Deep Learning for Automatic Lung Disease Analysis in Chest X-rays

Vom Promotionsausschuss der
Technischen Universität Hamburg

zur Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

genehmigte Dissertation

von
Ivo Matteo Baltruschat

aus
Eckernförde

2021

Vorsitzender des Prüfungsausschusses:
Prof. Dr.-Ing. Rolf-Rainer Grigat

Gutachter:
Prof. Dr.-Ing. Tobias Knopp
PD Dr. rer. nat. habil Michael Grass

Tag der mündlichen Prüfung:
Mittwoch, 05. May 2021

DOI:
https://doi.org/10.15480/882.3511

ORCID von Ivo Matteo Baltruschat:
https://orcid.org/0000-0002-8748-3820

Abstract

Chest X-ray (CXR) imaging is the most common examination type in a radiology department, today. Automatic disease classification can assist the radiologists to reduce workload and to improve the quality of patient care. Medical image analysis has undergone a paradigm shift over the last decade, which is largely due to the tremendous success of convolutional neural networks (CNNs) that achieve superhuman performance in many image classification, segmentation, and quantification tasks. CNNs are being applied to CXR images, but the high spatial resolution, the lack of large datasets with reliable ground truth, and the large variety of diseases are significant research challenges when moving towards application in the clinical environment. Notably, these challenges motivate the novel contributions made throughout this thesis. Systematic evaluation and analysis of four major design decision for CNNs were performed: loss functions, weight initialization, network architectures, and non-image feature integration. To leverage the information such as age, gender, and view position, a novel architecture integrating this information, as well as the learned image representation, was proposed and resulted in state-of-the-art results for the ChestX-ray14 dataset. Furthermore, two advanced image preprocessing techniques were investigated to improve the performance of CNNs: bone suppression—an algorithm to artificially remove the rib cage from CXRs—and automatic lung field cropping—a method to increase the input resolution for CNNs. Both methods combined slightly increased the average results for the OpenI dataset. Finally, a framework is developed to investigate whether CNNs for smart worklist prioritization can optimize the radiology workflow and reduce report turnaround times (RTAT) for critical findings in CXRs. The simulations demonstrate that urgency prioritization with CNNs can reduce the average RTAT for critical findings such as pneumothorax by a factor of two. In conclusion, improvements to specific design decision such as the network architecture, image preprocessing, and training with small datasets for CXR analysis were made. The results were used to demonstrate a significant reduction in the average RTAT for critical findings, which can substantially improve the quality of patient care.

Acknowledgements

This dissertation is dedicated to my grandfather, who always wanted to study but did not have the privilege of growing up in a country without war.

At this point I would like to thank all those who have contributed to the success of this work by their support and cooperation:

To Tobias Knopp for his excellent personal and professional support, for his always open door, for his great confidence in my work and for the freedom that he allowed me to realize my own ideas. His enthusiasm for the chest X-ray project has always been a great incentive for me.

Axel Saalbach's and Hannes Nickisch's invaluable advice and the valuable time they devoted to me and the chest X-ray project. During my doctoral studies I benefited especially from their immense knowledge in the field of machine learning and their continues support. Many ideas of this work have emerged from exciting discussions with them.

To René Werner, Frederic Madesta, Thilo Sentker and Nils Gessert for the excellent cooperation within DAISYlabs, for many valuable discussions, for their enormous commitment to the project and for the very pleasant working atmosphere. I have always enjoyed working in this team, have made enormous professional progress, and will miss the time spent at conferences with them.

For all PhD students, employees, and friends of the Institute of Biomedical Imaging, not only for the cooperation in research, but especially for the nice conversations during breaks.

And – above all – I thank my family for supporting me in pursuing my dreams.

Contents

List of Figures

List of Tables

1 Introduction

In the United Kingdom, the Care Quality Commission has recently reported that—over the preceding 12 months—a total of 26,345 chest X-rays (CXRs) and 2,167 abdomen X-rays have not been formally reviewed by a trained expert radiologist at Queen Alexandra Hospital alone. As a result, three patients with lung cancer have suffered significant harm because their chest X-rays were not properly assessed [Care Quality Commission, 2017].

As a diagnostic tool, medical imaging is one of the most revolutionary advances in medicine in recent decades. By providing a visual representation of the inside of the human body, medical imaging helps radiologists make earlier and more accurate diagnoses. Thus, diseases can be treated more effectively to improve the quality of patient care. Throughout the years, medical imaging has improved in terms of measurement speed, spatial resolution, and contrast. Having this useful tool results in the need for sufficient capacity to have expert radiologists assess the relevant data. We already have situations where there is insufficient capacity to have all X-ray images reviewed by radiologists [Care Quality Commission, 2017; Royal College of Radiologists, 2018]. With the increasing amount of data generated by various medical imaging modalities [Kesner et al., 2018] and the growing world population [United Nations DESA, 2019], it is expected that the demand for expert reading capacity will increase. Among the imaging modalities available in radiology departments, plain radiography is the most common, while chest X-rays are the most frequent examination type [Bundesamt für Strahlenschutz, 2020; NHS England, 2020].

Automatic image analysis tools allow radiologists to significantly reduce their workload and increase the quality of patient care. Earlier methods often combined handcrafted feature representation and classifiers. Unfortunately, developing methods for the feature extraction requires enormous domain expertise and is often a time-consuming process. However, deep learning potentially changes such requirements.

In the year 2012, Krizhevsky et al. [2012] presented AlexNet—a convolutional neural network—for image classification in computer vision and won the ImageNet challenge by a large margin. This was possible due to the increased computing power (i.e., the parallel computing of graphical processing units (GPUs)) and the enormous amount of available data. Such success helped revive neural networks as a method of machine learning, which is a subfield of artificial intelligence (AI). In computer vision, deep learning has already proven its ability to analyze images with superhuman accuracy [He et al., 2016; Simonyan et al., 2015; Szegedy et al., 2014; Tan et al., 2019]. The field of medical image analysis is now intensely exploring deep learning.

The following paragraphs outline the structure of this thesis and provide an overview of each chapter and its contributions. Chapters 2 to 4 summarize the background information and important literature. Then, Chapters 4 through 7 present the research conducted for this thesis. Finally, Chapter 8 concludes this thesis with a summary and outlook for the future.

Chapter 2 briefly introduces medical imaging and its automated analysis. Thereafter, a comprehensive review of chest X-ray analysis with deep learning is presented. As one of the most important enablers of rapid progress in deep learning, open source datasets such as ChestX-ray14 [Wang et al., 2017] and OpenI [Demner-Fushman et al., 2016] are discussed. This is followed by a discussion of the challenges posed by noisy annotation generated by natural language processing (NLP) as well as high-resolution chest X-ray data. Finally, we examine the clinical application of chest X-ray classification in the context of current challenges.

Chapter 3 outlines the historical motivation and chronological progression of neural networks. Their basic element—an artificial neuron—is explained, and different types of activation functions are discussed. Subsequently, the principles of a feed-forward neural network and the differences between classification vs. regression tasks are explained. To calculate the optimal weight parameter changes—and as an updated rule for neural networks—Rumelhart et al. [1986] proposed back-propagation. Finally, this chapter explains how gradient descent is used as an optimization technique for neural networks and outlines significant improvements to this method for the optimization of neural networks.

Chapter 4 describes the major changes to standard feed-forward neural networks that led to deep neural networks and their successful application to high-dimensional

signals—especially in image processing. The basic understanding of convolutional neural networks as hierarchical feature extractors and the application to high-dimensional images are explained. To achieve this, important building blocks of state-of-the-art network architectures (e.g., convolutional, pooling, and normalization layers) are presented. Optimization with gradient descent bears the risk of gradients exploding and vanishing when naively stacking layers in a very deep network. Gradient vanishing is addressed by residual connections and densely connected architectures—both of which allow the stacking of additional layers. Such advanced models typically have millions of parameters to train; therefore, they can easily overfit to the training data. For this reason, data augmentation is often used to artificially enlarge datasets. This also helps to improve the generalizability of a neural network because the model becomes invariant to affine transformations. After training a model, it is important to assess its generalization capability and performance. First, different resampling methods (e.g., k-fold cross-validation or Monte Carlo subsampling) can split a dataset into training-testing subsets, which facilitates generalization assessment. Second, evaluation metrics such as the receiver operating curve and precision-recall curve are used to quantify model performance in disease classification.

Chapter 5 provides insight into different training approaches and their applications to chest X-ray disease classification. Building on prior work in this domain, transfer learning is considered with and without fine-tuning and the training of a dedicated X-ray network from scratch. Due to the high spatial resolution of X-ray data, we propose an adapted ResNet-50 architecture with a larger input size and demonstrate its superior performance when compared to other models [Baltruschat et al., 2019c]. Since radiologists usually include much more information than merely a chest X-ray for their diagnoses, the model architecture is further changed and a novel model is introduced to include non-image features that facilitate patient information acquisition. Finally, the limitations of the ChestX-ray14 dataset are highlighted by analyzing the model with Grad-CAM. These findings motivate the contributions of the following chapters.

Chapter 6 deals with the normalization of chest X-ray data to train on a small dataset (i.e., with only a few thousand samples)—the OpenI dataset [Demner-Fushman et al., 2016]. In addition, the effect of increased input data resolution for neural networks is investigated. Manually-labeled datasets typically have a small sample size—although the OpenI dataset is one of the largest (3,125 images)—which complicates the training of deep neural networks from scratch. As a first preprocessing method, lung field

cropping based on segmentation and bounding box calculation is proposed. This step greatly reduces variation in the appearance of chest X-rays and increases their resolution as an input image, as the factor of downscaling is also reduced. The second method is bone suppression, which can be used to reduce information superposition by removing the bone structure from a chest X-ray. Notably, both methods contribute to improving disease classification performance [Baltruschat et al., 2019e]. Moreover, this chapter outlines the process of annotation generation by expert radiologists for chest X-rays as well as problems related to inter-observer variability [Ittrich et al., 2018; Steinmeister et al., 2019].

Chapter 7 presents the translation of disease classification with deep learning into a specific clinical application. After chest X-rays are acquired, they are usually sorted into a worklist. Depending on the workflow in each radiology department, this worklist is sorted by acquisition time or manual priority labels and, to a large extent, radiologists process their worklist items sequentially. Therefore, the worklist is only processed according to the first-in-fist-out principle. A state-of-the-art classification algorithm for chest X-ray diseases can automatically assign priority labels, which can greatly improve worklist sorting. This chapter presents a novel simulation framework for modeling a clinical workday, which highlights the effects of an automatically prioritized worklist. The framework uses empirical data from the University Medical Center Hamburg-Eppendorf and can simulate a clinical workday, which includes the chest X-ray generation process, the automatic disease classification of chest X-rays, and the time needed for final report generation by a radiologist [Baltruschat et al., 2020b]. Notably, the improved methods proposed in Chapters 5 and 6 for the classification of chest X-ray diseases are used.

Chapter 8 concludes the thesis and its main contributions. It also presents new questions that have arisen from this thesis.

2 Motivation and challenges of lung disease classification

This chapter reviews the current challenges, limitations, and potential of lung disease classification for clinical applications. It begins with a brief introduction to the general concepts of medical imaging and medical image analysis. Conventional radiographic imaging is then explained in relation to chest X-ray. The important factors leading to a paradigm shift in automated image analysis are then outlined. Relevant literature and open source datasets for chest X-ray disease analysis are also presented.

Medical imaging refers to the generation of (two- or three-dimensional) images that non-invasively visualize organs and structures within the human body. As one of the major milestones in 20th-century medical progress, medical imaging has made a fundamental contribution to improving our understanding of human anatomy, physiology, and disease patterns. The evaluation of medical images provides clinicians with an objective basis for the diagnosis of diseases, which has significantly improved the treatment of patients [Heinrich, 2013].

Several medical imaging modalities exist, which can be divided based on two characteristics: projection imaging methods and sectional (tomographic) imaging methods. Projection imaging methods generally have a low cost per examination and a rapid acquisition time. Since only one image is required per examination, image reconstruction is computationally easy to handle. However, projection imaging only produces two-dimensional images. On the other hand, tomographic imaging methods can reconstruct volumetric three-dimensional images, but at the expense of solving a complex mathematical problem and a longer image acquisition time. The algorithms for reconstruction generally have high computational complexity since several measurements are combined. Second, the methods can be divided into non-ionizing and ionizing radiation methods. Magnetic resonance imaging, ultrasound, and magnetic

particle imaging use non-ionizing radiation for image acquisition, which is considered harmless to patients. In contrast, the ionizing radiation employed by conventional radiography (also called X-ray), computed tomography (CT), and positron emission tomography (PET) can cause cell mutation. Nevertheless, advantages such as high spatial resolution, bone structure contrast, and metabolic process visualization by X-ray, CT, and PET, respectively, outweigh the risks. This thesis focuses solely on the projection X-ray images of radiography systems, which represent the most common form of imaging in everyday clinical practice. Notably, Section 2.1 provides a more detailed introduction to this imaging modality.

Deriving clinically useful information for the detection, diagnosis, and treatment of diseases from such images is the main task of radiology. Radiology also includes surgical intervention (e.g., stent placement), where real-time imaging is used to guide radiologists through blood vessels, arteries, and organs to the target internal structures of the body. Increased computational resources and the establishment of medical imaging as a fundamental diagnostic tool have resulted in the emergence of medical image analysis. The goal of medical image analysis is the development of techniques that provide radiologists with relevant information derived from images. These techniques facilitate reproducible, quantitative, and objective assessments of medical scans. Medical image analysis is a useful tool for experts, who typically judge images qualitatively and subjectively.

Medical image analysis can be roughly divided in three major areas.

Image classification: Assigning the correct class of a set of categories to a new image is the process of image classification. In medical imaging, classifying whether or not a pathology is present represents an important task.

Image registration: Aligning two (or more) images to achieve the anatomical correspondence is the process of image registration. In medical imaging, CT and PET can visualize anatomical structures and metabolic information, respectively. To display both scans in an overlay, image registration is required to align both scans.

Image segmentation: Delineating different structures in an image is the process of image segmentation. In medical imaging, the segmentation of different organs, pathologies, or tissue classes is often of great interest. This is useful for further processing, such as measuring the size or describing the shape or texture of organs in medical image.

Medical image analysis has undergone a paradigm shift over the last decade, which is

largely due to the tremendous success of deep learning methods that achieve super-human performance in many tasks. This thesis focuses on the automated analysis of chest X-ray with deep learning. Section 2.2 presents a brief introduction to medical image analysis and provides a comprehensive literature review on chest X-ray analysis. Additionally, Section 2.4 outlines the main methodological challenges for automated chest X-ray analysis with deep learning and its potential clinical applications, which motivate the novel contributions of the present thesis. Chapters 3 and 4 introduce the concept of artificial neural networks for image processing, while Chapters 5 to 7 discuss these methods in greater detail.

2.1 Conventional radiography imaging

In 1895, Wilhelm Röntgen discovered X-rays and was the first to take a two-dimensional X-ray image of a human body part (see Figure 2.1 (a)). This discovery started a new era in medical imaging, which has rapidly evolved into the most common examination type today [Bundesamt für Strahlenschutz, 2020; NHS England, 2020]. Conventional radiography is a two-dimensional projection imaging technique that involves projecting an object onto a detector. The X-ray tube generates X-radiation, which passes through objects. The intensity of X-radiation is scattered or attenuated depending on the different densities and attenuation coefficients of materials (i.e., bones, tissues, and fluids).

Today, most radiography systems use a digital X-ray detector to convert X-radiation into an image. A typical detector has an active image area of 34.48 cm × 42.12 cm, and the resulting image has a matrix size of 2330 × 2846 pixels and a bit depth of 14 bits [Philips Healthcare, 2020]. Digital detectors can be separated into direct and indirect radiation conversion groups, while the latter is more common. A digital detector with direct conversion directly converts the absorbed X-rays into electric current. In contrast, indirect conversion uses a scintillator layer to convert the X-radiation into light. Photodiodes then capture the light for the final conversion into an electric current. Digital X-ray images are usually displayed as inverted, where a high signal (i.e., low X- radiation absorption) appears black and a low signal (i.e., high X-radiation absorption) appears white (see Figure 2.1 (b)).

Compared to other imaging techniques, the benefits of conventional radiogra-

(a) One of the first X-rays **(b)** Modern medical X-ray

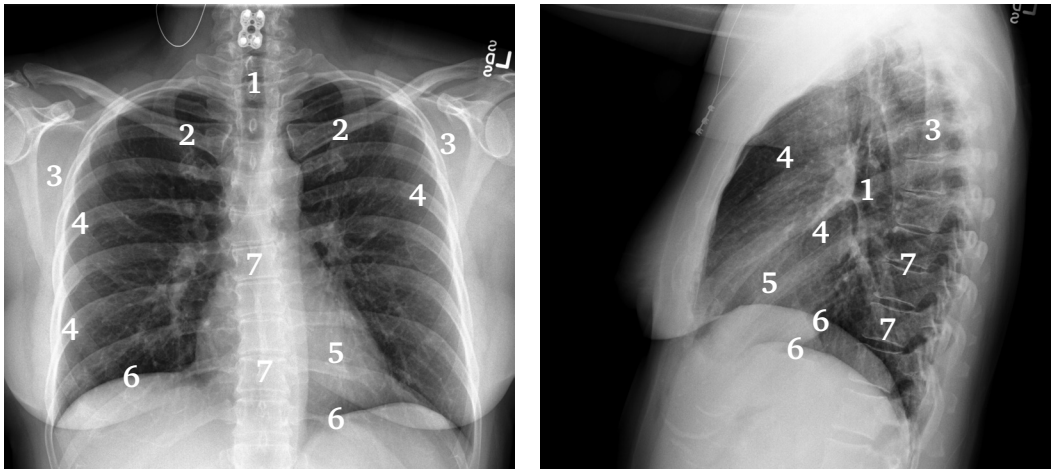
Figure 2.1: One of the first X-rays by Wilhelm Röntgen of Anna Bertha Ludwig's hand [Image source: http://www.zeno.org/nid/20001894587; public domain] (a) and a modern medical X-ray (acquired after an accident) of Ivo Matteo Baltruschat's hand, captured by a Philips DigitalDiagnost (b).

phy include its rapid examination time, high spatial resolution (commonly up to 3.4 line pairs/mm [Philips Healthcare, 2020]), relative lack of artifacts (e.g., motion or reconstruction artifacts), and low cost per image. Additionally, intensive care units can use mobile radiography systems to acquire X-ray images without the need to move patients. The large variety of applications for different body parts and pathologies make conventional radiography the most impotent imaging modalities in medicine. A more detailed introduction to medical imaging is presented in [Van Metter et al., 2000], which also provides a good overview of the physics of different imaging modalities.

Chest X-ray: Daffner [1999] called the chest X-ray a "mirror of health and disease". Twenty-one years later, chest X-ray is the most common examination type in radiology departments [Bundesamt für Strahlenschutz, 2020; NHS England, 2020] and the statement of Daffner [1999] remains true. In [Brant et al., 2007; Lange et al., 2007], and [Darby et al., 2012], the fundamentals of chest X-ray interpretation and diseases are presented. The following provides a short introduction to frequently used terminology in chest radiography as well as an overview of the anatomical structure in a chest X-ray.

Chest X-rays are commonly named based on how the radiation beam passes through the patient. They can be roughly divided into three projection types: posteroanterior (PA), lateral, and anteroposterior (AP). PA and lateral are the basic examinations of the thorax (see Figures 2.2 (a) and 2.2 (b), respectively). For a PA examination, the patient stands upright, positions the front (anterior) of his chest against the detector, and places his hands on his hips or the handles of the device. Thus, the radiation beam passes through the back (posterior) to the anterior portion of the patient's chest. A lateral examination is made while the patient stands with his left side against the detector and arms raised [Lange et al., 2007]. AP examinations are typically used for patients who cannot stand or are bedridden. In contrast to PA, the patient positions his posterior chest against the detector (i.e., the radiation beam passes through the anterior to the posterior portion of the patient's chest). This positioning leads to a magnification of internal structures in the X-ray image since the distance between organs and the detector increases.

In a standard PA and lateral chest X-rays (see Figure 2.2 (a) and 2.2 (b), respectively), readers can typically observe the trachea, clavicles, scapulae, ribs, heart, diaphragm, and vertebrae forming the spine. Both X-rays shown in Figure 2.2 are of the same



(a) Frontal PA chest X-ray **(b)** Lateral chest X-ray

Figure 2.2: Typical examination type in which two corresponding X-ray images of the chest are taken from one patient. (a) shows the frontal PA chest X-ray and (b) the lateral chest X-ray. In both X-rays, one can see anatomical structures: (1) trachea, (2) clavicle, (3) scapulae, (4) ribs, (5) heart, (6) diaphragm, and (7) vertebrae forming the spine. Example images are taken from the OpenI dataset [Demner-Fushman et al., 2016]

healthy patient. Notably, lung diseases can significantly alter the appearance of a chest X-ray.

The nature of chest X-ray images—being a projection imaging modality—makes them very difficult to interpret. This is largely due to the overlapping of anatomical structure and diseases. Another problem can be the distinction between visually similar diseases or diseased and healthy structures (i.e., infiltration and normal blood vessel structure within the lungs). After learning the basics of chest X-ray analysis, radiologists typically improve their understanding of chest X-rays and diagnostic skills by viewing a large number of X-rays. Over many years, radiologists learn what the chest X-ray of a healthy patient looks like and compare each new patient with this memorized representation. This is a subjective process that often leads to large interobserver and intraobserver differences when radiologists diagnose chest X-rays [Albaum et al., 1996; Johnson et al., 2010]. Chapter 5 discusses the appearance of common chest X-ray diseases such as pneumothorax or pleural effusion. Based on the effort to create a multi-reader annotated dataset, Sections 2.4.2 and 6.2 discuss the problems of inter- and intraobserver differences.

2.2 Medical image analysis

Due to its high clinical impact and remaining challenges, medical image analysis has become a broad and active area of research in recent decades. Notably, Beutel et al. [2000] provide an introduction to medical image analysis. Moreover, van Ginneken et al. [2001] and van Ginneken et al. [2001] present a comprehensive review of chest X-ray analysis that includes rule-based methodological approaches. However, since these reviews do not cover the most recent methodological changes to deep learning, this section of the thesis provides an overview of recent deep learning methods for chest X-ray analysis.

The literature review is limited to the period from January 2017 to December 2019. Two websites were used to find suitable literature: Arxiv Sanity Pre-server (http://www.arxiv-sanity.com) and Google Scholar (https://scholar.google.com). The following terms were used to search for suitable literature: “X-ray”, “chest”, “lung”, “deep learning”, and “neural network”.

For the literature summarized in Tables 2.1 and 2.2, chest X-ray analysis with deep learning can be divided into four main areas: classification, localization, segmentation, and report generation. Additionally, NLP with neural networks is attracting increasing interest among researchers. NLP offers the possibility to use old reports for chest X-ray analysis by converting them into labels. The existing labels can then be used for the supervised learning (see Section 3.2) of a neural network. Table 2.1 only summarizes work on disease classification, while Table 2.2 groups work presenting methods for other chest X-ray analysis areas. The following paragraphs discuss the tables and then highlight some important work related to this thesis.

For the literature summarized in Table 2.1 and 2.2, chest X-ray analysis with deep learning can be spitted in four main areas: classification, localization, segmentation, and report generation. Additionally, NLP with neural networks is becoming increasingly interesting for researchers. NLP offers the possibility to use old reports for a chest X-ray analysis by converting the report into labels. The existing labels can then be used for supervised learning (see Section 3.5) of a neural network. Table 2.1 only summarizes work on disease classification, while Table 2.2 groups work presenting methods for other chest X-ray analysis areas. First, the tables are discussed, and then some important works for this thesis are highlighted.

Table 2.1: The research papers shown in this table are sorted by their year of publication in descending order. Despite other datasets being released earlier, the ChestX-ray14 dataset is used for training in most of these works. This could be due to older open source datasets having one or two orders of magnitude fewer images (see 2.3 and Table 2.3). Moreover, nearly half of the 19 papers used some form of internal data to either train their network or obtain a clean test dataset. In terms of neural network architecture, most used either ResNet [He et al., 2015a] or DenseNet [Huang et al., 2017]. Notably, both of these architectures have among the most powerful classification networks for the ImageNet challenge [Russakovsky et al., 2014]. Furthermore, only two papers employed the older VGG-19 [Simonyan et al., 2015] architecture. Additionally, 10 of the 19 papers used a model pre-trained on ImageNet.

Table 2.2: This table presents papers on disease classification as well as localization, segmentation, report generation, and NLP. The table is sorted according to tasks in the same order as the previous enumeration to better group the papers. Compared to Table 2.1, 9 of the 20 papers present methods for classification; however, they usually combine their methods with a second task such as localization, segmentation, or report generation. Only three of these nine papers use the ChestX-ray14 dataset, even though they report a classification method. The total number of papers on segmentation, localization, and report generation are eight, seven, and five, respectively, which suggests that all tasks are of similar interest. Among the papers featured in this table, ResNet is the most commonly used neural network architecture, while VGG and the DenseNet are only used twice and once, respectively. Furthermore, 5 of the 20 papers used a model pre-trained on ImageNet.

Important works: Bar et al. [2015] proposed the use of a convolutional neural network trained on natural images as a feature extractor since medical data with annotation were rare at that time. Combining the extracted image features with well-known descriptors such as GIST [Oliva et al., 2001] or bag-of-visual-words (BoVW) [Csurka et al., 2004] has slightly increased model performance when compared to using each feature descriptor on its own.

With the release of the large dataset ChestX-ray14 in 2017, the classification of diseases in X-ray images has gained a lot of attention among researchers. Rajpurkar et al. [2017] have reported that a DenseNet-121 architecture—with no substantial archi-

tectural changes—pre-trained on ImageNet and fine-tuned to ChestX-ray14 can detect pneumonia with a higher F1 score than radiologists. To support this claim, they have compared their method with four radiologists of different experience levels. Furthermore, they have presented results for 13 other pathological findings, which are part of ChestX-ray14 (see Section 5.1). Here, they have reported the commonly used area under the receiver operation curve (AUROC) and achieved superior performance for all 14 findings when compared to two initial works using the same data.

Since the F1 score is the harmonic mean of precision and recall, it changes when the prevalence changes. Unfortunately, their presented materials and results are incomplete, which makes it difficult to verify the F1 results. Since Rajpurkar et al. [2017] have not reported the recall or precision, it is impossible to tell where the differences between the results originate from. Moreover, they have not reported the prevalence of their test data set.

The output of neural networks are typically continues numbers, and Rajpurkar et al. [2017] must use a threshold value to binarize the neural network output (see Section 3.1). Rajpurkar et al. [2017] have reported neither the threshold value nor the precision-recall curve for their neural networks. Chapter 5 discusses these results in greater detail and presents a novel and superior architecture that includes meta-information.

Kim et al. [2018] have presented an approach to reduce the problem of catastrophic forgetting when a neural network is trained sequentially. After the deployment of a neural network, it is often unfeasible to retrain the neural network from scratch when new data becomes available. Hence, neural networks are trained sequentially. To preserve the knowledge gained from training on old data, Kim et al. [2018] added a reconstruction loss to the standard training loss, thereby forcing the latent space to be informative about earlier training stages. Furthermore, they have shown that their method works for both natural images and chest X-ray disease classification.

Table 2.1: Overview of recent literature on chest X-ray for disease classification with deep learning. The results of each paper are not presented because they often cannot be compared to each other. Instead, this table provides information about the dataset, neural network architecture, and some additional notes for deep learning experts. In the architecture columns, we encoded the number of layers by using a specific symbol for each neural network. The "ResNet" column uses "x" for 50, "o" for 101, and "#" for 18 layers. In the "DenseNet" column, "x" means 121 layers. An "x" in the "Pretrained" column indicates that the model was pretrained on ImageNet.

	Dataset									Architecture				
	JSRT	PLCO-Lung	MC	Shenzhen	SCR	OpenI	ChestX-ray14	MIMIC-CXR	PadChest	Inhouse	VGG	ResNet	DenseNet	Pretrained	Additional notes
[Bar et al., 2015]										443				x	Shallow CNN; combining CNN features with GIST and BoVW; three classes
[Rajpurkar et al., 2017]							x			400			x	x	
[Yao et al., 2017]							x						x		RNN for modeling multi-label dependencies
[Ypsilantis et al., 2017]										100k	x	x			Encoder with RAM
[Zech et al., 2018]						x	x			42k			x		Generalization across hospitals
[Ge et al., 2018]							x				x	#	x	x	Two networks; three losses: MSM-loss for label interdependency; bilinear pooling -> fine-grained; CE-loss
[Yan et al., 2018]							x						x	x	SE block (from scratch) + 1x1 conv. before final max-min pooling
[Guendel et al., 2018]		x					x						x	x	High-resolution input + class. of loc. label
[Guan et al., 2018]							x					x		x	Global/Local-net: CAM to generate "weakly" location -> crop image to this area for local; concatenation global/local features
[Laserson et al., 2018]										959k			x		Two networks: concatenation lat. + frontal img. features
[Santeramo et al., 2018]										337k				x	Inception-v3 + RNN for longitudinal detection
[Rubin et al., 2018]								x					x		Two networks: concatenation lat. + frontal img. features
[Putha et al., 2018]										2300k					Company paper without technical information
[Kim et al., 2018]										10.5k					Continual Learning
[Wang et al., 2019]							x					o		x	Grad-CAM attention
[Calli et al., 2019]							x			15k		x		x	Free rejection of out-of-distribution samples
[Baltruschat et al., 2019c]							x					x		x	Architecture including meta-data
[Bertrand et al., 2019]									x				x		Comparison of frontal and lat. classification
[Baltruschat et al., 2019e]						x	x			3125		x			Advanced preprocessing

Table 2.2: Overview of recent literature for chest X-ray analysis with deep learning methods. This table provides information about the specific tasks addressed in the paper as well as the dataset, neural network architecture, and some additional notes. In the architecture columns, we encoded the number of layers by using a symbol for each neural network. The "ResNet" column uses "x" for 50, "o" for 101, and "#" for 18 layers. In the "DenseNet" column, "x" means 121 layers. An "x" in the "Pretrained" column indicates that the model was pretrained on ImageNet.

	Task					Dataset										Architecture				
	Class.	Loc.	Seg.	Rep.-gen.	NLP	JSRT	PLCO-Lung	MC	Shenzhen	SCR	OpenI	ChestX-ray14	MIMIC-CXR	PadChest	Inhouse	VGG	ResNet	DenseNet	Pretrained	Additional notes
[Gooßen et al., 2019b]	x	x	x												1003					Comparison study of MIL, class. and seg.
[Tang et al., 2018]	x	x										x					x			Fine-tuning by severity sorted batches and binary class. + CAM attention
[Yao et al., 2018]	x	x										x								U-Net (adapted) + saliency map generation (weak supervision)
[Islam et al., 2017]	x	x				x			x	x	x					x	x		x	Loc. by black square occlusion
[Pesce et al., 2019]	x	x													305k	o				1x1 conv. attention feedback (loc.) vs. + RAMAF (loc.)
[Imran et al., 2019]	x		x					x	x	x										APPAU-Net: Generator for seg. and discriminator for class.
[Mahapatra et al., 2018]	x		x			x				x					400		#		x	cGAN data augmentation
[Wang et al., 2018]	x			x							x	x			900		x		x	RNN with multi-level saliency attention
[Shin et al., 2016]	x			x							x									GoogLeNet + RNN for context generation
[Datta et al., 2020]					x						x									Short review of papers working with OpenI; NLP with spatial role labeling
[Cai et al., 2018]		x										x					x			Multi-scale aggregation at the end; combining AT with KP
[Xing et al., 2019]		x										x								Pix2Pix-GAN for data augmentation; only augmenting non-disease area
[Chen et al., 2018]			x					x		x							v		x	U-Net; CycleGan + semantic-aware loss for domain adaption
[Hwang et al., 2017]			x					x		x										U-Net with atrous conv.; Two-stage training: 1. rough segmentation 2. Concat. original img. + rough segmentation
[Nishio et al., 2019]			x					x		x					65					U-Net hyperparameter optimization for lung seg.
[Novikov et al., 2018]			x							x										InvertedNet with ELU (U-Net variation)
[Dong et al., 2018]			x							x					221		#			GAN for seg.
[Gasimova, 2019]				x							x						x		x	RNN for report generation
[Harzig et al., 2019]				x							x						o			Two RNNs for normal and abnormal
[Liu et al., 2019]				x							x		x					x		RNN + RNN combined with reinforcement learning

2.3 Open source chest X-ray datasets

In contrast to rule-based methodological approaches based on predefined features or classical machine learning methods, the performance of deep learning algorithms scales with data [Sun et al., 2017]. With ongoing research efforts and an increasing amount of medical data being generated daily, more data should be available for research and clinical application development. Today, open datasets are one of the main factors influencing the rapid progress of research in medical image analysis with deep learning. Hence, we provide a summary of the available chest X-ray datasets in Table 2.3 and also present a list of supplementary annotations to these datasets in Table 2.4.

