The overall answer to the question is that deep learning makes it possible for computer vision to deepen its primary capabilities and, above all, to understand the image being analyzed. A deep neural network is found to be the technology behind deep learning models. Shorten and Khoshgoftaar [128] provide important insights into the role of deep neural networks in CV tasks such as image classification, image segmentation, and object detection. One could say that there are many neural network variants. An exhaustive list of every neural network variant would be infeasible, and outside the scope of this research. Therefore, Section 2.5.1 made an effort to cover the world of convolutional neural networks and their applications.

2.5.1 Convolutional Neural Networks

The specific type of artificial neural network that accomplishes image classification, object detection, and other tasks is called a convolutional neural network (CNN). CNN has become the de facto standard for various computer vision and machine learning operations and has been widely used in various computer visual recognition tasks and has achieved outstanding results compared with traditional methods[140, 141]. CNN-based computer vision has enabled the scientific community to accomplish incredible applications that have been impossible in the past such as autonomous vehicles, intelligent medical treatment, and face recognition[142].

The groundbreaking advances in the CV field are posed by the ever-increasing computing power and the massive amounts of data. Perhaps rigorous theoretical justifications and real-world examples on CNN have been provided. The question is how does CNN work? CNN works by dividing an image into smaller groups of pixels called a filter. Each filter is a pixel matrix, and the network performs a series of calculations on these pixels which compare them to pixels in a particular pattern. CNN consists of a series of layers that learn to extract relevant features from images. The most common building blocks encountered in CNN architectures are the convolution, fully-connected, and pooling layers. They are feature extractors, classification, and dimensionality reduction layers respectively. Several different image filters (also called convolutional kernels) are applied to the input image by the convolution layer. This means that when training a CNN, the network will apply filters that capture sort of visual information, such as edge, orientation, and then, in a higher layer of the network, entire patterns[143].

Convolution Layer

Convolutional layers are the parts of CNN where convolution operation is executed. The image through the convolution layer can be seen as a process of extracting features of the image. Unlike human vision, which identifies objects visually by brightness, size, and contour, computer vision identifies the image of an object through mathematical operations. The convolution operation is implemented by using certain sized filters (commonly called window size). Each filter slid over the image, the activation function is applied to the output, and a feature map is created with these calculated values. The multiplication of these layers within the neural network will make it possible to extract increasingly complex features that will ultimately make it possible to predict the class items in the image. These convolutional filters are trainable parameters. Filter weights are trained by the amount of error that falls on the relevant feature map during backpropagation[144, 145].

Pooling Layer

The purpose of the pooling layer is to compress the spatial size of the image representation captured by the convolutional layer while preserving their important characteristics. It mainly simplifies the information collected and creates a condensed version of the same information. In other words, the operation is used for down-sampling the feature maps. The most popular form of pooling is max-pooling. The max-pooling layer slides a window over its input and takes the max value in the window, discarding all other values[146].

Fully Connected Layer

The image is first passed through a series of convolution for feature extraction followed by a pooling layer to reduce the size of feature maps and then is passed through fully connected layers for classification. The main function of a fully connected layer is to perform classification on the features detected and extracted by the series of convolutional layers and pooling layers. The output of pooling and convolution layers are 3D volumes, and therefore, the features maps are flattened into a single 1D vector because a fully connected layer takes 1D vector numbers as input. It is generally used as a classifier in CNN structures[145, 146].

Taken together, in the first layer of a CNN, it can detect high-level patterns for instance rough edges and curves. As the network performs more convolutions, it can begin to identify specific objects like faces and animals. DL has been applied to analyze not only images but also in video processing because, at its core, a video is just a series of image frames.

2.5.2 CNN Architectures

There are several architectures in the field of convolutional networks pioneering with LeNet in 1998 which was experimented on handwritten classifications. However, the history of deeper CNN architecture in CV began with Alex-Net(in 2012), which won the far more difficult ImageNet challenge for visual object recognition called the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [147]. It is only since the work of AlexNet that the study of CNN architecture has gained momentum. Alex-Net takes an input size of 227×227 images of RGB channels. It consists of five convolutional layers and is then followed by three fully connected layers. There are 96 to 384 filters within each convolution layer and their filter size varies from 33 to 1111, with 3 to 256 channels each[142]. It was then followed by the ZFNet architecture as shown in Figure 2.4[141].

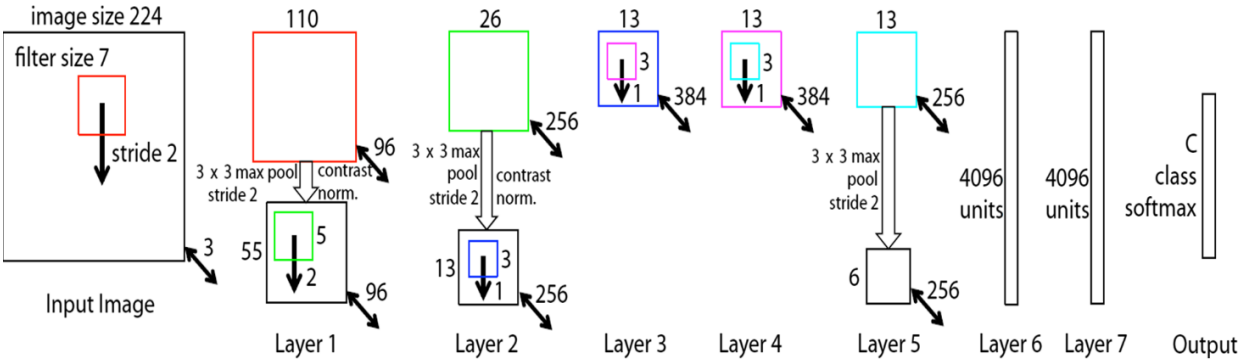

Figure 2.4: ZFNet Architecture [141]

ZFNet (stands for Zeiler & Fergus Network) emerged as an improvement on Alex-Net by modifying the hyperparameters of the architecture, in particular by reducing the size of the filters (uses 7x7 convolution kernels instead of 11x11) to significantly reduce the number of parameters[147]. As shown in figure 2.4 [141], ZFNet architecture takes an input image of size 224×224 with RGB channels. Similar to AlexNet, the ZFNet network consists of five convolutional layers stacked together to extract high-level features. Max-pooling and drop out are also applied and followed by three fully connected layers. The last layer (final layer) of the ZFNet is the SoftMax layers used to translate the score to a probability that the image belongs to each group (1000 categories in ImageNet dataset). Some other notable examples of CNN state-of-art architectures with outstanding classification performance include DenseNet, GoogleNet, and Res-Net [148]. GoogleNet was a particular incarnation of the inception model with the goal of dimension-reduction by eliminating a large number of network parameters that have a huge computational burden and overfitting problem[149]. Res-Net (short for Residual Network) features skip-connections (shortcut connections) that outperform the GoogleNet inception V3[150]. Similar to Residual Network (ResNet), DenseNet was developed to mitigate the vanishing gradient problem. It enables cross-layer depth-wise connectivity by concatenating previous layers' information before assigning to the new transformation layer rather than adding them[151].

There has been substantial work on image classification and recognition using convolutional neural networks that have been discussed in several publications of diverse fields. Although these findings should be interpreted with caution, these studies have several strengths that give birth to promising new architectures. A series of vigorous challenges and improvements followed each other based on experimental findings. One major advance to current CNN architectures was found in a paper called EfficientNet[152]. Tan and Le [152], consider the computing resources and scalability issues to develop scaling up a convolutional network to overcome computing hardware resource constraints.

Various improvements in CNN architecture have indeed been made since 1989. These advancements can be classified as regularization, optimization, and structural reformulation. However, most innovations come from the new design of CNN blocks [151]. Overall, CNN architectures have been a continuing source of encouragement and optimism in the field of computer vision.

2.6 AI in Healthcare

This section seeks to assess the far-reaching impacts of AI in healthcare. The conjunction of the rapid advances in the field of AI with socially impactful projects can influence the vast areas of our lives. One might be wondering where AI fits in the world of healthcare. In the Covid-19 pandemic, for example, there are AI-based initiatives to support the fight against the viral spread. Covid-19(coronavirus) was recently discovered in Wuhan, China's Hubei Province, and then spreads throughout China and other parts of the world[153]. To counter the pandemic, Majeed et al.[154] harness the potential of AI to facilitate radiologists in accelerating the speed of the diagnosing process. Its main focus was to investigate the applicability of AI to detect Covid-19 in chest X-ray images. The striking lesson when analyzing all the research efforts is the wide variety of fields AI is involved. AI appears everywhere powering everything from viral screening to drug discovery[115]. There are, however, several barriers to bringing advances in AI to the clinical setting, including the problem of generalization, limitations in the available training data, lack of interpretability, and lack standards for reproducibility of publications[155].

Google health breast cancer paper, published in 2020 by McKinney et al. [156], provides an in-depth analysis of the work of AI systems showing its relevance to an early screening of breast cancer. What is remarkable about the study is that it made an effort to link breast cancer large representative data from different clinical settings (USA and UK) and assess the generalizability of the AI system. McKinney et al. [156] found a major problem in previous researches because little or no evidence of AI systems translate among different screening settings and populations without additional training data. Another recent clinically relevant application of AI that can benefit patients and doctors comes from researchers at Google AI Healthcare who developed an algorithm that analyzed histology of stained tissue samples [157]. Such algorithms can be used by health-professionals to interpret patient data more quickly, without sacrificing performance. Turning now to the image-based algorithm, published in the summer of 2019, Liu et al.[157] created a learning algorithm called LYNA (short for Lymph Node Assistant) to identify metastatic breast cancer tumors from lymph node biopsies. The algorithm can identify suspicious regions indistinguishable to the human eye in the biopsy samples provided for physicians to examine more closely as shown in Figure 2.5 [157].

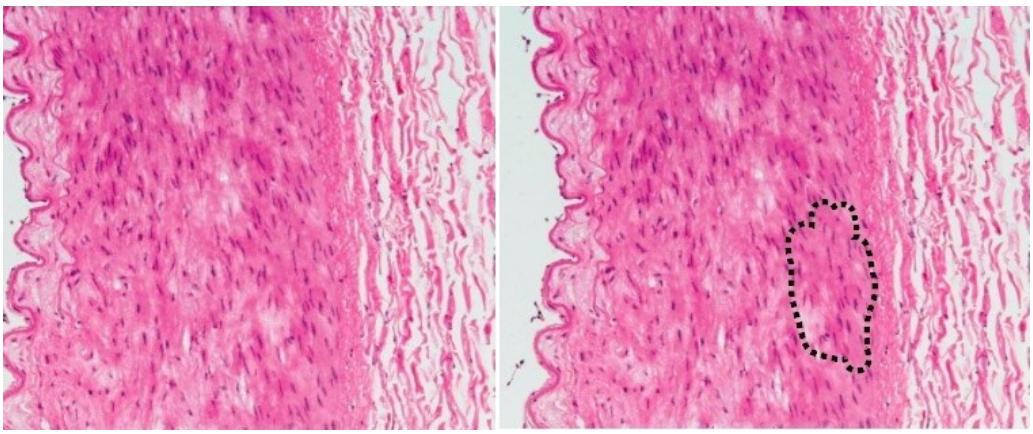

Figure 2.5: Image-based Detection Algorithm [157]

In another major study, Zhou et al.[158] illustrates the increased value of AI in diagnosing liver disease and continued to focus its applications and clinical challenges. This upward turn in popularity of AI, especially deep learning has shown promising aid in the evaluation and detection of liver diagnosis. As argued by Zhou et al. [158], automatic quantitative robust assessment of medical images by clinical deep learning applications is far more cost-effective and therefore, gives rise to promising solutions to health care facilities and patient care. It opens up new perspectives and taking a whole new dimension in the detection of disease for physicians to save time and optimize the diagnosis because physicians, or more generally health professionals can suffer from fatigue and tiredness. Moreover, the effects of frustrated clinicians have been closely examined by Rajkomar et al.[159], in which the time to read patient history and medical errors is so high.

Such health-oriented technologies can be valuable resources to tackle problems and bring great value and actionable insights into the entire healthcare ecosystem. Due to the high volume of data generated as well as the increasing prevalence of medical devices and digital record systems, medicine and healthcare stand to benefit tremendously from DL[160]. DL algorithms and techniques allow automatic detection of structures of a human body such as liver[158], kidney injury [161], cancer[156], and bone fractures in an image ranging from mild illness to life-threatening complications. DL represents the recent iteration in the technological advancement of AI that has undoubtedly permitted machines to imitate human intelligence in highly sophisticated and independent ways[162].

The rise of AI in healthcare applications is, therefore, fantastically valuable in the modern health ecosystem. Bohr and Memarzadeh [163] found quite revealing applications of AI algorithms in several ways enabling a new form of healthcare delivery from patient proactive experience (early radiological diagnostics) to the development of precision medicine (personalized medicine based on individuals disease profile). With this line of reasoning, Bohr and Memarzadeh [163] have also added several examples of AI applications that are at the heart of the growth of the health industry such as genetics-based solutions, drug discovery, biological activity prediction, surgery, and disease prediction. Healthcare is undergoing profound changes to optimize radiological diagnostics, follow-ups, and improve clinician workflow efficiencies[164]. The central focus of the study [164], was to assess the potential impacts of AI technologies on the future of healthcare and found a lack of empathy between the new technologies and patients in today's society. This indicates a need to have face-to-face communication with patients, and therefore, provide empathic care. However, a 2019 publication from Becker [165] states the current use of AI since its infancy and future promises in healthcare is remarkable. The most important limitation of the study by Blease et al. [164] lies in the fact that the findings in biomedical informaticians differ from physicians. AI-based technologies are a far-fetching expanding field and still an ongoing area of reach, however, that will never come to a point in time at which it will be completely independent of a human physician [165]. Becker [165] argues that AI is a powerful tool that should make our lives better not replace us. There is, therefore, a growing body of literature that provides a relevant response to the major current and future challenges in digital health that bring added value to both patients and health professionals such as reducing waiting time in hospitals while offering more precise detection of disease.

The radiology space represents one area of healthcare in which AI has been largely influential. AI has been appealing to radiology and the whole point of it is to detect radiographic features (characteristics) contained in radiologic images or generally medical imaging obscured to human viewers that suggest future treatment and diagnostic options[166]. As a field, radiology needs greater perceptual expertise in radiological image interpretations for radiologists to recognize abnormalities (subtle shapes in noisy backgrounds and textures) to improve accuracy, reduce long-standing errors (33% error rate), and enhance patient care[167]. According to Waite et al.[167], the interpretation errors that are introduced due to a lack of perceptual expertise can eventually lead to a

missed (inaccurate) diagnosis in the clinical diagnosis. Furthermore, radiologists face excess workload in routine medical imaging examinations such as X-ray, CT (short for computed tomography), and magnetic resonance imaging (MRI) which collectively contribute to errors[168]. However, DL algorithms do have enormous potential to analyze a rich set of features, which include features not previously been considered by radiologists, and to reach a repeatable conclusion within a reasonable time [169]. The diagnostic features can include location, texture, signal intensity, size, heterogeneity, shape, borders, and some other features present in the image data[169]. one of several existing examples of an algorithm in image classification tasks is the work of Nam et al.[170] from the National University Hospital researchers in 2018 who have developed an AI algorithm called DLAD (Deep Learning-based Automatic Detection - Automatic Detection Based Learning Deep) to analyze chest X-rays and detect abnormal cell growths. Spotting dangerous items in radiograph images is impactful and attention-grabbing for inviting other researchers to brought clinically relevant algorithms.

One of the promising use of deep learning in healthcare includes the works of Stephen et al.[171] who developed a CNN model to extract distinguishing features from radiographic images of the chest and thereafter determine whether a person is infected with pneumonia. As infectious pneumonia can be observed on x-ray images, deep learning can be employed for automatic analysis of radiological imaging data related to assisting in the diagnosis of pneumonia. Deep learning is preferred because there are huge variations and complexities among the medical images as well as the nature of human fatigue can give rise to limited performance and error-prone[172]. Moreover, detection and diagnosis of disease in radiographic images is a challenging task that heavily relies on highly skilled professional radiologists with clinical diagnostic experiences[173]. There is additionally a shortage of radiologist experts in the radiology field who are skillful to read chest radiographs, especially in rural areas. Delays in the diagnosis process and interpretation errors lead to a significant number of patient deaths in hospitals [172].

DL and its cutting-edge researches constitute an important solution, and therefore, DL in the medical field is undeniable. For example, it makes possible the earlier detection of cancerous tumors[156]. It transformed healthcare into a new era by speeding up the diagnosis and reveal new structural patterns to diagnose a hernia, effusion, atelectasis, cancer, pneumonia, and other diseases [172]. Techniques in DL for detecting abnormalities in X-ray images were first introduced in 2015 using the GoogleNet algorithm.

In radiology, the techniques of DL include semantic segmentation (converting images into parts), object detection (extract a particular area from an image or finding abnormalities), image processing, and automatic manipulations of radiology reports using natural language processing(NLP)[174]. Automatically detecting abnormalities with great accuracy can significantly improve diagnostic processes. However, the lack of standard datasets makes the detection method difficult in real-world settings (the reality is quite different) [175].

The researches to date have not been undertaken in a variety of healthcare settings though it should be validated in real clinical situations[176]. Another practical and realistic constraint is that small-scale health databases remain indeed a major bottleneck in the healthcare field for DL researches[177]. Generally, there has been substantial work on medical image analysis using DL models. Xie et al.[177] discussed and categorized different aspects of health knowledge that can be incorporated into ML (specifically DL models). These are the training pattern, the general diagnostic patterns radiologists view in images (because radiologists usually follow certain patterns when reading images), specific areas of focus, and the features (e.g., characteristics, structures, shapes) to which radiologists give special attention. However, this comes with its own limitation and open problems because human experience including radiologists could go subjective. Second, representing the radiologist's knowledge which is in the form of description remains a major challenge. In general, incorporating expertise of the health domain into DL has a price to pay [177]. Although DL techniques have shown astounding results in image-based diagnostic tasks, these algorithms are vulnerable to bias introduced when insufficient or diverse data sets are used to train algorithms [178]. There is a wealth of literature to suggest that medical data changes over time and location, as clinical practices are not constant and vary considerably among hospitals, therefore, leads to performance decline[179].

Returning to the question posed at the beginning of this section, it is now possible to state that AI and healthcare well go together for improving the efficiency of diagnosis, especially for the tedious and time-consuming manual tasks. Healthcare is likely to be profoundly transformed by AI, specifically by its subfield (DL). Continued to focus on its applications, DL has shown a renaissance owning significant improvements in solving clinical problems [156]. Through this area of scientific research, therefore, offers numerous health benefits, such as patient care, aid in radiological diagnostics, treatment, and prevention of different diseases from recent advances in object detection,

segmentation, and object classification. On the other hand, despite these significant advantages, DL techniques should consider capturing the high variability within data to handle latent confounders (discover unseen casual factors of variation in the data)[180]. This is to extract new clinical features that can be discovered within the data to facilitate structural pattern recognition. In the pages that follow, more specifically, latent representation learning will be discussed in the Bayesian deep learning section (Section 2.7).

2.7 Bayesian Deep Learning

DL, probabilistic programming (PP), and the Big Data boom are among the major AI-oriented technological rising trends. This paves the way to innovative technologies that open up new possibilities for AI at an accelerated pace. The surprising flood in the volume of data (termed Big Data) provides DL models a fertile ground to discover interesting patterns and facilitate knowledge discovery. To extract unknown correlations and meaningful value from data, ML techniques leverage statistical models via probabilistic theories. Therefore, probabilistic programming emerged to model complex real-life tasks with reasoning capabilities[181]. The core idea behind the probabilistic paradigm is that machines can develop representational capability (reasoning) of latent or unobserved data that can impact uncertain consequences. This is crucial because observed data is consistent with many models and also useless on its own unless inferences or knowledge is extracted from it[182]. In its most general sense, probabilistic programming refers to the idea of writing statistical models based on machine learning and provides a basis (reasoning) to justify these models[183]. A model can be quite simple and inflexible, such as a classic statistical linear regression model, or a deep and flexible model with lots of parameters such as deep neural networks[182].

DL and ML models have their limitations and strengths. On strength evidence of ML, Dusenberry et al.[184] argue that ML models such as the Bayesian neural network continue to improve the soundness and uncertainty of quantitative evaluation of contemporary DL techniques. However, despite their improved robustness, these machine learning systems suffer from underfitting problems and struggle with parameter efficiency. This gives rise to the emergence of deep learning as an alternative however, deep learning also faces a poor generalization dilemma[184]. The lack of generalization capability is particularly problematic for real-world applications such as self-driving cars, trade, finance and medical diagnostics where failure may even lead to catastrophic outcomes[185].

With a deep questioning mind and logical reasoning, one can ask a penetrating question about the notion of combining the strength of DL and ML approaches. More specifically, begs the question of bridging the gap between DL and PP. Wang and Yeung [186] provide a more plausible explanation and speculation on Bayesian deep learning that seeks to combine the merits of both approaches. The Bayesian paradigm is therefore a crucial topic. However, before diving into the technical depth of Bayesian deep learning, readers could be interested in knowing the more general principles and underlying theories, theorems, and theoretical motives behind the paradigm.