The first two publicly available chest X-ray datasets were published in 2000: the "JSRT" dataset from Shiraishi et al. [2000] and the "PLCO-Lung" dataset from Team PLCO Project et al. [2000]. Notably, Shiraishi et al. [2000] have released a small dataset with 247 images for lung nodule classification. The chest X-rays are digitalized film images and have an image size of 2048 × 2048 pixels with a 12-bit gray level. The PLCO-Lung dataset is relatively large (236,000 images from 70,000 patients) and has detailed annotation (i.e., location descriptions and the total count for each pathology) for 13 pathologies. The images are provided as TIFF files with an image size of 2500 × 2100 pixels and a 16-bit gray level.

Shortly after this release, Jaeger et al. [2014] provided another two open datasets for tuberculosis (TB) classification. The "Montgomery County" (MC) dataset includes 138 frontal chest X-rays. The images are provided as PNG files, have a 12-bit gray level, and an image size of 4020 × 4892 or 4892 × 4020 pixels. Besides the TB label, segmentation masks for the left/right lung are also provided. The second dataset, "Shenzhen", contains 662 frontal chest X-rays. While these images are also PNG files with a 12-bit gray level, they have an image size of approximately 3000 × 3000 pixels. Furthermore, the Shenzhen dataset only contains labels for TB and no segmentation masks.

In 2016, a new dataset known as "OpenI" was released by Demner-Fushman et al. [2016]—the first dataset to include frontal and corresponding lateral chest X-rays. The OpenI dataset includes 7,702 images from 3,851 patients and their corresponding reports. The images are provided in the standard data DICOM format with no preprocessing. In addition to the reports, annotation labels for image retrieval are

provided based on the Medical Subject Headings (MeSH) vocabulary. Shortly thereafter, the popular "ChestX-ray14" dataset was released by Wang et al. [2017]. At this time, the ChestX-ray14 dataset was one of the largest datasets, with 112,120 images from 30,805 patients. In this dataset, Wang et al. [2017] provide 14 labels, which were automatically generated by applying NLP to the reports. The images are preprocessed to obtain an image size of 1024×1024 pixels and have an 8-bit gray level. The file format is PNG.

In 2019, three more datasets known as "CheXpert", "PadChest", and "MIMIC-CXR-JPG" were released by Irvin et al. [2019], Bustos et al. [2020], and Johnson et al. [2019], respectively. CheXpert and MIMIC-CXR-JPG have the same 14 labels and similar NLP methods were used to generate them. In comparison to ChestX-ray14, MIMIC-CXR-JPG and CheXpert provide a binary label for each finding—"present" (i.e., 1) or "not present" (i.e., 0)—and also include "uncertain/ambiguous language" (i.e., -1) and "missing" (i.e., no mention of the label in the report). Furthermore, all three data sets have nine labels in common. CheXpert includes 224,316 frontal and lateral images from 65,240 patients. The images are preprocessed by a histogram equalization and converted to JPG files with an 8-bit gray level. The image size is unchanged by the preprocessing. The MIMIC-CXR-JPG dataset contains 377,110 images from 64,586 patients with frontal and lateral chest X-rays. Johnson et al. [2019] have used a preprocessing method similar to that of CheXpert. Hence, the images are converted to 8-bit gray level JPG files without altering the original image size. Additionally, the PadChest dataset comprises 160,868 frontal and lateral chest X-rays from 67,625 patients. In contrast to the other datasets, the reports for PadChest were released—instead of only automatically generated labels. The present thesis utilizes the ChestX-ray14 and OpenI datasets. At the time of writing, ChestX-ray14 was the largest available dataset with images selected from the daily routine; therefore, it provides a good basis for the experiments performed in this work. On the other hand, the OpenI dataset (the third largest) is the only one to provide images in DICOM format, which facilitates the use of its own preprocessing steps. Furthermore, the OpenI dataset also provides chest X-rays in two projections: frontal and lateral. Both datasets are discussed in Sections 5.1 and 6.2.

Supplementary annotations have been published for some of the presented open source datasets, which are shown in Table 2.4. Several years after the publication of the JSRT dataset, van Ginneken et al. [2006] provided segmentation masks for the lungs, heart, and clavicles across the entire dataset—known as "Segmentation in

Chest Radiographs" (SCR). Two major competitions have been based on the ChestX-ray14 dataset and provided specific annotations for their tasks. First, the Radiology Society of North America (RSNA) hosted a pneumonia detection competition and released over 30,000 additional annotations with labels and bounding boxes [RSNA, 2020]. Second, the Society for Imaging Informatics in Medicine (SIIM) and the American College of Radiology (ACR) hosted a pneumothorax segmentation competition in 2019 [SIIM, 2019]. They provided pixel-level pneumothorax segmentation masks for 12,047 images.

Since many researchers have pointed out that noisy labels generated by NLP can have a serious impact on the training and testing of neural networks, Majkowska et al. [2020] released 4,376 images from the ChestX-ray14 dataset with annotations by three expert radiologists. However, the original 14 classes were not used for their annotation; instead, only four classes were used: pneumothorax, nodule/mass, airspace opacity, and fracture.

Chapter 5 discusses the problems with NLP-generated labels, especially those related to pneumothorax. To create clean labels with minimal noise as the gold standard, two expert radiologists from the University Medical Center Hamburg-Eppendorf have reannotated the entire OpenI dataset. Section 6.2 discusses the annotation process and presents the results.

Table 2.3: Overview of open source datasets. We include information about the number of patients and images as well as the types of projection and labeling. For the column "Class.", "x" indicates manual labeling, while "o" means natural language processing-generated labels.

	Name	Patiens	Images	Frontal	Lateral	Class.	Bbox.	Seg.	Preproc.	Additional notes
[Shiraishi et al., 2000]	JSRT	247	247	x		x				Nodule
[Team PLCO Project et al., 2000]	PLCO-Lung	70,632	236,447	x		x				13 classes, loc. description + count
[Jaeger et al., 2014]	MC	138	138	x		x		x		Lung mask; tuberculosis
[Jaeger et al., 2014]	Shenzhen	662	662	x		x				Tuberculosis
[Demner-Fushman et al., 2016]	OpenI	3,851	7,702	x	x	x				Eeports; MeSH labels
[Wang et al., 2017]	ChestX-ray14	30,805	112,120	x		o	x		x	14 classes, bboxs only for small subset
[Irvin et al., 2019]	CheXpert	65,240	224,316	x	x	o			x	14 classes, uncertainty label
[Bustos et al., 2020]	PadChest	67,625	160,868	x	x				x	Reports
[Johnson et al., 2019]	MIMIC-CXR-JPG	64,586	377,110	x	x	o			x	14 classes

Table 2.4: Overview of supplements for open source datasets.

	Supplement	Name	Patiens	Images	Class.	Bbox.	Seg.	Additional notes
[van Ginneken et al., 2006]	JSRT	SCR	247	247			x	Lung, heart, clavicles
[RSNA, 2020]	ChestX-ray14	RSNA-Pneu	26,684	26,684	x	x		Pneumonia; 30,227 bbox. annotation
[SIIM, 2019]	ChestX-ray14	SIIM-PTX	5,688	12,047	x		x	Pneumothorax
[Majkowska et al., 2020]	ChestX-ray14	Google-CXR14	1,695	4,376	x			Pneumothorax, nodule/mass, airspace opacity, fracture; Three expert radiologists
Inhouse	OpenI	UKE-OpenI	3,125	6,250	x			Eight classes; two expert radiologists; frontal and lateral images

2.4 Challenges of lung disease classification

The following subsections discuss challenges for lung disease classification with deep learning as well as issues related to its translation into clinical applications. Although the first research on chest X-ray analysis began in the 1960s [Becker et al., 1964], the automatic analysis of chest X-ray images remains a complex problem that has not yet been solved. The supervised training of deep neural networks for lung disease classification has three main problems: mismatch between the small input size of the neural network and the large image size of chest X-rays (i.e., high spatial resolution), the lack of large-scale, annotated, and reliable ground truth data, and the wide variety of diagnoses.

2.4.1 High spatial resolution of image data

Spatial resolution defines the ability of an imaging system to visualize two adjacent structures as distinct from each other. Notably, low spatial resolution can lead to a visual blurring of the image. To measure the resolution of an imaging system, the line spread function and modulation transfer function are used. For the line spread function, a thin line (or slit) of a known spatial size is imaged. Thereafter, the blur degree of this line can be measured as the full width at half maximum. The same measured slit can also be used to calculate the modulation transfer function by calculating the absolute values of the Fourier transformation [Sawant et al., 2007].

Modern chest X-rays today typically have an image size of 2000 pixels to 3000 pixels to the square [Philips Healthcare, 2020] due to their high spatial resolution of 3.4 line pairs/mm and an active image area of 34 cm to 42 cm to the square. This image area is required to fully image the chest, while the high spatial resolution is required by radiologists to distinguish the small details of various lung pathologies [Huda et al., 2015]. For example, a pneumothorax is one of the most critical findings on a chest X-ray and typically requires immediate clinical intervention. Its visual appearance is subtle because the edge of the pleura appears as merely a thin line in a high-resolution X-ray image. Figure 2.3 (a) presents a high-resolution chest X-ray image with the full image size of 2828×2320 pixels, while Figure 2.3 (b) presents a 10x magnification of two highlighted image areas (i.e., blue and red boxes). The yellow arrows point to the

pleura edge, indicating that pneumothorax is present in this example image. Without a high spatial resolution, this edge would be blurred and not visible.

The input size of common convolutional neural networks for image classification in computer vision is approximately 224 pixels to 299 pixels to the square [He et al., 2015a]. To correct the discrepancy between the original image size and the input size, the original image is often downscaled to the input size via bilinear interpolation [Bar et al., 2015; Rajpurkar et al., 2017; Yao et al., 2017]. Such downscaling reduces the spatial resolution and can severely compromise the visibility of important image features (e.g., the pleura edge). Figure 2.3 (c) demonstrates the severe effects of such downscaling. The chest X-ray in Figure 2.3 (a) is downscaled by bilinear interpolation from 2828×2320 pixels to 256×256 pixels (i.e., reducing the width and height by a factor of 11 and 9, respectively). Figure 2.3 (d) presents the same image areas shown in Figure 2.3 (b), but after downscaling. The edge of the pleura is no longer visible, thus making it significantly more difficult to detect the pneumothorax.

To address this problem, a specially adapted ResNet with increased input size is presented in Section 5.2.3. Furthermore, in Section 6.1.2, lung field cropping is proposed as a method to increase the spatial resolution of the input image for the neural network.

2.4.2 Annotation of clinical data

The performance of deep learning methods remains strongly limited by the availability of reliable annotations in the medical domain [Greenspan et al., 2016]. While annotations by individual radiologists from a dataset are desirable, this is time-consuming, costly, and complicated. Moreover, while crowdsourced annotation is common in the computer vision domain, it is not possible to overcome the lack of annotation by using the same method for most medical problems. This method can be used in computer vision because it is easy for adults to recognize objects such as a table, house, or car. However, in the medical field, it is not typically possible for individuals to recognize signs of various diseases in a chest X-ray without possessing a lengthy medical education.

For chest X-rays, another challenge arises when labels for the supervised training of a neural network are created by radiologists. As explained in Section 2.1, a chest X-ray

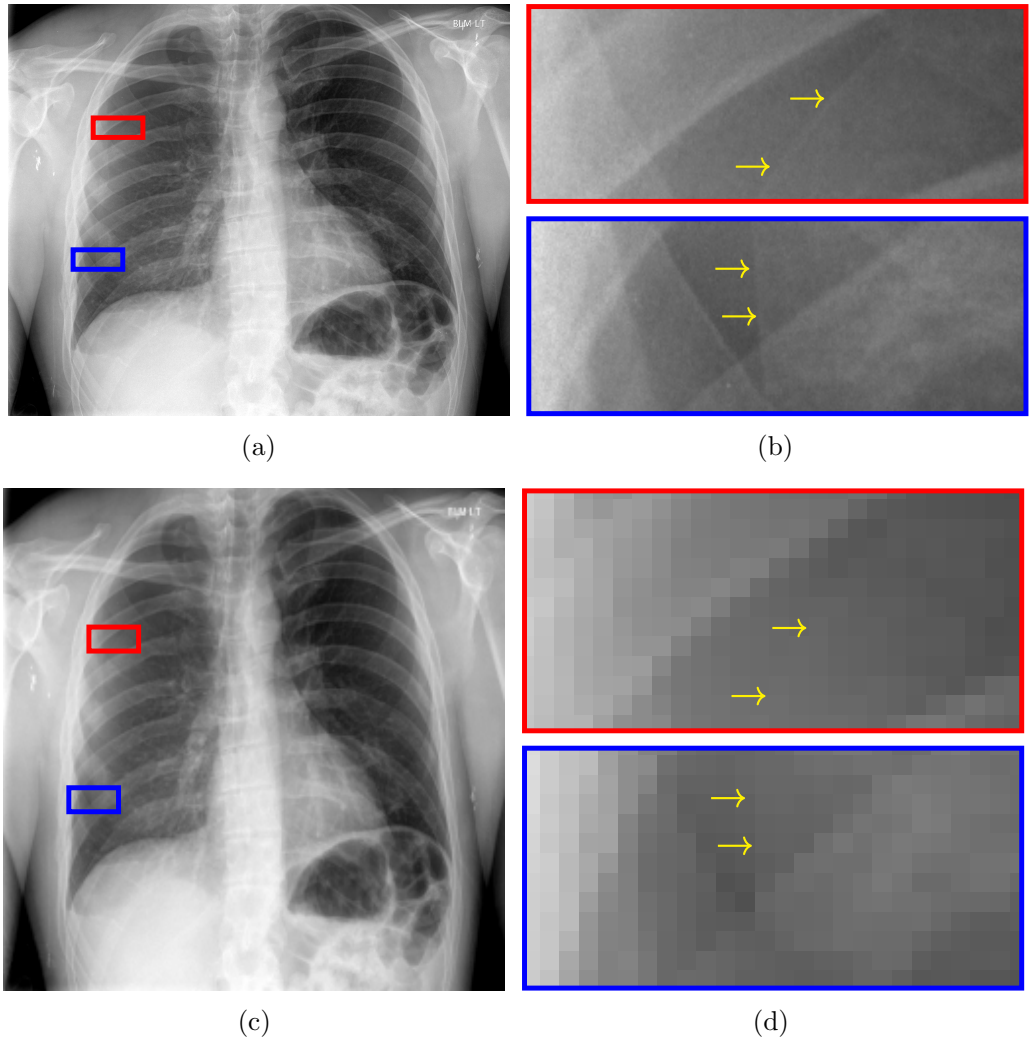


Figure 2.3: Comparison of a high- and low-resolution chest X-ray based on a pneumothorax. (a) shows the original chest X-ray in the full image size of 2828 × 2320 pixels. In (b), two areas of (a) are shown, magnified by a factor of ten. The yellow arrows point to the edge of the pleura, which indicates the pneumothorax. For comparison, (c) shows (a) downscaled by bilinear interpolation to an image size of 256 × 256 pixels. (d) shows the same magnified areas as (b) and the pleura edge is no longer visible. The example image was taken from the OpenI dataset [Demner-Fushman et al., 2016] (ID: 3378).

is a projection image. This implies that three-dimensional information is projected onto two dimensions. Such projections invariably involve the loss of information that cannot be recovered. This information loss also complicates image interpretation for trained radiologists. Moreover, the supervised training of neural networks for classification requires discrete labels (see Section 3.3), such as "pathology present" or "pathology not present". For a radiologist that normally uses descriptive text, it is difficult to make such final decisions. Notably, such decisions often heavily depend on the radiologist. For example, cardiomegaly (i.e., enlarged heart) is defined by the ratio of

the horizontal width of the heart to the maximum width of the lung area. If this ratio is above 0.5, the patient has cardiomegaly. While this seems to be a good criterion for defining the presence of cardiomegaly, it has certain problems that remain unaccounted for. For example, this ratio is highly dependent on the amount of inhalation or the examination type (e.g., AP or PA). Considering this additional information can lead to different results between radiologists since it is often not a binary classification task for them. The interpretation of conventional radiographs (e.g., chest X-rays) is strongly affected by the individual experience and education of the radiologist, which leads to measurable inter- or even intra-rater variability among radiologists [Albaum et al., 1996; Bloomfield et al., 1999; Hopstaken et al., 2004; Johnson et al., 2010; Neuman et al., 2012; Novack et al., 2006; Tudor et al., 1997].

Annotations for public datasets are often obtained by automated report analysis using NLP. Although NLP methods have steadily improved over the last decade, they continue to struggle with the complexity of free-text radiology reports and their inter-institutional transferability remains questionable [Collobert et al., 2011; Hripcsak et al., 2002; Hripcsak et al., 1998].

Wang et al. [2017] have released one of the first very large open-source datasets with frontal chest X-rays. Notably, they present a method to improve NLP for label extraction from free-text radiology reports. Nevertheless, they also report a label noise of approximately 10 % for each of the 14 findings, which implies that at least 10 % of the images have a false label (i.e., one or more labels are wrong in a single image). As demonstrated in our experiments (see Section 55.3) and mentioned by Oakden-Rayner [2017], applying the supervised training of neural networks with these NLP-generated labels presents another major problem, especially for the critical finding "pneumothorax". Most pneumothorax cases in the ChestX-ray14 dataset are already treated, meaning there is a chest tube in the image. Without addressing this problem, a neural network will use this tube as the main feature for classification since the tube is much easier to recognize than the pneumothorax.

Section 6.2 presents a set of new annotations for the public OpenI dataset generated by two expert radiologists of the University Medical Center Hamburg-Eppendorf. This enables a real performance evaluation of our methods (i.e., testing methods on reliable labels without noise) and also allows the identification of any biases in the dataset. Unfortunately, manually labeled datasets often have a small sample size due to the enormous effort involved. In Chapter 6, a solution based on pretraining a model

on the noisy—but very large—ChestX-ray14 dataset and fine-tuning it on normalized images is used. The image normalization is performed by lung field cropping and bone suppression (see Section 6.1) [Baltruschat et al., 2019e].

2.4.3 Abnormal findings in chest X-rays

MacMahon et al. [1991] have evaluated the frequency of abnormal findings in chest X-rays. For this purpose, they have defined 10 main abnormal findings and 30 subcategories but did not consider the degree of manifestation. Table 2.5 shows the defined finding categories, while Table 2.6 presents the results of the frequency analysis.

Two challenges for image processing arise from this. First, the large variety of findings makes it nearly impossible to develop an automatic image analysis that classifies most findings based on handcrafted features. This explains why researchers often only concentrate on abnormal individual findings when using handcrafted feature extraction methods. Section 2.4.4 discusses the resulting implications for a clinical applications. With deep learning, researchers no longer need to focus on individual findings because feature engineering is now obsolete.

Furthermore, the training of neural networks for the medical field is complicated because most abnormal findings have a low prevalence. The results of MacMahon et al. [1991] (see Table 2.6) indicate that only five abnormal findings have a prevalence greater than 10 %. A similar problem was identified by two expert radiologists from the University Medical Center Hamburg-Eppendorf while creating the new annotations for the OpenI dataset shown in Section 6.2. This problem of an imbalanced dataset is often addressed by oversampling the minority classes or by employing a weighted binary cross-entropy loss function. Section 5.2.1 explores various weighting methods for the loss function to deal with this problem.

2.4.4 Translation into clinical applications

While the goal of most research in the field of chest X-ray analysis is translation into clinical applications, only a few software solutions are currently available for automatic chest X-ray analysis. The problems with clinical applications for lung disease

Table 2.5: Overview of abnormal findings in chest X-rays for classification [MacMahon et al., 1991].

Cardiovascular	**Pleura**	**Mediastinum**
Cardiac size/contour	Scarring	Masses
Cardiac calcification	Effusion	Air collections
Pulmonary vessels	Masses	Shift/contour
Aorta	Pneumothorax	Tracheal deviation
Hila	**Bones**	**Diaphragm**
Masses	Ribs	Abnormal contour/elevation
Vascular	Spine	
Calcified nodes	Other	
Lung	**Hardware**	**Extrathoracic**
Nodules	Catheters	
Masses	Endotracheal/tracheostomy tubes	**Other**
Calcified granulomas	Drainage catheters and tubes	
Infiltrate	Prosthetic valves	
Linear atelectasis/scar	Pacemakers	
Bullae		

Table 2.6: Abnormal finding distribution in chest X-rays [MacMahon et al., 1991].

Frequency ranking	**Finding**	**Count**	**% of all images ($N = 1089$)**	**% of all abnormal ($N = 877$)**
1.	Pulmonary infiltrates	482	44%	55%
2.	I. V. catheters	291	27%	33%
3.	Heart size/contour	239	22%	27%
4.	Endotracheal/tracheostomy tubes	193	18%	22%
5.	Pleural effusions	130	12%	12%
6.	Linear atelectasis/scar	86	8%	10%
7.	Drainage catheters and tubes	78	7%	9%
8.	Pulmonary vascularity	77	7%	9%
9.	Pleural scarring	69	6%	8%
10.	Rib lesions	65	6%	7%
11.	Mediastinal masses	56	5%	6%
12.	Diaphragm	44	4%	5%
13.	Calcified granulomas	43	4%	5%
14.	Pneumothorax	42	4%	5%
15.	Lung nodules	40	4%	5%
16.	Extrathoracic abnormalities	36	3%	4%
17.	Lung masses	17	2%	2%
18.	Calcified nodes	13	1%	1%
19.	Mediastinal shift/contour	13	1%	1%
20.	Cardiac pacemakers	12	1%	1%

classification primarily arise from a combination of the challenges discussed in the previous sections, the current clinical situation, and regulatory questions.

Since chest X-ray is the most common type of examination in a radiology department [Bundesamt für Strahlenschutz, 2020; NHS England, 2020], the growing workload in radiology and decreasing revenue indicate the need for software support. While fully automated chest X-ray analysis—where a radiologist only has to cross-check the results—is the ultimate goal, there are many other clinical applications.

At present, most research (e.g., all 39 papers presented in Section 2.2) concentrates on only a subset of all diseases in chest X-rays since there is no public dataset available (see Section 2.3) to train a neural network for all diseases. However, the research presented in the literature review can be used for useful clinical applications other than fully automated chest X-ray analysis. First, the detection of all normal chest X-rays (i.e., no abnormal finding on the chest X-ray) can greatly reduce the workload in a radiology department. The results of Section7.2.1 show that approximately 30 % of all chest X-rays at the University Medical Center Hamburg-Eppendorf are normal. Hence, such software could reduce the workload by approximately 30 %.

Additionally, a system with only a subset of all diseases could be used to automatically pre-fill radiological reports [Laserson et al., 2018]. However, the issue with such pre-filling is that radiologists must read it and look for additional findings. Furthermore, the classification of chest X-ray diseases can also be used to develop a worklist prioritization system for a radiology department. This application could use classification results to sort patients according to the urgency of their condition (e.g., a patient with a pneumothorax requires urgent medical assistance or he may suffer significant harm). For most of these applications, it remains questionable whether the current training datasetmyColorMainAand especially the test datasetmyColorMainAis sufficiently labeled to determine the performance of an automated image analysis system.

Chapter 7 presents the first known simulation framework to model a clinical workday and show significant improvements to smart worklist ordering using a convolutional neural network. This clinical application is evaluated in the context of various operating points of the classification algorithm. Notably, the simulation shows a significant problem with imperfect classification since false-negative predictions result in significantly longer reporting times for some patients (i.e., the images are sorted to the end

of the worklist). Hence, Chapter 7 presents the use of a novel thresholding of the maximum waiting time to address this problem.

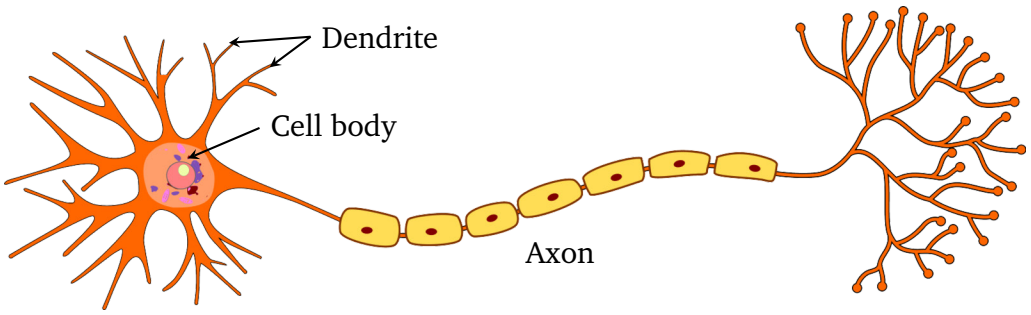
3 Artificial neural networks

This chapter introduces some core concepts related to artificial neural networks and provides a brief overview of their history. First, the basic theory of an artificial neuron and its biological analogue are explained. Notably, the combination of multiple artificial neurons results in an artificial neural network. Section 3.1 introduces a common artificial network type known as a feed-forward neural network. Then, Sections 3.5 and 3.6 introduce back-propagation and optimization methods for artificial neural networks, respectively. Finally, Section 3.8 discusses the various activation functions of an artificial neuron.

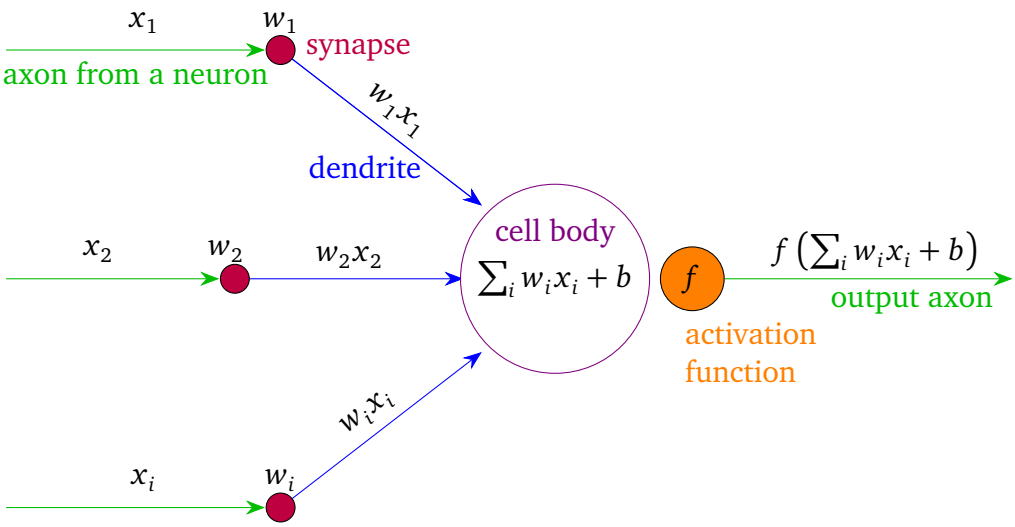
The human brain is a massive biological neural network in which over 86 billion biological neurons form a complex, nonlinear, parallel computer [Keller et al., 2016]. The neuron is the basic signal unit of our brain structure, while the simplification of a neuron into a mathematical model forms the foundation for building an artificial neural network (see Section 3.1). Figure 3.1 (a) illustrates a biological neuron and Figure 3.1 (b) presents an artificial neuron. Biological neurons receive signals through synapses located at dendrites. Based on synaptic strength w_i, the input signal x_i is multiplicatively weighted. The dendrites pass the weighted input signal $w_i x_i$ to the cell body, where all signals are summed. The cell body fires a signal (i.e., activation function f) if the sum reaches a specific threshold, which can be shifted by a bias. This output signal y is distributed by the axon and splits into multiple branches that are connected to the input of other neurons. In short, this behavior is modeled by an artificial neuron and can be expressed by the following formula:

$$y = f(a) = f\left(\sum_{i=1}^{I} w_i x_i + b\right) \tag{3.1}$$

where is i is the running index over the total number of input signals I, a is the artificial neuron's output before an activation function f (see Section 3.8) is employed, and b is the bias. Artificial neural network theory is based on the notion that the parameters w_i and b are trainable and control the influence that one artificial neuron has over



(a) Biological



(b) Artificial

Figure 3.1: Illustrations of a biological neuron (a) [Image source: [Wikimedia Commons, 2018]] and an artificial neuron (b) [Image based on [Karpathy, 2014]]. The artificial model was inspired by the biological neuron.

other artificial neurons. Hereafter, the terms “neuron” and “neural network” always refer to the artificial model.

3.1 Feed-forward neural network

A feed-forward neural network is a radically simplified representation of the brain structure. It consists of connected artificial neurons and can be represented in a weighted directed graph (see Figure 3.2). As shown by McCulloch et al. [1943] and Minsky et al. [1969], only the connection of multiple artificial neurons in a neural network can solve any logical function (see Section 3.4). Later, Cybenko [1989] proved that neural networks are also a universal function approximator.

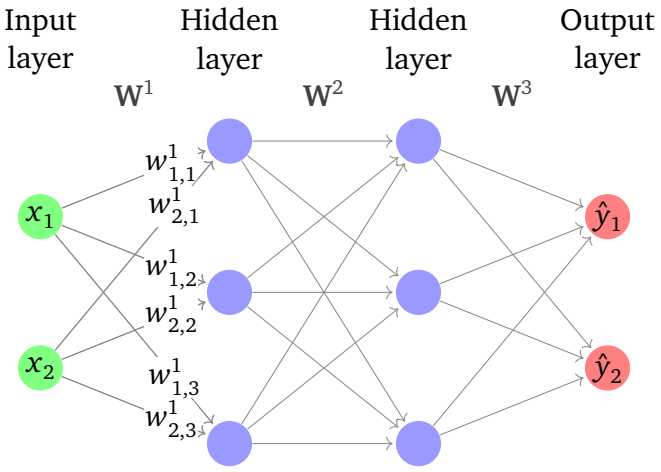


Figure 3.2: A feed-forward neural network with two hidden layers. The input and output layers have two neurons, while each hidden layer has three neurons.

Each node shown in Figure 3.2 represents an artificial neuron, while the arrows indicate the connections between them. As previously noted, each neuron has a weight $w^l_{i,j}$, where $l \in \{1,2,\dots,L\}$ is the l-th layer in a network with L layers, $i \in \{1,2,\dots,I\}$ is the i-th neuron of layer $(l-1)$ with I neurons, and $j \in \{1,2,\dots,J\}$ is the i-th neuron of layer l with J neurons. Hence, the pair i,j is the connection between the i-th neuron of layer $(l-1)$ and the j-th neuron of layer l. Neurons are arranged layer-wise and those of the same layer share no connections. Neurons of the input layer $(x^{\text{IN}}_1, x^{\text{IN}}_2, \dots, x^{\text{IN}}_m)^\top = \mathbf{x}^{\text{IN}}$ with $\mathbf{x}^{\text{IN}} \in \mathbb{R}^m$ only pass information into the network and perform no computations. The output neurons $(\hat{y}_1, \hat{y}_2, \dots, \hat{y}_n)^\top = \hat{\mathbf{y}}$ with $\hat{\mathbf{y}} \in \mathbb{R}^n$ rarely have a nonlinear activation function. The layers between input and output layers are called hidden layers. The total number of neurons in the hidden layers and the number of layers are known as the width and depth of a neural network, respectively [Goodfellow et al., 2016]. One can express the layers of a neural network using matrix vector notation. First, the weights of each neuron can be combined into a vector $(w^l_{1,j}, w^l_{2,j}, \dots, w^l_{I,j})^\top = \mathbf{w}^l_j$ with $\mathbf{w}^l_j \in \mathbb{R}^I$. Thereafter, we can combine the weights of all j neurons of the l-th layer into a single weight matrix $\mathbf{W}^l$ with $\mathbf{W}^l \in \mathbb{R}^{I \times J}$. We can do the same for the bias of each neuron and combine them into a vector $\mathbf{b}^l$ with $\mathbf{b}^l \in \mathbb{R}^J$. For simplicity, the bias vector can be merged into the weight matrix by extending the input vector $\mathbf{x}$ to a layer with a 1 and adding $\mathbf{b}^l$ as an additional column to $\mathbf{W}^l$. Hence, $\mathbf{x}^l = (x_1, x_2, \dots, x_I, 1)^\top$ and $\mathbf{W}^l$ have the dimensions $\mathbf{W}^l \in \mathbb{R}^{I \times (J+1)}$. To calculate the output of the l-layer, we use matrix vector multiplication and apply the vector function $\mathbf{g}(\mathbf{a}^l) = (f(a^l_1), f(a^l_2), \dots, f(a^l_j))^\top, \mathbf{g} : \mathbb{R}^J \to \mathbb{R}^J$, where f is the activation function. Hence, the output of the l-layer is $\mathbf{g}(\mathbf{a}) = \mathbf{g}(\mathbf{W}^l \mathbf{x}^l)$.
Therefore, all weights can be combined into matrices $\mathbf{W}^1, \mathbf{W}^2, \dots, \mathbf{W}^L$ layer by layer.

Hereafter, θ stands for all parameters of a neural network. The output $\hat{\mathbf{y}}$ is calculated as follows:

$$\hat{\mathbf{y}} = \mathbf{W}^L \cdot \mathbf{g}(\dots \mathbf{g}(\mathbf{W}^2(\mathbf{g}(\mathbf{W}^1\mathbf{x})))\dots) \,. \tag{3.2}$$

The example presented in Figure 3.2 shows fully-connected layers (also known as dense layers) where each neuron in a layer receives an input from all neurons of the previous layer. In a feed-forward network, the input information passes through the hidden layers to the output layer and no loops are permitted in between. The number of hidden layers (i.e., the depth) and the sizes of the hidden layers (i.e., the width) are *hyperparameters* of the network. All parameters that are not optimized during training are hyperparameters and must be manually selected.

3.2 Learning types

The learning of neural networks can be divided into four methods: supervised, semi-supervised, unsupervised, and reinforcement learning. This thesis only employs supervised learning, which is explained in greater detail in the following section. Comprehensive introductions to semi-supervised, unsupervised, and reinforcement learning are provided in [Goodfellow et al., 2016] and [Burkov, 2019].

In supervised learning, the dataset contains N sample-label pairs $(\mathbf{x}_i, y_i), i \in \{1, 2, \dots, N\}$. Each sample $\mathbf{x}_i \in \mathbb{R}^D$ is a vector with $\mathbf{x}_i = (x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(D)})$, where each entry $\mathbf{x}^{(j)}$ describes the sample in some manner. For example, in image analysis, each sample $\mathbf{x}_i$ can be an image, while the values describe the intensity value of each pixel. The label y_i can be a real number or an element of a set of classes $\{1, 2, \dots, k\}$, or a vector. The type of label depends on the problem at hand. For instance, if the samples are images of a single-digit number and the problem is number classification, then the set of classes is $K = \{0, 1, \dots, 9\}$ and each label $y_i \in K$ is one of those numbers. In Figure 3.3 (a), multiple samples of the MNIST (Modified National Institute of Standards and Technology) dataset [Deng, 2012; LeCun et al., 1995] are combined into one image to illustrate some examples of $\mathbf{x}_i$. The MNIST dataset contains handwritten digits with the appropriate label describing the number included in the sample.

The aim of a supervised learning algorithm is to use the sample-label pairs to train a

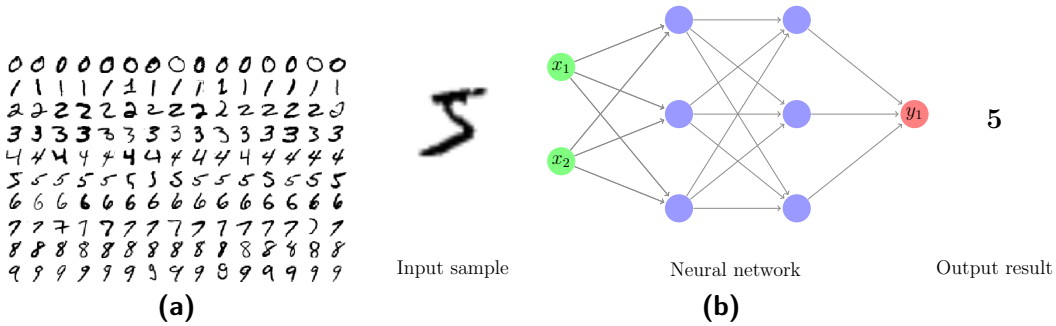


Figure 3.3: The sample-label pairs (b) of the MNIST dataset [Deng, 2012; LeCun et al., 1995] can be used in supervised learning to train a neural network. Based on the dataset (b), the neural networks learn to classify images of handwritten digits (a).

neural network so that the model can solve a specific task. Training means that the weights θ are tuned to map the input $\mathbf{x}_i$ to the output y_i. In the MNIST example, the model learns to classify images of handwritten digits (see Figure 3.3 (b)). Additional information about the training (also known as optimization) is provided in Sections 3.5 and 3.6.

3.3 Classification vs. regression problems

While neural networks can be used to solve many types of tasks, this thesis mainly focuses on classification and regression problems.

Classification describes the task of determining which class of k classes an input $\mathbf{x}_i$ belongs to. To solve this task, the neural network is optimized (see Section 3.6) to approximate a function $f_{NN} : \mathbb{R}^D \rightarrow \mathbb{R}$. The trained model then uses $f_{NN}(\mathbf{x}_i) = \hat{y}$ to assign a number to each input example $\mathbf{x}_i$. For example, the number $\hat{y}$ (a probability) is used to derive the class (note that the terms “class”, “category” and “label” are used interchangeably).

When the number of classes k is two, it represents a binary classification problem (e.g., “healthy” or “sick”). When the number of classes k is three or more, it represents a multiclass classification problem. This should not be confused with multilabel classification. In binary or multiclass classification, only a single class is assigned to the input $\mathbf{x}_i$; however, for multilabel classification, more than one class can be assigned to

the input $\mathbf{x}_i$ by a function $f_{NN} : \mathbb{R}^D \to \mathbb{R}^m$. In this thesis, only multilabel classification problems are explored.

While regression is similar to classification, with regression a real value is assigned to an input instead of a discrete label. Therefore, a neural network is optimized to approximate a function $f_{NN} : \mathbb{R}^D \to \mathbb{R}$ and the output is the direct final result. For example, we investigate the regression task of predicting the age of a patient based on their chest X-ray image in Chapter 5. The next section briefly summarizes the history of artificial neural networks to provide a better understanding of where they began and how they evolved.

3.4 Artificial neural network as a computational tool

The origins of using artificial neural networks as a computational tool began as early as the 1940s, when Warren McCulloch and Walter Pitts published the first paper [McCulloch et al., 1943] on the possible functioning of artificial neurons. These researchers showed that single-layer artificial neural networks can solve problems that are linearly separable [McCulloch et al., 1943]. In the 1950s, the Mark 1 Perceptron machine (see Figure 3.4) was the first successful hardware implementation of the perceptron—an artificial neuron with a step function as the activation function—algorithm [Rosenblatt, 1962]. Notably, it was a single-layer neural network. Rosenblatt and his colleagues demonstrated that it was possible to correctly recognize the letters of the alphabet. The input image was 20×20 pixels in size and the activation function was a step function (see Section 3.8), as per Equation 3.1.

In 1960, Widrow invented ADALINE: Adaptive linear neuron [Widrow, 1960]. Instead of using the output of the activation function for weight adjustment as Rosenblatt had done, Widrow used the weighted summation of the inputs for error calculation. The advantage was that the derivative of the input error could be calculated with respect to each weight to determine the optimal weights by minimizing the error. This was possible because Widrow did not employ an activation function. Later, Widrow and Hoff presented MADALINE: Multiple adaptive linear neurons [Widrow et al., 1960]. This was the first stacked multilayer perceptron network. MADALINE is a fully-connected, feed-forward neural network that consists of three stacked layers.

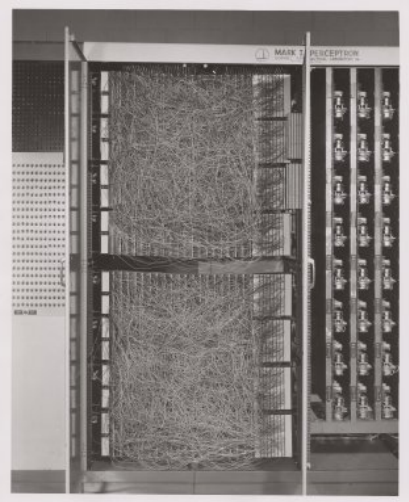


Figure 3.4: The Mark 1 Perceptron machine was the first successful hardware implementation of the perceptron algorithm.

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \sum_{i=1}^{I} w_i x_i > 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

d

Equation 3.1: Rosenblatt's Perceptron with $\mathbf{x}$ as the input vector and $\mathbf{w}$ as adjustable weights.

As with many so-called "perceptron networks", the problem was that the network was unable to solve complex problems without a nonlinear activation function. In 1969, Minsky and Papert published the book "Perceptrons" [Minsky et al., 1969], which addressed the perceptron's limitations. For example, a single perceptron could not approximate the Exclusive-OR logic function. For this reason, it was concluded that only a multilayer perceptron network could learn arbitrary logical functions and that it was not possible to train a network using Rosenblatt's learning algorithm [Minsky et al., 1969].

Unfortunately, despite showing early promise, artificial neural networks lost popularity until their resurgence in the 1980s, when David Rumelhart, Geoffrey Hinton, and Ronald Williams demonstrated that training a multilayer neural network with back-propagation (see Section 3.5) was possible [Rumelhart et al., 1986]. This achievement rekindled research efforts into artificial neural networks and brought many new researchers to the field.
In the next section, back-propagation and optimization (see Section 3.6) are presented. Then, different activation functions (see Section 3.8) are introduced as important components for neural network training.

3.5 Back-propagation

Back-propagation [Rumelhart et al., 1986] was a breakthrough in training neural networks. Prior to its development, it was a huge effort to train the weights of the hidden layers in a neural network. Before discussing back-propagation, it is important to consider the theory behind training neural networks. In supervised learning, a given training set $X = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_P, y_P)\}$ contains P pairs of input vectors and target scalars. For a specific set of parameters θ and the input $\mathbf{x}_i$, the neural network produces the output $\hat{y}_i$. The training process attempts to minimize a certain error E between the model output $\hat{y}_i$ and the target y_k by optimizing the parameters θ. To calculate the error E, one uses a loss function L. For example, the squared error

$$E = L(\hat{y}, y,) = (\hat{y} - y)^2 \tag{3.4}$$

is commonly employed for regression tasks, while the cross-entropy error

$$E = L(\hat{y}, y) = y \log(\hat{y}) + (1 - y) \log(1 - \hat{y}) \tag{3.5}$$

is often used for classification problems. Notably, we typically calculate the average error over all training sample-label pairs P and thus have a sum minimization problem $E = \frac{1}{P} \sum_{i=1}^{P} L(\hat{y}_i, y_i)$. The influence of changing a single parameter $w_{i,j}^l$ on the error E is measured by the partial derivative $\frac{\partial E}{\partial w_{i,j}^l}$. Thus, the gradient of the error function ∇E becomes the vector of partial derivatives with respect to all parameters θ:

$$\nabla E = \left(\frac{\partial E}{\partial w_{1,1}^1}, \frac{\partial E}{\partial w_{1,2}^1}, \frac{\partial E}{\partial w_{2,1}^1}, \frac{\partial E}{\partial w_{1,1}^2}, \ldots, \frac{\partial E}{\partial w_{i,j}^l}\right) \text{ with } \nabla E \in \mathbb{R}^{IJL} \ . \tag{3.6}$$

The gradient is used by the optimization algorithm to minimize the error, which is explained in Section 3.6.

Back-propagation entails calculating the partial derivatives of the error function by applying the chain rule

$$\frac{\partial E}{\partial w_{i,j}^l} = \frac{\partial E}{\partial s_j^l} \frac{\partial s_j^l}{\partial w_{i,j}^l} \tag{3.7}$$

with s_j^l being the output of the j-th neuron in the l-th layer:

$$s_j^l = f(a_j^l) = f(\mathbf{W}^l \mathbf{s}^{l-1}) = f(\sum_{i=1}^{I} w_{i,j} s_i^{l-1}) \,. \tag{3.8}$$

To compute the partial derivative $\frac{\partial s_j^l}{\partial w_{i,j}^l}$, the chain rule is used again:

$$\frac{\partial s_j^l}{\partial w_{i,j}^l} = \frac{\partial s_j^l}{\partial a_j^l} \frac{\partial a_j^l}{\partial w_{i,j}^l} = f'(a_j^l) s_j^{l-1} \,. \tag{3.9}$$

The partial derivative $\frac{\partial E}{\partial s_j^l}$ is the influence of the neuron s_j^l on the error E. Two distinct cases arise from this method. First, the neuron s_j^l is an output neuron s_j^L. Thus, the partial derivative can be directly calculated:

$$\frac{\partial E}{\partial s_j^L} = \frac{\partial E}{\partial \hat{y}} \,. \tag{3.10}$$

For example, if the square error is used as the loss function, then:

$$\frac{\partial E}{\partial s_j^L} = \frac{\partial E}{\partial \hat{y}} = \frac{\partial}{\partial \hat{y}} (\hat{y} - y)^2 = 2(\hat{y} - y) \,. \tag{3.11}$$

Secondly, if s_j^l is not an output neuron, the partial derivative is calculated as follows:

$$\begin{aligned} \frac{\partial E}{\partial s_j^l} &= \sum_{b=l+1}^{L} \sum_{i=1}^{I} \frac{\partial E}{\partial s_i^b} \frac{\partial s_i^b}{\partial s_j^l} \\ &= \sum_{b=l+1}^{L} \sum_{i=1}^{I} \frac{\partial E}{\partial s_i^b} \frac{\partial s_i^b}{\partial a_i^b} \frac{\partial a_i^b}{\partial s_j^l} \\ &= \sum_{b=l+1}^{L} \sum_{i=1}^{I} \frac{\partial E}{\partial s_j^b} f'(a_i^b) w_{i,j}^b \end{aligned} \tag{3.12}$$

where b represents the count of all subsequent neurons of layer l. To solve Equation (3.12), the partial derivative $\frac{\partial E}{\partial s_j^b}$ of all previous neurons s_j^b must be known. This can be computed by starting the calculation at the output neurons and propagating the information backward toward the input neurons (hence the term “back-propagation”).

3.6 Optimization

Optimization is the process of minimizing the error E by tuning the parameters θ. The most common optimization method in deep learning is the gradient descent algorithm—a first-order iterative method to find the local or global minimum [Cauchy, 1847; Curry, 1944]. A simple gradient descent algorithm updates the parameters θ via a step in the opposite direction of the error function gradient ∇E:

$$\theta \leftarrow \theta - \eta \nabla E = \theta - \eta \frac{1}{P} \sum_{i=1}^{P} \nabla L(\hat{y}_i, y_i) \tag{3.13}$$

where η is the learning rate. Here, the update is calculated on the entire training set $\mathbb{X}$, which is also known as *batch gradient descent*. If η is sufficiently small, it is guaranteed that batch gradient descent converges to a local—but not necessarily to the global—minimum for a non-convex error function, and the global minimum for a convex error function [Goodfellow et al., 2016]. The calculation time required for a single iteration step (i.e., a single parameter update) can be long for very large datasets because the sum of all gradients is calculated. Thus, if the training set increases by m, the computational cost also increases with $O(m)$. This calculation time should not be confused with the number of iterations required to converge to a sufficiently small error.

Data-wise, *stochastic gradient descent* (SGD) [Kiefer et al., 1952; Robbins et al., 1951] is the opposite of batch gradient descent and performs a weight update for each training example $(\mathbf{x}_i, y_i)$ as follows:

$$\theta \leftarrow \theta - \eta \cdot \nabla L(\hat{y}_i, y_i) \,. \tag{3.14}$$

When compared to batch gradient descent, SGD does not easily converge to a local or global minimum and typically requires more update steps than batch gradient descent [Goodfellow et al., 2016]. Nevertheless, the cost of a single update does not increase with training set size m, which implies that the computational cost of a single update step is only $O(1)$. This is especially important in the case of big data. If m increases near infinity, we can argue that SGD converges to the best possible error without using all training data at a computational cost of $O(1)$ for a single update [Goodfellow et al., 2016].

Notably, there exists a hybrid version of gradient descent that combines the advantages

of both methods. Known as *Mini-batch gradient descent*, this version performs the parameter update on a small subset of the training set. At each step, a random mini-batch of examples $B = \{(\mathbf{x}_1, t_1), \ldots, (\mathbf{x}_O, y_O)\}$ is sampled from the training set X. The gradient is then estimated, and the parameters are updated as follows:

$$\theta \leftarrow \theta - \eta \frac{1}{O} \nabla \sum_{i=1}^{O} L(\hat{y}_i, y_i) \,. \tag{3.15}$$

where n is the size of the subset (i.e., also known as *mini-batch size*). This results in a more stable convergence to a local or global minimum because the variance of the updates is reduced [Goodfellow et al., 2016]. Similar to SGD, if the training set size m grows, the computational cost per update step is $O(1)$ with respect to m.

Optimization methods are an ongoing topic of research. In addition to basic mini-batch gradient descent, there are many other approaches. In this thesis, only *adaptive moment estimation* (ADAM) [Kingma et al., 2015] is explained since, at the time of writing, this is often a good algorithm choice [Schmidt et al., 2020]. Notably, ADAM is used for our experiments in Chapters 5, 6, and 7. In [Ruder, 2016; Schmidt et al., 2020], a comprehensive overview and benchmark of other algorithms is provided.

Adaptive moment estimation

ADAM [Kingma et al., 2015] adapts the learning rate for each parameter at every update step by employing the first and second momentum of the gradients. Momentum helps the algorithm converge faster because it accumulates the weight updates using a moving average [Qian, 1999]. Thus, updates in the same direction as prior updates are preferred. ADAM updates the parameters using the following formula:

$$\theta \leftarrow \theta - \eta \frac{\hat{\mathbf{m}}_u}{\sqrt{\hat{\mathbf{v}}_u} + \epsilon} \tag{3.16}$$

where ϵ is a small constant value to prevent a division by zero, $\hat{\mathbf{m}}_u$ is the bias-corrected first momentum, and $\hat{\mathbf{v}}_u$ is the bias-corrected second momentum. First, we define the first momentum $\mathbf{m}_u$ and second momentum $\mathbf{v}_u$:

$$\mathbf{m}_u = \beta_1 \mathbf{m}_{u-1} + (1 - \beta_1)\mathbf{g}_u \tag{3.17}$$

$$\mathbf{v}_u = \beta_2 \mathbf{v}_{u-1} + (1 - \beta_2)\mathbf{g}_u^2 \tag{3.18}$$

where $\mathbf{g}_u$ is the gradient ∇E, $\mathbf{g}_u^2$ is the elementwise square of $\mathbf{g}_u \odot \mathbf{g}_u$, $\beta_1, \beta_2 \in [0, 1)$ are hyperparameters, and u represents the current update step. Notably, the hyperparameters control the exponential decay of the moving average.

Since the vectors $\mathbf{m}_0, \mathbf{v}_0$ are initialized with zero, a bias correction of $\mathbf{m}_u$ and $\mathbf{v}_u$ must be applied:

$$\hat{\mathbf{m}}_u = \frac{\mathbf{m}_u}{1-\beta_1^u} \tag{3.19}$$

$$\hat{\mathbf{v}}_u = \frac{\mathbf{v}_u}{1-\beta_2^u} \,. \tag{3.20}$$

Kingma et al. [2015] suggested a default value of 0.9 for β_1, 0.999 for β_2, and 10^{-8} for ϵ. ADAM combines the advantages of AdaGrad [Duchi et al., 2010] and RMSProp [Tieleman et al., 2012], which are very popular optimization algorithms [Kingma et al., 2015; Ruder, 2016].

3.7 Generalization assessment of neural networks

Generalization error relates to the prediction capability of the neural network on independent test data (i.e., new data that was not seen before). To evaluate the generalization error, the dataset must be split prior to starting the training. The best approach is to randomly split the dataset into three parts: training, validation, and test data [Hastie et al., 2005]. For example, Figure 3.5 illustrates a split of 50% for training, 25% for validation, and 25% for testing.

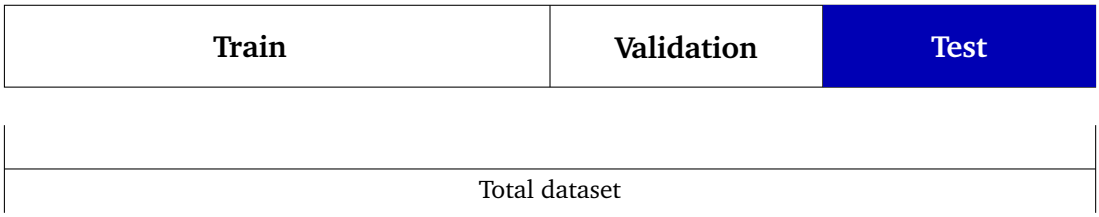


Figure 3.5: The dataset of size N is randomly split into three subsets: training, validation, and test data. The parameters are learned on the training set, while hyperparameter tuning is performed on the validation set. Finally, the generalization performance is measured on the test set.

Stratified sampling preserves the proportion of classes in each subset as compared to

the original dataset. If the dataset classes are balanced, each subset contains balanced classes—which means the problem of training with imbalanced data does not arise. For example, a dataset contains n_A samples of class A and n_B samples of class B. The ratio of class A to class B is $r = \frac{n_A}{n_B}$. If the dataset is split as in Figure 3.5, each set contains approximately the ratio r [Kohavi et al., 1995].

As explained in Section 3, neural networks have many learnable parameters (i.e., parameters changed during optimization) and hyperparameters (i.e., selected manually before starting the optimization). Both types of parameters can and are optimized. The training data is first used to learn the parameters of the neural network; thereafter, the error rate is measured on the validation set. To achieve the optimal error rate, hyperparameters are manually tuned on the validation set. This setup ensures the possibility to measure the generalization error of the neural network on an independent test set [Hastie et al., 2005]. If hyperparameter tuning were performed on test data, the model would overfit to the test data and lose generalization capability [Hastie et al., 2005].

3.7.1 Under- and overfitting

Two common problems can arise when a neural network model is trained to approximate f_{NN} by reducing the training error E: underfitting and overfitting.

Underfitting refers to a problem that occurs when neural network model cannot solve the desired task because the complexity of the neural network is not sufficient for the problem. This means that the training error and test error (i.e., generalization error) are always high, and neither of them converge toward a small error. To counteract this problem, one typically increases the number of parameters of the neural network, which is often considered equivalent to an increase in model complexity [Goodfellow et al., 2016].

On the other hand, overfitting implies that the neural network can solve the desired task for the training data sufficiently well with a small error; however, the error for test data (i.e., data not yet seen) is also very large. Here, the neural network's degree of freedom to adapt to the training data is too high (i.e., the neural network's complexity is too high). Both problems are closely related to dataset size. If we keep the model complexity unchanged (i.e., do not change the number of parameters or add regu-

larization techniques) and reduce the amount of training data, the model will likely overfit to this reduced training set. However, if we substantially increase the amount of training data, the model will most likely underfit the data.

Figure 3.6 presents an exemplary illustration of this problem and the trade-offs one must make. Normally, we aim to find a solution where both the training and testing error are low. At the top of Figure 3.6, three examples of regression models are shown. In all examples, the regression model, training examples, and true function are shown as a red line, red circles, and blue line, respectively. First, a linear regression model is shown that cannot fit the training samples because the complexity (i.e., degree of freedom) of the model is too small. Second, a regression model with a polynomial degree of 4 can fit the true function almost perfectly with low training and testing error. Finally, the complexity of the final polynomial regression model is too high and it overfits the training data with low training error and high testing error.

The generalization error typically contains two types of error: bias and variance. Bias represents the difference between the average prediction of the neural network and the correct value we are attempting to predict [Goodfellow et al., 2016]. High bias neural networks tend to pay little attention to the training data and oversimplify the problem (see Figure 3.6).
Variance refers to the variability of the prediction for a given data point or a value that tells us the spread of the data [Goodfellow et al., 2016]. High variance neural networks pay close attention to training data and do not generalize to data they have not seen before. Thus, we must always make a trade-off between minimizing bias and variance [Hastie et al., 2005; Kohavi et al., 1995].

3.7.2 Sampling methods for dataset splitting

Where the size of the data set is limited, a more sophisticated sampling method is required for generalization error. This is because splitting a small dataset into three parts most likely results in too little training data for optimizing a neural network. Several resampling methods exist that can be used to overcome this problem. These methods split the dataset into training-testing partitions several times. A neural network is trained and tested for each partition and the error rates are averaged. This provides an estimate of the generalization error of the neural network. The most common meth-

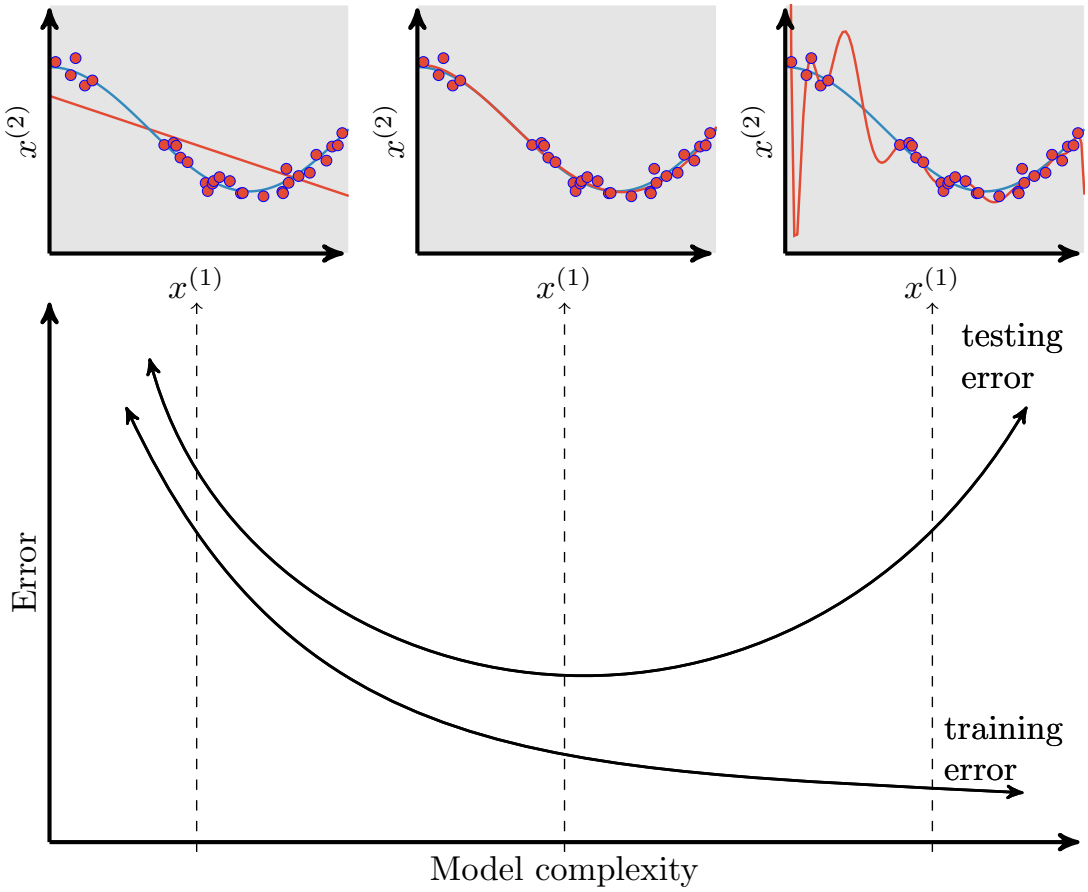


Figure 3.6: Illustration of under- and overfitting. In the upper part of this figure, three examples of regression models (red lines) are shown. Below that, the corresponding errors of the regression models are shown. Each of the three examples show the fitted regression model, true function (blue lines), and training samples (red circles). When the model complexity is too low for the training data and the model cannot be fitted to the data (i.e., first example at the top; the linear regression model failed to fit the true function of the training data), underfitting has occurred. The other end of the spectrum is overfitting, where the model can fit the training data almost perfectly because the model complexity is sufficiently large. At the top right, an example of such overfitting with a polynomial regression model is shown. (Image source: [Fortmann-Roe, 2012] and [scikit-learn, 2020])

ods are K-fold cross-validation and random subsampling (also known as Monte Carlo cross-validation). Both methods are explained in the following subsection.

3.7.2.1 K-fold cross-validation

In K-fold cross-validation, the dataset of size N is randomly split into K subsets, also known as folds. Those subsets have roughly the same size and are mutually exclusive. An illustration of a $K = 5$ split is presented in Figure 3.7. The k-th fold ($k = 4$ in Figure 3.7) is used to measure the error rate, while the other $K - 1$ folds are used for training. This is repeated for $k = \{1, 2, \ldots, K\}$ and the K error rates are averaged to estimate the generalization error. Parameter $K \in [2, N]$ controls the bias-variance trade-off [Molinaro et al., 2005]. A small K results in less training data and thus high bias with low variance. As K becomes larger (to a maximum of $K = N$), the variance increases while the bias decreases. Many studies suggest that favorable choices of K (i.e., 5 or 10) result in acceptable trade-offs between bias and variance [Hastie et al., 2005; Kohavi et al., 1995].

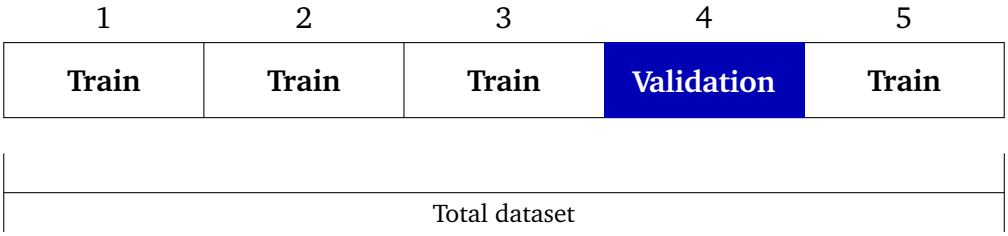


Figure 3.7: Illustration of data splitting using K-fold cross-validation. In this example, the dataset is randomly split into five subsets. The 4th fold is used to measure the error rate, while the other folds are used for training. This is repeated for each subset. The generalization performance can thus be measured without a test set.

3.7.2.2 Random subsampling

Random subsampling splits the dataset into a training and validation set and is often repeated several times I to measure the generalization error. The training set contains $n_t = Np$ samples, whereas the validation set contains $n_v = N - n_t$ samples. The parameter $p \in (0, 1)$ regulates the bias-variance trade-off. Each repetition a

new training-validation partition is randomly sampled without replacement from the dataset, and the error rate of the the validation set is measured. All error rates are averaged to estimate the generalization error. For example, a split of $p = \frac{2}{3}$ and $I = 3$ is presented in Figure 3.8. A large value for the parameter p leads to lower bias and higher variance. In the literature, a p of $\frac{2}{3}$ is recommended as an acceptable trade-off [Molinaro et al., 2005; Xu et al., 2004].