The flourishing Bayesian statistical approach revolves around the use of the theorem of Bayes and the rules of probability sum and product. Probability is a mathematical phenomenon used to express the degree to which the chances of a particular event occur. Probability reasoning, therefore, uses probability and logic in making decisions especially in times of uncertainty[181]. As shown in Equation (1), Bayes' Theorem is a mathematical formula that refers to a statistical probability that an event will occur based on a previous event, also called conditional probability[182].

$$\boldsymbol{P(y|x) = \frac{P(x|y)P(y)}{P(x)}} \qquad (1)$$

Where x represents the observed data (for example images), and y corresponds to uncertain quantities. P(y|x) is the posterior probability of y given the observed value of x. P(y) is the original probability of the class **y** (also known as a prior probability). P(x|y) is the probability of the observed data (predicator) given a class. P(x) is the original probability of the predictor (evidence). P(x) can be expressed as the sum rule $\sum_{y \in Y} \boldsymbol{P(x, y)}$ which represents the marginal of x obtained by summing over the joint y (or does integration for continuous variables). The probability theory can be applied to ML and therefore, the formula can be written as Equation (2) [182].

$$\boldsymbol{P(\theta|D, m) = \frac{P(D|\theta, m)P(\theta|m)}{P(D|m)}} \qquad (2)$$

Where **θ** denotes unknown model parameters, **m** is the class of probabilistic model, P(D|θ,m) denotes the likelihood of parameters **θ** in model **m,** P(θ|D, m) is the posterior of **θ** given the observed data **D**, and P(θ|m) is the prior probability of **θ.**

The Bayes theorem establishes a relationship between ML and probability theory. The Bayesian statistical approach operates with certain knowledge (evidence). It combines

subjectively accumulated evidence (hypothesis) with objective information from the data. In the presence of continuous variables and multiple models $\boldsymbol{\theta}$ and its associated probability, P($\boldsymbol{\theta}$) rather than a point estimate, the Bayes theorem can be written as follows to enforce the parameter independence with input data[185].

$$\boldsymbol{P}(\boldsymbol{\theta}\,|D) = \frac{p(D\boldsymbol{y}\,|D\boldsymbol{x}\,,\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int \boldsymbol{\theta}\, p(D\boldsymbol{y}\,|D\boldsymbol{x}\,,\boldsymbol{\theta}\,')p(\boldsymbol{\theta}\,')\boldsymbol{d\theta}\,'} \propto p(D\boldsymbol{y}\,|D\boldsymbol{x}\,,\theta)p(\theta) \qquad (3)$$

Where P($\boldsymbol{\theta}$ |D) is the Bayesian posterior, **D**x denotes the training features, **D**y designates the training labels, the integration $\int \boldsymbol{\theta}\, p(D\boldsymbol{y}|D\boldsymbol{x},\, \boldsymbol{\theta}\,')p(\boldsymbol{\theta}\,')d\boldsymbol{\theta}\,'$ is the evidence.

Equation (3) considers practical implications especially in applications such as medical diagnostics and self-driving cars where their predictions have big impacts. Whereas, artificial neural networks tend to lack the capability of making statistical inferences, usually comes with an "I don't know" answer for out-of-training data distributions. When acquiring new data, the Bayes theorem provides a way to update previous knowledge by incorporating new data. The Bayesian paradigm is therefore a reference to effective learning, from where the knowledge fits into a dynamic procedure. Despite the robustness of the Bayesian statistical approach, computing the evidence $\int \boldsymbol{\theta}\, p(D\boldsymbol{y}|D\boldsymbol{x},\, \boldsymbol{\theta}\,')p(\boldsymbol{\theta}\,')d\boldsymbol{\theta}\,'$ and sampling from the computed distribution is extremely challenging[185].

Having discussed the difficulties with statistical inferences, Jospin et al.[185] argue for explanatory theories that can address these problems. According to the authors, the first approach is applying an algorithm called Markov Chain Monte Carlo (MCMC) and the second possible approach is to use a variational inference. In its strictest sense, the term inference denotes the process of finding unobserved variables (latent variables) based on the given parameters (observations). The process of looking at these parameters is termed learning [186]. The variational inference is an approach to estimate posterior distributions by learning a variational distribution. In the presence of complex models, not only the evidence computation but also sampling is difficult because the sampling space is high dimensional. However, both MCMC and the variational inference bypass the evidence $\int \boldsymbol{\theta}\, p(D\boldsymbol{y}|D\boldsymbol{x},\, \boldsymbol{\theta}\,')p(\boldsymbol{\theta}\,')d\boldsymbol{\theta}\,'$ computation [185].

Gibbs sampling is a powerful MCMC sampling algorithm in the MCMC scheme, on the other hand, it can be slow (long training time -computational effort) for large datasets and therefore, its applicability is limited to small datasets. Furthermore, there is another

drawback because the datasets can be very large and do not fit in memory. To address this problem, variational autoencoders (VAE) and stochastic variational inference emerged to provide an optimal solution to scale up the inference. VAE has rooted in the Bayesian inference mechanism in the calculation of the parameters (weights) of the neural network [187]. The lack of scalability capabilities of MCMC algorithms gives rise to variational inference especially to deep learning models that require huge size datasets. The core idea behind the Bayesian variational inference (variational Bayes) is to approximate the posterior probability function say P(H |D) with an arbitrary distribution (or recognition distribution) q_ϕ(H). The random variable $\boldsymbol{\phi}$ (also called variational distribution) learns (inferred) when the difference between the probability distributions decreases. The measure of proximity is given by the Kullback-Leibler (KL) divergence. KL-divergence is a function of $\boldsymbol{\phi}$ *w*hich measures the closeness between distributions [185].

$$\boldsymbol{DKL(q\phi\,||P)} = \int_{\boldsymbol{H}} \boldsymbol{q\phi(H')log}\left(\frac{\boldsymbol{q\phi(H')}}{\boldsymbol{P(H'|D)}}\right)\boldsymbol{dH'} \qquad (4)$$

Where $\boldsymbol{\phi}$ denotes sets of parameters, D_{KL} (qϕ ||P) is KL-divergence, it can be seen as a metric of the distance between two distributions, with the exception that it is an asymmetric operator and, therefore, it is not a distance in the strict sense.

If qϕ(H) is a perfect approximation of the posterior probability P(H |D), mathematically (q_ϕ(H)= P(H |D)), then the term D_{KL} (q_ϕ ||P) is equal to zero. The fundamental insight here is that variational methods reformulate the problem of computing the posterior probability P(H |D) as an optimization problem instead of computing P(H |D). As calculating the KL divergence written in Equation (4) is not feasible, the goal is to use an alternative called evidence lower bound (ELBO) as Equation (5) [185].

$$\mathbf{E}\boldsymbol{LBO} = \int_{\boldsymbol{H}} \boldsymbol{q\phi(H')log}\left(\frac{\boldsymbol{q\phi(H')}}{\boldsymbol{P\left(H'\middle|D\right)}}\right)\boldsymbol{dH'} = \mathbf{log}\,(\boldsymbol{P(D)}) - \boldsymbol{DKL(q\phi\,||P)} \quad (5)$$

Where log(P(D)) is the marginal likelihood, which is only dependent on a prior observation

To ensure expressiveness and scalability for large data sets, variational inference algorithms are, therefore, derived to approximate the distribution of the hidden variables of different models. In the pages that follow, a theoretical base and roles of variational autoencoder (VAE) will be discussed.

Variational Autoencoders

Variational autoencoders (VAE) are based on Bayesian inference techniques. However, it is necessary here to clarify exactly what is meant by an autoencoder before diving into the technical depth of variational autoencoders. Autoencoders are essentially neural networks for reducing dimensionality and consist of encoder and decoder operations. The general idea of autoencoders is quite intuitive. First, the neural network feeds the encoder with some data and converts it into more compact representations or lower-dimensional space (latent space representation) and then followed by decoder operations that reconstruct the input data from the compressed vector. The key variational autoencoder operation resides in these three components (encoder, latent variable, and decoder) as shown in Figure 2.6. The ultimate goal of reducing dimensionality is to reduce that number of dimensions while preserving much information as possible from the data structure in the reduced representations. Neural networks are used for learning the autoencoders from unlabeled data in an unsupervised manner. In such a way, the decoded output is compared with the initial data and backpropagates the error in the autoencoder architecture to update the weights of the networks[188].

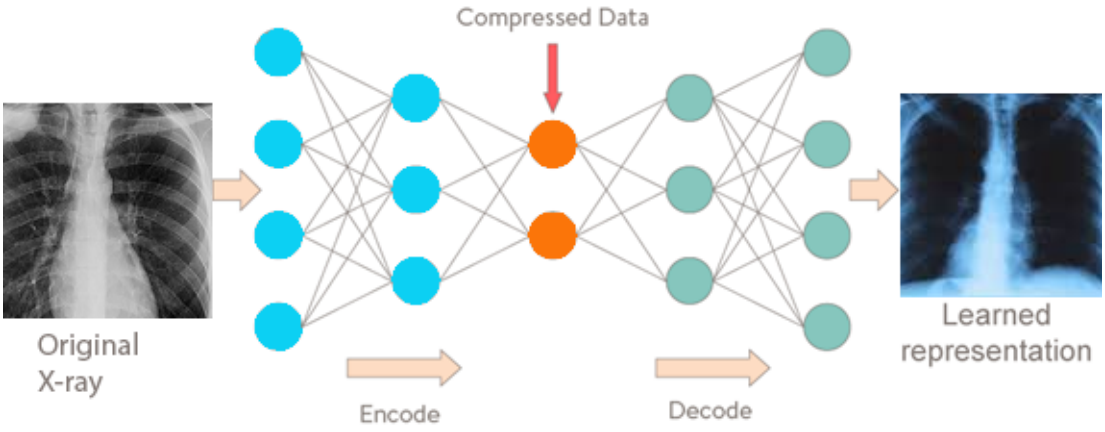

Figure 2.6: Autoencoder Architecture

Unsupervised learning appears to solve real-world problems when sufficient training data is rare to find. The paradigm of extracting meaningful representations from unlabeled data uncovers crucial value and makes it suitable in the era of Big Data because acquiring sufficient training is often expensive[189]. This remains an interesting research area as data continue to expand and AI-oriented technologies evolve. A published article in fall 2019 by Zadeh et al.[190] have revealed an important role played by variational autoencoders in machine learning researches, more specifically, in the realm of incomplete

real-world scenarios where messy, missing data values is widespread. To illustrate this point, Zadeh et al. [190] conduct a rigorous study on fundamental notions of variational autoencoders for dealing with real-world datasets. MCMC-based approaches for calculating the evidence (integral term) and sampling require exponential time during training. Moreover, there is another drawback for large datasets because they may not fit in memory (computationally expensive). These reasons make the variational autoencoder preferable[190].

VAEs are generally generative models. Since neural networks are based on mathematical principles, this discussion is from a mathematical point of view because VAE is clearly probabilistic. It learns a probability distribution of the latent space and, therefore, can be used for generative tasks. Particularly, the encoder generates latent variables z in the form of distribution (usually Gaussian) from the input data x using Bayesian inferences. The goal of training is to minimize the gap between the encoded and decoded data. Therefore, the encoder can be expressed as $q_\phi(z|x)$ or posterior function to approximate the intractable posterior $p_\theta(z|x)$, where $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ are parameters(weights) of the encoder and decoder respectively[188, 191].

Given a prior distribution $p_\theta(z)$, the loss function of the variational autoencoder can be written as follows[191].

$$\mathcal{L}(\theta, \phi; \mathbf{x}, \mathbf{z}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z})\right) \quad (6)$$

Where $\mathcal{L}(\theta; \phi; \mathbf{x}; \mathbf{z})$ is the loss function of the VAE, D_{KL} represents the KL-divergence (non-negative), and the other term with expectation operator is called reconstruction loss. Here, $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ are learnable parameters by propagation.

Minimize KL-divergence is equivalent to maximizing ELBO. Once ELBO is maximized, the distribution will come closer, which means the KL-divergence is minimized. Maximizing the ELBO is also maximizing the loss function. Furthermore, to make the network trainable end to end, a technique called reparameterization is used[188, 191]. The Reparameterization technique is to optimize the objective in Equation (6) and some assumptions are made to make it tractable (computable). Provided posterior $q_\phi(z|x)$ and prior p(z) are parametrized as Gaussians particularly with Gaussian $N(0; 1)$. This allows the reparameterization technique to approximate gradients of the ELBO concerning the parameter $\boldsymbol{\phi}$. Therefore, for each random variable $\mathbf{z}_i$ in the latent representation with noise variable ε, is parametrized as written in Equation (7) [191].

$$z_i = \mu_i + \sigma_i \epsilon \qquad (7)$$

Where **μ** is the mean distribution, and **σ** denotes the standard deviation of the distribution.

There is a new class of variational autoencoder known as disentangled variational autoencoder which has astonishing results. The fundamental idea behind disentanglement is to assure the different neurons of the latent distribution in the neural network to learn something different about the input data. In such a way that, it disintegrates or divides each feature into specific variables and encodes them as distinct dimensions. One neuron can learn features independent of other neurons (uncorrelated). This is developed by adjusting the VAE through one hyperparameter **β** in the loss function which weighs how much the KL divergence in the loss function is present and therefore, the autoencoder will use a specific latent variable[191, 192].

$$\mathcal{L}(\theta, \phi; \mathbf{x}, \mathbf{z}, \beta) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \beta\, D_{KL}\big(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z})\big) \qquad (8)$$

Where **β** is a disentangled representation

A higher value of β (usually when **β**>1), leads to more distinct latent representations by penalizing the distance between the posterior $q_\phi(z|x)$ and prior p(z). It is now, therefore, a trade-off between the degree of disentanglement and the reconstruction quality. Larger **β** encourages stronger disentanglement. However, when beta(**β=1),** it would be equivalent to the original variational autoencoder[192].

For an unknown number of features, an elegant non-parametric Bayesian approach called the Indian buffet process (IBP) is proposed that enables model size to grow with a dataset. One drawback of many models is that latent representation has to have a fixed dimension, irrespective of the size and complexity of the data[193]. IBP also called the Beta-Bernoulli process provides a flexible approach to approximate latent features automatically from the training data in an unsupervised way[187]. Unsupervised learning aims to identify latent variables required to produce the observed properties of objects. IBP offers a mechanism for choosing appropriate overlapping features[194]. It assumes that there are infinitely numerous features that are essential for representing the input data and the ultimate goal is to adapt the complexity of the data while producing compact representations[54].

Briefly, IBP defines a distribution over binary matrix $\mathbf{Z} \in \{0, 1\}^{N \times K+}$ indicating which features are possessed by what objects. This is for N objects and K features. Then, the binary variable $Z_{n,k}$ equals 1 if the object has feature K, and 0 otherwise[187].

$$\nu \sim Beta(\alpha, 1); \pi_k = \prod_{i=1}^{k} \nu_i$$
$$z_{n,k}|\pi_k \sim Bernoulli(\pi_k) \quad (9)$$

Where π_k is the relative probability of each feature, $\boldsymbol{\alpha}$ designates the expected number of features, Beta distribution (as a conjugate prior to Bernoulli distribution) is utilized.

Provided a prior over latent feature allocations Z each ranging from {1…N), a generative model can be specified for data X, $\{x_n\}^N_{n=1}$, where $\mathbf{X}i \in \mathbb{R}^D$ with a likelihood p(**X**| **Z**) as shown in Equation (10) [187, 193].

$$\mathbf{Z}, \boldsymbol{\nu} \sim \text{IBP}(\alpha); \quad \mathbf{A}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{K^+}); \quad \mathbf{x}_n \sim p_\theta(\mathbf{x}_n|\mathbf{Z}_n \odot \mathbf{A}_n) \quad (10)$$

Where **A** represents a matrix of infinite rows and D columns, $\odot$ is an elementwise product for $n \in \{1…N\}$, **I** represents an identity covariance matrix.

2.8 Summary

This chapter provides a review of current literature on X-ray image analysis using neural-network and non-neural-network approaches. It specifically briefly covered computer-aided detection approaches, current state-of-the-art (SOTA) ML and DL-based approaches to determine cardiorespiratory disease features that are useful for X-ray image transfer learning via statistical analysis of backpropagation gradients during network fine-tuning.

Chapter 3: Related Work

With the rapidly expanding field of CV[136, 138, 195], major advances in Convolutional Neural Networks (CNNs)[100, 196] and what has come to be known as transfer learning (TL) [197]–[199] have allowed stunning transformations of deep learning (DL) algorithms and applications built on radiographic images. Within the medical field, radiology represents one of the pioneer areas of healthcare to incorporate AI technologies for visual image analysis and interpretations [200]–[202]. This is especially useful because manual analysis and interpretations of images are tedious and error-prone. Therefore, to assess the potential of AI algorithms in supporting the process of diagnosis and treatment, various medical imaging techniques such as MRI, CT scan, ultrasound, and radiography are used to generate images[203]. Despite the great advance in its use in recent years, there are still long-standing problems related to image segmentation, clinically significant radiology reports extraction, and automated detection of abnormalities[204]. Spotting dangerous items (abnormality detection) in radiography images has particularly drawn our attention as a research area. It formed a central focus of our study and there is now a substantial wealth of literature on cardiorespiratory disease detection and repeatedly faced unique complexities, thereby reveal several contrasting themes.

Cardiorespiratory detection methods predominantly fall under two major categories[194, 195, 196]: (1) Classical machine learning (ML) approaches and (2) Deep learning (DL) based approaches. Having defined the distinctive categories of detection methods, this chapter, therefore, move on to present the long-standing issues of earlier studies, briefly touch upon existing approaches to detect cardiorespiratory diseases in X-ray images.

3.1 Classical ML Approaches

Chan et al.[208] proposed an automatic detection of pneumothorax from X-ray images using a support vector machine (SVM). SVM was applied to identify radiological patterns after discriminating features are extracted with the LBP (short for local binary pattern). At its core, texture analysis was carried out to find the abnormal regions and boundaries of the rib were also identified via the Sobel operator (edge detector). This approach effectively analyzes cardiorespiratory problems in the chest X-ray image and facilitates the prospective diagnosis of pneumothorax. A total of 42 radiographic images was used for training with only a single class label, however, this indicates a high degree of limited sample size. The real-world clinical scenarios remain poorly represented.

3.2 DL Approaches

Wang et al. [43] originally introduced the NIH chest X-ray dataset and develop a DL algorithm for detecting multiple cardiorespiratory diseases. It was ground-breaking and impressive academic research into ChestX-ray14 images. The X-ray images were directly extracted from the PACS system (short for a picture archiving and communication system) with an image size of 1024x1024 and utilize NLP techniques to exploit radiological reports. The entire ChestX-ray14 dataset was randomly shuffled into training, validation, and testing for the disease classification task of 30,805 distinct patients. Four different CNN architectures namely VGG, ResNet-50, AlexNet, and GoogleNet was used for multi-label pathology classification. The quantitative performance among the architectures vary significantly and ResNet-50 was found to outperform all pre-trained models. Overall, the algorithm was shown to achieve an AUC (area under the ROC Curve) value of 0.7451. The study however was limited in its application to critically assess the effect of the unbalanced dataset and various hyperparameter values for supervised learning. The NIH dataset inherently possesses unequal distribution patterns between groups, where certain classes are rare to find while others have become so dominant. It may, therefore, have been a more reasonable approach to tackle the skewed distributions. Most importantly, the CNN model capacity, a fundamental classification performance factor should have been closely investigated.

Yao et al.[44] proposed a DL model based upon two strands of research, a combined recurrent neural network called LSTM networks (short for the Long Short Term Memory) and CNN approaches to capture statistical label dependencies, thus obtain a more accurate recognition rate. As the backbone of the CNN model, a vigorous variant of DenseNet[209] architecture, known as DenseNet10 was adopted for disease classification. The finding has prompted the plausible speculation that considerable progress was made over Wang et al. [43], outperforming on 12 of 14 diseases and achieve a mean score of 0. 761 AUC value. Despite its promising improvement towards developing automated X-ray models, the research lacks to consider the class imbalance problem, and similarly, the methodological limitation reasoning goes to Wang et al. [43] also applies to this study. It is important not to overlook the potential impact of hyperparameter optimization and knowledge adaptation approaches to DL algorithms because failing to perform an adequate investigation of these fruitful areas can bring serious drawbacks, declining model performance.

Guendel et al.[48] propose a DL-based method to classify cardiorespiratory diseases on a total of 86,876 patients. Supplementary X-ray images were further collected from the PLCO dataset25 in addition to the chest X-ray14 dataset and obtain a mean AUC score of 0.807. For the purpose of analysis, the authors incorporate spatial (location) information into X-ray images using a location-aware network called DNetLoc before the DenseNet network. For leveraging spatial information, two convolution layers of 3 × 3 filers and a stride of 2 was utilized. During training, the X-ray image resolution was downscaled to 128 × 128 sizes. A greater focus on fine-tuning the DenseNet network, however, could establish a natural progression of this work on this matter, by determining the underlying cause of misclassification biases.

Baltruschat et al.[210] investigate the ResNet-50 architecture and conduct a certain knowledge adaptation (transfer learning) with a fine-tuning approach on X-ray images and significant improvement has made over a model trained from a scratch. Experimentation was carried out based on data from the NIH ChestX-ray14 dataset for predictive health analysis and increased the training data by data augmentation techniques. The overall classifier performance was compared to earlier studies of state-of-the-art results and found to reach a mean score of 0.806 AUC value. Difficulties arise, however, when an attempt is made to implement the model due to a 10% label noise in the dataset, which adversely affects the evaluation of the network performance. The study raises an important issue that warrants further investigation and it was recommended to have a noise-free test set for clinical impact assessment. One of the main strengths of this line of reasoning is that no scientific investigations were undertaken for evaluating whether the approach generalized to a different healthcare context (new observations) to the best of our knowledge. What remains less clear is the nature of generalizability and its causal factors, still raising major theoretical issues in the field. This indicates a need to assess models to real-world clinical scenarios.