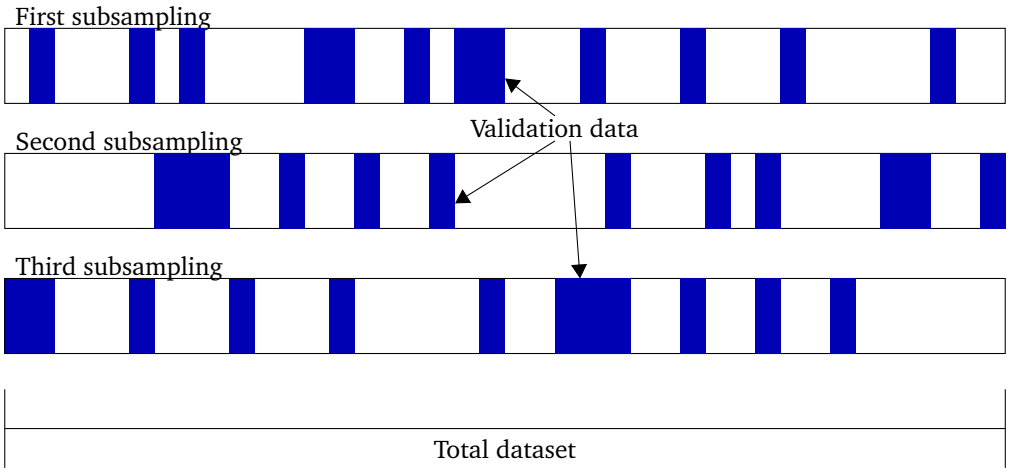
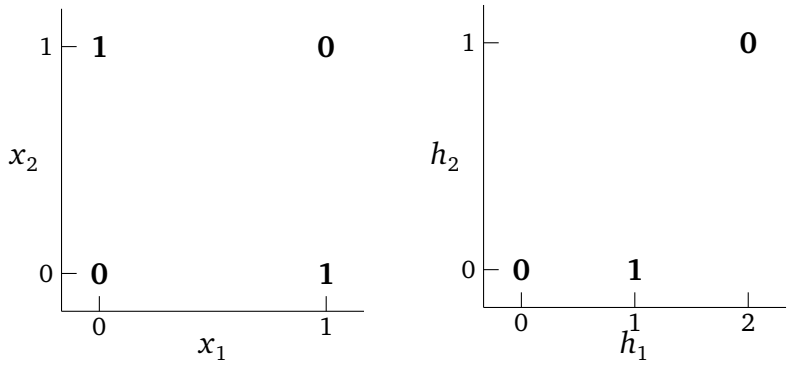


Figure 3.8: Illustration of data splitting using the random subsampling approach which is repeated several times I. The total dataset contains N samples. The training set, with a size of $n_t = Np$, is randomly sampled without replacement from the total dataset. The remaining data $n_v = N - n_t$ are used for validation, i.e., $p = \frac{2}{3}$ and $I = 3$. The validation data are show in blue.

3.8 Activation function

The activation function is an important part of artificial neural networks. However, there are some limitations to this function when we want to solve complex problems using an artificial neural network. If the activation function is linear, it is not possible to approximate nonlinear functions with an artificial neural network. A well-known and simple example of this is the XOR function [Minsky et al., 1969]. If we think of the XOR problem as a classification problem in which we aim to separate “0” and “1”, then it becomes clear that they are not linearly separable (i.e., no single line can separate the “0” and “1”) (see Figure 3.9 (a)). However, a two-layer neural network with a nonlinear activation function (e.g., the step function) can find a transformation of the original space (a) into the feature space (b). By using a nonlinear activation



(a) Original space **(b)** Transformed space

Figure 3.9: Illustration of the XOR problem. (a) shows the results of the XOR function, where the bold numbers indicate the resulting value for the two inputs x_1 and x_2. Since we cannot separate the numbers with a single line, the XOR problem is not linearly separable. A two-layer neural network with a nonlinear activation function can transform the XOR problem to become linearly separable, as shown in (b) [Goodfellow et al., 2016]. In (b), both "1"s are mapped to the same point (1, 0).

function, the capacity of the model (i.e., the number of functions a neural network can approximate) increases [Goodfellow et al., 2016]. Since many real-world problems are nonlinear, it is common to use a nonlinear activation function. Furthermore, the activation function must be almost everywhere differentiable since the optimization algorithm requires the derivatives of the activation function (see Sections 3.5 and 3.6).

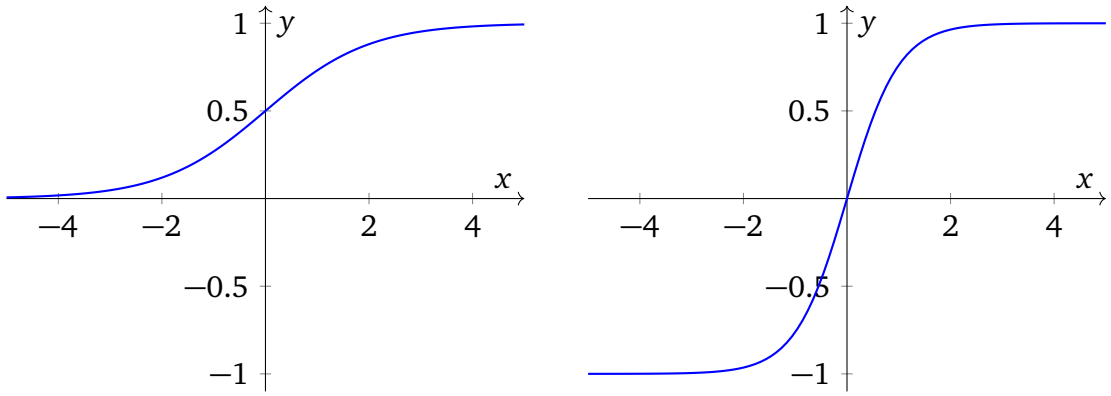
In the very beginning of neural networks (see Section 3.4), the step function was commonly used as an activation function. With the introduction of SGD as optimization method, it was necessary to find alternatives for the step function since the derivative is always zero. In the 1980s, the fist alternatives were the sigmoid

$$\sigma(a) = \frac{1}{1+e^{-a}} \tag{3.21}$$

or hyperbolic tangent

$$\tanh(a) = \frac{e^{a}-e^{-a}}{e^{-a}+e^{-a}} \tag{3.22}$$

functions. An illustration of these functions is presented in Figure 3.10. For the sigmoid activation function, large values of a are mapped to one and small values to zero. While this is a good approximation of a biological neuron's activation rate, the saturation at one and zero leads to the *vanishing gradient* problem in artificial neural networks [Srivastava et al., 2015]. If a neuron is in the saturation area, it has a gra-



(a) Sigmoid function **(b)** Hyperbolic tangent

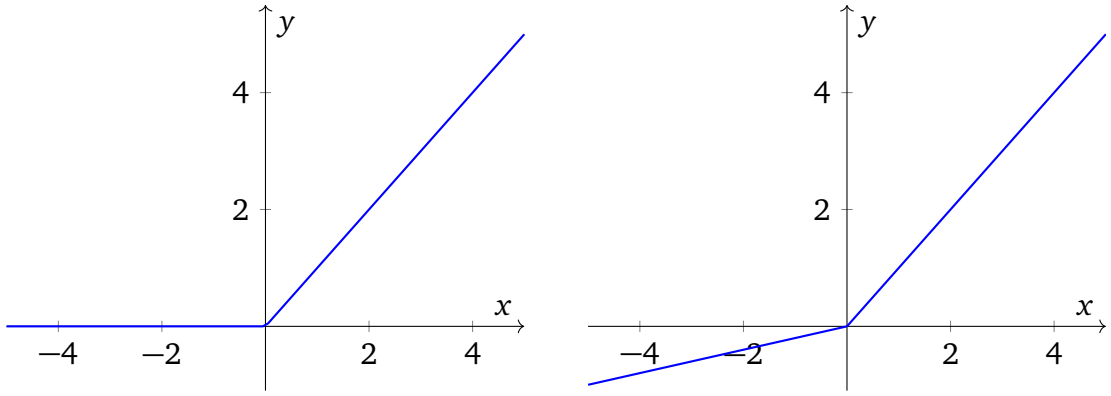
Figure 3.10: Plot of two traditional activation functions: sigmoid and hyperbolic tangent

dient of nearly zero. If a neuron's gradient approaches zero, its weights are no longer updated during back-propagation (see Section 3.5). Furthermore, the gradients in a multilayer neural network are multiplied during back-propagation, which can also increase the gradient vanishing problem.

For this reason, non-saturated activation functions are commonly used. One very popular activation function is the rectified linear unit (ReLU) [Nair et al., 2010]:

$$f(a) = \begin{cases} a & \text{if } a > 0 \\ 0 & \text{if } a \leq 0 \end{cases}. \tag{3.23}$$

Krizhevsky et al. [2012] reported that the optimization of a four-layer convolutional neural network with ReLU activation functions converges six times faster when compared to the same network with tanh activation functions. The time required for the optimization (also known as training) process of neural networks is typically one of many problems. Krizhevsky et al. [2012] noted that even with the improvement of neural networks, their final training took five to six days.
Additionally, the ReLU function can be implemented through a simple thresholding and does not require expensive operations such as exponentials. However, a neuron can "die" when training a neural network, which implies that it becomes inactive for all inputs. As illustrated in Figure 3.11 (a), the negative side of the ReLU function has a gradient of zero. If every input of the neuron is negative, the neuron is considered "dead" because the gradient flow is zero. This leads to a sparse representation of the neural network, which is somewhat useful since it reduces calculation complexity and



(a) ReLU function **(b)** PReLU function

Figure 3.11: Plot of two modern activation functions: the rectified linear unit (ReLU) function [Nair et al., 2010] (a) and the parametric ReLU [He et al., 2015b] with $\beta = 0.2$ (b).

helps to reduce the problem of overfitting for complex models [Glorot et al., 2011].

To counter the dying neuron problem, leaky [Maas et al., 2013] and parametric ReLU [He et al., 2015b] functions were proposed. Both can be expressed with the following function:

$$f(a) = \max(0, a) + \beta \min(0, a) = \begin{cases} a & \text{if } a > 0 \\ 0 & \text{if } \beta a \leq 0 \end{cases} . \tag{3.24}$$

While leaky ReLU has a constant small positive slope β in the negative area, parametric ReLU (PReLU) has a learnable argument β for the slope in the negative area. Thus, β is optimized during the training of the neural network to find the optimal value for β. Maas et al. [2013] investigated different constant slopes for leaky ReLU and found that $\beta = 0.01$ led to the fastest convergence for model training. In [Xu et al., 2015], it was shown that parametric ReLU can perform better for large datasets when compared to ReLU and leaky ReLU; however, it is prone to overfitting for small datasets. In this work, the ReLU activation function was primarily used due to its simplicity and good performance. However, finding the optimal activation function remains an active area of research. In [Rasamoelina et al., 2020], a short review of advanced methods like Swish [Ramachandran et al., 2017] and Mish [Misra, 2019] is provided.

4 Deep neural networks

In this chapter, we present important enablers for training multilayer neural networks—also known as deep neural networks—and explain some state-of-the-art concepts related to improving neural networks. First, the concept of convolutional neural networks, which are specifically designed for image processing, is described. Thereafter, the concept of batch normalization is explained. In Section 4.5, residual (also known as shortcut) connections are presented, which help to solve the vanishing gradient problem for very deep neural networks. The supervised training of modern deep neural networks requires a lot of data. Thus, data augmentation methods are commonly employed to artificially increase the dataset and create a model invariant to feature changes (see Section 4.6).

In the last decade, tremendous improvements have been made in applying neural networks to computer vision tasks such as image classification, image generation, and object detection. This was possible because certain fundamental changes to the traditional neural network have been proposed. In the following section, we explain and highlight the most important changes for this thesis.

4.1 Convolutional neural networks

In Section 3.1, we introduced the concept of fully-connected layers. Image processing often deals with high dimensional input data (e.g., a chest X-ray can have an image size of 2330×2846 pixels). If only fully-connected layers are used, the number of parameters grows rapidly and becomes extremely high, making optimization very computationally intensive or even impossible. For this reason, new layer types were introduced to reduce the number of parameters. In image processing, we can reduce the number of parameters by incorporating prior knowledge about the strong local correlation of neighboring pixels into the layer structure. Hence, neurons are

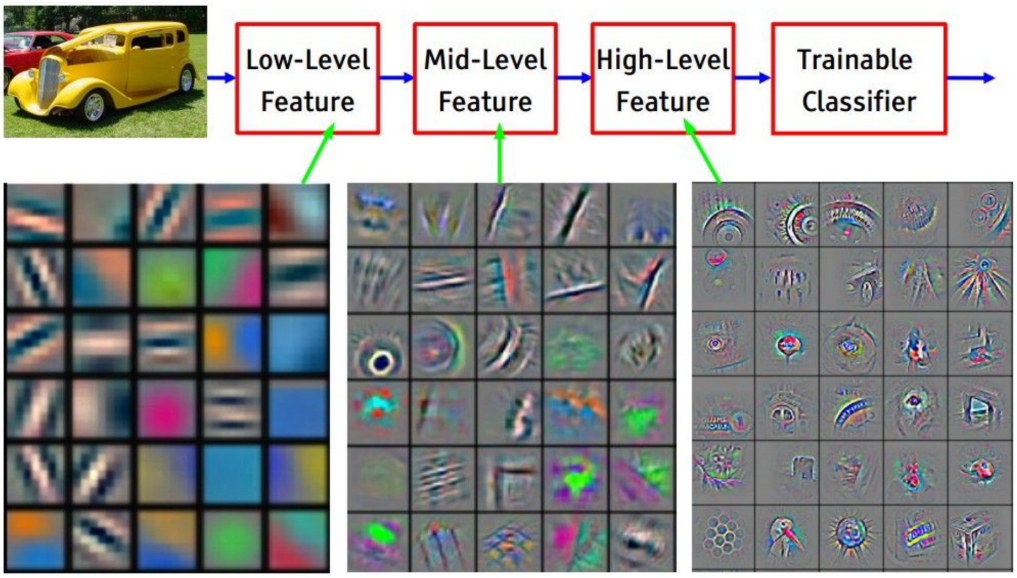


Figure 4.1: Hierarchical feature extraction of a convolutional neural network. The bottom row presents the feature visualization of a convolutional network trained on the ImageNet dataset [Russakovsky et al., 2014]. The top row illustrates the layers of a convolutional neural network. [Image source: [Zeiler et al., 2014]]

only connected to a small local area of the input and are no longer not connected to all other input neurons. As a result, convolutional layers (see Section 4.2) were introduced and replaced most fully-connected layers in neural networks [LeCun et al., 1989]; hence the name *convolutional neural network*. At the time of writing, these networks are often built by stacking convolutional layers, pooling layers (see Section 4.3), batch normalization layers (see Section 4.4), and a final fully-connected layer. In the following section, we briefly explain how convolutional neural networks extract information from an image and also discuss the different layer types.

In a convolutional neural network, information is extracted hierarchically [Zeiler et al., 2014]. The first layers extract simple features such as edges or color blobs. Deeper layers extract feature combinations from previous layers based on the linear combination of previously extracted features. In the final convolutional layers, high-level features are extracted from the image. Figure 4.1 presents a hierarchical feature extraction. The top row illustrates a convolutional neural network with multiple layers. Each layer extracts some low-level features, which are shown underneath. For example, the first layers extract color blobs and edges, while the middle layers extract combinations such as circles. Thereafter, certain objects are extracted that are hopefully linearly separable by a classifier (i.e., the final fully-connected layer).

4.2 Convolutional layer

The convolutional layer is motivated by the fact that, in an image, the information of each pixel has a strong local correlation to neighboring pixels (e.g., edges are an important feature formed by local correlations). Since features can be present in several areas of an image, a filter needs to slide over the complete input data to extract them. The local correlations are utilized by convolving a small filter $\mathbf{K}$ with the input data. The filter often has a symmetric kernel size of $k \times k$. Although the layer is called a convolutional layer, the cross-correlation is typically calculated because this helps to omit kernel flipping. For a two-dimensional input matrix $\mathbf{I}$ and filter $\mathbf{K}$, the two-dimensional cross-correlation is calculated as follows:

$$\mathbf{C}(i,j) = (\mathbf{I} \star \mathbf{K})(i,j) = \sum_{m=1}^{k}\sum_{n=1}^{k} I_{i+m,j+n} K_{m,n} \,. \tag{4.1}$$

Notably we calculate a valid cross-correlation. This means that the calculation area is constrained to pixels (i,j), where the filter $\mathbf{K} \in \mathbb{R}^{k\times k}$ is fully within the input matrix $\mathbf{I} \in \mathbb{R}^{p\times q}$. Let $h = \lfloor k/2 \rfloor$, where $\lfloor\cdot\rfloor$ is the integer division. Thus, we can define the calculation area with $i \in \{h, h+1, \ldots, p-h\}$ and $j \in \{h, h+1, \ldots, q-h\}$. The parameters of the filters are learned during training of the neural network.

In the following section, we explain a so-called two-dimensional convolutional layer and provide an illustration of this layer in Figure 4.2. The feature map $\mathbf{F}^l \in \mathbb{R}^{w_l \times h_l \times d_l}$ is the output of the l-th convolutional layer with width w_l, height h_l, and depth d_l. While the width w_l and height h_l depend on the size of the input map $\mathbf{F}^{l-1}$, the depth d_l is the number of filters a convolutional layer can learn during optimization. Moreover, the depth d_l is a hyperparameter that is often defined before training. Let v and a be the run indexes over the depth d_l and d_{l-1}, respectively. Thus, we can extend the equation (4.1) for a three-dimensional case:

$$\mathbf{F}^l(i,j,v) = \sum_{a=1}^{d_{l-1}}\sum_{m=1}^{k}\sum_{n=1}^{k} \mathbf{F}^{l-1}_{i+m,j+n,a} K^l_{v,m,n,a} \tag{4.2}$$

where $\mathbf{F}^{l-1}$ and $\mathbf{K}^l_v$ are now three-dimensional with $\mathbf{F}^{l-1} \in \mathbb{R}^{w_{l-1}\times h_{l-1}\times d_{l-1}}$ and $\mathbf{K}^l_v \in \mathbb{R}^{k\times k\times d_{l-1}}$, respectively.

In comparison to the fully-connected layers, it is easier to consider that the neurons are structured in a matrix and not as a vector. The total number of neurons N in a

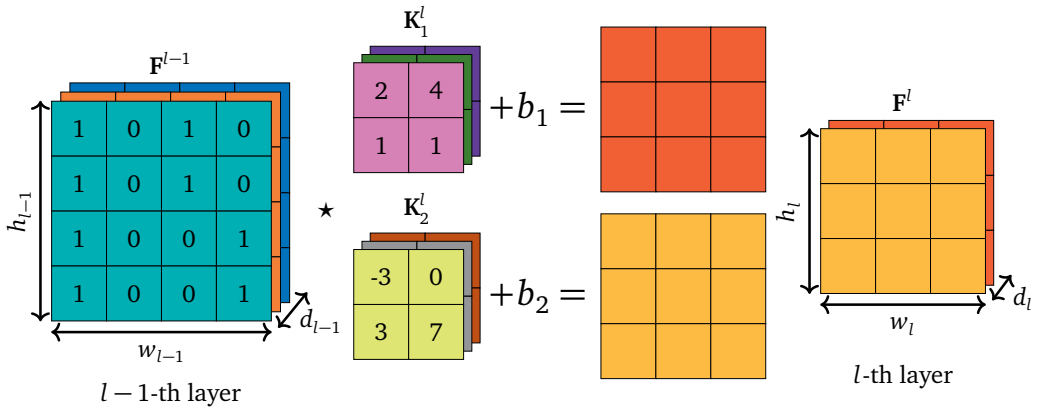


Figure 4.2: Illustration of a convolutional layer with stride $s_l = 2$ and padding $p_l = 1$. The color emphasizes the difference between each tensor.

convolutional layer equals the size of the feature map; therefore, $N = w_l h_l d_l$. Two key components are required to realize the convolution in a neural network and reduce the number of parameters: local receptive field and weight sharing.

Local receptive field: Each neuron of the l-th convolutional layer is only connected to local area $\mathbf{R}^l_{i,j,v}$ in the $l-1$-th layer with the size $k_l \times k_l \times d_{l-1}$, where d_{l-1} is the depth of the input layer to the convolutional layer. This local area or local receptive field describes the size of the region in the input that contributed to the feature calculation. As such, each local receptive field can learn its own filter $\mathbf{K}^l_{i,j,v}$ with the same size as $\mathbf{R}^l_{i,j,v}$. The displacement of each local receptive field in a convolutional layer is defined by the stride $s_l \in \mathbb{N}^*$.
Without weight sharing (which is explained next), each of the N neurons would have $k_l k_l d_{l-1} + 1$ parameters, while the convolutional layer would have $N(k_l k_l d_{l-1} + 1)$ parameters in total. Notably, one parameter is added due to the bias b of each neuron.

Weight sharing: Since the same feature can appear at multiple locations, the concept of weight sharing was proposed. This makes it unnecessary to learn the same feature extractor multiple times and reduces the parameters significantly. Weight sharing implies that all neurons belonging to the same slice v have the same filter $\mathbf{K}^l_v$. Therefore, the depth d_l controls how many filters can be learned. This reduces the total parameters of the convolutional layer by $w_l h_l$; hence, the layer only has $d_l(k_l k_l d_{l-1} + 1)$ parameters. In Figure 4.3, we provide a simple example of a convolutional layer with stride $s_l = 2$ and kernel size $k_l = 2$, $\mathbf{K}^l \in \mathbb{K}^{2\times 2\times 1}$. To calculate the final results, we use the cross-correlation in Equation 4.1 and add the bias b. For example, in the top row

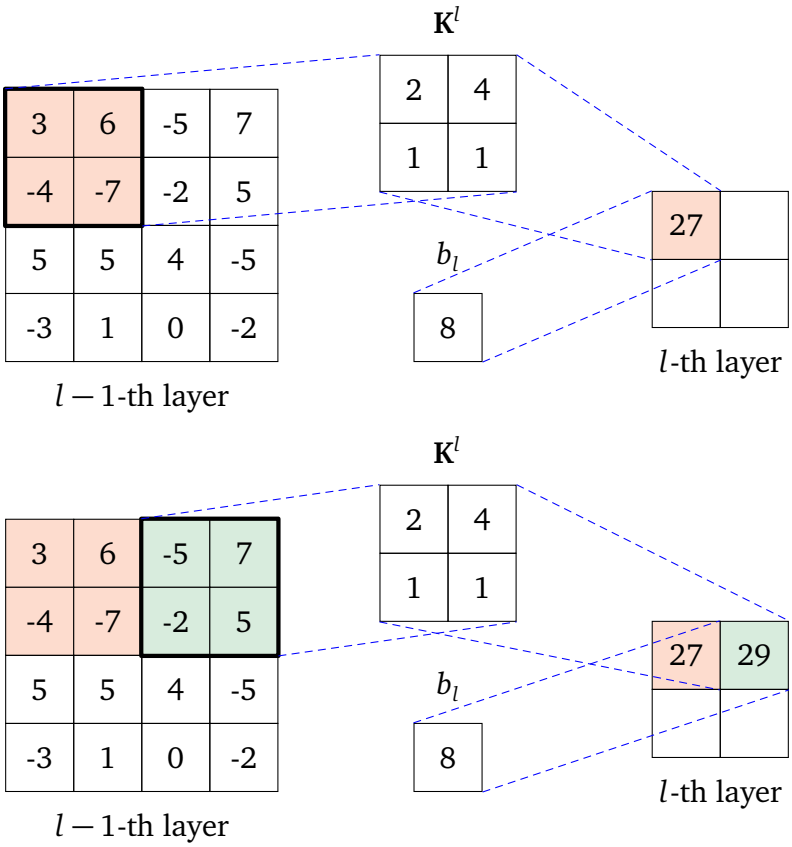


Figure 4.3: Example of a valid cross-correlation calculation with stride $s_l = 2$ and without zero-padding. Only the first two steps are shown. First, the filter K^l (size: 2×2) is applied to the top left area of the $l-1$-th layer (i.e., the light red area). Thereafter, the bias b_l is added and the result is the top left pixel of the l-th layer (i.e., the light red pixel). Then, the filter is shifted by the stride s_l to the right and the same calculation is performed again. This calculation is shown as the light green area and pixels.

of Figure 4.3, we calculate the result for the first cell as follows:

$$\mathbf{F}^l(1,1) = (\mathbf{F}^{l-1} \star \mathbf{K}^l)(1,1) + b_l = (3 \cdot 2) + (6 \cdot 4) + (-4 \cdot 1) + (-7 \cdot 1) + 8 = 27 \,.$$

The local receptive field must be fully connected to the input. Thus, the size of feature map F_l can be calculated by:

$$h_l = (h_{l-1} - k_l)/s_l + 1 \tag{4.3}$$

$$w_l = (w_{l-1} - k_l)/s_l + 1 \,. \tag{4.4}$$

This would always reduce the size of the input tensor by at least $k_l + 1$. Therefore, padding was introduced. Padding artificially increases the size of the $l-1$-th layer by

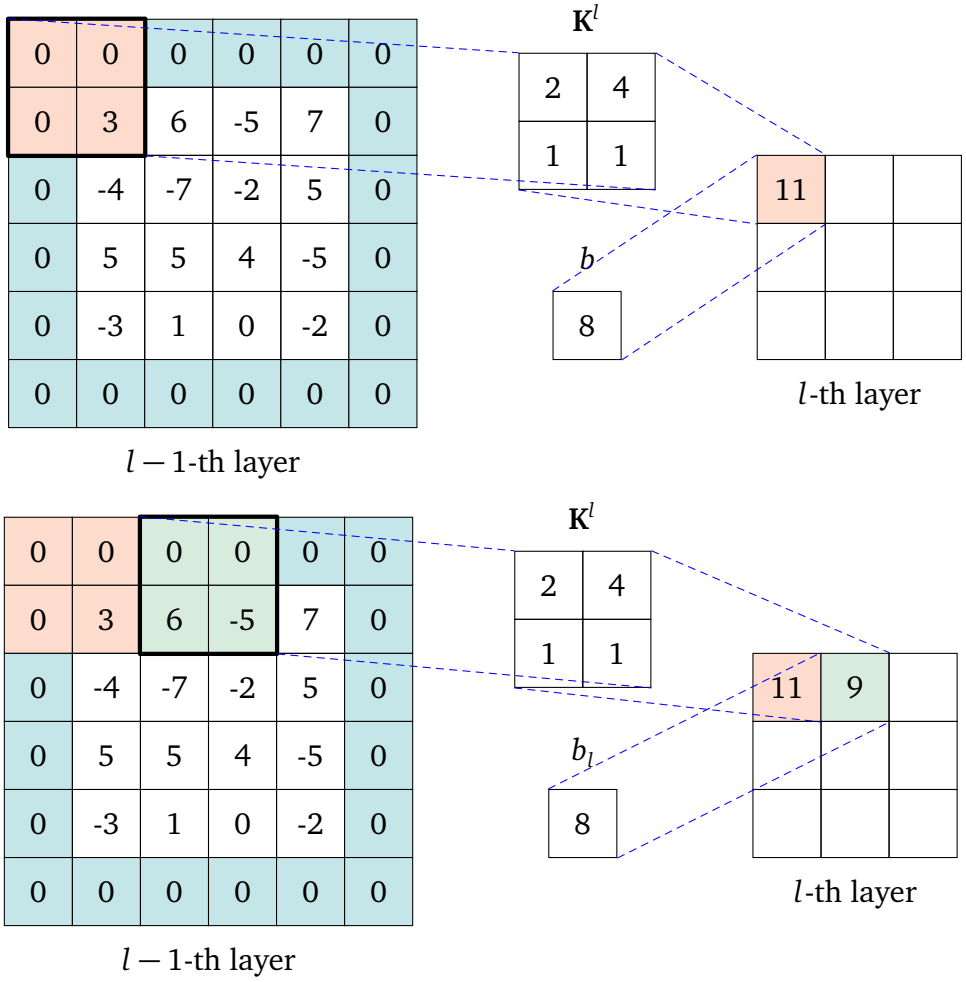


Figure 4.4: Example of a valid cross-correlation calculation with zero-padding $p_l = 1$ and stride $s_l = 2$. The zeros with a light green background are added because of the zero-padding.

adding a border around the input tensor. The size of the border is defined by $p_l \in \mathbb{N}$ and the added border typically contains only zeros. Hence, padding is also known as zero-padding. In Figure 4.4, we illustrate zero-padding with padding $p_l = 1$ and stride $s_l = 2$ for an example matrix. The width and height are then calculated as follows:

$$h_l = (h_{l-1} + 2p_l - k_l)/s_l + 1 \tag{4.5}$$

$$w_l = (w_{l-1} + 2p_l - k_l)/s_l + 1\ . \tag{4.6}$$

4.3 Pooling layer

The pooling layers are used to reduce the spatial dimensions and are defined by three aspects: a specific operation applied to the filter area, the filter size $k_l \times k_l$, and the stride s_l. The most common operations are maximum and average pooling. While maximum pooling [Zhou et al., 1988] (max-pooling) calculates the maximum of the filter area, average pooling calculates the average of the filter area. Average pooling is often used as the last layer to reduce the spatial dimensions before the fully-connected layer is employed. Usually, only the dimensions width and height are reduced—but not the depth of the input tensor. An illustration of max-pooling with filter size $k_l = 2$ and stride $s_l = 2$ is shown in Figure 4.5.

Pooling layers help a model become invariant for small translations of the input; however, the spatial meaning of a pixel is lost [Goodfellow et al., 2016]. In this context, invariant means that most output values of the pooling layer do not change if the input is shifted (i.e., translated) by a small amount.

In the past, pooling layers were integrated into neural networks many times because they are an efficient way to reduce the total parameters. This acts as a regularization method and can counter overfitting on small datasets [Krizhevsky et al., 2012]. Due to increased computing power and data availability, Springenberg et al. [2014] suggests that pooling layers should be replaced by convolutional layers or omitted. For example, the convolutional neural networks in our experiments only contain two or three pooling layers.

Without this regularization method to counteract overfitting, other methods such as batch normalization [Ioffe et al., 2015], dropout [Srivastava et al., 2014], data augmentation [Krizhevsky et al., 2012], and weight decay [Krogh et al., 1992] are currently used (and often required for a small dataset). Good overviews and explanations of common regularization methods can be found in [Kukačka et al., 2017] and [Goodfellow et al., 2016]. Since only batch normalization and data augmentation were employed in this thesis, these methods are explained in more detail in Section 4.4 and 4.6.

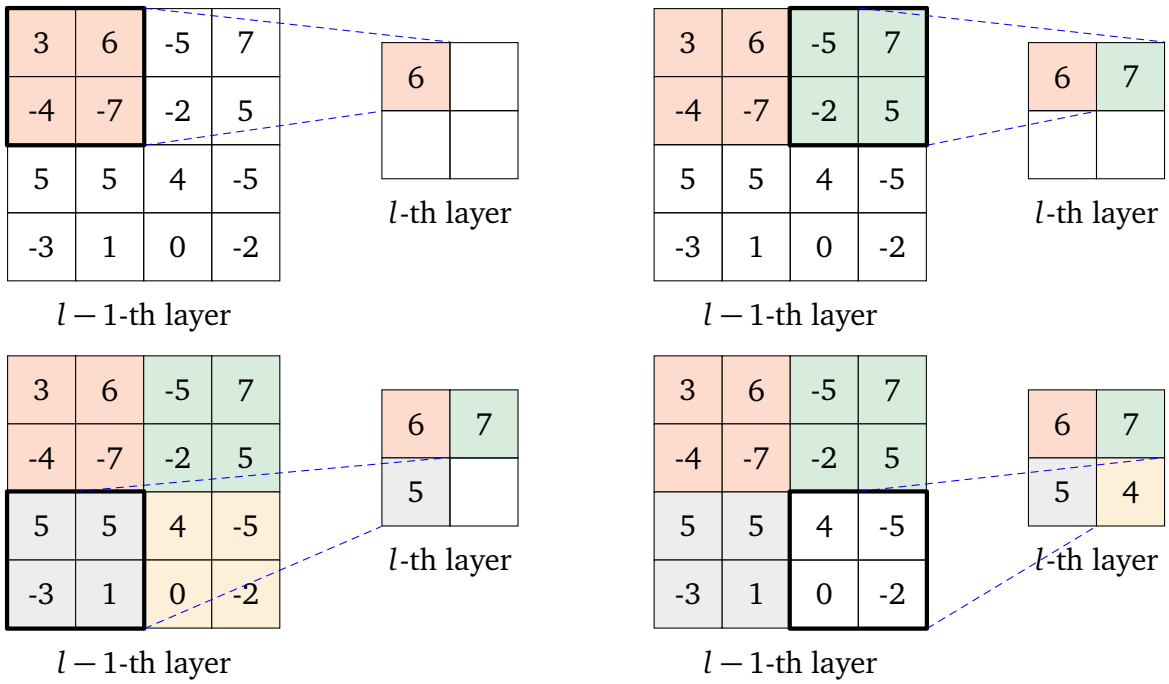


Figure 4.5: Illustration of a pooling layer example. The input layer (size: $4 \times 4 \times 1$) is max-pooled with filter size $k_l = 2$ and stride $s_l = 2$ into an output layer of size $2 \times 2 \times 1$.

4.4 Batch normalization

Batch normalization counters several problems that arise when training deep neural networks. First, it accelerates the training processes by a substantial margin due to improved convergence properties [Ioffe et al., 2015]. Secondly, it allows higher learning rates and a less careful weight initialization [Ioffe et al., 2015]. Thirdly, it can act as a regularization and reduce the need for dropout [Goodfellow et al., 2016]. Currently, several normalization methods are available [Ba et al., 2016; Miyato et al., 2018; Salimans et al., 2016; Ulyanov et al., 2016; Wu et al., 2018]. Since we use batch normalization in this thesis, it is explained in the this section.

Ioffe et al. [2015] identified internal covariate shift as a problem for slow convergence. In neural networks, the inputs of internal layers are affected by the parameters of all previous layers. A small adjustment to parameters in the beginning becomes amplified as the networks become deeper. If the parameters change due to training, the distribution of the layer input also changes. The layer must be adapted and coordinated to this change, which is known as an internal covariate shift.

An internal covariate shift can be reduced by normalizing the activation of a layer by making it have a mean of zero and a unit variance. Consider a layer with a d-dimensional activation vector $\mathbf{s} = (s_1, \ldots, s_d)$ and a mini-batch size of m being used for gradient descent. In this case, each input has d activations. We can arrange this in the activation matrix $\mathbf{S} \in \mathbb{R}^{m \times d}$, where the row represents the samples of the mini-batch m and the columns are the corresponding activations s_d. The values of $\mathbf{S}$ are normalized by column (i.e., the d-dimension are independent) as follows

$$\hat{\mathbf{S}}_{i,j} = \frac{\mathbf{S}_{i,j} - \mu_j}{\sigma_j} \tag{4.7}$$

where μ_j and σ_j are the mean and standard deviation for each column, respectively. Mean and variance are computed over the mini-batch by

$$\mu_j = \frac{1}{m} \sum_{i=1}^{m} \mathbf{S}_{i,j} \tag{4.8}$$

$$\sigma_j = \sqrt{\epsilon + \frac{1}{m} \sum_{i=1}^{m} (\mathbf{S}_{i,j} - \mu_j)^2} \tag{4.9}$$

A simple normalization can reduce the representation power of a neural network. For example, a normalized input to a sigmoid nonlinearity would constrain the function to the linear area. Therefore, two additional parameters are used to apply a linear transformation:

$$\hat{\mathbf{S}}^t_{i,j} = \gamma_j \hat{\mathbf{S}}_{i,j} + \beta_j \tag{4.10}$$

where γ_j and β_j are parameters of the neural network that are optimized during gradient descent. This allows the neural network to restore the original activation by driving γ_j to σ_j and β_j to μ_j.

4.5 Residual connections

The findings of Eldan et al. [2016] show that deeper neural networks are desirable since they can better approximate functions. Based on the findings of He et al. [2015a], it can be argued that when compared to a shallow network, a deeper network should have the same or better error for the same test set. However, naive stacking of layers (i.e., adding more layers to a neural network) does not usually help the optimization method find a solution with a lower error. Therefore, residual connections

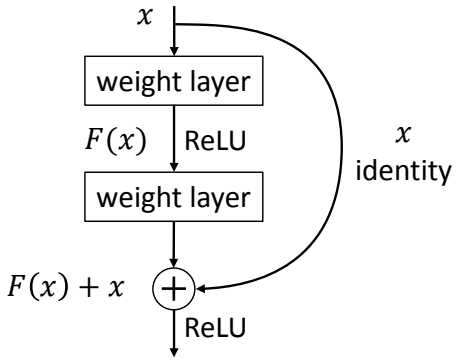


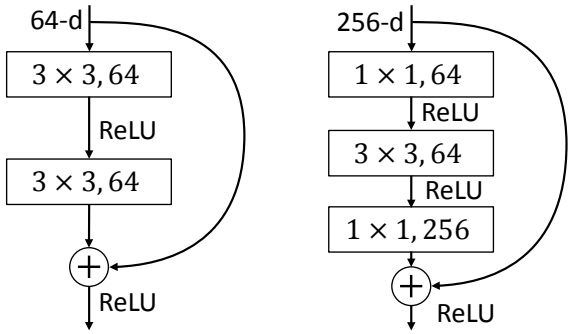
Figure 4.6: Illustration of a residual connection, which is the shortcut from x to the sum $F(x) + x$ (i.e., the identity connection).

(also known as skip connections) were proposed by He et al. [2015a] to deal with this problem. At the time of writing, deep neural networks with residual connections represent state-of-the-art networks for many tasks.

He et al. [2015a] concluded that the optimizer often faces difficulties in finding a favorable solution with a small error for deep neural networks. As a result, He et al. [2015a] introduced residual connections to ease the optimization process for very deep neural networks. Figure 4.6 illustrates the basic concept of a residual connection.

A residual connection is often implemented in deep neural networks by adding connections that act as a shortcut over one or more stacked layers and forward the identity $\mathbf{x}$ to the output of the stacked layers. Let $H(\mathbf{x})$ be the desired mapping. Instead of driving $F(\mathbf{x})$ to $H(\mathbf{x})$, we can reformulate the problem so that $F(\mathbf{x}) := H(\mathbf{x}) - \mathbf{x}$ fits the residual mapping. Thus, the desired mapping $H(\mathbf{x})$ is $F(\mathbf{x}) + \mathbf{x}$. This is realized by the shortcut connection (as seen in Figure 4.6 (a)) and is motivated by the fact that it might be more difficult for deeper layers to learn an identity mapping than to drive $F(\mathbf{x})$ to zero [He et al., 2016].

A bottleneck architecture was also proposed to reduce computation complexity in terms of floating-point operations (FLOPs) since complexity does not scale well by adding more layers to a neural network. For example, training a 200-layer ResNet with bottleneck architecture on ImageNet takes approximately three weeks on eight graphics processing units (GPUs) and would not otherwise be possible [He et al., 2016]. In a bottleneck architecture, a block of two convolutional layers is replaced with three convolutional layers. While this may seem counterintuitive at first due to the additional convolutional layer, it has a major impact on computational complexity. The convolutional layers perform the following three steps.



(a) Original. **(b)** Bottleneck.

Figure 4.7: Comparison of the standard residual connection design with the bottleneck design. The bottleneck has a four times greater input dimension when compared to the standard design. However, the time complexity is the same for both designs.

First, a convolutional layer with a filter size of $1 \times 1 \times d_l$ is employed to reduce the depth dimension of the input [Lin et al., 2013]. As explained in Section 4.2, the convolutional layer can reduce the depth dimension d_{l-1} of the input map $\mathbf{F}^{l-1} \in \mathbb{R}^{m_{l-1} \times n_{l-1} \times d_{l-1}}$ to d_l by having only d_l filters. This is illustrated in Figure 4.7 (b), where the input map with $d_{l-1} = 256$ is reduced to $d_l = 64$. Secondly, the time-consuming $3 \times 3 \times d_l$ convolution is only calculated on the reduced dimensions d_l. Finally, the last convolutional layer restores the depth dimension d_{l-1} by also performing again a $1 \times 1 \times d_{l-1}$ convolution. The depth dimension d_{l-1} is restored via the same method used to reduce the depth dimension in the first layer; however, the number of filters is now greater than the input depth.

For the example, in Figure 4.7, the number of parameters are 73.728 and 69.632 for the old and bottleneck design, respectively. While both have similar complexity in terms of FLOPs, the bottleneck design calculates with an input that has a four times greater depth dimension.

4.6 Data augmentation

Data augmentation can be used to artificially increase the size of a dataset. The training process requires numerous images to counter overfitting. Since labeled data is often rare (particularly in medical image processing), one must use data augmentation. Common data augmentation methods include intensity and geometric transfor-

mation. Since we only apply geometric transformations to our data in this thesis, they are explained in the following section.

A medical X-ray image f is a gray-value image that assigns an intensity value $f(\mathbf{x}) \in [a,b] \subset \mathbb{R}$ to a point $\mathbf{x} \in \Omega \subset \mathbb{R}^2$. The bit depth of an image defines a and b. For example, most raw X-ray images have the data type unsigned short with 16-bit; therefore, $a = 0$ and $b = 2^{16} - 1$. In short, the image is defined by

$$f : \Omega \subset \mathbb{R}^2 \to [a,b] \subset \mathbb{R} \tag{4.11}$$

and Ω is a domain representing the connected open subset of a finite-dimensional vector space.
Furthermore, the transformed image $\hat{f}$ is given by

$$\hat{f} : \Omega \to [a,b], \mathbf{x} \mapsto f(T(\mathbf{x})) \tag{4.12}$$

with a geometric transformation $T : \Omega \to \Omega$.

4.6.1 Rotation

A transformation $T_\theta^{\text{rot}} : \mathbb{R}^2 \to \mathbb{R}^2$ with

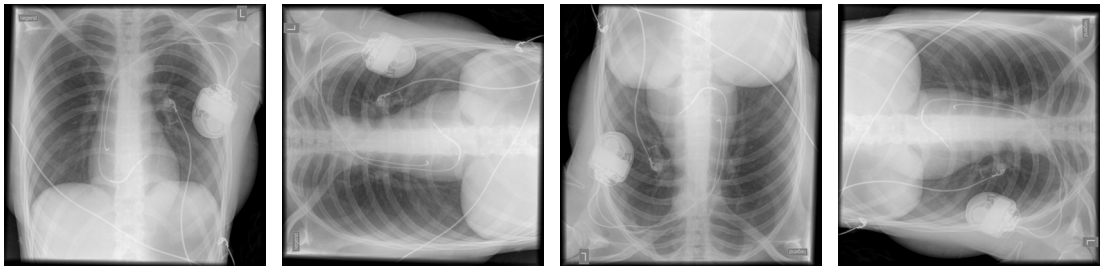
$$T_\theta^{\text{rot}}(\mathbf{x}) = R_\theta \mathbf{x} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{4.13}$$

and $\theta \in [0, 2\pi)$ is called a rotation. As an example, we rotate the images by $\theta = \{90°, 180°, 270°\}$ around the center of the image. Figure 4.8 presents these three rotations on a chest X-ray image.

4.6.2 Reflection

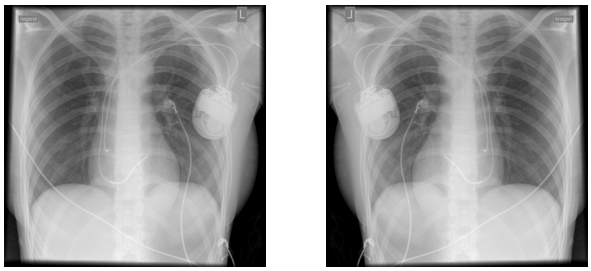
A reflection $T_l^{\text{ref}} : \mathbb{R}^2 \to \mathbb{R}^2$ about the line l having a counterclockwise angle θ with respect to the x-axis is given by

$$T_l^{\text{ref}}(\mathbf{x}) = R_l \mathbf{x} = \begin{pmatrix} \cos(2\theta) & \sin(2\theta) \\ \sin(2\theta) & -\cos(2\theta) \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}. \tag{4.14}$$



(a) Original. **(b)** First. **(c)** Second. **(d)** Third.

Figure 4.8: The image (a) is rotated for data augmentation in 90° steps. Thus, (b) is a rotation by 90°, (c) is a rotation by 180° and (d) is a rotation by 270°.



(a) Original. **(b)** Horizontal flip.

Figure 4.9: Illustration of a reflection for data augmentation. The original image is on the left, while the horizontally flipped (i.e., reflection about the y-axis) image is on the right.

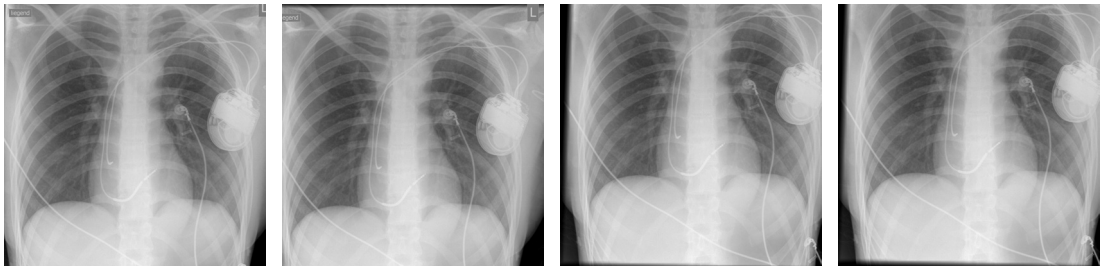
For a geometric transformation, the line l can be the y-axis; therefore, $\theta = 90°$. Hence, Equation (4.14) is simplified by using θ and results in the following:

$$T_l^{\text{ref}}(\mathbf{x}) = R_l\mathbf{x} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} -x_1 \\ x_2 \end{pmatrix}. \tag{4.15}$$

Figure 4.9 shows an example of the reflection transformation in Equation 4.15.

4.6.3 Random cropping

Another data augmentation method involves the random cropping of an input image, which acts like a regularization and increases the dataset by a large factor. Random cropping means that patches with a size $w' \times h'$ are randomly taken from the original image size $w \times h$. While the positions of the patches are random, the patch normally does not exceed the image boundaries [Zheng et al., 2016]. To calculate the position of a patch, two random numbers $w_{\text{off}}, h_{\text{off}}$ (i.e., defining the top left position of the



(a) (b) (c) (d)

Figure 4.10: The image (a) is randomly cropped three times to illustrate the effect of random cropping. (b) - (c) show the results of the random cropping of (a).

patch) are sampled from the integer intervals

$$[0, w_{\text{max,off}}] = \{x \in \mathbb{N} \,|\, 0 \leq x \leq w_{\text{max,off}}\} \text{ and}$$
$$[0, h_{\text{max,off}}] = \{x \in \mathbb{N} \,|\, 0 \leq x \leq h_{\text{max,off}}\}$$

where $w_{\text{max,off}} = w - w'$ and $h_{\text{max,off}} = h - h'$.

The input layer of a neural network often has specific dimensions (i.e., the spatial size of the input layer). For example, ResNet [He et al., 2015a] and VGGNet [Simonyan et al., 2015] have input dimensions of $w' \times h' = 224 \times 224$ and random cropping is performed on images (often downscaled by bilinear interpolation) with the size $w \times h = 256 \times 256$. Thus, we can calculate the interval boundaries $w_{\text{max,off}} = 256 - 224 = 32 = h_{\text{max,off}}$. An example of this method is shown in Figure 4.10. Here, the chest X-ray (a) is randomly cropped three times (b)-(d).

5 Chest X-ray disease classification with convolutional neural networks

This chapter aims to investigate the possible applications of deep learning for chest X-ray classification. Standard network architectures are examined and two new architectures that consider the specifics of chest X-ray data are proposed. First, the size of the input images processed by the convolutional neural network is doubled to address the problem of information loss during downscaling. Secondly, a new network architecture is presented that mimics the workflow of a radiologist by incorporating additional feature information, including the age and gender of the patient and the view position of image acquisition.
Most of the methods and results described in this chapter have been published by Baltruschat et al. [2018b, 2019c].

In computer vision, deep learning has already shown its power for image classification with superhuman accuracy [He et al., 2016; Krizhevsky et al., 2012; Simonyan et al., 2015; Szegedy et al., 2014]. Additionally, the medical image processing field is intensely exploring deep learning. However, one major problem in the medical domain is the availability of large datasets with reliable ground truth annotation. Therefore, transfer learning approaches—as proposed by Bar et al. [2015]—were often considered as a means to overcome such problems.

In 2017, two chest X-ray datasets became available: the OpenI dataset released by Demner-Fushman et al. [2016] and the ChestX-ray14 dataset from the National Institutes of Health Clinical Center [Wang et al., 2017]. Due to its size, the ChestX-ray14 dataset—consisting of 112,120 frontal chest X-ray images from 30,805 unique patients—attracted considerable attention in the deep learning community. Triggered

by the work of Wang et al. [2017] using convolutional neural networks from the computer vision domain, several research groups have begun to address the application of convolutional neural networks for chest X-ray disease classification. Notably, Yao et al. [2017] presented a combination of convolutional neural networks and a recurrent neural network to exploit label dependencies. They used a DenseNet [Huang et al., 2017] model as a convolutional neural network backbone, which was adapted and trained entirely on X-ray data. Li et al. [2017] presented a framework for pathology classification and localization using convolutional neural networks. More recently, Rajpurkar et al. [2017] proposed a transfer learning approach by fine-tuning a DenseNet-121 [Huang et al., 2017] on the ChestX-ray14 dataset, which improved the state-of-the-art AUROC results for multilabel disease classification.

Unfortunately, a faithful comparison of approaches remains difficult. Most reported results were obtained with different experimental setups. This includes (among others) the employed network architecture, loss function, and data augmentation. Additionally, differing dataset splits were used and only Li et al. [2017] reported five-fold cross-validated results. In contrast to these results, the experiments (see Section 5.3) demonstrate that the performance of a network depends significantly on the selected split. To achieve a fair comparison, Wang et al. [2017] published an official split a few months after their initial release of the ChestX-ray14 dataset. Yao et al. [2018] and Guendel et al. [2018] reported results for the official split, while Guendel et al. [2018] achieved state-of-the-art results in all 14 classes with a location-aware DenseNet-121.

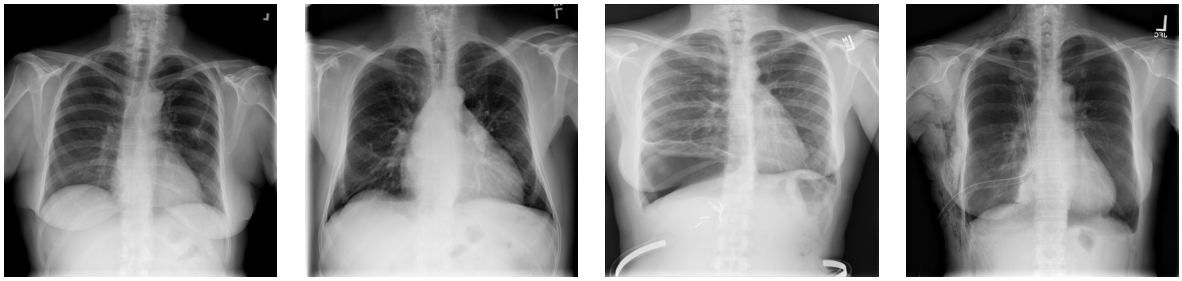
To provide more detailed insights into the effects of distinct design decisions for deep learning, a systematic evaluation using a five-time subsampling scheme is performed. Four major topics are empirically analyzed:

1. Loss functions such as binary cross-entropy (BCE), class-weighted BCE, and positive/negative-weighted BCE (see Section 5.2.1)

2. Weight initialization, pre-training and transfer learning (see Section 5.2.2)

3. Network architectures such as ResNet-50 with large input sizes (see Section 5.2.3)

4. Non-image features such as age, gender, and view position (see Section 5.2.4)



(a) "no finding" **(b)** "cardiomegaly" **(c)** "pneumothorax" **(d)** "pneumothorax"

Figure 5.1: Four examples of the ChestX-ray14 dataset. ChestX-ray14 consists of 112,120 frontal chest X-rays from 30,805 patients. All images were labeled with up to 14 pathologies or "no finding". Under each image (a) to (c), we show the label. The dataset does not only include acute findings, as per the pneumothorax in Figure (c), but also treated patients with a drain as "pneumothorax" (d).

Prior work on ChestX-ray14 has been limited to the analysis of image data. However, radiologists employ a broad range of additional features during diagnosis in clinical practice. To leverage the complete information of the dataset (i.e., age, gender, and view position), a novel architecture integrating this information—in addition to the learned image representation—is proposed in Section 5.2.4.

5.1 ChestX-ray14 dataset

To train and evaluate the approaches for multilabel pathology classification, the entire corpus of ChestX-ray14 is employed. Figure 5.1 illustrates four selected examples from ChestX-ray14. In total, the dataset contains 112,120 frontal chest X-rays from 30,805 patients. The dataset contains only preprocessed images and not the original DICOM images. Wang et al. [2017] performed a simple preprocessing based on the encoded display settings, while the pixel depth was reduced to 8-bit. Additionally, each image was resized to 1024×1024 pixels without preserving the aspect ratio.

In Tables 5.1 and Table 5.2 as well as Figure 5.2, the distribution of each class and statistics for non-image information are shown. The prevalence of individual pathologies was generally low and varied between 0.2 % and 17.74 % (see Table 5.1).The distribution of patient gender and view position was quite even, with a ratio of 1.3 and 1.5, respectively (see Table 5.2). In Figure 5.2, the histogram shows the distribu-

Table 5.1: Summary of disease distribution in the ChestX-ray14 dataset. For each disease, the total number of "true" and "false" (i.e., whether the disease is present or not) and their prevalence are given. The last row shows the number of "true" and "false" items for the implicit label "No Findings"

Pathology	True	False	Prevalence [%] $N = 112,120$
Cardiomegaly	2,776	109,344	2.48
Emphysema	2,516	109,604	2.24
Edema	2,303	109,817	2.05
Hernia	227	111,893	0.20
Pneumothorax	5,302	106,818	4.73
Effusion	13,317	98,803	11.88
Mass	5,782	106,338	5.16
Fibrosis	1,686	110,434	1.50
Atelectasis	11,559	100,561	10.31
Consolidation	4,667	107,453	4.16
Pleural thickening	3,385	108,735	3.02
Nodule	6,331	105,789	5.65
Pneumonia	1,431	110,689	1.28
Infiltration	19,894	92,226	17.74
No findings	60,412	51,700	53.89

tion of patient age in ChestX-ray14. The average patient age was 46.87 years, with a standard deviation of 16.60 years.

5.2 Method

In the following sections, pathology classification is cast as a multilabel classification problem. All images $\mathbb{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}, \mathbf{x}_i \in \mathbb{X}$ were associated with a ground truth label vector $\mathbf{y}_i$, while we sought a classification function $f : \mathbb{X} \rightarrow \mathbb{Y}$ that minimizes a specific loss function L using N training sample-label pairs $(\mathbf{x}_i, \mathbf{y}_i), i = \{1, \ldots, N\}$. Here, the label for each image was encoded as binary vector $\mathbf{y} \in \{0,1\}^M = \mathbb{Y}$ (with M labels). As presented in Section 5.1, ChestX-ray14 usually had 14 labels per image. For the experiments in this thesis, the implicit label "No finding", which means that all other classes are not present in the image, was encoded as an explicit additional label. Hence, the total number of labels in the following experiments is $M = 15$.

Table 5.2: Distribution of patient gender and view position in the ChestX-ray14 dataset. For patien gender, the total count of female and male is show and, for view position, the total count of posterior-anterior (PA) and anterior-posterior (AP) is given. In the third column, the ratio between the first and second columns was calculated.

	Female	Male	Ratio
Patient gender	63,340	48,780	1.30
	PA	AP	Ratio
View position	67,310	44,810	1.50

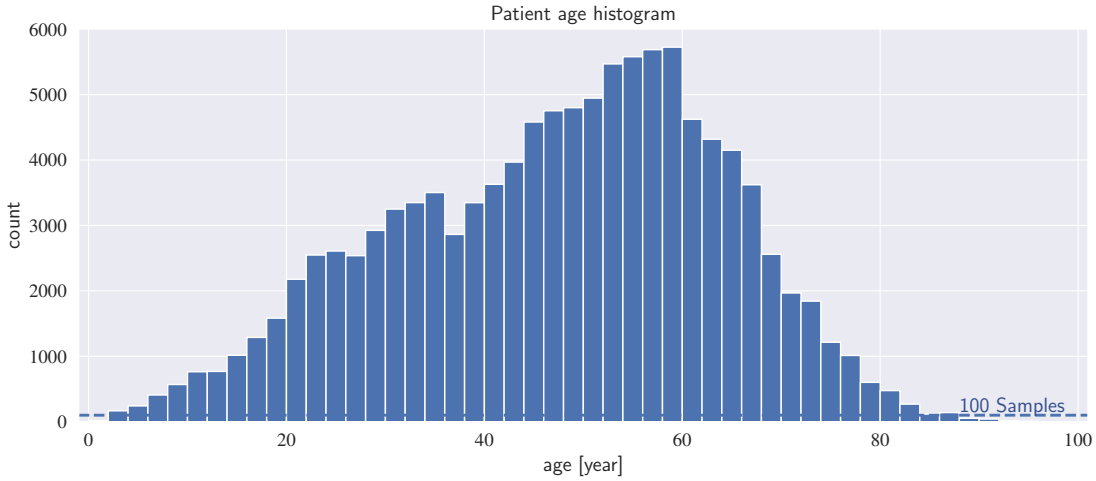


Figure 5.2: Distribution of patient age in the ChestX-ray14 dataset. Each bin covers a width of two years. The average patient age was 46.87 years, with a standard deviation of 16.60 years.

Previous work on the ChestX-ray14 dataset primarily concentrated on ResNet-50 [He et al., 2015a] and DenseNet-121 [Huang et al., 2017] architectures. Due to the excellent performance of ResNet-50 in computer vision [He et al., 2015a], this architecture was chosen for the experiments. The original ResNet-50 was trained on ImageNet [Krizhevsky et al., 2012] to classify 1000 classes. To adapt the network to the new task of multilabel pathology classification, the final fully-connected layer of the original architecture was replaced with a new fully-connected layer. The outputs matched the number of labels M and a sigmoid activation function for the multilabel problem was added (see Table 5.3).

5.2.1 Loss function exploration

Initially, the effect of different weightings for the BCE as a loss function was investigated. The baseline was the class-averaged BCE, which is defined as follows:

$$L_{\text{BCE}}(\mathbf{y}, f(\mathbf{x})) = \frac{1}{M}\sum_{m=1}^{M} H[y_m, f(x_m)] \tag{5.1}$$

with $H[y, f(x)] = -y \log f(x) - (1-y)\log(1-f(x))$.

Next—and similar to [Wang et al., 2017]—a positive/negative balancing was used to counteract the imbalanced distribution of "1"s (i.e., positive) and "0"s (i.e., negative). While Wang et al. [2017] defined two weighting factors based on the training batch, the α_P, α_N was calculated with respect to the whole training set in the present study. Hence, the positive/negative weighted loss function $L_{\text{PN-BCE}}$ is defined as:

$$L_{\text{PN-BCE}}(\mathbf{y}, f(\mathbf{x})) = \alpha_P \sum_{m=1}^{M} -y_m \log f(x_m) + \alpha_N \sum_{m=1}^{M} -(1-y_m)\log(1-f(x_m)) \tag{5.2}$$

where α_P is $\frac{|P|+|N|}{|P|}$ and α_N is $\frac{|P|+|N|}{|N|}$. $|P|$ and $|N|$ are the total number of "1"s and "0"s in the training set, respectively.

Finally, a class-weighted balancing was explored that resembles an oversampling of minority classes. In multiclass classification, it proved to be superior to the unweighted

Table 5.3: Architecture of the original, off-the-shelf, and fine-tuned ResNet-50. In the experiments, the ResNet-50 architecture was used. This table presents the differences between the original architecture and the new one (i.e., off-the-shelf and fine-tuned ResNet-50). If there was no difference from the original network, the word "same" is written in the table. The violet and bold text emphasizes which parts of the network were changed for our application. All layers employed automatic padding (i.e., depending on kernel size) to keep spatial size consistent. The conv3_0, conv4_0, and conv5_0 layers perform a down-sampling of the spatial size with a stride of 2.

Layer name	Output size	Original 50-layer	Off-the-shelf	Fine-tuned
conv1	112 × 112	7 × 7, 64-d, stride 2	Same	**Fine-tuned**
pooling1	56 × 56	3 × 3, 64-d, max pool, stride 2	Same	Same
conv2_x	56 × 56	$\begin{bmatrix} 1\times1,\ 64\text{-d, stride } 1 \\ 3\times3,\ 64\text{-d, stride } 1 \\ 1\times1,\ 256\text{-d, stride } 1 \end{bmatrix} \times 3$	Same	**Fine-tuned**
conv3_0	28 × 28	$\begin{bmatrix} 1\times1,\ 128\text{-d, stride } 2 \\ 3\times3,\ 128\text{-d, stride } 1 \\ 1\times1,\ 512\text{-d, stride } 1 \end{bmatrix}$	Same	**Fine-tuned**
conv3_x	28 × 28	$\begin{bmatrix} 1\times1,\ 128\text{-d, stride } 1 \\ 3\times3,\ 128\text{-d, stride } 1 \\ 1\times1,\ 512\text{-d, stride } 1 \end{bmatrix} \times 3$	Same	**Fine-tuned**
conv4_0	14 × 14	$\begin{bmatrix} 1\times1,\ 256\text{-d, stride } 2 \\ 3\times3,\ 256\text{-d, stride } 1 \\ 1\times1,\ 1024\text{-d, stride } 1 \end{bmatrix}$	Same	**Fine-tuned**
conv4_x	14 × 14	$\begin{bmatrix} 1\times1,\ 256\text{-d, stride } 1 \\ 3\times3,\ 256\text{-d, stride } 1 \\ 1\times1,\ 1024\text{-d, stride } 1 \end{bmatrix} \times 5$	Same	**Fine-tuned**
conv5_0	7 × 7	$\begin{bmatrix} 1\times1,\ 512\text{-d, stride } 2 \\ 3\times3,\ 512\text{-d, stride } 1 \\ 1\times1,\ 2048\text{-d, stride } 1 \end{bmatrix}$	Same	**Fine-tuned**
conv5_x	7 × 7	$\begin{bmatrix} 1\times1,\ 512\text{-d, stride } 1 \\ 3\times3,\ 512\text{-d, stride } 1 \\ 1\times1,\ 2048\text{-d, stride } 1 \end{bmatrix} \times 2$	Same	**Fine-tuned**
pooling2	1 × 1	7 × 7, 2048-d, average pool, stride 1	Same	Same
fully-connected	1 × 1	1000-d, FC-layer	**15-d, FC-layer**	
loss	1 × 1	1000-d, softmax	**15-d, sigmoid, BCE**	

BCE. In the following, the class-weighted BCE $L_{\text{CW-BCE}}$ is defined as:

$$L_{\text{CW-BCE}}(\mathbf{y}, f(\mathbf{x})) = \sum_{m=1}^{M} -\beta_m y_m \log f(x_m) - \beta_m (1 - y_m) \log(1 - f(x_m)) \tag{5.3}$$

where β_m is the inverse class frequency for each class with

$$\beta_m = \begin{cases} \frac{N}{P_m} & \text{if } y_m = 1 \\ \frac{N}{N_m} & \text{if } y_m = 0 \end{cases} . \tag{5.4}$$

Here, P_m is the total count of positives (i.e., "1"s) for class m, N_m is $N - P_m$, and N is the size of the training set.

Based on the results of the loss function exploration (presented in Section 5.3), the best performing loss function was employed in all other experiments.

5.2.2 Weight initialization and transfer learning

Two distinct initialization strategies for ResNet-50 were investigated. First, the same scheme as described by He et al. [2016] was employed, where the network parameters were initialized with random values; thus, the model was trained from scratch. Second, the network was initialized with pretrained weights, where knowledge was transferred from a different domain and task (also known as the transfer learning approach). Such initialization with pretrained weights can be differentiated into *off-the-shelf* (OTS) and *fine-tuning* (FT).