Sirazitdinov et al.[49] develop a model based upon the CNN Inception architecture[211], particularly an inception variant called Inception-Resnet-v2 network. The inception model was trained with a learning rate of 0.01 and momentum of 0.9 using an SGD optimizer for only the last layer of the network. A comparison of the findings with those of other researchers confirms, the authors achieved a higher AUC value (0.808). Data augmentation methods were incorporated to address the overfitting problem; however, classes remain unbalanced.

Wang and Xia [45] proposed an attention-based and classification DL model for fourteen cardiorespiratory diseases and the model was named ChestNet. At its core, ChestNet is comprised of a ResNet-152 CNN architecture (pre-trained model) that plays a central role in the classification of disease. The attention-branch was introduced to find correlations among disease labels and localize abnormal regions, thus indicates the location of diseases in an image. The ChestNet model was fine-tuned to enhance the performance of the model, and compared to previous state-of-art results, it demonstrated greater capability for chest X-ray image recognition. It detected the fourteen diseases with an average per-class AUC of 0.781, providing robust evidence of their new architecture outperforming the ResNet model. Despite that, the ChestNet approaches suffer from serious weaknesses. One of the limitations of this explanation is that it does not consider conducting a hyperparameter search to determine learning rates and hidden layer dimensionality that are relevant for the detection task. Similarly, the existing class imbalance problem in the ChestX-ray14 dataset remains unanswered.

Ma et al. [50] adopted various prominent CNN architectures and tackle the class imbalance issue using a data augmentation technique for the X-ray images. This is certainly useful in the case of unbalanced scenarios. The effectiveness of the data augmentation technique has been exemplified and demonstrated in their experimental analysis; however, it was carried out in an online manner. An online image augmentation method[212] was employed to address the overfitting issue and significantly expand the quantity of training data to maximize the classifier model performance. This does not mean, however, that the online mode adversely affected the training time and computing resources. Image augmentation may rather have been performed in an offline mode [213] during the pre-processing stages instead of the default operations in the training stage.

Kumar et al. [47] implemented ensemble approaches (boosted cascade CNN network) for multiple cardiorespiratory disease detection. It was focused to combine several DL models and draw together principal features of state-of-art networks to build a powerful ensemble. Choosing compatible loss functions has also formed an important part of their work for training the CNN network on the NIH ChestX-ray14 dataset. Their methods yielded higher performance compared to a single classifier and appear to reach an average per-class AUC of 0.795 scores.

3.3 Summary

In this chapter, we have discussed various CNN architectures and approaches which are relevant to the current work. Different methods have been proposed to classify multiple cardiorespiratory diseases from X-ray images and each approach has numerous advantages and limitations. While the performance of existing approaches has been improved substantially, there is still abundant room for further progress on the current topic. Wang and Xia[45] proposed a deep architecture called ChestNet from the ChestX-ray14 dataset and outperform previous state-of-art results. ChestNet was shown to yield a mean AUC score of 0.7810, comparatively higher than earlier findings presented in [43, 44]. However, ChestNet model fine-tuning involved retraining only the last fully connected layer of a pre-trained DenseNet[209]. The authors of ChestNet also note that no additional regularization was imposed on this layer, indicating that the fine-tuned network generalizability could still be improved. We believe that additional analysis is warranted of the prospect of a breakthrough on several critical issues. To the best of our knowledge, learning the structure of the fully-connected layer (or indeed other layers of the pre-trained network) for transfer learning has not yet been attempted. To support this line of reasoning, we will develop a novel IBP based nonparametric Bayesian statistical model to modulate the number of features in neural networks. The objective of this investigation is twofold. One is to assess the effect of learning the dimensionality of the fully connected layers for prediction because different cardiorespiratory diseases might require different numbers of features for detection and therefore, could improve the generalization during evaluation. Second, conduct a more thorough analysis of pre-trained networks for distinguishing features, fitting the training data by statistical analysis of backpropagation gradients during fine-tuning, that could then improve the standard classifier model performance.

Current detectors[43, 44, 45, 47, 48, 210] failed to consider the underlying class imbalance problem in the ChestX-ray14 dataset. In light of this, we will closely examine the effect of an unbalanced set and develop an algorithm to overcome these imbalances. To date, no or little attention has been paid to hyperparameter choices, therefore, we will conduct hyperparameter searches for the detection task. Last but not least, the authors of ChestNet[45] note that a limitation of their work is its application in X-ray images from only one clinical setting and location, which is especially evident because of the label noise [43, 210]. To this effect, we will demonstrate our approach from another clinical setting.

Chapter 4: Model Design for Cardiorespiratory Disease Detection

This chapter describes the specific methods to design and develop a model for multiple cardiorespiratory disease detection. It draws together the various strands of the thesis and addresses each of the research problems. This chapter, therefore, moves on to discuss each component of the model including data acquisition and preprocessing, the CNN classifier, and algorithms used to implement the model.

4.1 The Cardiorespiratory Diseases Detection Model

The cardiorespiratory disease detection model consists of two main components, namely, a deep transfer learning (DTL) component and the detection component. The DTL component consists of preprocessing, pre-trained models (adapting CNN architectures), and unsupervised visual extraction through a variational autoencoder (VAE) coupled with a non-parametric stochastic process called the Indian buffet process. Whereas the detection component is comprised of splitting the dataset into three parts (training, validation, and test sets), validation on two different datasets, image preprocessing and data augmentation for the training data, CNN classifier model, and the predictive model. It also presents fine distinctions among different hyperparameter values for the CNN classifier. The detection components are adapted from [214, 215]. This multitiered model is used to classify the 14 cardiorespiratory diseases from X-ray images as shown in Figure 4.1.

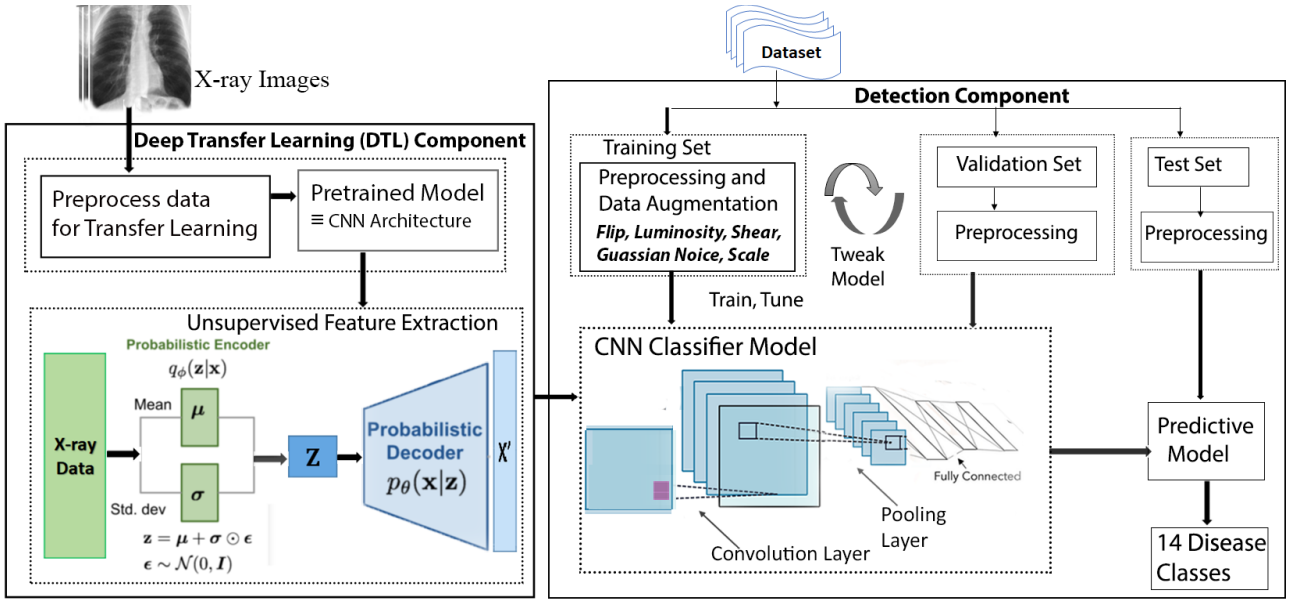

Figure 4.1: Cardiorespiratory Diseases Detection Model

4.2 Deep Transfer Learning Component

The DTL component is the principal force behind the model. It is where previously acquired knowledge is further investigated in unsupervised learning to maximize the transfer of knowledge and generality. Knowledge adaptation is a long-standing problem and an active research area. CNN models require substantial duration for training examples, especially from scratch. This case, therefore, reveals the need for further investigation because fine-tuning a complete model requires more training examples, otherwise, leads to an overfitting problem. Furthermore, high-quality annotated medical images are rare to find in the real world. We employed DTL for X-ray image classification. Our foremost consideration is to find optimal relationships between layers of these deep neural networks and the data in an unsupervised fashion. Instead of choosing the dimensionality of feature space arbitrarily, we rely on data-driven approaches. Therefore, the soundness of our approach goes beyond the ability to replicate the known knowledge but rather to discover previously unknown knowledge by applying concepts from statistical probabilistic programming and data mining field.

4.2.1 Data Preprocessing

This is the first stage of the DTL component to facilitate the next steps. The goal is to enhance the data quality of the X-ray images which are driven from two datasets. Our main source of data is drawn from the National Institute of Health (NIH) which contains 112,120 radiographic images of size 46GB data with 1024 x 1024 resolution [43]. There are four important reasons to choose the NIH dataset. (1) The NIH dataset is purposely developed for the diagnosis of diseases and becomes an active area of research including in Kaggle platform detection challenges (2) The dataset comprises 30,000 unique patients and multiple findings that represent realistic real-world clinical settings and challenges. (3) The NIH dataset provides us a framework for developing a single model that can detect multiple complications. (4) The dataset also takes great relevance for creating a robust deep learning detection architecture.

We also collected data from Tikur Anbesa Specialized Hospital to validate how well these deep learning approaches will generalize to another geographical region or clinical population. To begin the preprocessing pipeline, we extract the NIH dataset for unsupervised feature extraction and improved relevant image characteristics such as normalization and altering resolution techniques. A series of further image preprocessing

were performed because the preprocessed unlabeled radiographic images will be fed into one of the popular CNN architectures, namely, AlexNet, DenseNet, and EfficientNet, and then applied a novel approach to disentangled features using variational autoencoder and a Beta-Bernoulli distribution (also called Indian buffet process). The expected input dimension of these architectures is 224 x 224 x 3 images (RGB format). Therefore, each x-ray image is converted into an image format for fine-tuning the pre-trained models. Moreover, images are not only resized but also transformed into a desirable format (or put the data into tensors) and the image data are then loaded into a suitable structure. For this reason, images should be preprocessed so that pre-trained models can accept them. The mode of preprocessing is similar to what other researchers do because deep learning models learn from data. Since the DTL component deals with unsupervised learning, the data preprocessing is made only for images. However, there is a separate section of preprocessing (both images and labels) for supervised learning in the detection component.

4.2.2 Pretrained Models

Pretrained computer vision models play a paramount role to transfer knowledge between domains. It overcomes the demand for massive volumes of labeled images for training. There are two different scenarios to use pre-trained models, the choice of which usually depends on the size of the dataset and the availability of computational resources. The first approach is fine turning where the final classifier layer of a network is replaced to fit the current dataset classification and maintain the parameters learned from other layers. Another possible alternative is to freeze some first layers of the network. Freezing has taken a different approach by focusing on the last layers and therefore, requires relatively lower computational power. Despite its common usage, the parameters of the pre-trained networks are not modified in these first layers.

In our case, the pre-trained model forms a basis for the unsupervised feature extraction. The pre-trained model is then trained in an end-to-end manner to maximize the performance of the model with respect to the state of art. Therefore, before, the training stage, CNN architectures with millions of parameters are fed into our variational autoencoder model for compact and robust representation. The approach can accept different types of CNN architectures such as GoogleNet, AlexNet, ResNet, VGGNet, and many more. For our investigation, we started from the weights of the AlexNet network, which paved the way for revolutionary research in the CNN architectures as it is now. We, therefore gradually begin to evaluate our approach to the recent state-of-art

architectures such as EfficientNet and conduct fine-tuning on these models. That is, all layers of the network are further refined to match the model capacity with our dataset.

4.2.3 Unsupervised Feature Extraction

This feature extraction component formed a central role in regularizing network complexities and provide a framework for data-driven transfer learning of useful features for cardiorespiratory disease detection. Our proposed model applies a nonparametric structure learning Bayesian factor model for searching optimal architecture that grows automatically with the size of the dataset. In this work, we studied to determine the effect of applying DTL methods to the image classification based on an unsupervised variational autoencoder neural network algorithm. This is essential because a set of unlabeled training examples is relatively easy to find and, in particular, when it is difficult to access a large number of labels. We introduced an elegant Indian buffet process approach (**β**-VAE) to the popular CNN architectures more specifically EfficientNet to moderate the network complexities. We design a VAE that forces a compressed knowledge representation of the original X-ray image input as shown in Figure 4.2.

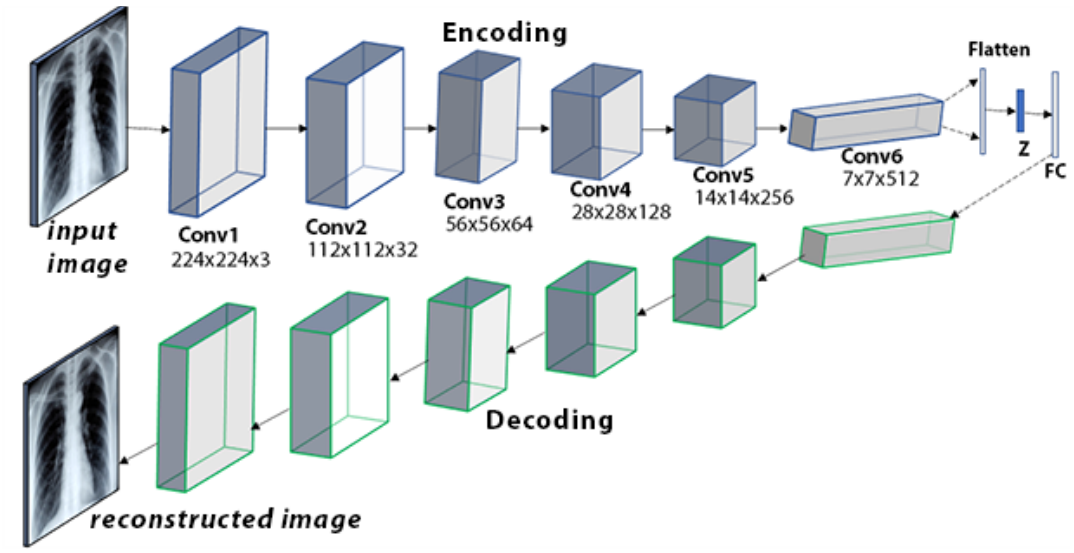

Figure 4.2: Unsupervised Feature Extractor

This type of neural network consists of two parts, which are encoder and decoder. An encoder is part of the network to compress the input into a latent variable (**z**). Our first step in this process was defining a series of convolutions with an activation function ReLU and pooling layer of MaxPooling. Each convolution layer of the encoder network is comprised of a kernel (filter) size of 3, padding 1, and a stride of 1.

We design the network, to begin with, 224 x 224 RGB format, and then continued to be compressed until we obtain 512 output channels of 3 x 3 resolution. Following the convolutional layers, we added two fully connected layers mapping ultimately into 14 disease categories. Once the X-ray input images are encoded as a distribution in the latent space, an essential next step is sampling from the conditional probability distribution with mean ($\boldsymbol{\mu}$), and standard deviation (covariance $\boldsymbol{\sigma}$) of the distribution. The sampling process should be expressed in a way that allows the error to be propagated by the network. The sampling operation in distributions makes the backpropagation algorithm possible or allowing the training of the neural networks. In doing so, it is, therefore, necessary to carry out the backpropagation process to minimize the reconstruction errors between the original X-ray image and the reconstructed output. The challenge now is to reconstruct the original X-ray images with a minimum error and retain the most important patterns. Therefore, it is quite difficult to guarantee, the latent space represents the distribution of the data in the initial space while minimizing the complexity of CNN architectures which contains millions of parameters.

We followed a reasonable approach to tackle this problem by having a powerful encoder-decoder architecture definition. There are certain drawbacks associated with the use of complex networks because it raises the prospect of the overfitting problem. Similarly, there are obvious difficulties in choosing a latent space dimension. There is, therefore, a definite need to design the network with caution. Returning to the modeling dynamics, our proposed model is comprised of six convolutional layers. The first convolutional layer is comprised of nn.Conv2d(3, 32, kernel_size = (3,3)) where the first three represents the number of input channels (a color RGB format) and 32 is the feature map (the number of channels). Each convolution is followed by Batch normalization, ReLU, and downsampling (a max-pooling operation to compress the input image). The pattern is repeated until a small image size of three by three is produced. Finally, the output of the convolution layer is flattened before the transition into fully connected layers. In the latent layer, Kulback-Leibler divergence is applied to approximate the distributions in the encoding-decoding scheme. The decoder also utilizes Beta-Bernoulli distributions that can be expressed as means and covariance matrices of two distributions to reconstruct the images.

The standard implementation of this is illustrated by Tomczak and Welling[216]. Difficulties arise, however, when an attempt is made to implement the beta (β) distribution. To address this problem, the Kumaraswamy distribution[187] is then introduced for approximating beta distributions (beta random variables). This way images are reconstructed from a Bayesian perspective and features are extracted that would be fed into a supervised image classifier model in the detection component.

4.3 The Detection Component

The detection component generally consists of the following component: (1) Training, validation, and test datasets. (2) Preprocessing for each dataset. (3) Data augmentation (4) The CNN classifier model (5) finally, the predictive model (classifier). It appears from the aforementioned investigations that most attention has been paid to these components of the detection component. To the best of our knowledge, little or no references in the literature systematically describe the effect of deep transfer learning (DTL) methods on the detection component. Realizing the gap in the existing literature, we critically examine the effects of not only DTL approaches but also each detection component to further enhance the performance of the image classification for cardiorespiratory disease detection. In the detection component, we also conducted hyperparameter and tuning techniques to provide a more in-depth investigation regarding the effects of neural network parameters. The principal objective of this thesis is, therefore, addressed in this section.

4.3.1 Training, Validation, and Test Sets

The first two steps in the process of building neural network applications are the collection of data related to the problem and its separation into a training set and a set of tests. In addition to this division, a subdivision of the training set can also be used, creating a validation set, used to check the efficiency of the network in terms of its generalization capacity (to new situations) during training. Therefore, we randomly split the data into three parts and a further step towards the validation set is the use of more radiographic images from another clinical population more specifically Tikur Anbesa Specialized Hospital. To provide a solid evidence base, our approach to split the dataset is similar to that used by other researchers including the creators of the NIH dataset [43]. There are 25,227 X-ray images for validation and 75,682 images for training and 11,213 images for testing. Section 4.2.1 justifies why we choose the NIH dataset.

4.3.2 Preprocessing and Data Augmentation

Preprocessing:- The principal objective of data preprocessing is to deal with unwanted effects or variability in the data because there could be several artifacts such as missing data, noise (random errors and outliers), and inconsistencies [217]. Careful removal of data affecting the performance of DTL models should therefore be undertaken. In line with that, we conducted different preprocessing techniques for our datasets. Unlike the DTL component, preprocessing in the detection component focuses on the process of modifying target labels and image manipulations in supervised learning tasks. Therefore, labels are adjusted to any compatible format for achieving the end result. As part of the preprocessing, we remove irrelevant columns and remove attributes with null records, convert categorical data (disease categories) into numerical values or binary vector arrays.

In the case of the Tikur Anbesa Specialized Hospital dataset, we purposely match the data preparation with the NIH dataset so that it can be used in our proposed model, as shown in Algorithm 4.1. We have 14 separate categories and the goal is to transform the list of these text categories into a vector. This is, therefore, to ensure that, if an item belongs to that category, it will be assigned to 1, otherwise 0. Previous researches have indicated approaches for converting string categorical values into real number vectors. From the journal of Big data, in spring 2020, Hancock and Khoshgoftaar [218] described a method called one-hot encoding for mapping categorical values to vectors of real numbers as desirable inputs for neural networks. Since, our dataset is originally prepared with target columns that have multiple disease categories such as atelectasis, cardiomegaly, effusion, nodule, pneumonia, and others. There are obvious difficulties for later comparison of the convolutional neural network results because CNN models give out as vectors of 0's and 1's. For example, if a feature can take on three different values, it will be represented as [1 0 0], [0 1 0], and [0 0 1]. There is, therefore, a definite need for representing these categorical values in terms of real number vectors as shown in Algorithm 4.1 for values drawn from the 14 sets of diseases. For the images, we apply normalization (scale the data values over a specified range), resize the images into 224 x 224 RGB formats, and loading the NIH dataset into an object to easily access the underlying data. Data normalization is among the pre-processing techniques where the data is transformed to make an equal proportion of each feature for improving data quality [219]. We, therefore, transform the data with a statistical measure of standard deviation and mean of 1 and 0. 5 respectively.