A major drawback in medical image processing with deep learning is the limited size of datasets when compared to the computer vision domain. Hence, training a convolutional neural network from scratch is often not feasible. One solution to this challenge is transfer learning, which can be described by a domain $\mathbb{D}$ and a task $\mathbb{T}$. Following the notation of Pan et al. [2010], a domain $\mathbb{D}$ contains images $\mathbb{X}$ and a marginal probability distribution $P(\mathbb{X})$. A task $\mathbb{T}$ contains labels $\mathbb{Y}$ and a prediction function $f(\cdot)$, which is learned from the training data. Moreover, a source domain $\mathbb{D}_s = \{\mathbb{X}_s, P_s(X_s)\}$ with task $\mathbb{T}_s = \{\mathbb{Y}_s, f_s(\cdot)\}$ and a target domain $\mathbb{D}_t = \{\mathbb{X}_t, P_t(X_t)\}$ with task $\mathbb{T}_t = \{\mathbb{Y}_t, f_t(\cdot)\}$ are given. The constraint to employ transfer learning is $\mathbb{D}_s \neq \mathbb{D}_t$ and/or $\mathbb{T}_s \neq \mathbb{T}_t$. In

transfer learning, the knowledge gained in $\mathbb{D}_s$ and $\mathbb{T}_s$ is used to help learn a prediction function $f_t(\cdot)$ in $\mathbb{D}_t$.

When employing an OTS approach, the pretrained network is used as a feature extractor, and only the weights of the final (classifier) layer are adapted [Razavian et al., 2014; Yosinski et al., 2014]. In fine-tuning, one chooses to re-train one or more layers with samples from the new domain. For both approaches, the weights of a ResNet-50 network trained on ImageNet [Russakovsky et al., 2014] are used as a starting point. In the fine-tuning experiment, all convolutional layers were retrained as shown in Table 5.3.

5.2.3 Architecture adaptations

In addition to the original ResNet-50 architecture, two variants were employed. First, the number of input channels was reduced to one, which reduced the total parameters and facilitated the training of an X-ray-specific convolutional neural network. ResNet-50 was original designed for processing RGB images (i.e., images with three channels for the colors red, green, and yellow) from the ImageNet dataset. Second, the input size was increased by a factor of two (i.e., 448 × 448). A higher resolution could be beneficial for the detection of small structures, which could be indicative of some pathologies (e.g., masses, nodules or pneumothorax). Figure 5.3 shows the severe effect of downscaling a chest X-ray with a pneumothorax to 256×256 pixels by bilinear interpolation (see 5.3 (a, b)). Upon comparing it with a 480 × 480 pixels downscaled version (see 5.3 (c, d)), it becomes clear that the pleura is only visible in the larger image (c, d) and not in the smaller one (a, b).

To maintain similar model architectures, only a new max-pooling layer was added after the first bottleneck block. This max-pooling layer had the same parameters as the "pooling1" layer (i.e., kernel size $k \times k$, $k = 3$, stride $s = 2$, and zero-padding $p = 1$). In Figure 5.4, the changes are illustrated at the image branch. In the following, the postfix "-1channel" and "-large" is used to refer to the model changes.

Finally, of these three models, the best setup was further investigated by changing the model depths.First, a shallower ResNet-38 was implemented where the number of bottleneck blocks for conv2_x, conv3_x, and conv4_x down is reduced to two, two,

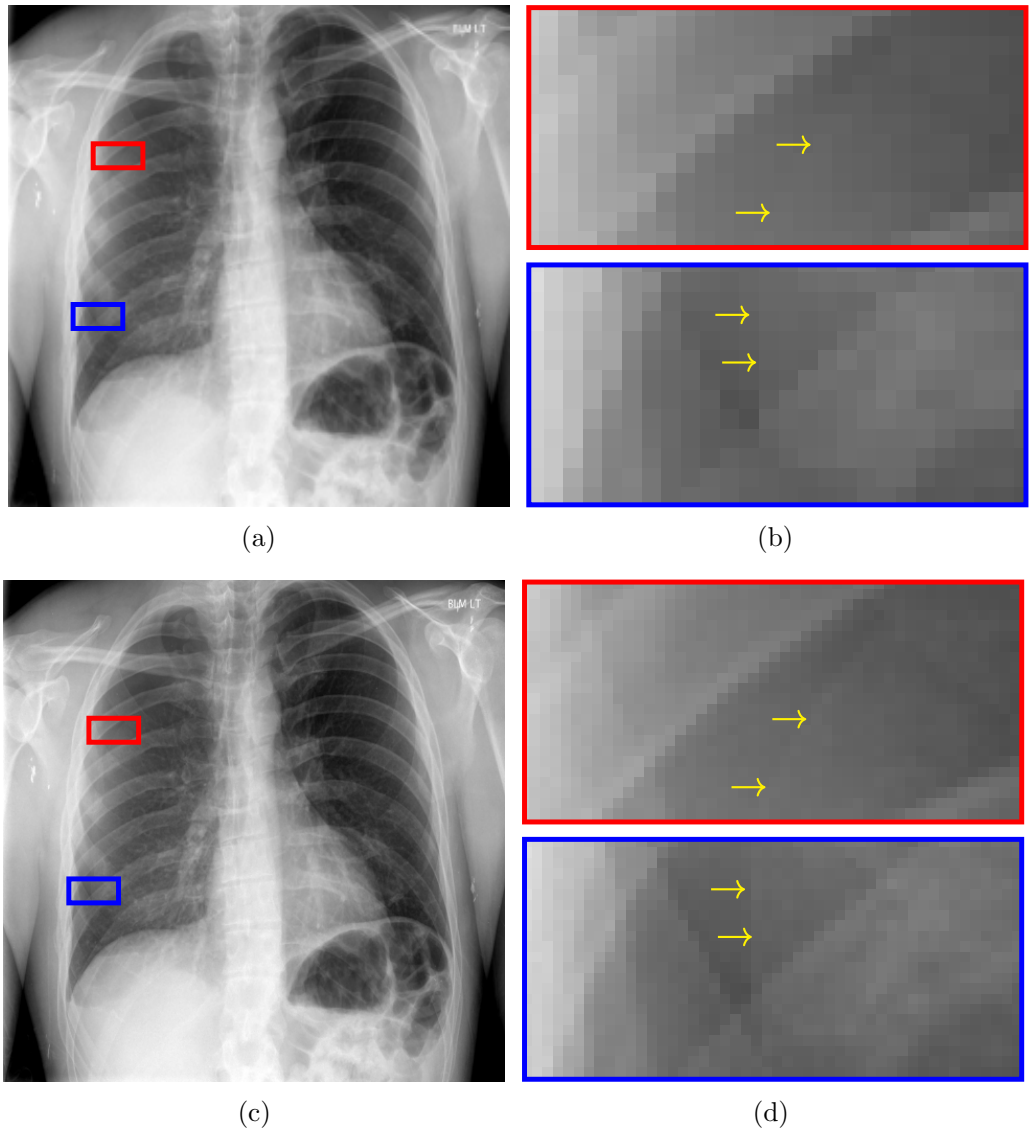


Figure 5.3: Comparison of a low- and medium-resolution chest X-ray based on a pneumothorax. (a) shows a chest X-ray downscaled by bilinear interpolation to an image size of 256 × 256 pixels. In (b), two areas of (a) are shown, magnified by a factor of ten. The yellow arrows point to the location where the pleura should be visible. For comparison, (c) shows the same chest X-ray downscaled to an image size of only 480 × 480 pixels. (d) shows the same magnified areas shown in (b). Again, the arrows indicate the position of the pleura, which is now visible. This example image was taken from the OpenI dataset and has the ID: 3378.

and three, respectively. Secondly, ResNet-101 was tested with the number of conv_3 blocks increased from 5 to 22 (when compared to ResNet-50).

5.2.4 Patient data inclusion

ChestX-ray14 contains information about patient age, gender, and view position (i.e., if the X-ray image is acquired posterior-anterior (PA) or anterior-posterior (AP)). However, radiologists also use information beyond the images to conclude which pathologies are present or not. Notably, the view position changes the expected position of organs in the X-ray images (i.e., PA images are horizontally flipped compared to AP). Additionally, organs (e.g., the heart) are magnified in an AP projection since the distance to the detector is increased.

As illustrated in Figure 5.4, the image feature vector (i.e., output of the last pooling layer with dimensions of 2024×1) was concatenated with the new non-image feature vector (with dimensions of 3×1). The view position and gender were encoded as $\{0, 1\}$. The age was linearly scaled $[\min(X_{\text{pa}}), \max(X_{\text{pa}})] \mapsto [0, 1]$ to avoid a bias toward features with a large range of values. In the experiments, the suffix "-meta" was used to refer to the model architecture with non-image features.

To determine whether the provided non-image features contained information for disease classification, an initial experiment was performed. A very simple multilayer perceptron (MLP) was trained with only the three non-image feature as input. The MLP performed the multilabel classification but with only three non-image features. The result was a low average AUROC of 0.61 for the MLP classifier; however, this indicates that the non-image features could help to improve classification results when provided to the novel model architecture.

5.3 Experiments and results

For an assessment of the generalization performance, a five-time random subsampling scheme was employed, as described in Section 3.7.2.2. Within each split, the data were divided into 70 % training, 10 % validation, and 20 % testing. Since individual

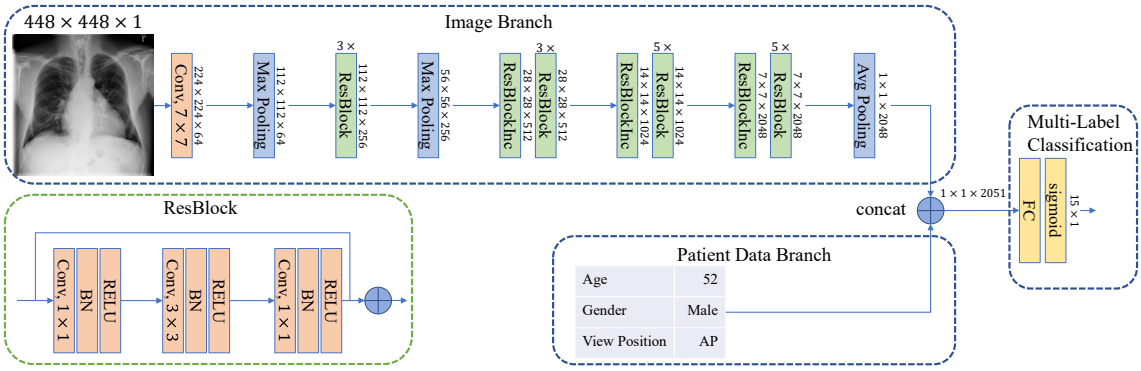


Figure 5.4: Patient data-adapted model architecture: ResNet-50-large-meta. Our architecture is based on the ResNet-50 model. Due to the enlarged input size, we added a max-pooling layer after the first three ResBlocks. Additionally, we fused image features and patient features at the end of our model to incorporate patient information.

patients had multiple follow-up acquisitions, all data from each patient were assigned to a single subset only. This led to a large diversity in patient numbers (e.g., split two had 5,817 patients and 22,420 images, whereas split five had 6,245 patients and the same number of images). The average validation loss over all five random subsamples was used to determine the best epoch e with the lowest error. Now, the models of epoch e were used to calculate the final results for the test sets. All results of the five random subsamples were then averaged. To achieve a fair comparison to other groups, an additional evaluation was conducted. Here, the best performing architecture with different depths was trained on the official split of Wang et al. [2017] (see Section 5.3.1).

Implementation: A fixed setup was used in all experiments. To extend ChestX-ray14, geometric data augmentation was used as described in Section 4.6. At training, patches of the image were sampled with sizes between 8 % and 100 % of the image area, while the aspect ratio of the patches was evenly distributed between $\frac{3}{4}$ and $\frac{4}{3}$. Additionally, horizontal flipping and random rotations between $-7°$ and $7°$ were employed. For validation and testing, images were rescaled to 256×256 and 480×480 pixels for small and large input sizes, respectively. Thereafter, we used the center crop as the input image. As per the work of He et al. [2016], dropout was not employed. ADAM [Kingma et al., 2015] (see Section 3.6) was used as an optimizer with the default parameters for $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate η was set to $\eta = 0.001$ and $\eta = 0.01$ for transfer learning and from scratch, respectively. While training, the learning rate was reduced by a factor of two when the validation

loss did not improve. Due to model architecture variations, the batch size was set to 16 and 8 for transfer learning and from scratch, respectively, with a large input size. The models were implemented in the Computational Network Toolkit (CNTK) from Microsoft [Seide et al., 2016]. CNTK is a computation-graph based deep learning framework for training and evaluating neural networks and is open-source. Moreover, the models were trained on Nvidia GeForce GTX 1080 GPUs with 8 GB of memory. The inference time was approximately 10 ms per image.

Loss function exploration: First, the results for the different weighting schemes of BCE were summarized. Based on the results, the loss function with the highest average AUROC was used for all other experiments.
Three ResNet-50-1 channels were trained from scratch without non-image features. Then, the results presented in Table 5.4 were used to evaluate the performance of the different loss functions. The training with L_{BCE} achieved the highest average AUROC value of 0.822. Compared to the two other loss functions $L_{\text{PN-BCE}}$ and $L_{\text{CW-BCE}}$, this is an increase of 0.49 % and 4.31 %, respectively. The difference between L_{BCE} and $L_{\text{PN-BCE}}$ was only marginal, while the difference to $L_{\text{CW-BCE}}$ was substantial.

Wang et al. [2017] stated that they needed the $L_{\text{PN-BCE}}$ because the training would otherwise be "overwhelmed with 0s and the model barely sees 1s" at training. This was not a problem in the case of the experiment in this thesis because—other than the most—"no finding" was explicitly used as the 15^{th} label. Hence, many 1's are already in the training data. For the experiment with $L_{\text{CW-BCE}}$, the label noise and strong data augmentation (which is necessary to avoid overfitting to the training data) could have been a problem. The large weighting (i.e., between ≈ 17 for "nodules" and ≈ 450 for "hernia") of positives likely causes too much false feedback since many of the labels are incorrect during training.

Based on these results and for the sake of simplicity, L_{BCE} was used in all of the following experiments.

Results of the weight initializations and architecture changes: Table 5.5 summarizes the outcomes of the evaluation. In total, eight different experimental setups with varying weight initialization schemes and network architectures—with and without non-image features—were evaluated. A receiver operating character-

Table 5.4: AUROC results for the loss function experiments. The results for each pathology of a single split are shown. For better comparison, the average AUROC over all pathologies is also presented in the last row. Bold text emphasizes the highest AUROC overall value. The leading 0 was omitted for convenience.

Pathology	L_{BCE}	L_{PN-BCE}	L_{CW-BCE}
Cardiomegaly	**.903**	.900	.889
Emphysema	**.879**	.868	.813
Edema	.891	**.893**	.879
Hernia	**.895**	.882	.856
Pneumothorax	**.855**	.851	.825
Effusion	**.878**	.873	.850
Mass	**.834**	.833	.772
Fibrosis	**.801**	.791	.760
Atelectasis	**.797**	.790	.769
Consolidation	**.804**	.802	.788
Pleural thickening	**.785**	.784	.749
Nodule	**.742**	.740	.678
Pneumonia	**.745**	.742	.722
Infiltration	**.704**	.701	.690
Average	**.822**	.818	.788
No findings	**.772**	.770	.755

Table 5.5: Overview of AUROC results for all experiments. This table presents the averaged results over all five splits and the calculated standard deviation (std) for each pathology. The experiments are divided into three categories: without and with non-image features, transfer learning with off-the-shelf (OTS), and fine-tuned (FT) models from scratch, where "1channel" refers to the same input size as in transfer learning but a different number of channels, while "large" implies that the input size was changed to $448 \times 448 \times 1$. For better comparison, the average AUROC and standard deviation over all pathologies are presented in the last row. Bold text emphasizes the highest overall AUROC value. The leading 0 was omitted for convenience.

	Without non-image features				With non-image features			
Pathology	OTS	FT	1channel	large	OTS	FT	1channel	large
Cardiomegaly	.727 ± .018	.885 ± .007	.889 ± .005	.897 ± .003	.759 ± .014	.884 ± .008	**.902 ± .004**	.898 ± .008
Emphysema	.778 ± .021	.892 ± .010	.870 ± .008	.883 ± .013	.798 ± .019	**.894 ± .012**	.874 ± .013	.891 ± .012
Edema	.844 ± .006	**.891 ± .004**	**.891 ± .006**	.888 ± .005	.857 ± .005	**.891 ± .007**	.890 ± .006	.889 ± .003
Hernia	.788 ± .014	.855 ± .038	.881 ± .042	.875 ± .045	.819 ± .025	.882 ± .032	.893 ± .044	**.896 ± .044**
Pneumothorax	.773 ± .013	**.870 ± .008**	.857 ± .009	.859 ± .009	.791 ± .012	.865 ± .006	.854 ± .007	.859 ± .011
Effusion	.794 ± .004	.871 ± .002	**.876 ± .002**	**.876 ± .002**	.806 ± .004	.872 ± .003	**.876 ± .002**	.873 ± .003
Mass	.668 ± .006	.822 ± .010	.833 ± .006	**.839 ± .009**	.686 ± .006	.822 ± .010	.833 ± .007	.832 ± .003
Fibrosis	.720 ± .009	**.800 ± .009**	.799 ± .008	.792 ± .016	.739 ± .008	**.800 ± .009**	.796 ± .005	.789 ± .005
Atelectasis	.718 ± .006	**.803 ± .007**	.799 ± .004	.792 ± .007	.732 ± .007	.801 ± .006	.793 ± .006	.791 ± .004
Consolidation	.743 ± .003	.795 ± .005	**.806 ± .004**	.800 ± .003	.753 ± .003	.796 ± .005	.804 ± .005	.800 ± .007
Pleural thickening	.688 ± .010	**.790 ± .007**	.784 ± .009	.780 ± .011	.708 ± .011	.786 ± .011	.782 ± .013	.771 ± .013
Nodule	.650 ± .008	.726 ± .009	.733 ± .008	.751 ± .013	.665 ± .007	.747 ± .006	.740 ± .007	**.758 ± .014**
Pneumonia	.664 ± .027	.744 ± .016	.743 ± .015	.753 ± .022	.683 ± .023	.733 ± .013	.748 ± .015	**.767 ± .015**
Infiltration	.659 ± .002	.699 ± .006	**.702 ± .003**	**.702 ± .005**	.670 ± .004	**.702 ± .002**	.701 ± .005	.700 ± .007
Average	.730 ± .011	.817 ± .010	.819 ± .009	.821 ± .012	.748 ± .011	.820 ± .009	.820 ± .010	**.822 ± .011**
No findings	.716 ± .003	.769 ± .005	**.773 ± .003**	.771 ± .004	.725 ± .003	.768 ± .004	.771 ± .004	.771 ± .003

istic curve analysis was performed using AUROC for all pathologies, the classifier scores were compared by Spearman's pairwise rank correlation coefficient [Spearman, 1961], and state-of-the-art method gradient-weighted class activation mapping (Grad-CAM) [Selvaraju et al., 2017] was employed to gain more insight into the trained convolutional neural networks. Grad-CAM is a method for visually assessing the model predictions of convolutional neural networks. This method highlights important regions in the input image for a specific classification result by using the gradient of the final convolutional layer.

The results indicated high variability in the outcome with respect to the selected dataset split. Especially for "Hernia", which is the class with the smallest number of positive samples, a standard deviation of up to 0.05 was observed. As a result, the assessment of existing approaches and comparison of their performance were difficult since prior work mainly focused on a single (random) split.

Regarding the different initialization schemes, acceptable results for OTS networks optimized on natural images were observed. Using fine-tuning techniques, the results were considerably improved (from 0.730 to 0.819 AUROC on average). The complete training of the ResNet-50-1channel using chest X-rays resulted in comparable performance. Only the high-resolution variant of ResNet-50-large outperformed the FT approach by an AUROC of 0.002, on average. For smaller pathologies (e.g., nodules and masses), improvements were observed (i.e., 0.018 and 0.006 AUROC increases, respectively). However, for other pathologies, similar or slightly lower performance was estimated. Finally, all experiments with an architecture including the non-image features observed slight increases in average AUROC when compared to their counterparts without non-image features. The ResNet-50-large-meta (trained from scratch) yielded the best overall performance, with an average AUROC of 0.822.

To gain better insights into why the non-image features only slightly increased the AUROC for the fine-tuned models and those trained from scratch, the capability to predict non-image features based on the extracted image features was investigated. The weights of the model trained from scratch (i.e., ResNet-50-large) were used as the initialization for three additional models. The three models (i.e., ResNet-50-large-age, ResNet-50-large-gender, ResNet-50-large-VP) were used to predict patient age, patient gender, and view position (VP). The same training setup as the OTS experiment was used.

First, the ResNet-50-large-VP model predicted the correct VP with a very high AUROC of 0.9983 ± 0.0002 (i.e., AP is encoded as true and PA as false). After choosing the optimal threshold based on the Youden index [Youden, 1950], sensitivity and specificity were calculated as 99.3 % and 99.1 %, respectively. Secondly, ResNet-50-large-gender also precisely predicted patient gender with a high AUROC of 0.9435 ± 0.0067. The sensitivity and specificity—87.8 % and 85.9 %, respectively)—were also high. Finally, to evaluate the performance of ResNet-50-large-age, the mean absolute error (MAE) and its standard deviation are reported because age prediction is a regression task (see Section 3.3). The model achieved an MAE of 9.13 ± 7.05 years. All three experiments and their results indicate that the image features already encoded information about the non-image features. This could explain why the proposed model architecture with the non-image features at hand did not increase the AUROC performance for multilable disease classification by a large margin.

Furthermore, the correlation between the predictions for individual findings was investigated. The Spearman's rank correlation coefficient was computed for the predictions of all model pairs and averaged over the folds. The pairwise correlation coefficients for the models are provided in Table 5.6. Based on the degree of correlation, three groups can be identified. First, the "from scratch models" (i.e., "1channel" and "large") without non-image features have the highest correlation of 0.93 among each other, followed by the fine-tuned models with 0.81 and 0.80 for "1channel" and "large", respectively. Secondly, the OTS model surprisingly had a higher correlation with the models trained from scratch than the fine-tuned model. Thirdly, no such correlation was observed for models with non-image features, with values between 0.32 and 0.47. This indicates that models trained exclusively on X-ray data achieve not only the highest accuracy but also the greatest consistency.

While the proposed network architecture achieved high AUROC values in all categories of the ChestX-ray14 dataset, the applicability of such technology in a clinical environment considerably depends on the availability of data for model training and evaluation. For the ChestX-ray14 dataset, the reported label noise [Wang et al., 2017] and medical interpretation of the labels represent important issues. As mention by Oakden-Rayner [2017], the class "pneumothorax" is often labeled for already treated pneumothorax cases (i.e., a chest drain is visible in the image) in the ChestX-ray14 dataset. Grad-CAM can be used in this scenario to gain insight into whether the trained convolutional neural network has captured drains as a main feature for "pneumothorax". Grad-CAM visualizes the areas that are most responsible for the final prediction

Table 5.6: Spearman's rank correlation coefficient was calculated between all model pairs and averaged over all five splits. Our experiments are grouped into three categories. First, "Without" and "With" non-image features. Second, transfer learning with off-the-shelf (OTS) and fine-tuned (FT) models. Third, models trained from scratch (i.e., "1channel") have the same input size as in transfer learning but with an altered number of channels. Notably, "large" implies that the input dimensions were changed to $448 \times 448 \times 1$. We identify three clusters: all models under "With", models trained from scratch and "Without", and the "OTS" model.

		Without				With			
		OTS	**FT**	**1channel**	**large**	**OTS**	**FT**	**1channel**	**large**
Without	**OTS**	-	0.65	0.74	0.73	0.46	0.38	0.40	0.59
	FT	0.65	-	0.81	0.80	0.38	0.42	0.43	0.64
	1channel	0.74	0.81	-	0.93	0.41	0.43	0.47	0.71
	large	0.73	0.80	0.93	-	0.40	0.43	0.47	0.71
With	**OTS**	0.46	0.38	0.41	0.40	-	0.32	0.33	0.39
	FT	0.38	0.42	0.43	0.43	0.32	-	0.35	0.42
	1channel	0.40	0.43	0.47	0.47	0.33	0.35	-	0.45
	large	0.59	0.64	0.71	0.71	0.39	0.42	0.45	-

as a heatmap. In Figure 5.5, two examples of our test set are shown. First, the top row shows a correct example where the highest activations are around the pneumothorax. Secondly, the bottom row shows a negative example where the highest activation is around the drain and the network falsely predicted a pneumothorax. This indicates that the network learned not only to detect an acute pneumothorax but also the presence of chest drains. Thus, the utility of the ChestX-ray14 dataset for the development of clinical applications remains an open issue.

5.3.1 Comparison to other approaches

In the evaluation of the experiments in this thesis, a considerable spread (i.e., the difference between the minimum and maximum) of the results in terms of AUROC values was observed. Next to the employed data splits, this could be attributed to the (random) initialization of the models and the stochastic nature of the optimization process.

When ChestX-ray14 was made publicly available, only images—with no official dataset splitting—were released. Hence, the researcher started to train and test their proposed methods on their own dataset splits. For the five different splits used in this thesis,

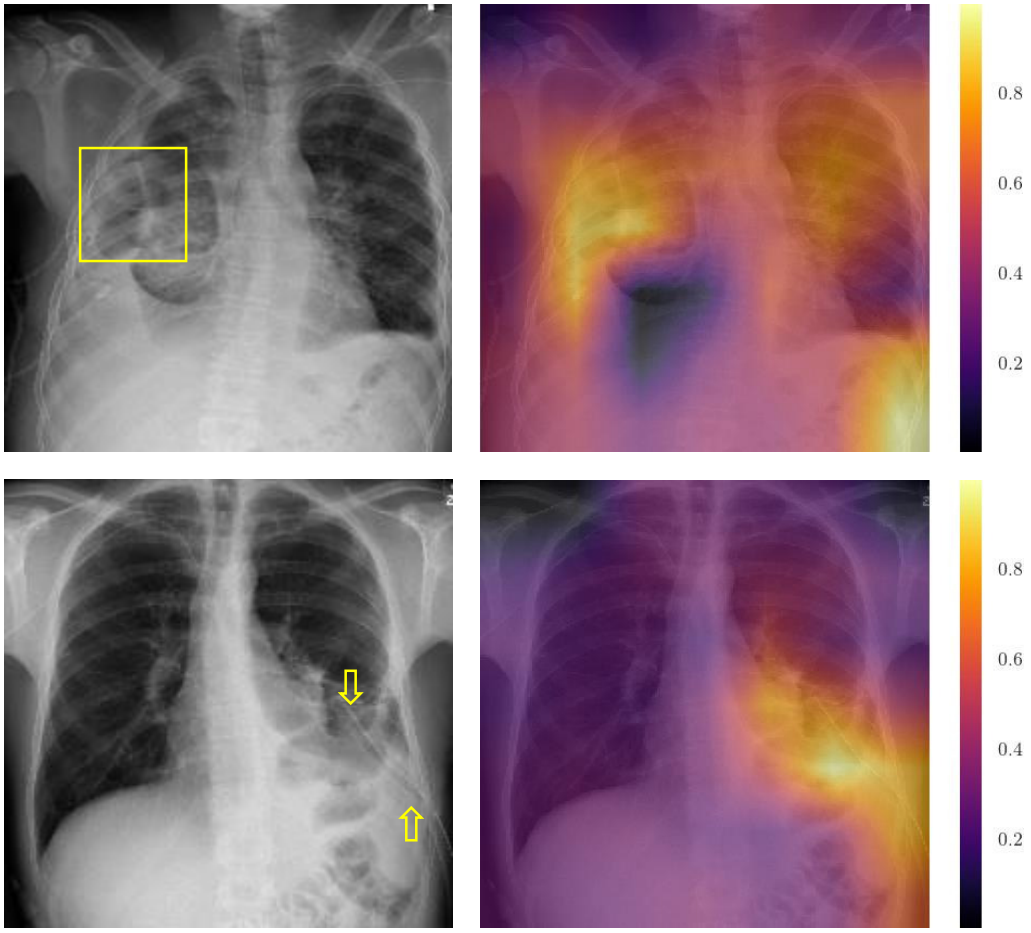


Figure 5.5: Grad-CAM results for two example images. In the top row, the location of the pneumothorax is marked with a yellow box. As shown in the Grad-CAM image next to it, the model's highest activation for the prediction is within the correct area. The second row shows a negative example where the highest activation, which was responsible for the final prediction of "pneumothorax", is at the drain. The drain is marked with yellow arrows. This indicates that the trained convolutional neural network detects drains as a main feature for "pneumothorax".

a large diversity in performance was observed. Therefore, a direct comparison to other groups might be misleading as state-of-the-art results. For example, Rajpurkar et al. [2017] reported state-of-the-art results for all 14 classes on their own split. In Figure 5.6, the best performing model architecture (i.e., ResNet-50-large-meta) of the subsampling experiments is compared to Rajpurkar et al. [2017] and other groups. For the ResNet-50-large-meta model, the minimum and maximum AUROC over all subsampling are plotted as error bars to illustrate the effect of random splitting.

State-of-the-art results for "effusion" and "consolidation" were achieved when directly comparing the AUROC (i.e., averaged over five-time subsampling) to former state-of-the-art results. However, the comparison of maximum AUROC over all subsampling splits resulted in state-of-the-art performance for "effusion", "pneumonia", "consolidation", "edema", and "hernia". This indicates that a fair comparison between groups without the same splitting might be inconclusive.

5.3.2 Official split and model depth

Later, Wang et al. [2017] released an official split of the ChestX-ray14 dataset. To achieve a fair comparison to other groups, the results of this split for the best performing architecture with different depths (i.e., ResNet-38-large-meta, ResNet-50-large-meta, and ResNet-101-large-meta) are report in Table 5.7.

The results were first compared to Wang et al. [2017] and Yao et al. [2018] because Guendel et al. [2018] used an additional dataset—the PLCO dataset [Team PLCO Project et al., 2000]—with 185,000 images. While ResNet-101-large-meta already has a higher average AUROC of 0.785 and (in 12 out of 14 classes) a higher individual AUROC, its performance is lower when compared to ResNet-38-large-meta and ResNet-50-larg-meta. Reducing the number of layers increased the averaged AUROC from 0.785 to 0.795 and 0.806 for ResNet50-large-meta and ResNet38-larg-meta, respectively. Hence, the results indicate that training a model with less parameters on ChestXray14 is beneficial to its overall performance. Secondly, Guendel et al. [2018] reported state-of-the-art results for the official split in all 14 classes with an averaged AUROC of 0.807. Although ResNet-38-large-meta is trained with 185,000 fewer images, it still achieved state-of-the-art results for "Emphysema", "Edema", "Hernia", "Consolidation", and "Pleural thickening" at a slightly lower average AUROC of 0.806.