Algorithm 4.1: Encoding Validation Dataset

Input: Annotated *Image Set*

Output: Vector of 0's and 1's value (occurrence)

```
Initialize csv_columns ← ['FileName', all the 14 diseases],
Initialize disease_categories ← {disease:key pair dict},
Initialize lsdata ← image lists, Initialize output ← {}
Initialize wf ← Open image labels in w+ mode
SET writer ← {csv.writer(wf)} // write lists into CSV file
for each m in lsdata do:
    writer[m].writerow(csv_columns) // write rows into file
end for
Initialize lion_lines ← Open CSV file// images labels
for each i in range(len(lion_lines)) do:
   value ← lion_lines[i].split(',')
   file_name ← split[0]
   SET label_str ← split[1]
   SET labels ← label_str.split('|')
   vector = [0 for _ in range(14)] // repeat for 14 disease
   for each label in labels do:
            if label IS NOT EQUAL "No Finding":
                vector[disease_categories[label]] ← 1
            end if
   end for
   output.append(file_name)
   output ← output + vector
   writer[m].writerow(output) // write output into file
end for
Return vector
```

Data Augmentation: - the general principle of data augmentation is to artificially expand training data by applying operations that reflect the real-world variation. The goal is to get more training data from the existing dataset because getting training data can be expensive and is not always possible in practice[220]. Therefore, we used data augmentation techniques to create more X-ray images from the training data. These techniques include rotating X-ray images by small-angle say by 20 degrees, flipping images horizontally and vertically, adding Gaussian blur (noise), crop images say 10 percent of their height and width, scale images to a certain size, adding luminosity, and other transformations. There is, therefore, an expanded training example that fed into the training algorithm to improve the performance of our network. Before the training phase, the model is fed with already augmented training examples. It is necessary here to clarify exactly why we followed prior offline augmentation instead of using the default deep learning libraries. Our approach carries profound significance to solve two critical problems. These are (1) addressing the imbalanced class problem (2) it reduces the time of training. Class imbalance[128] refers to a dataset with a skewed majority-to-minority ratio. In our case, the NIH dataset has highly unbalanced training sets ranging from 110 images of hernia to 9547(infiltration X-ray images). Therefore, the CNN model will favor the class with the highest probability of occurrence (the majority), resulting in a low detection rate for the minority groups. Because of this, we followed an optimal reasonable approach to tackle the imbalanced class problem by selectively augmenting the disease categories based on the specific characteristics of each disease. We conduct less data augmentation to the dominant groups such as infiltration (containing 9547 radiographic images) and atelectasis (with 4215 images). In contrast to the dominant groups, we carried out heavy data augmentation to classes (disease categories) that have the lowest probability of occurrence.

Another problem with the default deep learning augmentation approach is that it fails to take the training time into account because it does the image transformations during the training phase. This remains a challenging problem, especially with computing resource constraints. We, therefore, provide offline data augmentation to mitigate the imbalanced class problem and the training time.

Algorithm 4.2: Data Augmentation

```
Input: Training data size of n
Output: Augmented data of size m
DEFINE FUNCTION appendList (Filename, Number of elements)
    Open CSV file in append mode
    Create a writer object from the CSV module
    Add contents of the list to the last row in the CSV file
Open CSV file as csv_reader // values with comma delimiter
SET line_count TO 0 // initialize to zero
for each row in csv_reader do:
   IF row[0] EQUALS "FileName":
         Continue
   for each disease_occurrence in CSV file do:
        Open images of the specific Class
        Augment the images n times // rotate, flip, scale, etc.
        SET images_aug ← TO augmented images
       for I, aug in enumerate(images_aug) do:
            SET image_name ← TO row[0].split(".")[0]
            SET Image_Index ← TO image_name+"_"+str(I)+".png"
            Write images to the specific Folder
            SET each disease to corresponding rows
            INVOKE FUNCTION appendList // Add row contents
            SET line_count ← TO line_count + 1
            return line_count
        end for
   end for
end for
```

The data augmentation algorithm shown in Algorithm 4.2 tackles the class imbalance issue.

4.3.3 CNN Classifier Model

Our CNN classifier model is quite an important part of the detection component. What is appealing about this model is that it encompasses robust features from supervised and unsupervised learning techniques. The classifier model uses supervised learning to learn characteristics of the standards that define each class, based on the examples and their already known answers to infer future responses. The model also manages to take advantage of features extracted from pre-trained networks (previously trained parameters) in an unsupervised manner. We fine-tune the weights of models that have already been trained on large datasets (also called transfer learning) to adapt it to our problem. We, therefore, take the necessary time and effort to design the model because small variations can directly influence the time needed for training the model and the final classification performance. There have been ongoing bleeding-edge research efforts on the choice of CNN architectures. Designing CNN architecture involves choosing the number of layers and the number of units per layer. Our model design approach also involves a lot of empirical searches. As part of designing the model, we begin from small and then gradually increase to these models that have greater representational power. We have been increasing the model capacity (size of convolutional layers) until the validation error starts improving (generalization capability decreases). The proposed CNN classifier model is shown in Figure 4.3.

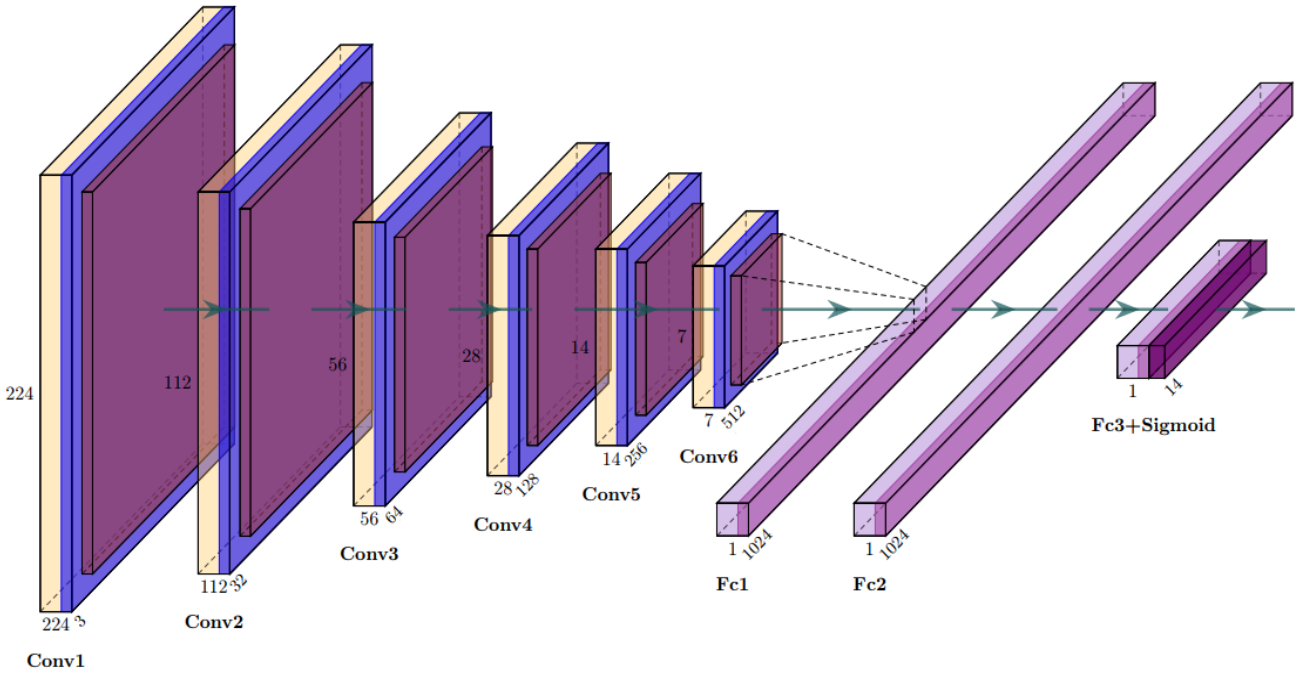

Figure 4.3: CNN Classifier Model

If we could focus for a moment in Figure 4.3, we can see that the model design starts with the input image of size 224 x 224 x 3, where 3 represents the number of input channels. It then reshaped into 112 x 112 through pooling layers after 3 x 3 filters (also called kernels) are applied. Furthermore, to each 6 convolution layers, multiple operations will be applied such as the pooling layer (a six max-pooling of a 2 x 2 window of stride 2 is used), ReLU operations, and Batch normalization for normalizing output features. Batch normalization (BN) is a method used to normalize activations in hidden layers (intermediate layers) of deep learning models [221]. Every convolutional layer also has padding and stride equal to 1. The final layer of connections on the network is called a fully connected layer. That is, this layer connects all neurons in the max-pooling layer to each of the 1024 output neurons. More precisely, all neurons in the network are connected to all neurons in adjacent layers. In total, we have three fully connected layers. Each of the first two layers has 1024 channels, and the third fully connected layer has 14 channels representing the class of 14 X-ray image classification (disease categories).

4.3.4 Predictive Model

In the learning stage, we carried out various hyperparameter searches to assess the effectiveness of our proposed CNN model. Therefore, based on variations in the values for the hyperparameters, DL models are evaluated upon how well they perform on the detection task or domain. What we know about hyperparameter is largely based on our empirical searches to know which values contributed most to the model's performance. This is performed before choosing the final model that can be used to predict the occurrence of the 14 cardiorespiratory diseases in radiographic images. To determine the occurrence of the specific 14 cardiorespiratory diseases within an X-ray image, the DTL algorithm capture characteristics (features) in the data using a series of convolution layers, and then the CNN model uses these image features to provide distinct class probabilities. Then, the sigmoid activation function maps the resulting predictive class probability of the image value within 0 and 1. This activation function has values within the range 0 and 1[212] and then passed to the loss function to compare against the actual labeled data. Finally, a model with a minimum loss and a higher AUC score is chosen for the prediction.

The predictive model component quantifies the outcomes of X-ray image classifications of new images based on the feature vectors extracted from the CNN classifier (DL) model. The predictive model is the learned model weight of the classifier. It is the best performing model from the training stage and is evaluated for its capability to generalize to new

observations (test dataset). The most common evaluation metric used in DL is prediction accuracy, in which the prediction of the DL models is compared with the actual observed value. Therefore, the selected model is measured via Area Under the ROC curve (AUC). AUC is the most widely used measure to evaluate the ability of probabilistic predictive models to discriminate among individuals who developed the disease or not [222]. The predictive model based on the DTL algorithm can, therefore, ultimately applied in the health field with the profound potential to assist health professionals in making decisions.

4.4 Summary

In this chapter, we explained the detection and deep transfer learning (DTL) components used for cardiorespiratory disease detection. The DTL component is employed for knowledge adaptation of CNN architectures (EfficientNet and DenseNet) to the detection component. The unsupervised feature extraction component plays an important role in regularizing the CNN architectures by changing the input data into a lower-dimensional space as discussed in Section 2.7. The detection component is also comprised of multiple components such as preprocessing stage for the class labeled data and a CNN classifier model which gives a learned model weight used to test the performance of the model for unseen image examples.

Chapter 5: Implementation and Evaluation

In this chapter, we describe data acquisition procedures, tools, and experimental settings and approaches utilized to implement the proposed model. It also presents analytical discussions and empirical results obtained from the experiments coming from the model training on the radiographic X-ray images. We also compare our model performance with that found by other researchers. Therefore, this chapter is divided into seven main sections, each of which contributes to addressing the research gaps and ties together the various theoretical and empirical strands to detect multiple cardiorespiratory diseases.

5.1 Dataset Preparation

Since data is the essence that allows DL algorithms to learn, we run the experiments on two image datasets: NIH clinical center, publicly available dataset[43], and the Tikur Anbesa Specialized Hospital dataset. We normalized the X-ray images to bring their pixel values within a range of 0.5057 mean and standard deviation of 1. These values are driven from the X-ray inputs as pre-processing to make equal distributions around the images and used for faster computation and improving data quality[223, 224].

The NIH dataset contains 112,120 1024 × 1024 RGB images and is associated with a label from 15 classes. These are atelectasis, cardiomegaly, effusion, infiltration, mass, nodule, pneumonia, pneumothorax, consolidation, edema, emphysema, fibrosis, pleura thickening, hernia, and no finding (a disease-free category). Although the labeled data is already in tabular format, we have converted the categorical findings into real-number vectors as shown in Algorithm 4.1 using Python libraries. Within the NIH dataset, we randomly split the data into three parts: a set of 75,682 images which we use for the training set, 25,227 for the validation set, and a separate test set of 11,213 images. The validation images will be used to evaluate the model during training. For example, the use of validation images particularly appears more feasible when the neural network is configured with certain hyperparameters, such as the learning rate. In general, therefore, we provide the CNN model with images and their corresponding class labels for classification. However, there are certain drawbacks associated with the use of NIH datasets for DL models. The NIH dataset generally emerged with highly unbalanced classes. To support this line of reasoning, we considered providing quantitative shreds of evidence that would usefully supplement and extend the discussion.

Table 5.1: The number of images per class

Disease	Total images
Hernia	110
Pneumonia	322
Edema	628
Fibrosis	727
Emphysema	892
Cardiomegaly	1093
Pleural Thickening	1126
Consolidation	1310
Mass	2139
Pneumothorax	2194
Nodule	2705
Effusion	3955
Atelectasis	4215
Infiltration	9547

If we turn for a moment to look at Table 5.1, we can clearly see statistically significant differences, especially between hernia and infiltration groups. None of the classes share the same number of images. It is rather apparent that each class shows different proportions of images to other groups with extreme variations.

We utilized imgaug[225] a python data augmentation library to increase the data to see its effect on the reduction of overfitting (improve network generalization) and imbalanced class problems as discussed in Section 4.3.2. The imbalance class problem occurs when the underlying training examples are made up of unequal distributions for each class, causing data from some classes to dominate. The cases of multiple minority classes negatively affect the general performance of the classifier on minority classes, resulting in poor generalization performance because of the strong bias towards the majority groups. Therefore, we have selectively carried out heavy data augmentations on the minority groups such as hernia which contains only 110 total images before we even split the data into three datasets. We first apply data augmentation and pass it to the learning algorithm for cardiorespiratory classifications. In this case, images are passed along corresponding class labels, showing a supervised learning approach. Whereas, the unsupervised feature extractor takes only images of size 224 x 224 because it is necessary to match with the pre-trained models. However, in the unsupervised approach, the training set is not further augmented, but rather the algorithm is fed with the original training image dataset since the objective is to reconstruct the original images while retaining the key characteristics of the data.

5.2 Tools and Experimental Setup

We carried out the experiments in the language of Python programming, one of the most widely used today for deep learning and data science, with several libraries implemented by third parties. Some of the powerful libraries are used in this research. The most popular frameworks launched with neural network functionality are PyTorch and Tensorflow. We used the PyTorch library to implement the CNN model. PyTorch appeared to support the creation of dynamic computation graphics (DCG) and also offers programmers to connect the vast ecosystem of Python libraries and GPU-accelerated deep neural network programming features[226]. PyTorch was specifically adapted for the GPU functionality in Python and was written using a combination of python, C++, CUDA, and the design principle is also made more pythonic (tightly integrated into python) [227]. However, PyTorch is not the only library used in our implementation, but rather many packages exist such as pandas, which is an open-source framework with structures for storing data, read directly from a text file, or generally for processing data flexibly and quickly. We also used NumPy[228], a python language package used to create data structures for multi-array dimensions and matrices, which supports highly complex mathematical functions, linear algebra, statistics, and other features.

Similarly, for image processing, we used ImageIO python library for reading and writing image files, ImgAug[229] for image augmentation, OpenCV-python(short for open-source computer vision), and another powerful python image library (PIL) for converting image formats [230]. Turning now to the experimental evidence on PyTorch because within that, there are many scientific computing python packages. PyTorch is now being adopted in DL researches covering several DL libraries such as Torch, a top-level package that contains all other packages, and tensor library. Within Torch, there exist a torch.nn (an umbrella term that encompasses extensible classes and modules such as layers, weights, and forward functions for building neural networks) and the torch.autograd which are the primarily workhorse packages of PyTorch. Torch.nn is where we built the CNN model. The torch.autograd handles the derivative calculations needed to optimize the neural network weights and provides the backward and forward pass functionalities [231]. Torch.nn.functional and torch.optim are sub-packages of the torch. The first one offers functional access to activation functions, convolution operations, and loss functions, and torch.optim (the optimizer) gives access to typical standard optimization algorithms such as Adam and SGD (short for stochastic gradient descent).

We run our experiments on premium Google Cloud Platform (GCP) infrastructure. A series of experiments were carried out on two distinct Google-managed computing services (VM instances). Our first experimental investigation specifically, the deep transfer learning (DTL) component was undertaken on NVIDIA Tesla K80 GPU optimized Debian with pre-installed Google DL images such as CUDA 10.0 and CuDNN. However, we continued to set up virtual environments and install many necessary external C++ library dependencies needed to support the GPU capability and unsupervised experimental approaches. Our second investigation (the supervised detection component) was performed in a Google Compute Engine of NVIDIA GK210GL Tesla K80 with 13 GB of RAM, 215 GB of HDD, and Intel(R) Xeon(R) CPU @ 2.30GHz model. We then install PyTorch 1.0 with python 3.7, CUDA 10.0, and cuDNN 7.4.2 (short for CUDA deep neural network).

The computational resource constraint has vigorously challenged us in the first stages of a prototype implementation. One of the main difficulties with this line of reasoning is that DL models require substantial training time, however, it is not necessarily true because it shows variations over different models. This would be certainly true, in the case of large models that we have discussed in Section 2.5.2. There is, therefore, a definite need for DL specialized computing resources particularly due to the requirement for processing scale and when large data is available for training. In our case, the image dataset was originally a size of 46 GB, however, it grows into a total of 143,576 images (52 GB size) after data augmentation is performed. This clearly shows a real-challenge that does not give the chance to fight the research. However, the initiative that Google brings now, has also stepped-up access to the DL infrastructure for researchers to run their experiments via optimized tensor processing units (abbreviated as TPUs). Much of what is already possible in our experimental investigations is due to Google's managed services. However, it has emerged with financial burdens because there are costs per service usage, and therefore it was far more expensive. On average, one epoch completes in 58 minutes for training our dataset. What is important for us to recognize here, is that we train our model for 60-100 epochs on the GCP, meaning it would take days to complete the entire process. In this way, we have been running our experiments on the Google platform for 6 months and sometimes in parallel mode of training on two VM instances. One VM instance for the DTL component and the second, for the supervised detection component (here, tweaking models, tuning hyperparameters).

5.3 Model Evaluation

The general procedure comprises two major steps. Our first experimental evaluation was carried out on the unsupervised complete fine-tuning for pre-trained models. The training involves variational autoencoders (VAE) with Beta-Bernoulli distribution. The ultimate goal is to balance the network complexity and reconstruction quality. As previously discussed, in the Bayesian deep learning section (Section 2.7), the loss function is minimized when training a VAE is composed of reconstruction and regularization terms as shown in Equation 11.

$$L(\theta,\varphi;x) = -D_{KL}(q_{\varphi}(\mathbf{z}|\mathbf{x})||p_{\theta}(\mathbf{z})) + E_{q_{\varphi}(\mathbf{z}|x)}[\log p_{\theta}(x|\mathbf{z})] \quad (11)$$

Where **θ** and **φ** are neural network parameters, **x** represents the input image (in our case, X-ray images) and **z** is the latent space while the last part in the equation is a term for reconstructing the input images with a given z that follows $p\theta(x|z)$.

The KL-divergence (D_{KL}) introduced here, is that the latent space **z** should be regularized, making the distributions returned by the encoder close to a standard Gaussian distribution in terms of the mean and the covariance matrices of two distributions. In its strictest sense, KL-divergence acts as a regularization term to approximate $q_{\varphi(\mathbf{z}|\mathbf{x})}$ and the true posterior $p\theta(\mathbf{z}|\mathbf{x})$ and the estimation difference is greater or equal to zero [232]. In a nutshell, based on the initial data distribution, the architectural design of the encoder network, and the choice of latent dimension (size), the number of features representing the original data decreases during the encoding process. Therefore, part of the model evaluation would check how the encoding-decoding architectural design retains the original information by looking at the reconstruction error during decoding (measure the differences between the data and the encoded-decoded data). Secondly, there is a measure of KL-divergence for reducing network dimensionalities. With this, we wrapped up evaluation approaches for the first empirical investigations of our research more specifically, the DTL component.

We, therefore, moved on to present the evaluation metric for the supervised detection component. We used the area under the ROC curve (AUC) measure to evaluate our proposed model performance. We choose AUC because it helps us compare various methodological approaches in the related existing literature for example the work of Wang et al.[233]. AUC provides a binary performance outcome for the classifier, measuring the level of the classifier's certainty about the classification made (for each class presented to the classifier)[234, 235].

5.4 Experimental Design

What is appealing about the cardiorespiratory disease detection model is that it is designed to address a twofold concern. Firstly, we want to investigate the effect of applying unsupervised training of pre-trained CNN architectures with variational autoencoders (VAEs) being the non-parametric Bayesian factor model at its core to X-ray image classification. Secondly, the empirical study aims to assess how the choice of various hyperparameters affects training performance in the detection of cardiorespiratory diseases via supervised learning.

In the pages that follow, it will be argued specifically in addressing the research gaps. We considered that quantitative measures would usefully supplement and extend the principal theoretical implications of existing literature findings. In fact, spotting dangerous items in X-images is an exciting field and the research to date on these aspects have often been discussed on an experimental basis, however, despite these promising results, questions remain. It is now necessary to explain the principal findings of the current investigation, together with, significant associations (causal factors) coupled with the detrimental effects to provide a solid evidence base for cardiorespiratory disease detection. It was carried out based on the weakness of the previous methods and then, we introduce elegant solutions and evaluate their results and compare them against the existing state-of-art methods. The abundant room for further progress in the existing literature has therefore appeared to shed light on finding fruitful ways of improvements when developing a predictive model for X-ray images. In the majority of cases, we begin with already existing approaches and parameter values and then continued to bring discriminative approaches of supervised learning.

Therefore, we precisely design the experiment by first training the NIH images without the class labeled data in an unsupervised manner with VAE architectural definition. Then, remove the decoder portion of the VAE and rather, add an extra classification layer on the top of the encoder and then conduct an end-to-end complete fine-tuning of the network parameter (weights) only over the labeled examples. During training, each model is provided with necessary epochs (the number of times the train sampling repeats) to obtain convergence or before the validation errors begin to increase. The unsupervised training acts as a baseline for the supervised detection component and assessed its methods for the impact on the classifier.

5.5 Implementation

The DTL model is developed according to the high-level neural network architectural definition shown in Figure 4.2. The input image to the model is resized to 224×224 and further downscaled to provide a smaller size of latent representation and compel learning through a variational inference scheme to fit our IBP based DTL model and infer the weights of the networks of fully-connected layers. This allows learning from a compressed version of X-ray images by the deep convolutional autoencoder. At each learning iteration, the model is feed first with encoders and followed by a decoder. The loss function then compares the reconstructed image with the original input data and backpropagate errors in the model to update the neural network weights.

Table 5.2: Convolutional encoder network details

Layer	CNN Encoder
CNN1	`Conv2d(3, 32, 3, stride=1, padding=1)` `Max pooling, ReLU`
CNN2	`Conv2d(32, 64, 3, stride=1, padding=1),` `Max pooling, ReLU`
CNN3	`Conv2d(64, 128, 3, stride=1, padding=1)` `Max Pooling, ReLU`
CNN4	`Conv2d(128, 256, 3, stride=1, padding=1)` `Max Pooling, ReLU`
CNN5	`Conv2d(256, 512, 3, stride=1, padding=1)` `Max Pooling, ReLU`
CNN6	`Conv2d(512, 512, 3, stride=1, padding=1)` `Max Pooling, ReLU`
Flatten (reshape)	`torch.Size([1, 512*3*3])`
FC1	`Linear Layer (512*3*3, 14)`
FC2	`Linear Layer (512*3*3, 14)`

Typically, we have used six convolution layers and followed by a series of fully-connected layers (FCs) as[236, 237] following the popular architecture of VGG[238]. It will then be decoded layer by layer in unsupervised learning after it runs for 50 epochs. Finally, the extracted features will be fine-tuned in an end-to-end manner as part of the detection component of multi-label classifications. For the detection, the decoder network is removed, and a classifier is added to the top of the encoding network. A linear transformation is carried out to map the input tensors into 14 disease output classes and followed by an activation function for detecting the particular disease. The training process is repeated for all image samples/batches and pass to the network for loss, gradient, and optimization computations with the PyTorch DL library for 100 epochs.

5.6 Training, Tuning, and Experimental Results

Our CNN classifier is trained with 128 mini-batch sizes for 100 epochs. We use the stochastic gradient descent (SGD) method for optimization with the following parameters: momentum ($\beta_1 = 0.9$), L^2 weight decay of 1e-4 and we adopted a dynamic reduction of the learning rate by a factor of 10 after three trials of no further improvement in the validation performance in the training phase. The learning rate can be expressed as a hyperparameter that governs the convergence step. The learning strategy we brought here to the supervised fine-tuning remains crucial because it allows the learning rate to be able to optimize according to the state of convergence of the model. In its strictest sense, the training begins with an initial learning rate of [0.1 - 0.4], and then it decreases to cross the plateau when it becomes difficult to converge to the global minimum by the gradient algorithm. Such periodical iterations and optimization of the learning rate help the training algorithm to escape saddle points as explained by Mukherjee et al.[239]. Saddle points are indeed serious obstacles to local optimization algorithms [240, 241]. Therefore, we followed a reasonable approach to tackle this issue by having a dynamic learning rate. According to our experimental investigation, the best learning rate configuration was found to be 0.2, resulting in an AUC score of 88.01, which is the metric used for model selection and optimization measures during the learning and testing stages. The overall performance of the classifier was generally assessed by the AUC, representing predicted probability ranging from 0 to 1. The model with a higher AUC (closer to 1) was, therefore, chosen for predicting the new observations (test data).

The choice of hyperparameter investigation was not limited to the learning rate. However, we continued to examine variations in the values of batch size as part of parameter optimization. For this reason, we were enthusiastically involved in looking at different values of batch size more specifically, values that fall within (1, 64, 80, 96, 100, 112, 128, and 256). In this way, we can eliminate several problems in previous approaches. This exhaustive search of all the possible values, however, poses several problems, the first being that computationally expensive. The second obvious difficulty is finding an optimal value remains less clear. Therefore, the hyperparameter search presents a difficult set of problems, especially in resource constraint settings. It often involves lots of trial and error, however, we limited our search to certain values as shown in Table 5.3.

Turning now to the experimental supervised fine-tuning model performance on cardiorespiratory disease detection, the result was found to score an AUC measure of 88 from the preliminary analysis of the EfficientNet CNN model. What stands out in this performance is the effect of unsupervised learning has shown statically significant dominance over the standard CNN classifier (without the pre-trained model). Needless to say, the standard CNN classifier achieved an AUC value of 80.98, which is significantly lower than the predicted value obtained by introducing the DTL component (unsupervised feature extraction) as shown in Figure 5.1 and Table 5.2.

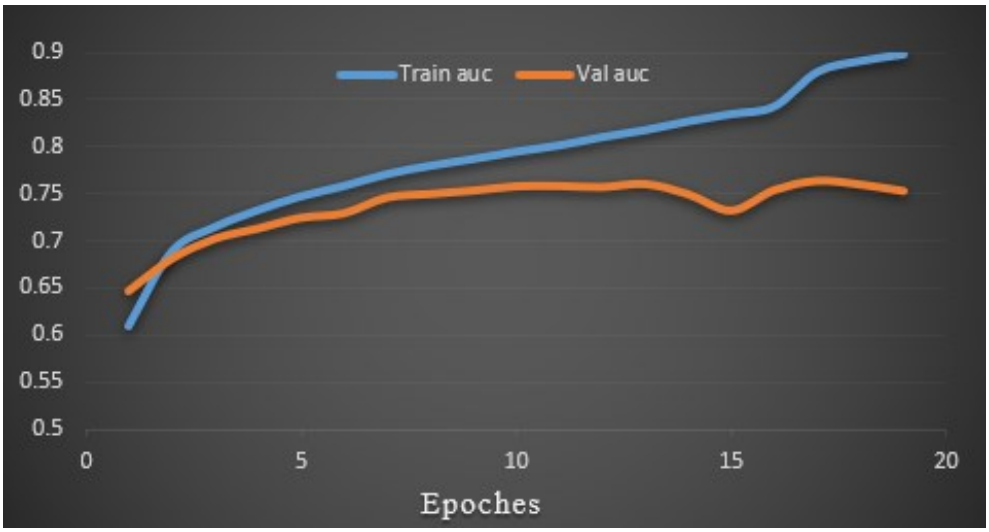

Figure 5.1: Accuracy progression

Table 5.3: Training accuracy of each class

Atelectasis	0.8017
Cardiomegaly	0.964
Effusion	0.9474
Infiltration	0.7328
Mass	0.905
Nodule	0.8959
Pneumonia	0.8816
Pneumothorax	0.9293
Consolidation	0.8611
Edema	0.9247
Emphysema	0.9188
Fibrosis	0.8529
Pleural Thickening	0.8323
Hernia	0.8739
Average AUC	**0.8801**

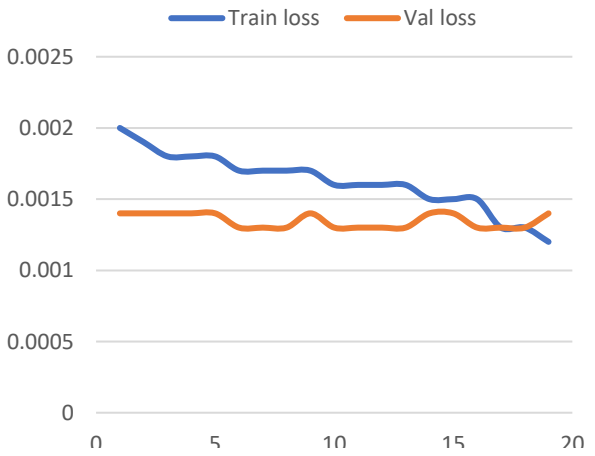

Figure 5.2: Training and validation loss progress

One statistic worth noting is that the quantitative performance for detecting the multi-label 14 cardiorespiratory diseases varies among different groups. The outcome of the experiment is particularly noticeable in Table 5.2, in which a couple of points are worth mentioning here: firstly, we are able to detect cardiomegaly (AUC =96.4), effusion (AUC=94.74), pneumothorax (AUC =92.93), edema (92.47), emphysema (AUC = 91.88) and mass (90.5) more accurately compared to other categories. Mass and nodule appear to show statically high detection results, thereby was not adversely affected to each other even though the underlying patterns are similar. One may speculate, a clear benefit of our unsupervised approach to cardiorespiratory diseases are immediately visible in contrast to earlier findings. For example, Wang et al.[233] detected mass with an AUC value of 56.09, appearing to confirm our approach is promising. We'll shortly comeback later on how our study casts doubt on earlier researches. This is due to a combination of factors that provide strong empirical evidence to the discriminatory power of the model performance. Turning now to the principal points drawn from Figure 5.2 and Table 5.2 without jumping to a conclusion and comparison of previous methods. Figure 5.2 reveals that there has been a steady drop in the train and validation loss until epoch 19, where validation error stops decrease. The model performance reached a peak value at epoch 19(AUC = 88.01), thereby no further improvement is shown afterward.

Figure 5.3: Various LR values for batch size 128

Conducting a hyperparameter search is another area of improvement, undertaken repeatedly for cardiorespiratory disease detection as shown in Figure 5.3. It marked a major turning point in the model performance. The results also lend strong support to the causal relationship the hyperparameter optimization would carry to our model.

For the sake of clarity, we have selected here a few parameters that merely reflect on the final model. What can be seen in Figure 5.3 is the general patterns the different learning rates have in training the model. Learning rate 0.2 represents the least error, a loss value of 0.0013. There are however several possible alternative explanations of our findings that would rather interpret the experimental results appealing as shown in Table 5.3.

Table 5.4: Summary of training result

Batch Size	Learning Rate	Epoch	Beta	Model	Train Images	Train Loss	Validation Loss	Measure AUC%
128	0.1	20	β=1	EfficientNet	108,708	0.0016	0.0013	82.12
	0.2	17			108,708	0.0013	0.0013	88.01
	0.3	8			108,708	0.0016	0.0013	79.26
	0.4	9			108,708	0.0015	0.0013	83.25
64	0.1	18			99,549	0.0029	0. 0026	84.39
100	0.1	26			99,549	0.0014	0.0013	85.75
128	0.1	27			93,377	0.0017	0.0016	85.59
256	0.1	39			99,549	0.0007	0.0007	83.81
128	0.2	6		DenseNet	108,708	0.0016	0.0013	80.98
	0.3	9			108,708	0.0014	0.0013	85.73
	0.1	10			108,708	0.0016	0.0013	82.50
	0.4	5			108,708	0.0016	0.0013	80.78
128	0.1	22	β=5	EfficientNet	108708	0.0015	0.0013	83.31
	0.2	14			108708	0.0015	0.0013	82.69
	0.3	14			108708	0.0013	0.0013	87.69

We used the SGD algorithm for optimization and due to its stochastic (random) nature, the randomness of the algorithm is a double-edged sword. One it helps escape the local minimum however, difficulties arise, it continuous to bounce (oscillate) rather than decreasing smoothly. We adopted one solution to this dilemma, which is to gradually reduce the learning rate by a factor of 10 after it shows a downward trend in validation performance. With this line of reasoning, the learning rates seen in Table 5.4 are initial values which are then followed by a steady decrease. For example, the learning rate value which starts at 0.2 was later changed to 0.02, to allow the algorithm to stabilize at the global minimum, thereby achieved an AUC value of 88.01 and a loss value of 0.0012. The algorithm at the first phase, with a larger learning rate (0.2) would rash to learn quickly (achieve faster convergence) whereas, with smaller learning rates (say 0.02), it would make small steps/jumps, avoid oscillations during training. One question that needs to be asked, however, is whether to calculate the gradients based on a batch (total number of training examples present in a single epoch) or mini-batches (in small random sets of instances). In our case, we used mini-batches of size 64,100,128, and 256. It is worth bearing in mind that, the combination of different hyperparameters precisely a batch size 128, a learning rate of 0.2 yields greater performance at epoch 17 during the training phase. As a result of these investigations, the final model is then evaluated for the new observations using a measure of the AUC, and the ROC curve is shown in Figure 5.4.

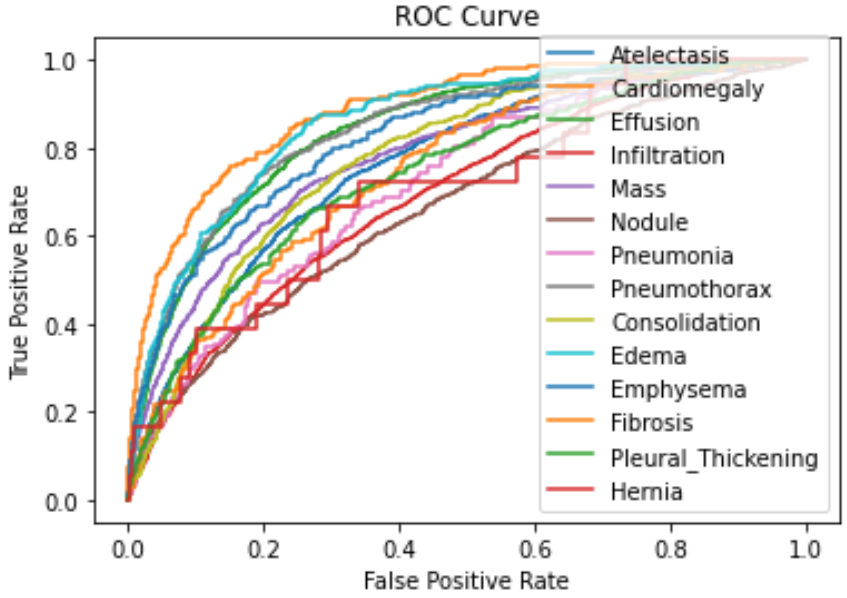

Figure 5.4: AUC ROC curve for 11,213 test images

5.7 Comparative Analysis

We compare our approaches quantitatively with those found by other researchers in this area. It is quite revealing in several ways. One way of interpreting the results is to compare the statistical AUC scores of particular diseases within the various methodologies. Another possible explanation of our findings is via the weighted average measures of AUC as shown in Table 5.4.

Table 5.5: Comparative Analysis

Disease	Wang [43]	Yao [44]	Kumar [47]	Guendel [48]	Baltruschat [210]	Xia [45]	Sirazitdinov [49]	Ma [50]	Our Method
Atelectasis	0.7003	0.733	0.762	0.767	0.763	0.7433	0.777	0.7627	**0.8017**
Cardiomegaly	0.81	0.856	0.913	0.883	0.875	0.8748	0.887	0.8835	**0.964**
Effusion	0.7585	0.806	0.864	0.828	0.822	0.8114	0.826	0.8159	**0.9474**
Infiltration	0.6614	0.673	0.692	0.709	0.694	0.6772	0.694	0.6786	**0.7328**
Mass	0.6933	0.718	0.75	0.821	0.820	0.7833	0.825	0.8012	**0.905**
Nodule	0.6687	0.777	0.666	0.758	0.747	0.6975	0.768	0.7293	**0.8959**
Pneumonia	0.658	0.684	0.715	0.731	0.714	0.6959	0.731	0.7097	**0.8816**
Pneumothorax	0.7993	0.805	0.859	0.846	0.840	0.8098	0.845	0.8377	**0.9293**
Consolidation	0.7032	0.711	0.784	0.745	0.749	0.7256	0.747	0.7443	**0.8611**
Edema	0.8052	0.806	0.888	0.835	0.749	0.8327	0.838	0.8414	**0.9247**
Emphysema	0.833	0.842	0.898	0.895	0.895	0.8222	0.906	0.8836	**0.9188**
Fibrosis	0.7859	0.743	0.756	0.818	0.816	0.8041	0.822	0.8007	**0.8529**
Pleural T.	0.6835	0.724	0.774	0.761	0.763	0.7513	0.782	0.7536	**0.8323**
Hernia	0.8717	0.775	0.802	0.896	**0.937**	0.8996	0.864	0.8763	0.8739
Average	0.7451	0.761	0.795	0.807	0.806	0.7810	0.808	0.7941	**.8801**

5.8 Discussion

The experimental results presented to evaluate the performance of the detection of the cardiorespiratory disease model have revealed several contrasting themes. What we know about the performance of the model is largely based upon empirical studies. As shown in Table 5.4, the comparison result demonstrates that our proposed model outperforms the existing methods such as the works of Wang et al. [43], Yao et al.[46], Kumar et al.[242], Guendel et al.[48], Baltruschat et al.[210], ChestNet[45], Sirazitdinov et al.[243] and Ma et al.[244]. A couple of points are worth mentioning here: firstly, the method adopted here shows a statistically significant AUC improvement over other groups for all the cardiorespiratory diseases except hernia. Baltruschat et al.[210] appear to show an increased hernia detection (AUC = **0.937**) in contrast to earlier findings and our proposed model. Comparison of the findings with those of other studies, however, confirms thirteen out of fourteen diseases yields higher detection rates in our proposed model compared to other works. Secondly, there is a profound difference in the mean score, particularly, we found an AUC value of 88.01, while the existing approaches are limited to a maximum of 80.07 AUC detection score.

Several factors could explain this observation. To the best of our knowledge, none of the existing approaches investigate the effect of deep transfer learning based on VAE neural network approaches in the first place. Finding co-occurred features in X-ray images through latent structure learning can provide objective grounds for knowledge transfer and then contribute to the model classifier performance. We also consider quantifying the various hyperparameter values and have drawn attention to the unbalanced scenarios in the dataset with the individual-based data augmentation techniques. Finally, important limitations that have contributed to the decline of model performance need to be considered. For this reason, there seems to be some evidence to indicate that ten percent (10%) of the misclassification biases (labeling errors) exist, incorporating incorporates an intrinsic potential source of bias in the NIH ChestX-ray14 image dataset [233]. Another difficulty lies in the underlying heterogeneity of the images because there exist several image quality factors as shown in Table 2.1 from Chapter two. Images vary and present different modifications that can lead to serious consequences on the visual interpretation of X-ray images. All these factors can impact the effectiveness of the cardiorespiratory disease detection approach.

Chapter 6: Conclusion and Future Work

6.1 Conclusion

In this work, we have discussed major technological advances in neural network-based DL and TL approaches to address public health problems caused by cardiorespiratory diseases from radiographic images (X-rays). DL-based TL algorithms are double-edged swords with their own strengths and shortcomings. The key strengths of TL remain in its power to overcome the training overhead of data-hungry DL algorithms and vigorously minimize the demand for massive training data by transferring knowledge from pre-trained models. On another spectrum, conventional fine-tuning tends to affect the target performance adversely. What part of the knowledge should be transferred from the source domain to the new task is still a longstanding problem. Current detectors also need labeled training data for detecting cardiorespiratory diseases, which is rare to find in real-world scenarios, especially within clinical settings. Besides, researches are limited to single clinical settings and might not apply to all situations due to the high bias in the training data. There is also a high variability within the data that needs further investigation.

At the heart of this critical juncture, there is, therefore, a definite need to combine powerful technologies to fill research gaps. Accordingly, we leverage unlabeled data using the unsupervised β-VAE algorithm for DTL from EfficientNet pre-trained model. At its core, there is the Bayes theorem which proves it's worthy to explain casual factors of variation in the data. Extracted features from the unsupervised algorithm are further trained using a standard supervised classifier model for multiple cardiorespiratory disease detection from class labeled X-ray images. The proposed solution is generally comprised of two principal components: One is the DTL component to examine the effectiveness of TL on X-ray images using β-VAE. The second part is the detection component that takes features from the DTL component and utilizes it for multiple cardiorespiratory disease detection. The experiments were run using Python programming and an open-source PyTorch framework on the Google cloud environment. We conduct a hyperparameter search to find appropriate values and develop an algorithm to solve the class imbalance problem. Further X-ray images were collected from Tikur Anbesa Hospital to externally validate our approach and make it more robust. What is striking about the experiment was that our approach achieves an 88% AUC Score which outperforms current SOTA performance results by around 8%. The rest of this chapter is organized as follows. Section 6.2, present the major contributions of our works. In Section 6.3, we conclude with possible future research directions.