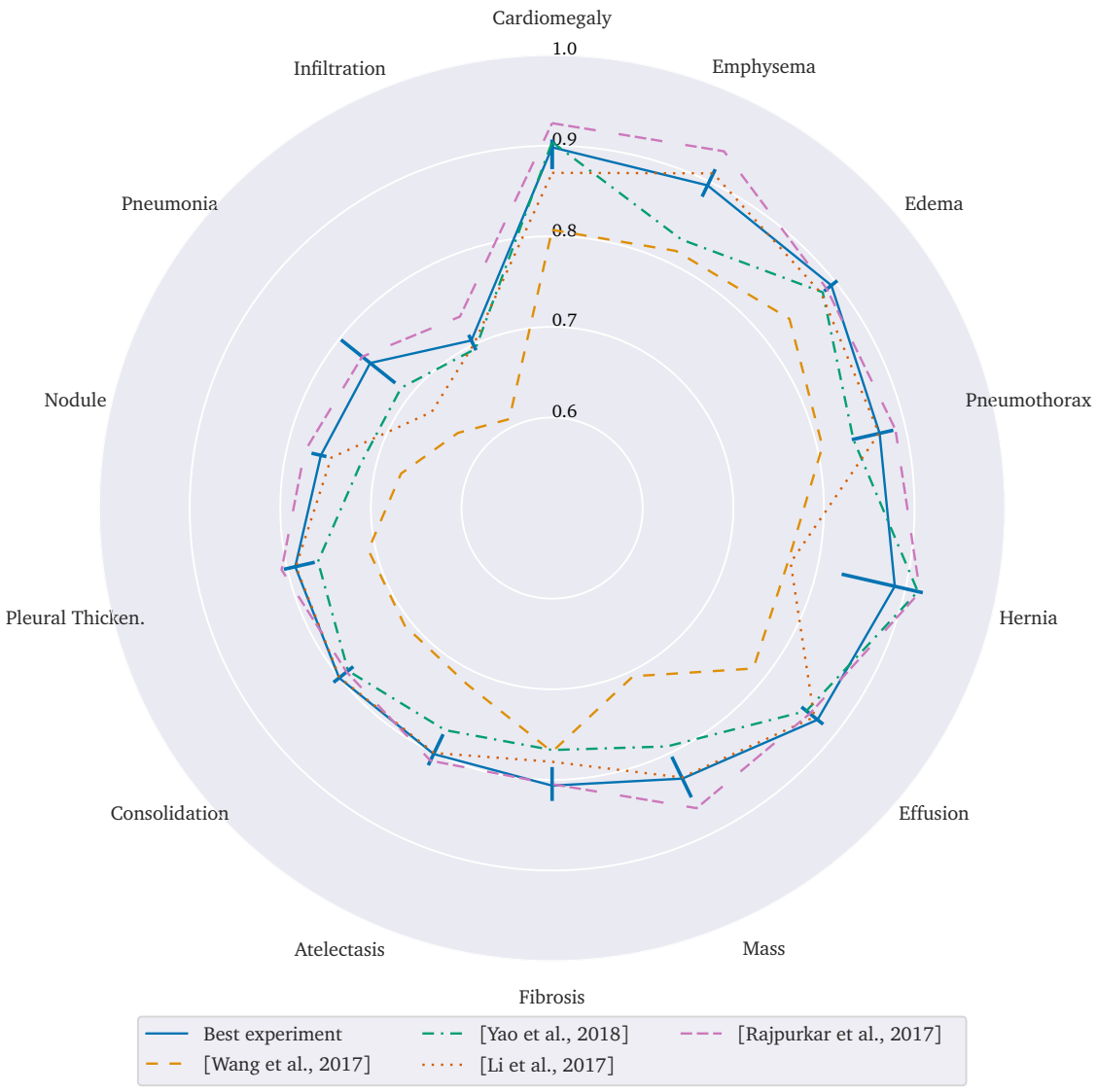


Figure 5.6: Comparison of the best model in this thesis to other groups. The pathologies were sorted with increasing average AUROC over all groups. For model presented in this thesis, the minimum and maximum AUROC over all folds are reported as error bar to illustrate the effect of random dataset splitting.

Table 5.7: Overview of AUROC results for experiments on the official split. In this table, results for the best performing architecture in this thesis with different depths (i.e., ResNet38-large-meta, ResNet50-large-meta, and ResNet101-large-meta) are presented and then compared to other groups. The average AUROC over all pathologies is provided in the final row. Bold text emphasizes the highest overall AUROC value.

				"large-meta"		
Pathology	[Wang et al., 2017]	[Yao et al., 2018]	[Guendel et al., 2018]	ResNet-38	ResNet-50	ResNet-101
Cardiomegaly	0.810	0.856	**0.883**	0.875	0.877	0.865
Emphysema	0.833	0.842	**0.895**	**0.895**	0.875	0.868
Edema	0.805	0.806	0.835	**0.846**	0.842	0.828
Hernia	0.872	0.775	0.896	**0.937**	0.916	0.855
Pneumothorax	0.799	0.805	**0.846**	0.840	0.819	0.839
Effusion	0.759	0.806	**0.828**	0.822	0.818	0.818
Mass	0.693	0.777	**0.821**	0.820	0.810	0.796
Fibrosis	0.786	0.743	**0.818**	0.816	0.800	0.778
Atelectasis	0.700	0.733	**0.767**	0.763	0.755	0.747
Consolidation	0.703	0.711	0.745	**0.749**	0.742	0.734
Pleural thickening	0.684	0.724	0.761	**0.763**	0.742	0.739
Nodule	0.669	0.724	**0.758**	0.747	0.736	0.738
Pneumonia	0.658	0.684	**0.731**	0.714	0.703	0.694
Infiltration	0.661	0.673	**0.709**	0.694	0.694	0.686
Average	0.745	0.761	**0.807**	0.806	0.795	0.785
No findings	-	-	-	0.727	0.725	0.720

5.4 Discussion

The optimized ResNet-38-large-meta architecture achieved state-of-the-art results in 5 out of 14 classes when compared to Guendel et al. [2018] (who achieved state-of-the-art results in all 14 classes on the official split). Notably, even higher scores are reported for other classes in the literature (e.g., [Rajpurkar et al., 2017]). However, a comparison of the different convolutional neural network methods with respect to their performance is inherently difficult since most evaluations have been performed on individual (random) dataset splittings. Substantial variability in the results was observed when different splits were considered. This was especially apparent for "Hernia", the class with the fewest samples in the dataset (see also Figure 5.6).

While the obtained results suggest that the training of deep neural networks in the medical domain is a viable option as more and more public datasets become available, the practical use of deep learning in clinical practice remains an open issue. For the ChestX-ray14 datasets, the rather high label noise of 10 % [Wang et al., 2017] makes the assessment of the true network performance difficult. Therefore, a clean test set without label noise is required for clinical impact evaluation. In addition to the presence of treated findings, Oakden-Rayner [2017] noted that the quality of (automatically generated) labels and their precise medical interpretation may be limiting factors. The Grad-CAM results in this thesis support Oakden-Rayner [2017] concerns regarding the "pneumothorax" label. A neural network trained solely on ChestX-ray14 would also respond to cases with a chest drain. However, in a clinical setting (i.e., for the detection of critical findings), the focus would be on the reliable identification of acute cases of pneumothorax. A solution to this problem might be to first remove all images with a chest drain that are simultaneously labeled as "pneumothorax" and then train on this clean dataset only.

5.5 Summary

This chapter presents a systematic evaluation of different approaches and model changes for multilabel chest X-ray disease classification using convolutional neural networks. While satisfactory results were obtained with neural networks optimized on the ImageNet dataset, the best overall results were reported for the proposed model

architecture incorporating non-image data (i.e., view position, patient age, and gender) and trained exclusively with chest X-rays.

First, common approaches for deep learning when working on a small- to medium-sized dataset were investigated (i.e., transfer learning). Thereafter, it was shown that the novel model architecture with large input size and non-image feature incorporation is superior to the baseline and other groups. Next, Grad-CAM was employed to investigate the trained models for the pathology "pneumothorax". Notably, it was found that the model uses medical tubes as a strong feature for classification, which makes its application for some clinical use cases questionable.

The next chapter deals with the problem of labels generated by natural language processing and also presents advanced preprocessing techniques that support the training of convolutional neural networks. Moreover, the advantages of ensembles are demonstrated by simultaneously utilizing all positive aspects of the different advanced preprocessing techniques.

6 Advanced preprocessing for convolutional neural networks

The objective of this chapter is to provide novel insights into two important areas when working with chest X-ray data: advanced preprocessing of chest X-ray images and retrospective labeling of chest X-ray datasets.
First, two preprocessing methods are proposed—bone suppression and lung field cropping—to improve the AUROC results of a convolutional neural network for multilabel classification. The contribution is a novel ensemble that uses the various information available through advanced preprocessing. Secondly, the retrospective labeling of a chest X-ray dataset by several radiologists is a complicated and time-consuming process. This chapter highlights and discusses some problems that arise during the process of retrospective labeling. For the experiment, multiple radiologists attempted to create retrospective labels with minimal noise for the OpenI dataset (with 3,125 images). As a result, one of the largest sets of manual annotations for a publicly available dataset was created.
Most of the methods and results described in this chapter have been published by Baltruschat et al. [2019a], Baltruschat et al. [2019d, 2019e], Baltruschat et al. [2019f], Grass et al. [2019], Ittrich et al. [2018], Steinmeister et al. [2019].

Recent developments in pathology classification have mainly focused on specific aspects of deep learning (e.g., novel network architectures). Early on, Shin et al. [2016] demonstrated that a convolutional neural network combined with a recurrent part can be applied for image captioning in chest X-rays. The increased availability of annotated chest X-ray datasets such as ChestX-ray14 [Wang et al., 2017] helped to accelerate progress in the field of pathology classification, detection, and localization.

In this rapidly evolving field, Li et al. [2017] presented a unified network architecture for pathology classification and localization, in which only limited annotation is required for localization. Moreover, Cai et al. [2018] proposed an attention mining

method for disease localization that works without localization annotation. Additionally, Wang et al. [2018] presented a classification and reporting method that involves leveraging radiologist reports in addition to images. Putha et al. [2018] demonstrated the effect of a very large dataset (with 1.2 million images) on pathology classification with convolutional neural networks.

In this context, only very simple preprocessing steps have been employed. Using preprocessing methods can reduce variation in image appearance, which facilitates a good approximation of the mapping function (i.e., low generalization error) by training a neural network. Notably, a sufficiently large dataset with $m \to \infty$ can make preprocessing obsolete since the additional data helps neural networks become invariant to variations in image appearance.

Motivated by prior work in the computer vision domain, preprocessing steps predominantly include intensity normalization as well as a rescaling of images to a model's input size. However, several methods have been developed in recent years to support radiologists in the diagnostic process. Two well-known techniques include bone suppression and lung field segmentation [von Berg et al., 2016; von Berg et al., 2015]. Bone suppression artificially removes the rib cage to facilitate the detection of small pathologies, while lung field segmentation can be used to standardize image appearance. The benefits of such image processing methods for various diseases have been shown in multiple studies [Li et al., 2012]. However, an obvious question arises: do bone suppression and lung field segmentation have the same beneficial effect on disease classification with convolutional neural networks?

One of the largest publicly available datasets—Chest X-ray14 [Wang et al., 2017] (as discussed in Sections 2.3 and 5.1)—is limited due to it consisting of down-sampled 8-bit PNG-images and having labels created by natural language processing. As stated by Wang et al. [2017], the generated labels are noisy (i.e., some of the generated labels are false) due to the natural language processing involved. Training convolutional neural network models on mid-sized datasets with noisy labels can degrade generalization performance. Furthermore, a final performance evaluation with noisy labeled data has an upper bound, depending on the noise.

To address these shortcomings, the OpenI dataset [Demner-Fushman et al., 2016] was used in this chapter. Moreover, two expert radiologists created labels via manual annotation. This technique has two beneficial properties: the reduction of errors

stemming from an NLP-based analysis and avoiding misreadings by including multiple readers. The OpenI dataset contains 3,996 DICOM images all from different patients and already provides some manually generated labels. Unfortunately, these labels were created for image retrieval and not for the development of computer-aided detection systems. Therefore, to obtain a DICOM image dataset with appropriate annotations for supervised training, two radiologists from the radiology department of the University Medical Center Hamburg-Eppendorf annotated the entire OpenI dataset. After initial consultation with the radiologists, the eight most important (i.e., for the University Medical Center Hamburg-Eppendorf) findings were annotated: pleural effusion, infiltrate, congestion, atelectasis, pneumothorax, cardiomegaly, mass/nodule (grouped together for the sake of simplicity), and foreign object.

In summary, this chapter evaluates the effect of advanced image preprocessing methods on training CNNs with a small dataset and on disease classification with CNNs. The effects of three methods are empirically analyzed:

1. Bone suppression (Section 6.1.1)

2. Lung field cropping (Section 6.1.2)

3. Ensembles with and without preprocessed trained models (Section 6.1.3)

In a methodologically comparable way to [Gordienko et al., 2018], the preprocessing methods are applied in three different scenarios: processing each image with bone suppression, cropping the images to segmented lung fields, and combining both processing steps. However, lung field segmentation was used to crop the images to the important area, whereas Gordienko et al. [2018] kept the image size equal and only set regions not belonging to the lung fields to zero. The hypothesis is that cropping increases convolutional neural network performance because it increases the effective spatial resolution of input images since the downscaling factor is smaller. Downscaling is required to reduce the original image size to the input size of the convolutional neural network. Furthermore, a novel ensemble architecture is proposed to leverage the complementary information of the different processed images, similar to a radiologist's workflow.

6.1 Method

Following the method and training setup in Chapter 5 [Baltruschat et al., 2019c], a ResNet-50 architecture with a larger input size of 448×448 pixels was used in the following experiments. The network was first pretrained on ChestX-ray14 before a final fine-tuning was performed on OpenI. Compared to existing network architectures and training strategies, the obtained model achieved the highest average AUROC value in the previous experiments (see Section 5.4). Due to the focus on eight specific pathologies, the last fully-connected layer of the converged model was replaced with a new fully-connected layer with eight outputs and a sigmoid activation function. Furthermore, fine-tuning was used to adapt the model to the new image domain and task.

6.1.1 Bone suppression

In a reported reader study [von Berg et al., 2015], the AUROC for the detection and localization of lung nodules increased for experienced human readers when using bone suppression images. Thus, deep learning with convolutional neural networks may also potentially benefit from suppressing a certain normal anatomy, which is tested in this thesis.

In the original OpenI images, the bones (i.e., ribs and clavicles) overlapping with the lung field are suppressed using the method presented by von Berg et al. [2016]. The goal was to generate I_{soft} by $I_{\text{soft}} = I - I_{\text{bone}}$ and to preserve the remaining details that were originally overlaid with the bones. The basic idea of this method is that we can take advantage of the fact that the original image I is a projected image (i.e., a superposition of several signals). This implies that I can be defined as $I = I_{\text{bone}} + I_{\text{soft}}$, where I_{bone} is the bone structure in the image and I_{soft} is the soft tissue.

The method of von Berg et al. [2016] uses five main steps to determine I_{bone} from the original image I so that I_{soft} can be calculated. First, an ST transformation $T_C : (x, y) \mapsto (s, t)$ is employed on I, which makes the contour of the ribs and clavicles appear as a straight line in ST space. Secondly, the partial derivative with respect to s of the transformed image $I_{\text{ST}} = T_C(I)$ is calculated as follows:

$$I'_{\text{ST}} = \frac{\partial I_{\text{ST}}}{\partial s} \,. \tag{6.1}$$

Since the bone edge is orientated orthogonal to s in the ST space, it substantially contributes to I'_{ST}, whereas any structure not orthogonal to s is suppressed. In the third step, a non-rotated anisotropic Gaussian filter $G_{\sigma_s,\sigma_t}(s,t)$ is used to further suppress non-bone structures. This filter is given by:

$$G_{\sigma_s,\sigma_t}(s,t) = \frac{1}{2\pi\sigma_s\sigma_t}\exp\left(-\frac{1}{2}\left(\frac{s^2}{\sigma_s^2}+\frac{t^2}{\sigma_t^2}\right)\right) \tag{6.2}$$

where σ_s and σ_t are the standard deviations that determine the size of the ellipsoid. Then, convolving the filter with the partial derivative I'_{ST} results in:

$$I^G_{\mathrm{ST}} = I'_{\mathrm{ST}} * G_{\sigma_s,\sigma_t} \ . \tag{6.3}$$

Since main steps to extract bone information from the original images are complete, one can invert the partial derivative and the ST transform. Thus, the fourth step involves the calculation of the integral:

$$I^R_{\mathrm{ST}}(s,t) = I_{\mathrm{ST}}(s_0,t) + \int_{s_0}^{s} I^G_{\mathrm{ST}}(s,t)ds \ . \tag{6.4}$$

Finally, the processed and integrated image $I^R_{\mathrm{ST}}(s,t)$ is transformed back into the image space by the inverse ST transformation T_c^{-1}. Since earlier processing steps may introduce artifacts like negative pixel values, the final image is clipped by zero using a max operation:

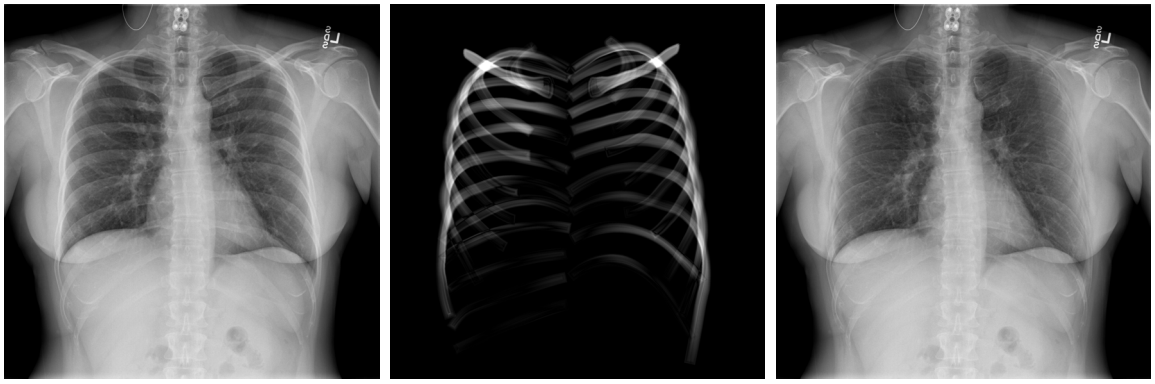
$$I_{\mathrm{bone}} = \max(0, T_c^{-1}(I^R_{\mathrm{ST}})) \ . \tag{6.5}$$

Figure 6.1 shows an example of a soft tissue image I_{soft} being generated by subtracting the bone structure I_{bone} from the original image I.

6.1.2 Lung field segmentation and cropping

Lung field cropping has two beneficial aspects. First, it reduces the amount of information loss due to downscaling via bilinear interpolation, which can be helpful for diseases with small appearances. Secondly, it also acts as a geometric normalization and can thus help in training a neural network using a small dataset.

Notably, segmentation of the lung field was first required. Then, the original images



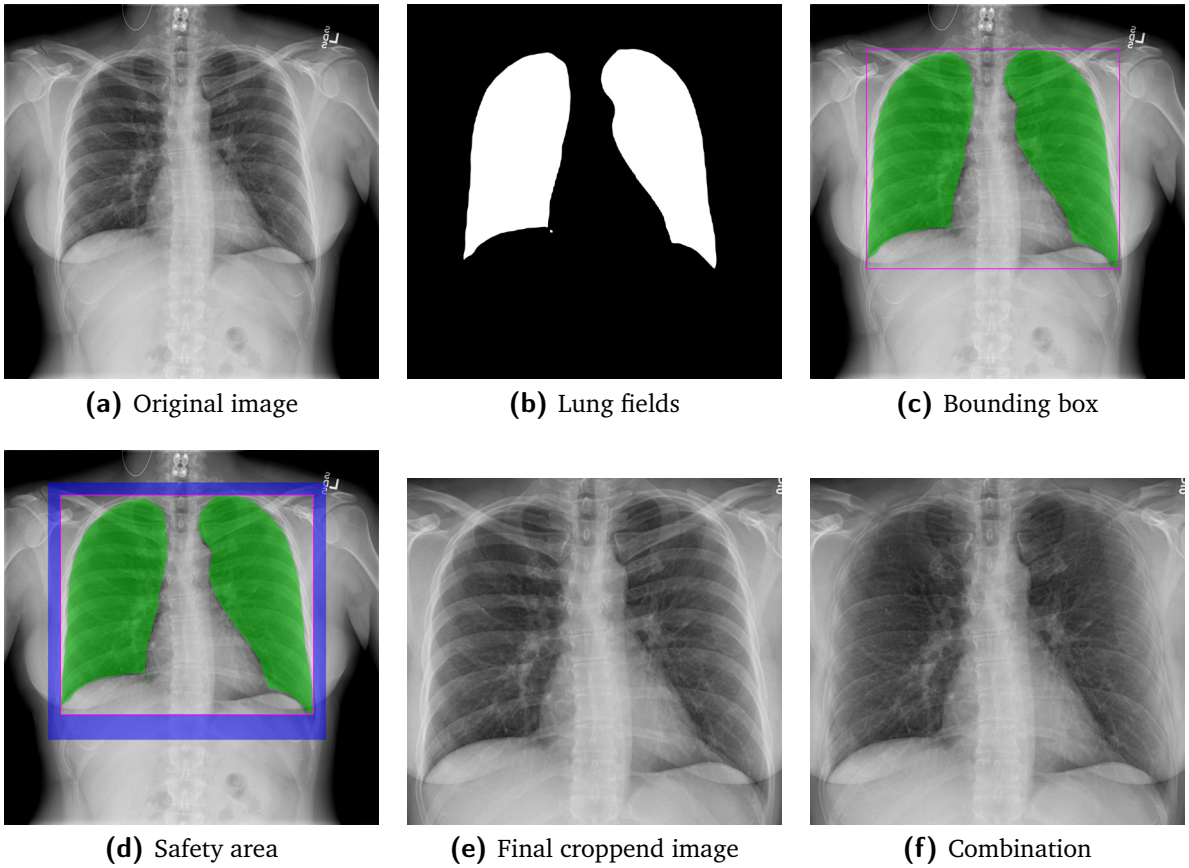
(a) Original image **(b)** Bone structure **(c)** Soft tissue image

Figure 6.1: The bone suppression method [von Berg et al., 2016] removes the bone structure (b) from the original image (a) by subtraction. The result is a soft tissue image (c) without bones.

could be cropped to the lung fields. To segment the lung fields (see Figure 6.2 (b)), a foveal convolutional neural network [Brosch et al., 2018] was used.

This neural network combined local information gained from high-resolution images with context from lower resolution images. This was done by directly providing multiple images with different resolution scales as input to the model. For lung field segmentation, an architecture with four inputs of different resolution levels was used [Sital et al., 2020]. Hence, each original image was processed by creating four images with different resolutions. The input to each resolution level was then processed by a feature extraction path, where each path was composed of three blocks. A single block was constructed by a convolutional layer, batch normalization layer, and an activation function [Brosch et al., 2018]. The outputs of all feature extraction paths were then combined in a single upsampling path [Brosch et al., 2018]. Starting with the lowest resolution output, the extracted features were processed through an additional block and then upsampled to the image size of the next feature extraction path. At the second-lowest resolution level, the extracted features were concatenated with the upsampled output of the layer before it [Sital et al., 2020]. This was repeated for each resolution level. For the last level, only a final convolutional layer with a softmax activation function was used to produce a segmentation probability map [Brosch et al., 2018]. The network was trained by semi-automatically annotated lung fields and then applied to the OpenI images.

After the initial lung field segmentation, post-processing steps were used to determine the final cropping area. First, all connected regions with eight connectivity pixels



(a) Original image **(b)** Lung fields **(c)** Bounding box

(d) Safety area **(e)** Final croppend image **(f)** Combination

Figure 6.2: Overview of the lung field cropping method. The original chest X-ray image (a) was processed by a foveal convolutional neural network to generate the lung field segmentation (b). (c) presents the calculated bounding box around the two largest connected regions in the color violet. In (d), the blue area emphasizes the safety area of the bounding box due to errors in the segmentation mask. (e) shows the final cropped image and (f) presents the combination of bone suppression and lung field cropping.

were identified. Since the two largest regions are most likely the left and right lungs, an initial bounding box around these two regions was calculated (see Figure 6.2 (c)). Thereafter, a small safety border of 100 pixels was added to the initial bounding box at the top, left, and right. To the bottom of the bounding box, a larger border of 200 pixels was added (see Figure 6.2 (d)). As a preprocessing step, each image was cropped to its individual bounding box (see Figure 6.2 (e)). In this thesis, the combination of bone suppression and lung field cropping was also considered (see Figure 6.2 (f)).

6.1.3 Ensemble with advanced preprocessed images

In many applications, combining different predictors can lead to improved classification results, which is known as ensemble generation [Hansen et al., 1990; Krogh et al., 1995]. Ensembling can be done in several ways and with any number of predictors. To determine whether the combination of several predictors could improve results, the Pearson correlation coefficient can be used. Ensembling predictors with a high correlation coefficient is unlikely to greatly improve results when compared to predictors with lower correlations. Methods for ensemble generation (i.e., combining the predictions of multiple predictors) include averaging, majority voting, and machine learning algorithms such as support vector machines.

This thesis focuses on using the averaging approach to limit the complexity of the experimental setup because the dataset used for the experiment is small. Since an ensemble approach will typically outperform an individual model, the individual models are not directly compared to an ensemble. Instead, the ensemble (EN-preprocessed, shown in Figure 6.3) was compared to another ensemble (EN-normal) built with four models trained on images without advanced preprocessing. The EN-preprocessed ensemble was built from models trained on the three different preprocessed images and the original images. This ensemble method could use all of the information available from different image types. Figure 6.3 presents an overview of our ensemble method, in which we combined four models that were trained on our four differently preprocessed input images.

6.2 OpenI dataset

The OpenI dataset contains 3,996 studies with DICOM images [Demner-Fushman et al., 2016]. In the first step, a revised dataset was created by removing studies with no associated images or labels (i.e., reference annotation). Next, studies that lacked either frontal or lateral acquisition were removed. Thus, the final dataset consisted of 3,125 studies.

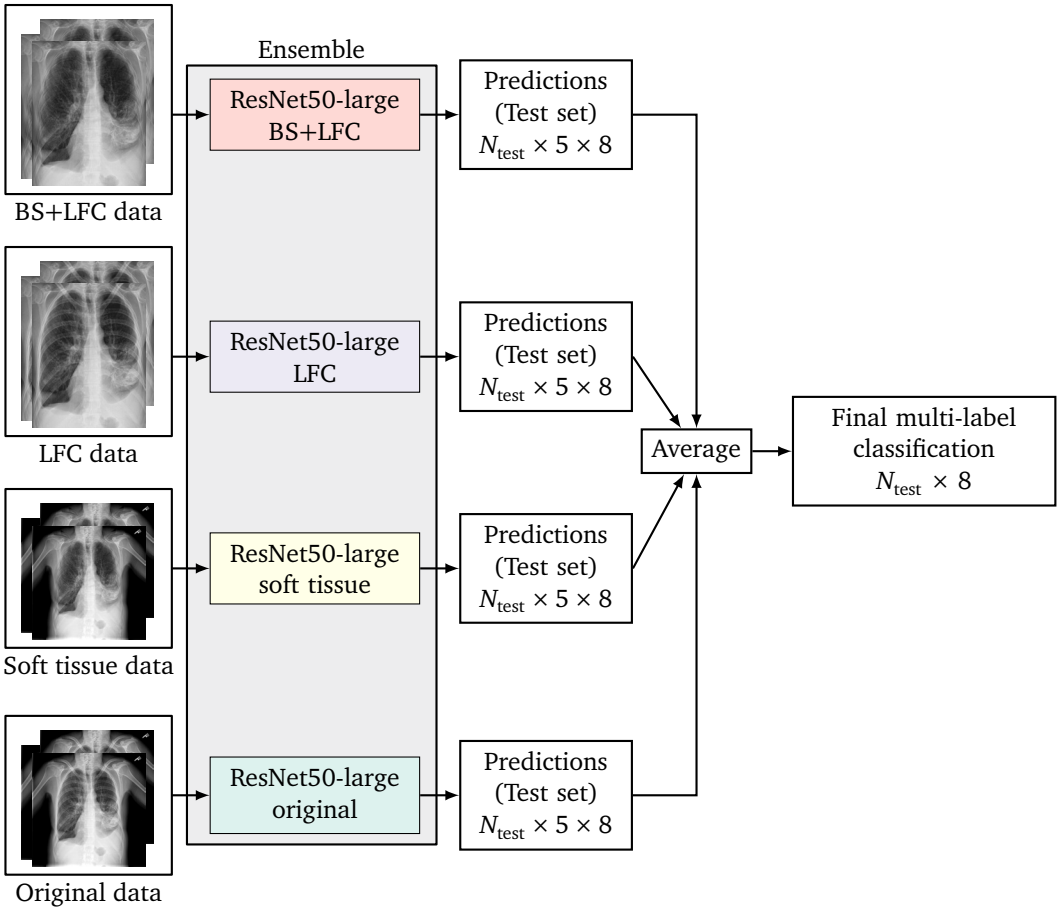


Figure 6.3: Ensemble method used to combine advanced preprocessed images. Four ResNet50-large models were trained on different image data: original, soft tissue, lung field cropped (LFC), and bone suppressed (BS) with LFC. Each model predicted the score for five cropped images (i.e., center and all four corners) in the test set N_{test} with eight classes. Thereafter, the predicted scores from all models were averaged to obtain the final multilabel classification result.

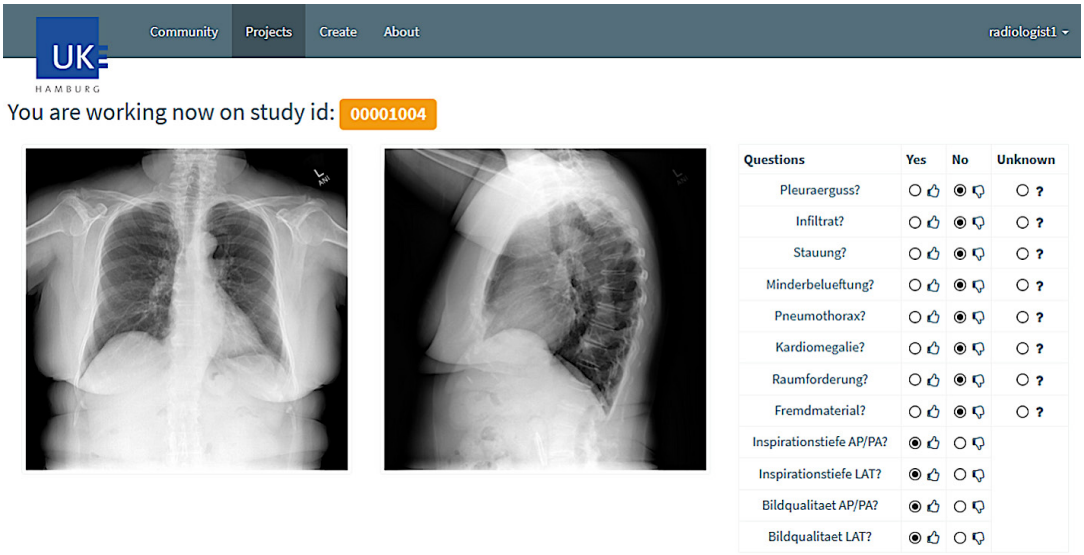


Figure 6.4: Web-based annotation tool for eight representative classes (i.e., pleural effusion, pneumothorax, infiltrate, congestion, atelectasis, cardiomegaly, mass/nodule, and foreign body). Radiologists had to actively indicate the presence of a finding with a "Yes", "No", or "Unknown" decision. For consensus building, labeling discrepancies between annotations from both radiologists were later highlighted as "mismatch" in the first column.

6.2.1 Annotation process

Two expert radiologists from the radiology department of the University Medical Center Hamburg-Eppendorf—radiologist1 and radiologist2, with 3 and 18 years of experience, respectively—reviewed all 3,125 cases. For the annotation process, the frontal and lateral CXRs were used by the radiologists. Annotations contained eight representative classes, which were considered to be most relevant in clinical practice: pleural effusion, infiltrate, congestion, atelectasis, pneumothorax, cardiomegaly, mass/nodule, and foreign object (i.e., all artificial objects like peacemaker, tubes or markers). Less frequent pathologies (e.g., pneumomediastinum or pneumoperitoneum) were not included in the annotation, which could potentially lead to severe complications. Reading was performed on a diagnostic workstation, while the radiologists were required to actively indicate the presence of a pathology in the implemented web-based annotation tool (see Figure 6.4).

As discussed in Section 2.4.2, high inter-rater variability is often observed in chest X-ray diagnosis and is especially high for the annotation process. Therefore, all examinations with labeling discrepancies between the two radiologists were jointly re-evaluated by both radiologists to establish a final consensus annotation for the ground

Table 6.1: Summary of the statistical distribution of all eight classes for the reannotated OpenI dataset.