6.2 Contribution

The major contributions of this thesis can be summarized as follows:

- We proposed a nonparametric Bayesian statistical IBP model of fully-connected layers for TL using the DTL (VAE based neural network) approach on X-ray images. This enables learning the number of latent factors for nodes in a network's hidden layer that fit the available data. Our IBP-based approach introduces a framework for data-driven transfer learning of useful features for cardiorespiratory disease detection.
- We developed an algorithm that works at different granularity levels to overcome the class imbalance problem of unbalanced NIH dataset distributions.
- We built a new dataset, called Tikur_ChestX-ray14, an open-source X-ray image dataset. We make it freely accessible to CV researchers to evaluate their models. ML researches in a single setting present a considerable challenge in healthcare as how well their approach generalizes to another clinical population remains unclear. To this effect, we acquired additional X-ray images from Tikur Anbesa Specialized Hospital and perform preprocessing to remove patient private details. Finally, we have released it on GitHub (https://github.com/kibromhft/Tikur_ChestX-ray14). In doing so, other researchers can validate their artifacts and contribute to the dataset.
- We extended our work to conduct a hyperparameter search to determine learning rates and hidden layer dimensionality that are relevant for the detection task.
- We outperform the current SOTA cardiorespiratory disease detection approaches.

6.3 Future Work

Radiologists do not always agree on what diseases are present in a given X-ray. Moreover, different diseases pattern tends to look similar, further complicating the detection task. This implies, there is still abundant room for further progress. Therefore, we will account for noisy labels by performing "Supertruth" enrichment: identifying and removing X-ray images that result in high predictive uncertainty for our models. We will then retrain our models using the enriched subset of training. We believe that additional analysis is warranted of the effect of removing noisy clinical-assigned labels. Additionally, since our publicly available dataset is limited in size, we intend to collect X-ray images to make it more representative. We will also consider optimizing the time required to run the DTL code as our proposed solution follows a combination of approaches.

References

[1] E. Ayan and H. M. Ünver, "Diagnosis of Pneumonia from Chest X-Ray Images Using Deep Learning," p. 5, Jun. 2019.

[2] C. S. Dela Cruz *et al.*, "Future Research Directions in Pneumonia. NHLBI Working Group Report," *Am. J. Respir. Crit. Care Med.*, vol. 198, no. 2, pp. 256–263, Jul. 2018, doi: 10.1164/rccm.201801-0139WS.

[3] A.-A. Seidu, K. S. Dickson, B. O. Ahinkorah, H. Amu, E. K. M. Darteh, and A. Kumi-Kyereme, "Prevalence and determinants of Acute Lower Respiratory Infections among children under-five years in sub–Saharan Africa: Evidence from demographic and health surveys," *SSM - Popul. Health*, vol. 8, p. 100443, Aug. 2019, doi: 10.1016/j.ssmph.2019.100443.

[4] H. N. Gouda *et al.*, "Burden of non-communicable diseases in sub-Saharan Africa, 1990–2017: results from the Global Burden of Disease Study 2017," *Lancet Glob. Health*, vol. 7, no. 10, Art. no. 10, Oct. 2019, doi: 10.1016/S2214-109X(19)30374-2.

[5] I. Dalju, "Occupational risk factors associated with respiratory symptoms among tannery workers in Mojo town, Southeast Ethiopia, 2018: a comparative cross-sectional study," p. 10, 2019.

[6] A. A. Asgedom, M. Bråtveit, and B. E. Moen, "High Prevalence of Respiratory Symptoms among Particleboard Workers in Ethiopia: A Cross-Sectional Study," *Int. J. Environ. Res. Public. Health*, vol. 16, no. 12, p. 2158, Jun. 2019, doi: 10.3390/ijerph16122158.

[7] M. Endriyas, "Burden of NCDs in SNNP region, Ethiopia: a retrospective study," p. 7, Jul. 2018.

[8] R. Ahmed, R. Robinson, and K. Mortimer, "The epidemiology of noncommunicable respiratory disease in sub-Saharan Africa, the Middle East, and North Africa," *Malawi Med. J.*, vol. 29, no. 2, p. 203, Aug. 2017, doi: 10.4314/mmj.v29i2.24.

[9] J. Ma, Y. Song, X. Tian, Y. Hua, R. Zhang, and J. Wu, "Survey on deep learning for pulmonary medical imaging," *Front. Med.*, vol. 14, no. 4, pp. 450–469, Aug. 2020, doi: 10.1007/s11684-019-0726-4.

[10] J. Hodler, R. A. Kubik-Huch, and G. K. von Schulthess, Eds., *Diseases of the Chest, Breast, Heart and Vessels 2019-2022: Diagnostic and Interventional Imaging*. Cham: Springer International Publishing, 2019.

[11] S. Dawkes and M. O'Reilly, "Chest X-ray interpretation," *Br. J. Card. Nurs.*, vol. 14, no. 5, pp. 1–9, May 2019, doi: 10.12968/bjca.2019.0004.

[12] J. Yanase and E. Triantaphyllou, "A systematic survey of computer-aided diagnosis in medicine: Past and present developments," *Expert Syst. Appl.*, vol. 138, p. 112821, Dec. 2019, doi: 10.1016/j.eswa.2019.112821.

[13] A. K. Bharodiya and A. M. Gonsai, "An improved edge detection algorithm for X-Ray images based on the statistical range," *Heliyon*, vol. 5, no. 10, p. e02743, Oct. 2019, doi: 10.1016/j.heliyon.2019.e02743.

[14] P. Rajpurkar *et al.*, "Deep learning for chest radiograph diagnosis: A retrospective comparison of the CheXNeXt algorithm to practicing radiologists," *PLOS Med.*, vol. 15, no. 11, p. e1002686, Nov. 2018, doi: 10.1371/journal.pmed.1002686.

[15] K. S. Rani, M. Kumari, and V. B. Singh, “Deep Learning with Big Data: An Emerging Trend,” p. 9, Oct. 2019.

[16] S. Akcay and T. Breckon, “Towards Automatic Threat Detection: A Survey of Advances of Deep Learning within X-ray Security Imaging,” *ArXiv200101293 Cs*, Jan. 2020, Accessed: Dec. 11, 2020. [Online]. Available: http://arxiv.org/abs/2001.01293.

[17] M. Raghu, J. Kleinberg, C. Zhang, and S. Bengio, “Transfusion: Understanding Transfer Learning for Medical Imaging,” p. 11, Feb. 2019.

[18] N. Best, J. Ott, and E. J. Linstead, “Exploring the efficacy of transfer learning in mining image-based software artifacts,” *J. Big Data*, vol. 7, no. 1, p. 59, Dec. 2020, doi: 10.1186/s40537-020-00335-4.

[19] L. Alzubaidi *et al.*, “Towards a Better Understanding of Transfer Learning for Medical Imaging: A Case Study,” *Appl. Sci.*, vol. 10, no. 13, p. 4523, Jun. 2020, doi: 10.3390/app10134523.

[20] H. Malik, M. S. Farooq, A. Khelifi, A. Abid, J. Nasir Qureshi, and M. Hussain, “A Comparison of Transfer Learning Performance Versus Health Experts in Disease Diagnosis From Medical Imaging,” *IEEE Access*, vol. 8, pp. 139367–139386, 2020, doi: 10.1109/ACCESS.2020.3004766.

[21] T. Liu, S. Alibhai, J. Wang, Q. Liu, X. He, and C. Wu, “Exploring Transfer Learning to Reduce Training Overhead of HPC Data in Machine Learning,” in *2019 IEEE International Conference on Networking, Architecture and Storage (NAS)*, EnShi, China, Aug. 2019, pp. 1–7, doi: 10.1109/NAS.2019.8834723.

[22] M. Tsiakmaki, G. Kostopoulos, S. Kotsiantis, and O. Ragos, “Transfer Learning from Deep Neural Networks for Predicting Student Performance,” *Appl. Sci.*, vol. 10, no. 6, p. 2145, Mar. 2020, doi: 10.3390/app10062145.

[23] S. Bharati, P. Podder, and M. R. H. Mondal, “Hybrid deep learning for detecting lung diseases from X-ray images,” *Inform. Med. Unlocked*, vol. 20, p. 100391, 2020, doi: 10.1016/j.imu.2020.100391.

[24] G. Liang and L. Zheng, “A transfer learning method with deep residual network for pediatric pneumonia diagnosis,” *Comput. Methods Programs Biomed.*, vol. 187, p. 104964, Apr. 2020, doi: 10.1016/j.cmpb.2019.06.023.

[25] T. Tan *et al.*, “Optimize Transfer Learning for Lung Diseases in Bronchoscopy Using a New Concept: Sequential Fine-Tuning,” *IEEE J. Transl. Eng. Health Med.*, vol. 6, pp. 1–8, 2018, doi: 10.1109/JTEHM.2018.2865787.

[26] S. K. Pandey and R. R. Janghel, “Recent Deep Learning Techniques, Challenges and Its Applications for Medical Healthcare System: A Review,” *Neural Process. Lett.*, vol. 50, no. 2, pp. 1907–1935, Oct. 2019, doi: 10.1007/s11063-018-09976-2.

[27] C. J. Kelly, A. Karthikesalingam, M. Suleyman, G. Corrado, and D. King, “Key challenges for delivering clinical impact with artificial intelligence,” *BMC Med.*, vol. 17, no. 1, p. 195, Dec. 2019, doi: 10.1186/s12916-019-1426-2.

[28] F. Zhuang *et al.*, “A Comprehensive Survey on Transfer Learning,” *ArXiv191102685 Cs Stat*, Jun. 2020, Accessed: Oct. 07, 2020. [Online]. Available: http://arxiv.org/abs/1911.02685.

[29] B. Zohuri and M. Moghaddam, “Deep Learning Limitations and Flaws,” vol. 2, no. 3, p. 11, Jan. 2020.

[30] N. Sünderhauf *et al.*, “The limits and potentials of deep learning for robotics,” *Int. J. Robot. Res.*, vol. 37, no. 4–5, pp. 405–420, Apr. 2018, doi: 10.1177/0278364918770733.

[31] D. Jin *et al.*, “Artificial intelligence in radiology,” in *Artificial Intelligence in Medicine*, Elsevier, 2021, pp. 265–289.

[32] I. Carbone and M. Anzidei, Eds., *Thoracic Radiology: A Guide for Beginners*. Cham: Springer International Publishing, 2020.

[33] S. Jalal, S. Nicolaou, and W. Parker, “Artificial Intelligence, Radiology, and the Way Forward,” *Can. Assoc. Radiol. J.*, vol. 70, no. 1, pp. 10–12, Feb. 2019, doi: 10.1016/j.carj.2018.09.004.

[34] S. Waite *et al.*, “Analysis of Perceptual Expertise in Radiology – Current Knowledge and a New Perspective,” *Front. Hum. Neurosci.*, vol. 13, p. 213, Jun. 2019, doi: 10.3389/fnhum.2019.00213.

[35] G. Maskell, “Virtual special issue: error in radiology,” *Clin. Radiol.*, vol. 75, no. 3, pp. 159–160, Mar. 2020, doi: 10.1016/j.crad.2019.11.018.

[36] H. Zhan, K. Schartz, M. E. Zygmont, J.-O. Johnson, and E. A. Krupinski, “The Impact of Fatigue on Complex CT Case Interpretation by Radiology Residents,” *Acad. Radiol.*, vol. 28, no. 3, pp. 424–432, Mar. 2021, doi: 10.1016/j.acra.2020.06.005.

[37] Ö. Önder, “Learning from mistakes: Errors, discrepancies and underlying bias in Radiology with case examples,” p. 1488 words, 2020, doi: 10.26044/ECR2020/C-06328.

[38] R. B. Gunderman and P. Patel, “Perception’s Crucial Role in Radiology Education,” *Acad. Radiol.*, vol. 26, no. 1, pp. 141–143, Jan. 2019, doi: 10.1016/j.acra.2018.08.004.

[39] R. Aggarwal, I. Mukhopadhyay, R. A. George, and A. Alam, “Patient safety in radiology: Our experience,” *Med. J. Armed Forces India*, p. S0377123720301805, Dec. 2020, doi: 10.1016/j.mjafi.2020.09.006.

[40] W. Xu, J. He, and Y. Shu, “Transfer Learning and Deep Domain Adaptation,” in *Advances and Applications in Deep Learning*, M. Antonio Aceves-Fernandez, Ed. IntechOpen, 2020.

[41] O. S. Pianykh *et al.*, “Continuous Learning AI in Radiology: Implementation Principles and Early Applications,” *Radiology*, vol. 297, no. 1, pp. 6–14, Oct. 2020, doi: 10.1148/radiol.2020200038.

[42] J. M. Ryan *et al.*, “Mortality due to cardiovascular disease, respiratory disease, and cancer in adults with cerebral palsy,” *Dev. Med. Child Neurol.*, vol. 61, no. 8, pp. 924–928, Aug. 2019, doi: 10.1111/dmcn.14176.

[43] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, “ChestX-ray8: Hospital-Scale Chest X-Ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases,” p. 10, May 2017.

[44] L. Yao, E. Poblenz, D. Dagunts, B. Covington, D. Bernard, and K. Lyman, “Learning to diagnose from scratch by exploiting dependencies among labels,”

ArXiv171010501 Cs, Feb. 2018, Accessed: Dec. 01, 2020. [Online]. Available: http://arxiv.org/abs/1710.10501.

[45] Y. Xia, "ChestNet: A Deep Neural Network for Classification of Thoracic Diseases on Chest Radiography," p. 8, Jul. 2018.

[46] L. Yao, E. Poblenz, D. Dagunts, B. Covington, D. Bernard, and K. Lyman, "Learning to diagnose from scratch by exploiting dependencies among labels," *ArXiv171010501 Cs*, Feb. 2018, Accessed: Nov. 29, 2020. [Online]. Available: http://arxiv.org/abs/1710.10501.

[47] P. Kumar, M. Grewal, and M. M. Srivastava, "Boosted Cascaded Convnets for Multilabel Classification of Thoracic Diseases in Chest Radiographs," *ArXiv171108760 Cs*, Nov. 2017, Accessed: Dec. 01, 2020. [Online]. Available: http://arxiv.org/abs/1711.08760.

[48] S. Guendel *et al.*, "Learning to recognize Abnormalities in Chest X-Rays with Location-Aware Dense Networks," *ArXiv180304565 Cs*, Mar. 2018, Accessed: Nov. 29, 2020. [Online]. Available: http://arxiv.org/abs/1803.04565.

[49] I. Sirazitdinov, M. Kholiavchenko, R. Kuleev, and B. Ibragimov, "Data Augmentation for Chest Pathologies Classification," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, Venice, Italy, Apr. 2019, pp. 1216–1219, doi: 10.1109/ISBI.2019.8759573.

[50] Y. Ma, Q. Zhou, X. Chen, H. Lu, and Y. Zhao, "Multi-attention Network for Thoracic Disease Classification and Localization," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, May 2019, pp. 1378–1382, doi: 10.1109/ICASSP.2019.8682952.

[51] I. M. Baltruschat, H. Nickisch, M. Grass, T. Knopp, and A. Saalbach, "Comparison of Deep Learning Approaches for Multi-Label Chest X-Ray Classification," *Sci. Rep.*, vol. 9, no. 1, p. 6381, Dec. 2019, doi: 10.1038/s41598-019-42294-8.

[52] A. Luque, A. Carrasco, A. Martín, and A. de las Heras, "The impact of class imbalance in classification performance metrics based on the binary confusion matrix," *Pattern Recognit.*, vol. 91, pp. 216–231, Jul. 2019, doi: 10.1016/j.patcog.2019.02.023.

[53] R. Xu, G. Li, J. Yang, and L. Lin, "Larger Norm More Transferable: An Adaptive Feature Norm Approach for Unsupervised Domain Adaptation," p. 10, Nov. 2018.

[54] J. Feng and T. Darrell, "Learning the Structure of Deep Convolutional Networks," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, Dec. 2015, pp. 2749–2757, doi: 10.1109/ICCV.2015.315.

[55] Y. Ni, P. Mueller, and Y. Ji, "Bayesian Double Feature Allocation for Phenotyping with Electronic Health Records," *ArXiv180908988 Stat*, Feb. 2019, Accessed: Dec. 15, 2020. [Online]. Available: http://arxiv.org/abs/1809.08988.

[56] R. Siddiqi, "Automated Pneumonia Diagnosis using a Customized Sequential Convolutional Neural Network," in *Proceedings of the 2019 3rd International Conference on Deep Learning Technologies - ICDLT 2019*, Xiamen, China, 2019, pp. 64–70, doi: 10.1145/3342999.3343001.

[57] R. Baskerville and J. Pries-Heje, “Projectability in Design Science Research,” vol. 20, no. 1, p. 24, Aug. 2019.

[58] W. J. Requia, M. D. Adams, A. Arain, S. Papatheodorou, P. Koutrakis, and M. Mahmoud, “Global Association of Air Pollution and Cardiorespiratory Diseases: A Systematic Review, Meta-Analysis, and Investigation of Modifier Variables,” *Am. J. Public Health*, vol. 108, no. S2, pp. S123–S130, Apr. 2018, doi: 10.2105/AJPH.2017.303839.

[59] H. N. Gouda *et al.*, “Burden of non-communicable diseases in sub-Saharan Africa, 1990–2017: results from the Global Burden of Disease Study 2017,” *Lancet Glob. Health*, vol. 7, no. 10, pp. e1375–e1387, Oct. 2019, doi: 10.1016/S2214-109X(19)30374-2.

[60] J. O. Benditt, “Pathophysiology of Neuromuscular Respiratory Diseases,” *Clin. Chest Med.*, vol. 39, no. 2, pp. 297–308, Jun. 2018, doi: 10.1016/j.ccm.2018.01.011.

[61] V. Sedláková, “Options for modeling the respiratory system: inserts, scaffolds and microfluidic chips,” *Drug Discov. Today*, vol. 24, no. 4, p. 12, 2019.

[62] M. P. Ekström *et al.*, “Minimal Clinically Important Differences and Feasibility of Dyspnea-12 and the Multidimensional Dyspnea Profile in Cardiorespiratory Disease,” *J. Pain Symptom Manage.*, vol. 60, no. 5, pp. 968-975.e1, Nov. 2020, doi: 10.1016/j.jpainsymman.2020.05.028.

[63] D. Kim, Z. Chen, L.-F. Zhou, and S.-X. Huang, “Air pollutants and early origins of respiratory diseases,” *Chronic Dis. Transl. Med.*, vol. 4, no. 2, pp. 75–94, Jun. 2018, doi: 10.1016/j.cdtm.2018.03.003.

[64] A. P. Nesterova *et al.*, “Diseases of the respiratory system,” in *Disease Pathways*, Elsevier, 2020, pp. 391–442.

[65] D. N. Bhatta and S. A. Glantz, “Association of E-Cigarette Use With Respiratory Disease Among Adults: A Longitudinal Analysis,” *Am. J. Prev. Med.*, vol. 58, no. 2, pp. 182–190, Feb. 2020, doi: 10.1016/j.amepre.2019.07.028.

[66] J. Pákó, H. Kiss, and I. Horváth, “Breathprinting-Based Diagnosis: Case Study: Respiratory Diseases,” in *Breath Analysis*, Elsevier, 2019, pp. 131–143.

[67] P. Rogliani, M. Cazzola, and L. Calzetta, “Cardiovascular Disease in Chronic Respiratory Disorders and Beyond,” *J. Am. Coll. Cardiol.*, vol. 73, no. 17, pp. 2178–2180, May 2019, doi: 10.1016/j.jacc.2018.11.068.

[68] F. Shaukat, G. Raja, and A. F. Frangi, “Computer-aided detection of lung nodules: a review,” *J. Med. Imaging*, vol. 6, no. 02, p. 1, Jun. 2019, doi: 10.1117/1.JMI.6.2.020901.

[69] Y.-D. Zhang *et al.*, “IEEE Access Special Section Editorial: Advanced Signal Processing Methods In Medical Imaging,” *IEEE Access*, vol. 6, pp. 61812–61818, 2018, doi: 10.1109/ACCESS.2018.2875308.

[70] I. Livieris, A. Kanavos, and P. Pintelas, “Detecting Lung Abnormalities From X-rays Using an Improved SSL Algorithm,” *Electron. Notes Theor. Comput. Sci.*, vol. 343, pp. 19–33, May 2019, doi: 10.1016/j.entcs.2019.04.008.

[71] J. A. Seibert, “PROJECTION X-RAY IMAGING: RADIOGRAPHY, MAMMOGRAPHY, FLUOROSCOPY,” *Health Phys.*, vol. 116, no. 2, p. 9, 2019.

[72] S. Waite *et al.*, "Analysis of Perceptual Expertise in Radiology – Current Knowledge and a New Perspective," *Front. Hum. Neurosci.*, vol. 13, p. 213, Jun. 2019, doi: 10.3389/fnhum.2019.00213.

[73] N. Maskell, *Pocket Tutor Chest X-Ray Interpretation.* London: JP Medical Ltd, 2012.

[74] E. Castro-Gutierrez, L. Estacio-Cerquin, J. Gallegos-Guillen, and J. D. Obando, "Detection of Acetabulum Fractures Using X-Ray Imaging and Processing Methods Focused on Noisy Images," in *2019 Amity International Conference on Artificial Intelligence (AICAI)*, Dubai, United Arab Emirates, Feb. 2019, pp. 296–302, doi: 10.1109/AICAI.2019.8701297.

[75] K. Kalantar-Zadeh, K. J. Berean, R. E. Burgell, J. G. Muir, and P. R. Gibson, "Intestinal gases: influence on gut disorders and the role of dietary manipulations," *Nat. Rev. Gastroenterol. Hepatol.*, vol. 16, no. 12, pp. 733–747, Dec. 2019, doi: 10.1038/s41575-019-0193-z.

[76] T. Torsy, R. Saman, K. Boeykens, M. Eriksson, S. Verhaeghe, and D. Beeckman, "Factors associated with insufficient nasogastric tube visibility on X-ray: a retrospective analysis," *Eur. Radiol.*, Oct. 2020, doi: 10.1007/s00330-020-07302-w.

[77] J. S. Klein and M. L. Rosado-de-Christenson, "A Systematic Approach to Chest Radiographic Analysis," in *Diseases of the Chest, Breast, Heart and Vessels 2019-2022*, J. Hodler, R. A. Kubik-Huch, and G. K. von Schulthess, Eds. Cham: Springer International Publishing, 2019, pp. 1–16.

[78] F. Andres Giraldo Vallejo, R. Romero, M. Mejia, and E. Quijano, "Primary Spontaneous Pneumothorax, a Clinical Challenge," in *Pneumothorax*, K. Amer, Ed. IntechOpen, 2019.

[79] M. O. Kattea and O. Lababede, "Differentiating Pneumothorax from the Common Radiographic Skinfold Artifact," *Ann. Am. Thorac. Soc.*, vol. 12, no. 6, pp. 928–931, Jun. 2015, doi: 10.1513/AnnalsATS.201412-576AS.

[80] M. C. Dominguez and B. R. Alvares, "Pulmonary atelectasis in newborns with clinically treatable diseases who are on mechanical ventilation: clinical and radiological aspects," *Radiol. Bras.*, vol. 51, no. 1, pp. 20–25, Jan. 2018, doi: 10.1590/0100-3984.2016.0157.

[81] I. Carbone and M. Anzidei, Eds., *Thoracic Radiology: A Guide for Beginners*. Cham: Springer International Publishing, 2020.

[82] G. Raghu *et al.*, "Diagnosis of Idiopathic Pulmonary Fibrosis. An Official ATS/ERS/JRS/ALAT Clinical Practice Guideline," *Am. J. Respir. Crit. Care Med.*, vol. 198, no. 5, pp. e44–e68, Sep. 2018, doi: 10.1164/rccm.201807-1255ST.

[83] R. J. Hallifax, A. Talwar, J. M. Wrightson, A. Edey, and F. V. Gleeson, "State-of-the-art: Radiological investigation of pleural disease," *Respir. Med.*, vol. 124, pp. 88–99, Mar. 2017, doi: 10.1016/j.rmed.2017.02.013.

[84] A. Saito *et al.*, "Pleural thickening on screening chest X-rays: a single institutional study," *Respir. Res.*, vol. 20, no. 1, p. 138, Dec. 2019, doi: 10.1186/s12931-019-1116-9.

[85] A. Tejpal and K. Kaur, "Machine Learning Survey," *Int. J. Comput. Sci. Eng.*, vol. 7, no. 2, pp. 453–457, Feb. 2019, doi: 10.26438/ijcse/v7i2.453457.

[86] H. Al-Sahaf *et al.*, “A survey on evolutionary machine learning,” *J. R. Soc. N. Z.*, vol. 49, no. 2, pp. 205–228, Apr. 2019, doi: 10.1080/03036758.2019.1609052.

[87] R. Elshawi, M. Maher, and S. Sakr, “Automated Machine Learning: State-of-The-Art and Open Challenges,” *ArXiv190602287 Cs Stat*, Jun. 2019, Accessed: Oct. 04, 2020. [Online]. Available: http://arxiv.org/abs/1906.02287.

[88] Q. Zhang, H. Yu, M. Barbiero, B. Wang, and M. Gu, “Artificial neural networks enabled by nanophotonics,” *Light Sci. Appl.*, vol. 8, no. 1, p. 42, Dec. 2019, doi: 10.1038/s41377-019-0151-0.

[89] J. Tang *et al.*, “Bridging Biological and Artificial Neural Networks with Emerging Neuromorphic Devices: Fundamentals, Progress, and Challenges,” *Adv. Mater.*, vol. 31, no. 49, p. 1902761, Dec. 2019, doi: 10.1002/adma.201902761.

[90] X. Fan and H. Markram, “A Brief History of Simulation Neuroscience,” *Front. Neuroinformatics*, vol. 13, p. 32, May 2019, doi: 10.3389/fninf.2019.00032.

[91] A. Y. Abyaneh, A. H. G. Foumani, and V. Pourahmadi, “Deep Neural Networks Meet CSI-Based Authentication,” *ArXiv181204715 Cs Eess Stat*, Nov. 2018, Accessed: Oct. 04, 2020. [Online]. Available: http://arxiv.org/abs/1812.04715.

[92] N. Kriegeskorte and T. Golan, “Neural network models and deep learning – a primer for biologists,” p. 14, 2019.

[93] R. M. Cichy and D. Kaiser, “Deep Neural Networks as Scientific Models,” *Trends Cogn. Sci.*, vol. 23, no. 4, pp. 305–317, Apr. 2019, doi: 10.1016/j.tics.2019.01.009.

[94] M. R. Tanhatalab, H. Yousefi, H. M. Hosseini, M. M. Bonab, V. Fakharian, and H. Abarghouei, “Deep RAN: A Scalable Data-driven platform to Detect Anomalies in Live Cellular Network Using Recurrent Convolutional Neural Network,” in *2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, Herlany, Slovakia, Jan. 2020, pp. 269–274, doi: 10.1109/SAMI48414.2020.9108742.

[95] J. Singh and R. Banerjee, “A Study on Single and Multi-layer Perceptron Neural Network,” in *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, Erode, India, Mar. 2019, pp. 35–40, doi: 10.1109/ICCMC.2019.8819775.

[96] B. T. Pham, M. D. Nguyen, K.-T. T. Bui, I. Prakash, K. Chapi, and D. T. Bui, “A novel artificial intelligence approach based on Multi-layer Perceptron Neural Network and Biogeography-based Optimization for predicting coefficient of consolidation of soil,” *CATENA*, vol. 173, pp. 302–311, Feb. 2019, doi: 10.1016/j.catena.2018.10.004.

[97] S. Abirami and M. Thilagavathi, “Classification of Fruit Diseases using Feed Forward Back Propagation Neural Network,” in *2019 International Conference on Communication and Signal Processing (ICCSP)*, Chennai, India, Apr. 2019, pp. 0765–0768, doi: 10.1109/ICCSP.2019.8698071.

[98] H. El-Amir and M. Hamdy, *Deep Learning Pipeline: Building a Deep Learning Model with TensorFlow*. Berkeley, CA: Apress, 2020.

[99] J. Dean, “1.1 The Deep Learning Revolution and Its Implications for Computer Architecture and Chip Design,” in *2020 IEEE International Solid- State Circuits Conference - (ISSCC)*, San Francisco, CA, USA, Feb. 2020, pp. 8–14, doi: 10.1109/ISSCC19947.2020.9063049.

[100] M. Z. Alom *et al.*, “A State-of-the-Art Survey on Deep Learning Theory and Architectures,” *Electronics*, vol. 8, no. 3, p. 292, Mar. 2019, doi: 10.3390/electronics8030292.

[101] M. A. Wani, F. A. Bhat, S. Afzal, and A. I. Khan, *Advances in Deep Learning*, vol. 57. Singapore: Springer Singapore, 2020.

[102] J. M. Johnson and T. M. Khoshgoftaar, “Survey on deep learning with class imbalance,” *J. Big Data*, vol. 6, no. 1, p. 27, Dec. 2019, doi: 10.1186/s40537-019-0192-5.

[103] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, “Deep learning and its applications to machine health monitoring,” *Mech. Syst. Signal Process.*, vol. 115, pp. 213–237, Jan. 2019, doi: 10.1016/j.ymssp.2018.05.050.

[104] J. Li, W. Wu, D. Xue, and P. Gao, “Multi-Source Deep Transfer Neural Network Algorithm,” *Sensors*, vol. 19, no. 18, p. 3992, Sep. 2019, doi: 10.3390/s19183992.

[105] W. Murphy *et al.*, “Using transfer learning for a deep learning model observer,” Mar. 2019, vol. 10952, doi: 10.1117/12.2511750.

[106] S. Lu, Z. Lu, and Y.-D. Zhang, “Pathological brain detection based on AlexNet and transfer learning,” *J. Comput. Sci.*, vol. 30, pp. 41–47, Jan. 2019, doi: 10.1016/j.jocs.2018.11.008.

[107] S. M. Salaken, A. Khosravi, T. Nguyen, and S. Nahavandi, “Seeded transfer learning for regression problems with deep learning,” *Expert Syst. Appl.*, vol. 115, pp. 565–577, Jan. 2019, doi: 10.1016/j.eswa.2018.08.041.

[108] Z. Wang, Z. Dai, B. Poczos, and J. Carbonell, “Characterizing and Avoiding Negative Transfer,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 11285–11294, doi: 10.1109/CVPR.2019.01155.

[109] S. Chen, K. Ma, and Y. Zheng, “Med3D: Transfer Learning for 3D Medical Image Analysis,” *ArXiv190400625 Cs*, Jul. 2019, Accessed: Oct. 07, 2020. [Online]. Available: http://arxiv.org/abs/1904.00625.

[110] H. Suryanto, C. Guan, and A. Voumard, “Transfer Learning in Credit Risk,” p. 16, Nov. 2019.

[111] M. Ahsan, R. Gomes, and A. Denton, “Application of a Convolutional Neural Network using transfer learning for tuberculosis detection,” in *2019 IEEE International Conference on Electro Information Technology (EIT)*, Brookings, SD, USA, May 2019, pp. 427–433, doi: 10.1109/EIT.2019.8833768.

[112] R. S. Kute, V. Vyas, and A. Anuse, “Component-based face recognition under transfer learning for forensic applications,” *Inf. Sci.*, vol. 476, pp. 176–191, Feb. 2019, doi: 10.1016/j.ins.2018.10.014.

[113] Q. Yu, X. Che, Y. Yang, and L. Wang, “A Transfer Learning Based Interpretable User Experience Model on Small Samples,” in *2019 IEEE 19th International Conference on Software Quality, Reliability and Security (QRS)*, Sofia, Bulgaria, Jul. 2019, pp. 186–196, doi: 10.1109/QRS.2019.00035.

[114] M.-T. Nguyen *et al.*, “Transfer Learning for Information Extraction with Limited Data,” *ArXiv200303064 Cs*, Jun. 2020, Accessed: Oct. 07, 2020. [Online]. Available: http://arxiv.org/abs/2003.03064.

[115] C. Cai *et al.*, “Transfer Learning for Drug Discovery,” *J. Med. Chem.*, vol. 63, no. 16, pp. 8683–8694, Aug. 2020, doi: 10.1021/acs.jmedchem.9b02147.

[116] L. Vu, Q. U. Nguyen, D. N. Nguyen, D. T. Hoang, and E. Dutkiewicz, “Deep Transfer Learning for IoT Attack Detection,” *IEEE Access*, vol. 8, pp. 107335–107344, 2020, doi: 10.1109/ACCESS.2020.3000476.

[117] L. Wen, L. Gao, and X. Li, “A New Deep Transfer Learning Based on Sparse Auto-Encoder for Fault Diagnosis,” *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 49, no. 1, pp. 136–144, Jan. 2019, doi: 10.1109/TSMC.2017.2754287.

[118] A. Sufian, A. Ghosh, A. S. Sadiq, and F. Smarandache, “A Survey on Deep Transfer Learning and Edge Computing for Mitigating the COVID-19 Pandemic,” preprint, Jul. 2020. doi: 10.36227/techrxiv.12535010.v3.

[119] J.-H. Han, D.-J. Choi, S.-U. Park, and S.-K. Hong, “Hyperparameter Optimization for Multi-Layer Data Input Using Genetic Algorithm,” in *2020 IEEE 7th International Conference on Industrial Engineering and Applications (ICIEA)*, Bangkok, Thailand, Apr. 2020, pp. 701–704, doi: 10.1109/ICIEA49774.2020.9101973.

[120] L. Yang and A. Shami, “On hyperparameter optimization of machine learning algorithms: Theory and practice,” *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020, doi: 10.1016/j.neucom.2020.07.061.

[121] X. Dong, J. Shen, W. Wang, L. Shao, H. Ling, and F. Porikli, “Dynamical Hyperparameter Optimization via Deep Reinforcement Learning in Tracking,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2019, doi: 10.1109/TPAMI.2019.2956703.

[122] Y. Yoo, “Hyperparameter optimization of deep neural network using univariate dynamic encoding algorithm for searches,” *Knowl.-Based Syst.*, vol. 178, pp. 74–83, Aug. 2019, doi: 10.1016/j.knosys.2019.04.019.

[123] J. Park, D. Yi, and S. Ji, “A Novel Learning Rate Schedule in Optimization for Neural Networks and It’s Convergence,” p. 16, 2020.

[124] A. Bakambekova and A. P. James, “Deep Learning Theory Simplified,” in *Deep Learning Classifiers with Memristive Networks*, vol. 14, A. P. James, Ed. Cham: Springer International Publishing, 2020, pp. 41–55.

[125] A. Shrestha and A. Mahmood, “Review of Deep Learning Algorithms and Architectures,” *IEEE Access*, vol. 7, pp. 53040–53065, 2019, doi: 10.1109/ACCESS.2019.2912200.

[126] A. V. Joshi, *Machine Learning and Artificial Intelligence*. Cham: Springer International Publishing, 2020.

[127] X. Ying, “An Overview of Overfitting and its Solutions,” *J. Phys. Conf. Ser.*, vol. 1168, p. 022022, Feb. 2019, doi: 10.1088/1742-6596/1168/2/022022.

[128] C. Shorten and T. M. Khoshgoftaar, “A survey on Image Data Augmentation for Deep Learning,” *J. Big Data*, vol. 6, no. 1, p. 60, Dec. 2019, doi: 10.1186/s40537-019-0197-0.

[129] R. Moradi, R. Berangi, and B. Minaei, “A survey of regularization strategies for deep models,” *Artif. Intell. Rev.*, vol. 53, no. 6, pp. 3947–3986, Aug. 2020, doi: 10.1007/s10462-019-09784-7.

[130] S. Maharjan, A. Alsadoon, P. W. C. Prasad, T. Al-Dalain, and O. H. Alsadoon, "A novel enhanced softmax loss function for brain tumour detection using deep learning," *J. Neurosci. Methods*, vol. 330, p. 108520, Jan. 2020, doi: 10.1016/j.jneumeth.2019.108520.

[131] Y. Chen, H. Wang, Q. Li, and D. Huang, "Data Driven Regularization for Convolutional Neural Networks on Image Classification," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, Sapporo, Japan, May 2019, pp. 1–5, doi: 10.1109/ISCAS.2019.8702224.

[132] S. H. Khan, M. Hayat, and F. Porikli, "Regularization of deep neural networks with spectral dropout," *Neural Netw.*, vol. 110, pp. 82–90, Feb. 2019, doi: 10.1016/j.neunet.2018.09.009.

[133] Q. Xu, M. Zhang, Z. Gu, and G. Pan, "Overfitting remedy by sparsifying regularization on fully-connected layers of CNNs," *Neurocomputing*, vol. 328, pp. 69–74, Feb. 2019, doi: 10.1016/j.neucom.2018.03.080.

[134] N. Lu, T. Zhang, G. Niu, and M. Sugiyama, "Mitigating Overfitting in Supervised Classification from Two Unlabeled Datasets: A Consistent Risk Correction Approach," p. 10.

[135] L. Qian, L. Hu, L. Zhao, T. Wang, and R. Jiang, "Sequence-dropout Block for Reducing Overfitting Problem in Image Classification," *IEEE Access*, vol. 4, p. 12, 2016.

[136] M. Hassaballah and K. M. Hosny, Eds., *Recent Advances in Computer Vision: Theories and Applications*, vol. 804. Cham: Springer International Publishing, 2019.

[137] M. R. Ibrahim, "Understanding cities with machine eyes_ A review of deep computer vision in urban analytics," p. 13, 2020.

[138] X. Feng, "Computer vision algorithms and hardware implementations: A survey," p. 12, 2019, doi: https://doi.org/10.1016/j.vlsi.2019.07.005.

[139] S. S. Chouhan, U. P. Singh, and S. Jain, "Applications of Computer Vision in Plant Pathology: A Survey," *Arch. Comput. Methods Eng.*, vol. 27, no. 2, pp. 611–632, Apr. 2020, doi: 10.1007/s11831-019-09324-0.

[140] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1D Convolutional Neural Networks and Applications – A Survey," p. 21.

[141] W. Wang and Y. Yang, "Development of convolutional neural network and its application in image classification: a survey," *Opt. Eng.*, vol. 58, no. 04, p. 1, Apr. 2019, doi: 10.1117/1.OE.58.4.040901.

[142] Z. Li, W. Yang, S. Peng, and F. Liu, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," p. 21, Apr. 2020.

[143] J. Teuwen and N. Moriakov, "Convolutional neural networks," in *Handbook of Medical Image Computing and Computer Assisted Intervention*, Elsevier, 2020, pp. 481–501.

[144] Y.-W. Chen and L. C. Jain, Eds., *Deep Learning in Healthcare: Paradigms and Applications*, vol. 171. Cham: Springer International Publishing, 2020.

[145] W. Pedrycz and S.-M. Chen, Eds., *Development and Analysis of Deep Learning Architectures*, vol. 867. Cham: Springer International Publishing, 2020.

[146] W. H. Lopez Pinaya, S. Vieira, R. Garcia-Dias, and A. Mechelli, “Convolutional neural networks,” in *Machine Learning*, Elsevier, 2020, pp. 173–191.

[147] Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, and M. S. Nasrin, “The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches,” p. 39.

[148] I. Strumberger, E. Tuba, N. Bacanin, R. Jovanovic, and M. Tuba, “Convolutional Neural Network Architecture Design by the Tree Growth Algorithm Framework,” in *2019 International Joint Conference on Neural Networks (IJCNN)*, Budapest, Hungary, Jul. 2019, pp. 1–8, doi: 10.1109/IJCNN.2019.8851755.

[149] W. Rawat and Z. Wang, “Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review,” *Neural Comput.*, vol. 29, no. 9, pp. 2352–2449, Sep. 2017, doi: 10.1162/neco_a_00990.

[150] Z. Li, W. Yang, S. Peng, and F. Liu, “A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects,” p. 21, Apr. 2020.

[151] A. Khan, “A survey of the recent architectures of deep convolutional neural networks,” p. 62, Jan. 2019.

[152] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” *ArXiv190511946 Cs Stat*, Sep. 2020, Accessed: Oct. 14, 2020. [Online]. Available: http://arxiv.org/abs/1905.11946.

[153] H. Zhang *et al.*, “Deep Learning Based Drug Screening for Novel Coronavirus 2019-nCov,” *Interdiscip. Sci. Comput. Life Sci.*, vol. 12, no. 3, pp. 368–376, Sep. 2020, doi: 10.1007/s12539-020-00376-6.

[154] T. Majeed, R. Rashid, D. Ali, and A. Asaad, “Problems of Deploying CNN Transfer Learning to Detect COVID-19 from Chest X-rays,” Radiology and Imaging, preprint, May 2020. doi: 10.1101/2020.05.12.20098954.

[155] K. Kallianos *et al.*, “How far have we come? Artificial intelligence for chest radiograph interpretation,” *Clin. Radiol.*, vol. 74, no. 5, Art. no. 5, May 2019, doi: 10.1016/j.crad.2018.12.015.

[156] S. M. McKinney *et al.*, “International evaluation of an AI system for breast cancer screening,” *Nature*, vol. 577, no. 7788, pp. 89–94, Jan. 2020, doi: 10.1038/s41586-019-1799-6.

[157] Y. Liu *et al.*, “Artificial Intelligence–Based Breast Cancer Nodal Metastasis Detection: Insights Into the Black Box for Pathologists,” *Arch. Pathol. Lab. Med.*, vol. 143, no. 7, pp. 859–868, Jul. 2019, doi: 10.5858/arpa.2018-0147-OA.

[158] L.-Q. Zhou *et al.*, “Artificial intelligence in medical imaging of the liver,” *World J. Gastroenterol.*, vol. 25, no. 6, pp. 672–682, Feb. 2019, doi: 10.3748/wjg.v25.i6.672.

[159] A. Rajkomar, J. Dean, and I. Kohane, “Machine Learning in Medicine,” *N. Engl. J. Med.*, vol. 380, no. 14, pp. 1347–1358, Apr. 2019, doi: 10.1056/NEJMra1814259.

[160] A. Esteva *et al.*, “A guide to deep learning in healthcare,” *Nat. Med.*, vol. 25, no. 1, pp. 24–29, Jan. 2019, doi: 10.1038/s41591-018-0316-z.

[161] N. Tomašev *et al.*, “A clinically applicable approach to continuous prediction of future acute kidney injury,” *Nature*, vol. 572, no. 7767, pp. 116–119, Aug. 2019, doi: 10.1038/s41586-019-1390-1.

[162] F. Wang, L. P. Casalino, and D. Khullar, “Deep Learning in Medicine—Promise, Progress, and Challenges,” *JAMA Intern. Med.*, vol. 179, no. 3, p. 293, Mar. 2019, doi: 10.1001/jamainternmed.2018.7117.

[163] A. Bohr and K. Memarzadeh, “The rise of artificial intelligence in healthcare applications,” in *Artificial Intelligence in Healthcare*, Elsevier, 2020, pp. 25–60.

[164] C. Blease, T. J. Kaptchuk, M. H. Bernstein, K. D. Mandl, J. D. Halamka, and C. M. DesRoches, “Artificial Intelligence and the Future of Primary Care: Exploratory Qualitative Study of UK General Practitioners’ Views,” *J. Med. Internet Res.*, vol. 21, no. 3, p. e12802, Mar. 2019, doi: 10.2196/12802.

[165] A. Becker, “Artificial intelligence in medicine: What is it doing for us today?,” *Health Policy Technol.*, vol. 8, no. 2, pp. 198–205, Jun. 2019, doi: 10.1016/j.hlpt.2019.03.004.

[166] L. Filipović-Grčić, “Artificial intelligence in radiology,” *Rad Hrvat. Akad. Znan. Umjet. Med. Znan.*, vol. 537, no. 46–47, pp. 55–59, 2019, doi: 10.21857/y26kec3o79.

[167] S. Waite *et al.*, “A Review of Perceptual Expertise in Radiology-How it develops, How we can test it, and Why humans still matter in the era of Artificial Intelligence,” *Acad. Radiol.*, vol. 27, no. 1, pp. 26–38, Jan. 2020, doi: 10.1016/j.acra.2019.08.018.

[168] Y. Xue *et al.*, “Multimodal Recurrent Model with Attention for Automated Radiology Report Generation,” in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, vol. 11070, A. F. Frangi, J. A. Schnabel, C. Davatzikos, C. Alberola-López, and G. Fichtinger, Eds. Cham: Springer International Publishing, 2018, pp. 457–466.

[169] M. A. Mazurowski, M. Buda, A. Saha, and M. R. Bashir, “Deep learning in radiology: An overview of the concepts and a survey of the state of the art with focus on MRI,” *J. Magn. Reson. Imaging*, vol. 49, no. 4, pp. 939–954, Apr. 2019, doi: 10.1002/jmri.26534.

[170] J. G. Nam *et al.*, “Development and Validation of Deep Learning–based Automatic Detection Algorithm for Malignant Pulmonary Nodules on Chest Radiographs,” *Radiology*, vol. 290, no. 1, pp. 218–228, Jan. 2019, doi: 10.1148/radiol.2018180237.

[171] O. Stephen, M. Sain, U. J. Maduh, and D.-U. Jeong, “An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare,” *J. Healthc. Eng.*, vol. 2019, pp. 1–7, Mar. 2019, doi: 10.1155/2019/4180949.

[172] R. Soni, R. Singh, S. Shah, and S. Malik, “Chest disease detection from human X-ray scans using Deep Learning,” *Deep Learn.*, p. 12, Feb. 2020.

[173] W. H. Hsu *et al.*, “Development of a Deep Learning Model for Chest X-Ray Screening,” p. 7, 2019.

[174] D. Ueda, A. Shimazaki, and Y. Miki, “Technical and clinical overview of deep learning in radiology,” *Jpn. J. Radiol.*, vol. 37, no. 1, pp. 15–33, Jan. 2019, doi: 10.1007/s11604-018-0795-3.

[175] M. T. Islam, M. A. Aowal, A. T. Minhaz, and K. Ashraf, “Abnormality Detection and Localization in Chest X-Rays using Deep Convolutional Neural Networks,” *ArXiv170509850 Cs*, Sep. 2017, Accessed: Oct. 16, 2020. [Online]. Available: http://arxiv.org/abs/1705.09850.

[176] E. J. Hwang *et al.*, "Deep Learning for Chest Radiograph Diagnosis in the Emergency Department," *Radiology*, vol. 293, no. 3, pp. 573–580, Dec. 2019, doi: 10.1148/radiol.2019191225.

[177] X. Xie, J. Niu, X. Liu, Z. Chen, and S. Tang, "A Survey on Domain Knowledge Powered Deep Learning for Medical Image Analysis," *ArXiv200412150 Cs Eess*, Apr. 2020, Accessed: Oct. 16, 2020. [Online]. Available: http://arxiv.org/abs/2004.12150.

[178] A. Kaushal, R. Altman, and C. Langlotz, "Geographic Distribution of US Cohorts Used to Train Deep Learning Algorithms," *JAMA*, vol. 324, no. 12, p. 1212, Sep. 2020, doi: 10.1001/jama.2020.12067.

[179] D. Bellamy, L. Celi, and A. L. Beam, "Evaluating Progress on Machine Learning for Longitudinal Electronic Healthcare Data," *ArXiv201001149 Cs Stat*, Oct. 2020, Accessed: Oct. 17, 2020. [Online]. Available: http://arxiv.org/abs/2010.01149.

[180] A. Esteva *et al.*, "A guide to deep learning in healthcare," *Nat. Med.*, vol. 25, no. 1, pp. 24–29, Jan. 2019, doi: 10.1038/s41591-018-0316-z.

[181] A. Anikwue and B. Kabaso, "Probabilistic Programming and Big Data," in *2019 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)*, Winterton, South Africa, Aug. 2019, pp. 1–6, doi: 10.1109/ICABCD.2019.8851053.

[182] Z. Ghahramani, "Probabilistic machine learning and artificial intelligence," *Nature*, vol. 521, no. 7553, pp. 452–459, May 2015, doi: 10.1038/nature14541.

[183] W. Lee, H. Yu, X. Rival, and H. Yang, "Towards verified stochastic variational inference for probabilistic programs," *Proc. ACM Program. Lang.*, vol. 4, no. POPL, pp. 1–33, Jan. 2020, doi: 10.1145/3371084.

[184] M. W. Dusenberry *et al.*, "Efficient and Scalable Bayesian Neural Nets with Rank-1 Factors," *ArXiv200507186 Cs Stat*, Aug. 2020, Accessed: Oct. 20, 2020. [Online]. Available: http://arxiv.org/abs/2005.07186.

[185] L. V. Jospin, W. Buntine, F. Boussaid, H. Laga, and M. Bennamoun, "Hands-on Bayesian Neural Networks -- a Tutorial for Deep Learning Users," *ArXiv200706823 Cs Stat*, Jul. 2020, Accessed: Oct. 20, 2020. [Online]. Available: http://arxiv.org/abs/2007.06823.

[186] H. Wang and D.-Y. Yeung, "A Survey on Bayesian Deep Learning," *ArXiv160401662 Cs Stat*, Jul. 2020, Accessed: Oct. 20, 2020. [Online]. Available: http://arxiv.org/abs/1604.01662.

[187] R. Singh, J. Ling, and F. Doshi-Velez, "Structured Variational Autoencoders for the Beta-Bernoulli Process," p. 9, 2017.

[188] C. Tang, N. Luktarhan, and Y. Zhao, "An Efficient Intrusion Detection Method Based on LightGBM and Autoencoder," *Symmetry*, vol. 12, no. 9, p. 1458, Sep. 2020, doi: 10.3390/sym12091458.

[189] Y. Sun, G. G. Yen, and Z. Yi, "Evolving Unsupervised Deep Neural Networks for Learning Meaningful Representations," *IEEE Trans. Evol. Comput.*, vol. 23, no. 1, pp. 89–103, Feb. 2019, doi: 10.1109/TEVC.2018.2808689.

[190] A. Zadeh, Y.-C. Lim, P. P. Liang, and L.-P. Morency, "Variational Auto-Decoder," *ArXiv190300840 Cs Stat*, May 2019, Accessed: Oct. 22, 2020. [Online]. Available: http://arxiv.org/abs/1903.00840.

[191] C. P. Burgess *et al.*, “Understanding disentangling in β-VAE,” *ArXiv180403599 Cs Stat*, Apr. 2018, Accessed: Oct. 22, 2020. [Online]. Available: http://arxiv.org/abs/1804.03599.

[192] H. Sikka, W. Zhong, J. Yin, and C. Pehlevan, “A Closer Look at Disentangling in β-VAE,” *ArXiv191205127 Cs Stat*, Dec. 2019, Accessed: Oct. 23, 2020. [Online]. Available: http://arxiv.org/abs/1912.05127.

[193] K. Xu, A. Srivastava, and C. Sutton, “Variational Russian Roulette for Deep Bayesian Nonparametrics,” p. 10, Oct. 2020.

[194] S. A. Williamson, M. M. Zhang, and P. Damien, “A New Class of Time Dependent Latent Factor Models with Applications,” *ArXiv190408548 Cs Stat*, Apr. 2019, Accessed: Oct. 23, 2020. [Online]. Available: http://arxiv.org/abs/1904.08548.

[195] J. Thevenot, M. B. Lopez, and A. Hadid, “A Survey on Computer Vision for Assistive Medical Diagnosis From Faces,” *IEEE J. Biomed. Health Inform.*, vol. 22, no. 5, pp. 1497–1511, Sep. 2018, doi: 10.1109/JBHI.2017.2754861.

[196] X. Yao, X. Wang, S.-H. Wang, and Y.-D. Zhang, “A comprehensive survey on convolutional neural network in medical image analysis,” *Multimed. Tools Appl.*, Aug. 2020, doi: 10.1007/s11042-020-09634-7.

[197] S. Tiwari, M. C. Trivedi, K. K. Mishra, A. K. Misra, K. K. Kumar, and E. Suryani, Eds., *Smart Innovations in Communication and Computational Sciences: Proceedings of ICSICCS 2020*, vol. 1168. Singapore: Springer Singapore, 2021.

[198] A. Choudhary, L. Tong, Y. Zhu, and M. D. Wang, “Advancing Medical Imaging Informatics by Deep Learning-Based Domain Adaptation,” p. 10, 2020.

[199] R. Ribani and M. Marengoni, “A Survey of Transfer Learning for Convolutional Neural Networks,” in *2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)*, Rio de Janeiro, Brazil, Oct. 2019, pp. 47–57, doi: 10.1109/SIBGRAPI-T.2019.00010.

[200] L. Filipović-Grčić, “Artificial intelligence in radiology,” *Rad Hrvat. Akad. Znan. Umjet. Med. Znan.*, vol. 537, no. 46–47, pp. 55–59, 2019, doi: 10.21857/y26kec3o79.

[201] Q. Waymel, S. Badr, X. Demondion, A. Cotten, and T. Jacques, “Impact of the rise of artificial intelligence in radiology: What do radiologists think?,” *Diagn. Interv. Imaging*, vol. 100, no. 6, pp. 327–336, Jun. 2019, doi: 10.1016/j.diii.2019.03.015.

[202] M. Reyes *et al.*, “On the Interpretability of Artificial Intelligence in Radiology: Challenges and Opportunities,” *Radiol. Artif. Intell.*, vol. 2, no. 3, p. e190043, May 2020, doi: 10.1148/ryai.2020190043.

[203] Y.-D. Zhang *et al.*, “IEEE Access Special Section Editorial: Advanced Signal Processing Methods In Medical Imaging,” *IEEE Access*, vol. 6, pp. 61812–61818, 2018, doi: 10.1109/ACCESS.2018.2875308.

[204] C. E. Kahn, “Artificial Intelligence, Real Radiology,” *Radiol. Artif. Intell.*, vol. 1, no. 1, p. e184001, Jan. 2019, doi: 10.1148/ryai.2019184001.

[205] N. O’Mahony *et al.*, “Deep Learning vs. Traditional Computer Vision,” in *Advances in Computer Vision*, vol. 943, K. Arai and S. Kapoor, Eds. Cham: Springer International Publishing, 2020, pp. 128–144.

[206] F. Pasa, V. Golkov, F. Pfeiffer, D. Cremers, and D. Pfeiffer, "Efficient Deep Network Architectures for Fast Chest X-Ray Tuberculosis Screening and Visualization," *Sci. Rep.*, vol. 9, no. 1, p. 6268, Dec. 2019, doi: 10.1038/s41598-019-42557-4.

[207] A. Sufian, A. Ghosh, A. S. Sadiq, and F. Smarandache, "A Survey on Deep Transfer Learning and Edge Computing for Mitigating the COVID-19 Pandemic," preprint, Jul. 2020. doi: 10.36227/techrxiv.12535010.v3.

[208] Y.-H. Chan, Y.-Z. Zeng, H.-C. Wu, M.-C. Wu, and H.-M. Sun, "Effective Pneumothorax Detection for Chest X-Ray Images Using Local Binary Pattern and Support Vector Machine," *J. Healthc. Eng.*, vol. 2018, pp. 1–11, 2018, doi: 10.1155/2018/2908517.

[209] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," *ArXiv160806993 Cs*, Jan. 2018, Accessed: Dec. 03, 2020. [Online]. Available: http://arxiv.org/abs/1608.06993.

[210] I. M. Baltruschat, H. Nickisch, M. Grass, T. Knopp, and A. Saalbach, "Comparison of Deep Learning Approaches for Multi-Label Chest X-Ray Classification," *Sci. Rep.*, vol. 9, no. 1, p. 6381, Dec. 2019, doi: 10.1038/s41598-019-42294-8.

[211] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning," *ArXiv160207261 Cs*, Aug. 2016, Accessed: Dec. 04, 2020. [Online]. Available: http://arxiv.org/abs/1602.07261.

[212] Z. Tang, Y. Gao, L. Karlinsky, P. Sattigeri, R. Feris, and D. Metaxas, "OnlineAugment: Online Data Augmentation with Less Domain Knowledge," *ArXiv200709271 Cs*, Aug. 2020, Accessed: Dec. 04, 2020. [Online]. Available: http://arxiv.org/abs/2007.09271.

[213] J. F. Ramirez Rochac, N. Zhang, L. Thompson, and T. Oladunni, "A Data Augmentation-Assisted Deep Learning Model for High Dimensional and Highly Imbalanced Hyperspectral Imaging Data," in *2019 9th International Conference on Information Science and Technology (ICIST)*, Hulunbuir, China, Aug. 2019, pp. 362–367, doi: 10.1109/ICIST.2019.8836913.

[214] T. J. Jun, H. M. Nguyen, D. Kang, D. Kim, D. Kim, and Y.-H. Kim, "ECG arrhythmia classification using a 2-D convolutional neural network," *ArXiv180406812 Cs*, Apr. 2018, Accessed: Mar. 17, 2021. [Online]. Available: http://arxiv.org/abs/1804.06812.

[215] B. Sahiner *et al.*, "Deep learning in medical imaging and radiation therapy," *Med. Phys.*, vol. 46, no. 1, pp. e1–e36, Jan. 2019, doi: 10.1002/mp.13264.

[216] J. M. Tomczak and M. Welling, "VAE with a VampPrior," *ArXiv170507120 Cs Stat*, Feb. 2018, Accessed: Nov. 04, 2020. [Online]. Available: http://arxiv.org/abs/1705.07120.

[217] P. Mishra, A. Biancolillo, J. M. Roger, F. Marini, and D. N. Rutledge, "New data preprocessing trends based on ensemble of multiple preprocessing techniques," *TrAC Trends Anal. Chem.*, vol. 132, p. 116045, Nov. 2020, doi: 10.1016/j.trac.2020.116045.

[218] J. T. Hancock and T. M. Khoshgoftaar, "Survey on categorical data for neural networks," *J. Big Data*, vol. 7, no. 1, p. 28, Dec. 2020, doi: 10.1186/s40537-020-00305-w.

[219] D. Singh and B. Singh, “Investigating the impact of data normalization on classification performance,” *Appl. Soft Comput.*, p. 105524, May 2019, doi: 10.1016/j.asoc.2019.105524.

[220] M. Talo, U. B. Baloglu, Ö. Yıldırım, and U. Rajendra Acharya, “Application of deep transfer learning for automated brain abnormality classification using MR images,” *Cogn. Syst. Res.*, vol. 54, pp. 176–188, May 2019, doi: 10.1016/j.cogsys.2018.12.007.

[221] N. Bjorck, C. P. Gomes, B. Selman, and K. Q. Weinberger, “Understanding Batch Normalization,” p. 12, Nov. 2018.

[222] A. C. J. W. Janssens and F. K. Martens, “Reflection on modern methods: Revisiting the area under the ROC Curve,” *Int. J. Epidemiol.*, vol. 49, no. 4, pp. 1397–1403, Aug. 2020, doi: 10.1093/ije/dyz274.

[223] D. Arpit, Y. Zhou, B. U. Kota, and V. Govindaraju, “Normalization Propagation: A Parametric Technique for Removing Internal Covariate Shift in Deep Networks,” p. 9, Jun. 2016.

[224] D. Singh and B. Singh, “Investigating the impact of data normalization on classification performance,” *Appl. Soft Comput.*, vol. 97, p. 105524, Dec. 2020, doi: 10.1016/j.asoc.2019.105524.

[225] “Imgaug,” [Online]. Available: Jung, A.B.; Wada, K.; Crall, J.; Tanaka, S.; Graving, J.; Yadav, S.; Banerjee, J.; Vecsei, G.; Kraft, A.; Borovec, J.; et al. Imgaug. 2020. Available online: https://github.com/aleju/imgaug (accessed on 31 January 2021).

[226] A. Paszke *et al.*, “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” *ArXiv191201703 Cs Stat*, Dec. 2019, Accessed: Nov. 16, 2020. [Online]. Available: http://arxiv.org/abs/1912.01703.

[227] M. N. Gevorkyan, A. V. Demidova, T. S. Demidova, and A. A. Sobolev, “Review and comparative analysis of machine learning libraries for machine learning,” *Discrete Contin. Models Appl. Comput. Sci.*, vol. 27, no. 4, pp. 305–315, Dec. 2019, doi: 10.22363/2658-4670-2019-27-4-305-315.

[228] C. R. Harris *et al.*, “Array programming with NumPy,” *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020, doi: 10.1038/s41586-020-2649-2.

[229] “imgaug,” [Online]. Available: Jung, A.B.; Wada, K.; Crall, J.; Tanaka, S.; Graving, J.; Yadav, S.; Banerjee, J.; Vecsei, G.; Kraft, A.; Borovec, J.; et al. Imgaug. 2020. Available online: https://github.com/aleju/imgaug (accessed on 31 January 2021).

[230] Y. Guan, F. Zhou, and J. Zhou, “Research and Practice of Image Processing Based on Python,” *J. Phys. Conf. Ser.*, vol. 1345, p. 022018, Nov. 2019, doi: 10.1088/1742-6596/1345/2/022018.

[231] S. C. van de Leemput, J. Teuwen, B. van Ginneken, and R. Manniesing, “MemCNN: A Python/PyTorch package for creating memory-efficient invertible neural networks,” *J. Open Source Softw.*, vol. 4, no. 39, p. 1576, Jul. 2019, doi: 10.21105/joss.01576.

[232] J. Jorge, J. Vieco, R. Paredes, J. A. Sanchez, and J. M. Benedı, “Empirical Evaluation of Variational Autoencoders for Data Augmentation,” p. 9, 2018.

[233] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "ChestX-ray8: Hospital-Scale Chest X-Ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases," p. 10.

[234] Z. Wang and R. Martin, "Model-free posterior inference on the area under the receiver operating characteristic curve," *J. Stat. Plan. Inference*, vol. 209, pp. 174–186, Dec. 2020, doi: 10.1016/j.jspi.2020.03.008.

[235] E. Bahn and M. Alber, "On the limitations of the area under the ROC curve for NTCP modelling," *Radiother. Oncol.*, vol. 144, pp. 148–151, Mar. 2020, doi: 10.1016/j.radonc.2019.11.018.

[236] J. Manit, A. Schweikard, and F. Ernst, "Deep convolutional neural network approach for forehead tissue thickness estimation," *Curr. Dir. Biomed. Eng.*, vol. 3, no. 2, pp. 103–107, Sep. 2017, doi: 10.1515/cdbme-2017-0022.

[237] S. Lu, Z. Lu, S. Aok, and L. Graham, "Fruit Classification Based on Six Layer Convolutional Neural Network," in *2018 IEEE 23rd International Conference on Digital Signal Processing (DSP)*, Shanghai, China, Nov. 2018, pp. 1–5, doi: 10.1109/ICDSP.2018.8631562.

[238] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *ArXiv14091556 Cs*, Apr. 2015, Accessed: Mar. 18, 2021. [Online]. Available: http://arxiv.org/abs/1409.1556.

[239] K. Mukherjee, A. Khare, and A. Verma, "A Simple Dynamic Learning Rate Tuning Algorithm For Automated Training of DNNs," *ArXiv191011605 Cs Stat*, Oct. 2019, Accessed: Nov. 20, 2020. [Online]. Available: http://arxiv.org/abs/1910.11605.

[240] C. Criscitiello and N. Boumal, "Efficiently escaping saddle points on manifolds," p. 11, Jun. 2019.

[241] A. Beznosikov, A. Sadiev, and A. Gasnikov, "Gradient-Free Methods for Saddle-Point Problem," *ArXiv200505913 Cs Math*, Sep. 2020, Accessed: Nov. 20, 2020. [Online]. Available: http://arxiv.org/abs/2005.05913.

[242] P. Kumar, M. Grewal, and M. M. Srivastava, "Boosted Cascaded Convnets for Multilabel Classification of Thoracic Diseases in Chest Radiographs," *ArXiv171108760 Cs*, Nov. 2017, Accessed: Nov. 29, 2020. [Online]. Available: http://arxiv.org/abs/1711.08760.

[243] I. Sirazitdinov, M. Kholiavchenko, R. Kuleev, and B. Ibragimov, "Data Augmentation for Chest Pathologies Classification," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, Venice, Italy, Apr. 2019, pp. 1216–1219, doi: 10.1109/ISBI.2019.8759573.

[244] Y. Ma, Q. Zhou, X. Chen, H. Lu, and Y. Zhao, "Multi-attention Network for Thoracic Disease Classification and Localization," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, May 2019, pp. 1378–1382, doi: 10.1109/ICASSP.2019.8682952.

Declaration

I, the undersigned, declare that this thesis is my original work and has not been presented for a degree in any other university, and that all source of materials used for the thesis have been duly acknowledged.

Declared by:

Name:

Signature: _____

Date: _____

Confirmed by advisor:

Name:

Signature: _____

Date: _____