Finding	True	False	Prevalence [%] $N = 3,125$
Pleural effusion	147	2,978	4.7
Infiltrate	152	2,973	4.9
Congestion	170	2,955	5.4
Atelectasis	212	2,913	6.8
Pneumothorax	11	3,114	0.4
Cardiomegaly	529	2,596	16.9
Mass/nodule	447	2,678	14.3
Foreign object	1,121	2,004	35.9
No findings	1,345	1,780	43.0

truth. Table 6.1 presents the distribution of each finding. As is common in the medical domain, most of the classes have a low prevalence. All findings except pneumothorax have more than 100 positive cases, whereas the finding pneumothorax has only 11 positive cases. The standard supervised training of a neural network with 11 cases is not possible. Therefore, the final evaluation (see Section 6.3) reports the results for pneumothorax only for completeness but does not discuss them.

6.2.2 Inter-observer variability

We evaluated the inter-rater reliability by using Cohen's kappa coefficient (κ) to determine the overall agreement between the two radiologists for each binary outcome (i.e., the presence or absence of pneumothorax), while "unknown" labels were excluded from analysis. The interpretation of kappa statistics was performed according to the benchmarks proposed by Landis et al. [1977] to classify the strength of agreement: poor (< 0.00), slight (0.00 to 0.20), fair (0.21 to 0.40), moderate (0.41 to 0.60), substantial (0.61 to 0.80), and almost perfect (0.81 to 1.00).

The analysis of inter-rater reliability revealed a substantial agreement between both radiologists for foreign object ($\kappa = 0.72$) and pleural effusion ($\kappa = 0.64$), a moderate agreement for pneumothorax ($\kappa = 0.55$), mass/nodule ($\kappa = 0.54$), and infiltrate ($\kappa = 0.52$), a fair agreement for congestion ($\kappa = 0.28$) and atelectasis ($\kappa = 0.28$), and a slight to fair agreement for cardiomegaly ($\kappa = 0.20$) (see Table 6.2). The confusion

Table 6.2: Inter-rater reliability between two radiologists from the radiology department of the University Medical Center Hamburg-Eppendorf evaluating the OpenI dataset (3,125 pairs of frontal and lateral chest X-rays) using Cohen's kappa coefficient (κ) and interpretation according to Landis et al. [1977].

Finding	κ	Agreement
Foreign object	0.72	Substantial
Pleural effusion	0.64	Substantial
Pneumothorax	0.55	Moderate
Mass/nodule	0.54	Moderate
Infiltrate	0.52	Moderate
Congestion	0.28	Fair
Atelectasis	0.28	Fair
Cardiomegaly	0.20	Slight/fair

matrix (see Figure 6.5) suggests that the less experienced radiologist (radiologist1) indicated the presence of pleural effusion ($n = 238$ vs. 130), infiltrate ($n = 175$ vs. 66), congestion ($n = 493$ vs. 130) and atelectasis ($n = 239$ vs. 127) significantly more often, whereas the more experienced radiologist (radiologist2) indicated cardiomegaly ($n = 94$ vs. 219) significantly more often for the same dataset. For mass/nodule, infiltrate, congestion, atelectasis, and pleural effusion, radiologist1 chose the label "unknown" more often than radiologist2.

6.3 Experiments and results

In this section, the experimental setup is described and the results are presented. For an assessment of the generalization performance (as discussed in Section 3.7), a five-time subsampling from the entire OpenI dataset was used. Each time, the dataset was split into 70 % training ($N_{\text{train}} = 2,188$) and 30 % testing ($N_{\text{test}} = 937$). The average error over all five random subsamples was calculated to estimate the optimal point for generalization. Finally, the results were calculated for each split on the test set and then averaged.

Implementation: DICOM images in the OpenI dataset have a 16-bit depth intensity range. To emphasize the anatomy of interest and to convert the images to 8-bit depth, a clipping to the interval $[a, b]$ with additional linear intensity transformation T :

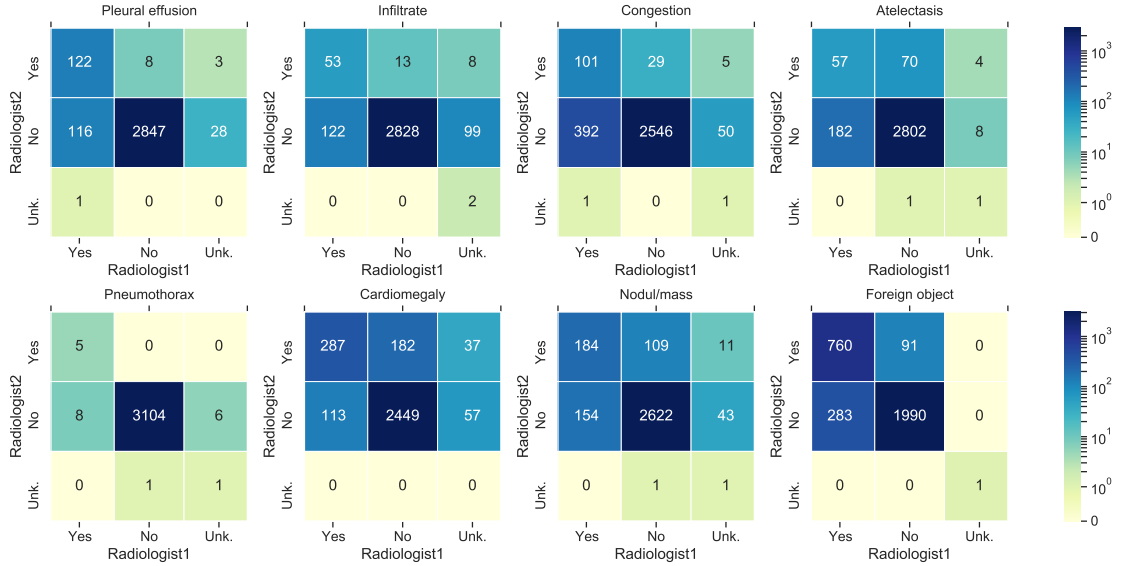


Figure 6.5: Confusion matrices of the annotation results for radiologist1 (3 years of experience) vs. Radiologist2 (18 years of experience). Both radiologists annotated the OpenI dataset (3,125 pairs of frontal and lateral chest X-ray) with respect to eight pathologies and used three different labels to indicate the presence of a finding: Yes, No or Unknown (Unk).

$[a,b] \rightarrow [c,d]$was applied to each image:

$$T(x) = (d-c)\frac{x-a}{b-a} + c \tag{6.6}$$

where $\mathbf{x} = (x_1, \ldots, x_n)\top$ and $[c,d] = [0,255]$ are the image and the output interval, respectively. For each image, the clipping interval $[a,b]$ was determined by generating a mask containing only pixels of the anatomical structure. This means that the direct radiation area and shutter area were not considered for determining the interval. The histogram was calculated while only considering the pixels in the mask. Here, the 1^{st} and 99^{th} percentiles were used for a and b, respectively.

Following the experimental setup presented in Chapter 5, an adapted ResNet-50 tailored to the chest X-ray domain was used. After replacing the fully-connected layer, the model was fine-tuned using the OpenI dataset. For training, a similar data augmentation to that presented in [Szegedy et al., 2014] was used. First, various patches were sampled with sizes between 80 % and 100 % of the image area. The patch aspect ratio was distributed evenly between $\frac{3}{4}$ and $\frac{4}{3}$. Additionally, each image was randomly horizontal flipped and randomly rotated between $-7°$ and $7°$. At testing, images were resized to 480×480 pixels and the average prediction of five cropped patches (i.e.,

center and all four corners) was used for the evaluation. In all experiments, ADAM [Kingma et al., 2015] was used as an optimizer with default parameters for $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate η was set to $\eta = 0.005$ and a batch size of 15 was used. While training, the learning rate was reduced by a factor of two if the validation loss did not improve. BCE was employed as loss functions. In the initial experiments with different loss functions to handle class imbalance, no performance difference was observed between standard BCE and class-weighted BCE (see Chapter 5). The models were implemented in CNTK [Seide et al., 2016]—an open-source deep learning toolkit—and trained on two Nvidia GeForce GTX 1080Ti GPUs with 11 GB of memory.

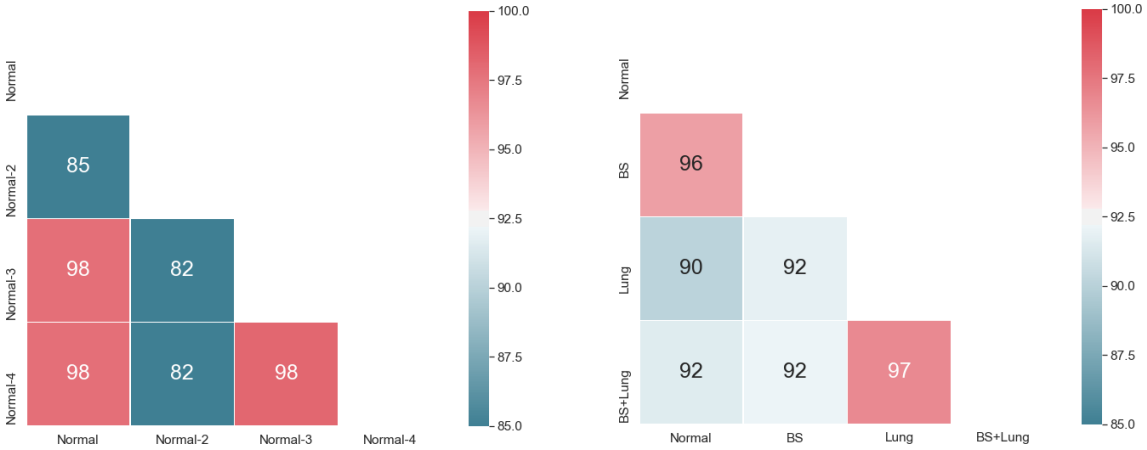
Six different experiments based on our proposed image preprocessing (see Sections 6.1.1 and 6.1.2) were performed. First, four models were trained using different training data: original images (i.e., no advanced preprocessing employed), bone suppressed images, lung cropped images, or images combining both preprocessing steps. Secondly, two ensembles were built: "EN-normal" and "EN-preprocessed". EN-normal was built upon four models trained similarly with original images but with different initializations as a baseline ensemble. EN-preprocessed was built with the three preprocessed trained models (i.e., bone suppressed, lung field cropped, and combined image trained) and one model trained with original images.

Results: Table 6.3 summarizes the outcome of the evaluation. To compare the experiments to each other, AUROC was calculated. The AUROC results are averaged over all five resamplings and presented with the standard deviation. For the ensemble experiment, the Pearson correlation coefficients between each model used for EN-normal and EN-preprocessed were calculated.

First, the experiments with different preprocessed images were compared based on their performance using AUROC results. In all experiments, five out of seven relevant classes had a high AUROC (over 0.9). Two of those five classes, "pleural effusion" and "cardiomegaly" had an AUROC over 0.95. Only the classes "mass/nodule" and "foreign object" had an AUROC below 0.9. Upon comparing the results of a model using bone suppression to the normal trained model, the AUROC for "foreign object" increased substantially from 0.795 ± 0.015 to 0.815 ± 0.013 with respect to the reported standard deviation. In all classes, the model trained with lung cropping had a higher AUROC and often a reduced standard deviation when compared to the baseline. How-

Table 6.3: Overview of AUROC results for all experiments. In this table, the averaged results over all five splits and the calculated standard deviation for each finding are presented. Furthermore, the average (AVG) AUROC over all findings is shown. The models were trained with four different input images. They were first trained with normal images, then with bone suppressed "BS" images, lung field cropped images "LFC", and a combination of bone suppressed and lung field cropped images "BS+LFC". Additionally, an ensemble with models trained on normal images "EN-normal" as well as an ensemble with the models trained on preprocessed images "EN-preprocessed" were created. Bold text emphasizes the highest overall AUROC value. The leading 0 was omitted for convenience. ★Pneumothorax was excluded due to its low positive count.

Finding	Normal	BS	LFC	BS+LFC	EN-normal	EN-preprocessed
Pleural effusion	.951 ± .008	.948 ± .009	.955 ± .007	.955 ± .009	**.960** ± .004	.957 ± .007
Infiltrate	.936 ± .012	.938 ± .012	.939 ± .007	.936 ± .014	**.944** ± .010	.943 ± .011
Congestion	.937 ± .013	.932 ± .015	.941 ± .014	.938 ± .014	.941 ± .012	**.946** ± .013
Atelectasis	.905 ± .020	.907 ± .016	.917 ± .017	.913 ± .020	.905 ± .020	**.923** ± .016
Cardiomegaly	.952 ± .006	.950 ± .006	.953 ± .005	.952 ± .003	.955 ± .004	**.959** ± .003
Mass/nodule	.764 ± .016	.766 ± .016	.821 ± .020	**.840** ± .011	.769 ± .014	.837 ± .014
Foreign object	.795 ± .015	.815 ± .013	.808 ± .013	.805 ± .015	.811 ± .018	**.821** ± .015
Pneumothorax	.731 ± .134	.789 ± .104	.813 ± .132	.794 ± .128	.736 ± .163	.792 ± .142
AVG ★	.891 ± .013	.894 ± .012	.905 ± .012	.906 ± .012	.898 ± .012	**.912** ± .011



(a) "Normal trained" **(b)** "Preprocessed trained"

Figure 6.6: Pearson correlation coefficients for normal trained models (a) and models trained with preprocessed images (b). The correlation between normal models was already high, except for the model "Normal-2", which seemed to converge to a different optimum. The models trained with preprocessed images have a lower correlation (approximately 92 %). This indicates that an ensemble of the models shown in (b) can have a greater impact on classification performance.

ever, the AUROC only increased substantially (from 0.766 ± 0.016 to 0.821 ± 0.020) for the class "mass". The increased spatial resolution for lung cropped images most likely helps the model to better detect small masses. This is consistent with the observations of radiologists, who reported that there are more small masses/nodules than large ones in the OpenI dataset. Combining both preprocessing steps resulted in the highest AUROC for "mass" and increased the AUROC by 9.95 %. No substantial changes for the other classes were observed.

In Figure 6.6, the Pearson correlation coefficients between the individual models for the two ensembles are reported. As expected, models for EN-normal were already highly correlated (i.e., values around 98 %)—except for the model "Normal-2", which seemed to converge to a different optimum. Upon comparing the Pearson correlation coefficients of the models for EN-preprocessed with the models for EN-normal, the coefficients are lower (approximately 92 %).

6.4 Discussion

The degree of agreement between both radiologists in the experiment differed based on the specific pathology. While the less experienced radiologist (radiologist1) rated more examinations as pathological for all seven classes apart from “cardiomegaly”, the more experienced radiologist (radiologist2) more often indicated “cardiomegaly”. For every pathology, both radiologists seemed to decide according to their own inner threshold. Nevertheless, in contrast to other publications, we achieved reasonable inter-observer reliability (between fair and substantial) for all classes. For example, Neuman et al. [2012] discovered only slight inter-observer reliability ($\kappa = 0.14$) between radiologists in detecting interstitial infiltrate in pediatric chest X-rays.

To minimize inter-observer variability, radiologists must be well instructed and clear cutoffs for every pathology must be defined for a “yes” or “no” decision before starting with an annotation. Thus, breaking down a complex radiology report into eight binary classes is a main limitation of the experiment in this thesis, especially since this procedure does not resemble a realistic clinical workflow and could be influenced by the concentration and motivation of the reader. Furthermore, open datasets do not commonly include clinical background information regarding the specific patient or the patient collective in general, thereby complicating image interpretation when compared to clinical reality [Berbaum et al., 1985; Potchen et al., 1979]. In particular, the spectrum of disease varies widely by geographic region, socioeconomic status, and ethnicity.

A solution for future studies to address the problem of disagreement between radiologists could involve the implementation of an annotation tool where the radiologist is not forced to perform binary reporting but can indicate a probability of the presence of a finding. For simplicity, this could be implemented with a slide controller in the web-based annotation tool. Such probabilities could then be used to directly train a neural network, or an attempt could be made to first normalize the probability distribution of each radiologist. Normalization can help if one of the radiologists always rates higher than the others.

The Pearson correlation results for the EN-preprocessed ensembles indicate that building an ensemble from those four models can impact AUROC results. This hypothesis was verified by the AUROC results presented in Table 6.3. EN-preprocessed ensembles considerably increase the AUROC for “cardiomegaly”, “foreign object”, and “atelecta-

sis" with respect to the reported standard deviation, whereas EN-normal ensembles do not. However, EN-normal has the highest—but only slightly higher—AUROC for "pleural effusion" and "infiltrate". In four out of seven classes, EN-preprocessed yields the best AUROC results and has the highest average AUROC of $.912 \pm .011$. However, its lower AUROC for "mass/nodule" when compared to the "BS+LFC" model indicates that the simple prediction averaging was not optimal. This is because the other three models in the ensemble have higher prediction confidence than the single model. A more advanced method to calculate the final prediction could help to solve this problem. This method could involve an additional neural network trained to find the optimal combination for the predictions of the four models.

6.5 Summary

In this chapter, the effects of two advanced preprocessing methods—bone suppression and lung field cropping—for multilabel disease classification in chest X-rays have been investigated. Notably, the superior performance of models trained on preprocessed images has been highlighted through a systematic evaluation. The best performance was achieved by an ensemble architecture leveraging all the information from the different advanced preprocessing methods. Moreover, substantial AUROC improvement for specific classes (e.g., "foreign object" and "cardiomegaly") has been achieved.

The next chapter introduces worklist prioritization for chest X-rays as a potential clinical application. For the first time, a unified framework to simulate a clinical workday in a radiology department is presented. The framework is implemented with empirical data from the radiology department of the University Medical Center Hamburg-Eppendorf (UKE) and used to demonstrate the significant impact of smart worklist ordering on report turnaround time (RTAT). The sorting is based on urgency levels determined by the predictions of a state-of-the-art convolutional neural network.

7 Simulation of chest X-ray worklist prioritization

This chapter aims to evaluate whether smart worklist prioritization by a deep learning-based sorting can optimize the radiology workflow and reduce RTATs for critical findings in chest X-rays.
A simulation framework was developed to model the current workflow in a radiology department by incorporating hospital-specific chest X-ray generation rates, reporting rates, and pathology distributions. Using this data, a standard worklist processing known as first-in, first-out (FIFO) was simulated and then compared to a worklist prioritization based on urgency. Examination prioritization was performed by a convolutional neural network that classified eight different pathological findings ranked in descending order of urgency: pneumothorax, pleural effusion, infiltrate, congestion, atelectasis, cardiomegaly, mass, and foreign object. Furthermore, a method to counteract the effect of false negative predictions by the convolutional neural network was proposed and investigated, resulting in a dangerously long RTAT since chest X-rays are sorted to the end of the worklist. The simulations demonstrate that smart worklist prioritization can reduce the average RTAT for critical findings in chest X-ray while maintaining a small maximum RTAT as per FIFO ordering.
Most of the methods and results described in this chapter have been published by Baltruschat et al. [2020a], Baltruschat et al. [2020b], Steinmeister et al. [2020].

Growing radiologic workload, a shortage of medical experts, and declining revenues often lead to potentially dangerous backlogs of unreported examinations, especially in publicly funded health care systems [Beardmore et al., 2016; Care Quality Commission, 2017; Royal College of Radiologists, 2018]. With the increasing demand for radiological imaging, the continuous acceleration of image acquisition, and the expansion of teleradiological care, radiologists are now working under increasing time pressure that cannot be relieved by improving radiology information system (RIS),

picture archiving and communication system (PACS) integration, or the use of speech recognition software [Reiner, 2013]. As reported by Beardmore et al. [2016], the average RTAT for plain X-rays in the United Kingdom is approximately 34 hours, while 74 % have an RTAT of less than 24 hours.

The delayed communication of critical findings to the referring physician bears the risk of delayed clinical intervention and impairs the outcome of medical treatment [Berlin, 2001; Hanna et al., 2005; Singh et al., 2007; The Joint Commission, 2020], especially in cases requiring immediate action (e.g., tension pneumothorax or misplaced catheters). For this reason, The Joint Commission defined the timely reporting of critical diagnostic results as an important goal for patient safety [The Joint Commission, 2020].

Many institutions process their examination worklists following the FIFO principle. However, the urgency information provided by the ordering physician is often incomplete or presented as an ambiguous and ill-defined priority level, such as “critical”, “ASAP” (as soon as possible), or “STAT” (short turnaround time) [Rachh et al., 2018; Wesp, 2006]. A recent study related to portable chest radiographs reported that 38 % of all STAT exams were not clinically urged [Gaskin et al., 2016].

While rule-based approaches that assign cases to specific worklists (e.g., emergency department or subspecialty) can help to optimize the overall workflow, they cannot take imaging findings into account. Furthermore, prioritization by radiographers after acquisition of a CXR has not found any application in clinical routines.

Deep learning methods such as convolutional neural networks offer promising options to streamline the clinical workflow. Automated disease classification systems based on convolutional neural networks can enable the real-time prioritization of worklists and reduce the RTAT [Ondategui-Parra et al., 2004] for critical findings by up to 60 %, which was demonstrated for head and neck CTs [Yaniv et al., 2018]. For chest X-ray examinations, a potential benefit of real-time triaging by convolutional neural networks has been reported in [Annarumma et al., 2019]; however, this study primarily focused on the development of a deep learning system without a real clinical simulation and does not present maximum RTAT values for critical findings.

The benefits of smart worklist prioritization need to be discussed not only based on the average RTAT but also in terms of the maximum RTAT. One problem with using

deep learning methods for smart worklist prioritization is that critical findings may be "overlooked" by the system (i.e., the false negative rate (FNR) of the prediction model is not zero). The higher the FNR, the more likely it is that individual examinations with critical findings will be sorted to the end of the worklist, which could increase the risk of delayed treatment.

In this chapter, we simulate multiple smart chest X-ray worklist prioritization methods for chest X-rays in a realistic clinical setting by using empirical data from the University Medical Center Hamburg-Eppendorf. We develop a realistic simulation framework and evaluate whether machine learning can reduce RTAT for critical findings by using smart worklist prioritization instead of the standard FIFO sorting. Furthermore, we propose a thresholding method for maximum waiting time to reduce the effect of false negative predictions by the neural network.

7.1 Method

Based on the work presented in Chapter 5 [Baltruschat et al., 2019c], a tailored ResNet-50 architecture with a larger input size of 448×448 pixels was used. Furthermore, each chest X-ray was preprocessed using two methods (i.e., lung field cropping and bone suppression). As shown in Chapter 6, the highest average AUROC value was achieved by combining both methods in an ensemble [Baltruschat et al., 2019e]. The neural network was pretrained on the publicly available ChestX-ray14 dataset [Wang et al., 2017] and, after replacing the last fully-connected layer of the converged neural network, it was fine-tuned on the open-source OpenI dataset [Demner-Fushman et al., 2016]. As presented in Section 6.2, two expert radiologists—with 5 and 19 years of experience in chest X-ray reporting—from the UKE radiology department annotated a revised OpenI dataset (containing 3125 chest X-rays) regarding eight findings: pneumothorax, congestion, pleural effusion, infiltrate, atelectasis, cardiomegaly, mass/nodule, foreign object.

Due to the importance of pneumothorax detection and the low number $n = 11$ of cases with "pneumothorax" in the OpenI dataset, the specifically adapted ResNet-50 of Gooßen et al. [2019b] was used and trained on a dedicated in-house dataset for pneumothorax detection. Notably, both datasets include different degrees of clinical manifestation for each finding. Therefore, the final neural network included two

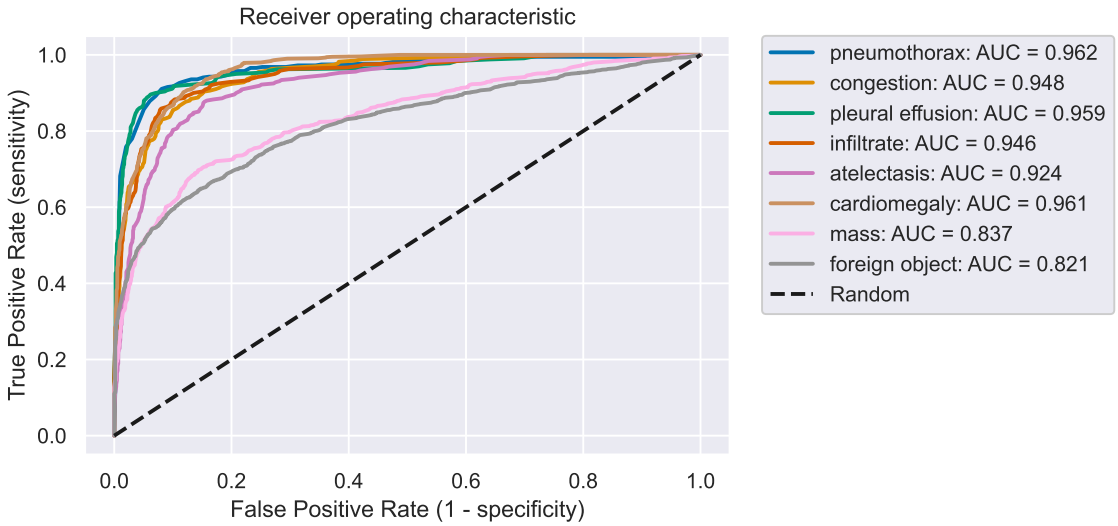


Figure 7.1: Receiver operating characteristic curves of the neural network for all eight findings.

separate convolutional neural networks, both of which obtained the highest average AUROC value (Figure 7.1) in previous experiments when compared to different network architectures.

The average inference time per image was approximately 21ms when using an Nvidia GeForce GTX 1080 GPU and 351ms with an Intel Xeon E5-2620 v4 8-core CPU. Both options add a negligible overhead to the reporting process.

7.1.1 Pathology triage

For triage, a ranking—reflecting the urgency for clinical action—of the pathologies was defined by two experienced radiologists. Since our annotations did not include different degrees of pathology manifestation, only the presence of pathology was considered for the prioritization. Furthermore, the impacts of different pathology combinations were not considered.

The following eight pathological findings were ranked in descending order of urgency: pneumothorax, pleural effusion, infiltrate, congestion, atelectasis, cardiomegaly, mass/nodule, and foreign object. Notably, this ranking only reflects the most relevant findings in the clinical routine, as defined by the two experienced radiologists.

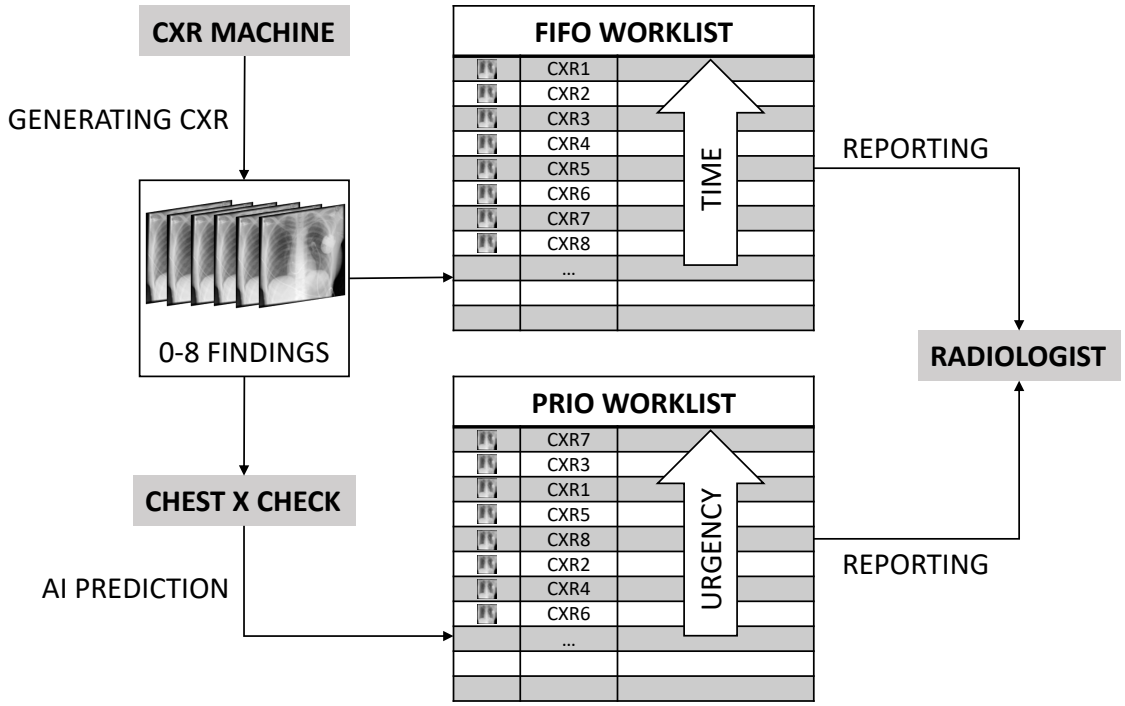


Figure 7.2: Workflow simulation overview. A conventional radiography system constantly generates chest X-rays. For each chest X-ray, between zero and eight findings were assigned. Chest X-rays were either sorted into the worklist chronologically (i.e., FIFO) or according to their urgency based on the predictions of a convolutional neural network (i.e., PRIO). Finally, worklists were processed by a virtual radiologist.

7.1.2 Workflow simulation

To evaluate the clinical effect of a chest X-ray worklist rearrangement by smart ordering under realistic conditions, the current workflow in the radiology department of the UKE was analyzed. Thereafter, the empirical data were transferred into a computer simulation (see Figure 7.2).

A framework consisting of four main parts was designed to perform the simulation. The first part is a discrete probability distribution of how often chest X-rays are generated $p_{\text{CXR}} : \Omega_{\text{CXR}} \mapsto [0, 1]$, where Ω_{CXR} is the sample space. The second part is the department-specific disease prevalence for eight findings to assign labels to the chest X-rays. The third part is the sensitivity and specificity for all eight findings of a state-of-the-art convolutional neural network to classify each chest X-ray. The fourth part is a second discrete probability distribution of how rapidly a radiologist finalizes a chest X-ray report $p_{\text{report}} : \Omega_{\text{report}} \mapsto [0, 1]$, where Ω_{report} is the sample space.

Now, the four main parts were used to model the clinical workload throughout the day using two asynchronous processes. One process (i.e., simulating the chest X-ray machines in the radiology department) generated new examinations that filled up the worklist after a specific time. The second process (i.e., simulating the reporting of a radiologist) took the top entry of the worklist and required a specific amount of time to complete the processing. To add information about the pathologies, each generated examination was assigned between zero and eight pathologies based on the a-priority probabilities from the pathology prevalence.

To describe the two discrete probability distributions, the sample space Ω was needed. Both sample spaces Ω_{CXR} and Ω_{report} were determined by monitoring the chest X-ray acquisition and reporting process of N samples. First, the sample space $\Omega_{\text{CXR}} = \{d_2, \ldots, d_N\}$, $d \in \mathbb{R}_{\geq 0}$, where d is the difference between the acquisition time stamps t_{acq} of two consecutive chest X-rays:

$$d_l = t_{\text{acq}}^{l} - t_{\text{acq}}^{l-1} \text{ , } l \in \{2, \ldots, N\} \text{ .} \tag{7.1}$$

Ω_{CXR} includes all effects, such as the different patient frequencies during day and night (see Figure 7.3).

Second, the same method was used for Ω_{report}, except the reporting times t_{report} of two subsequent chest X-rays were used to approximate the reporting speed of a radiologist r. Hence, $\Omega_{\text{report}} = \{r_2, \ldots, r_N\}$, $r \in \mathbb{R}_{\geq 0}$ with

$$r_l = t_{\text{report}}^{l} - t_{\text{report}}^{l-1} \text{ , } r \in \{2, \ldots, N\} \text{ .} \tag{7.2}$$

Ω_{report} includes the raw reporting speed for a chest X-ray as well as factors such as breaks or interruptions due to phone calls (see Figure 7.4).

As previously explained, this setup models a FIFO reporting scenario and is similar to the current clinical workflow used in the radiology department of the UKE. For the smart worklist prioritization, we included the neural network directly after the chest X-ray generation. For each chest X-ray, the neural network predicts whether a finding is present or not before it is sorted into the worklist.

By automatically predicting the presence of all eight pathological findings, a level of urgency could be assigned according to the urgency order defined by two the expert radiologists (see Section 7.1.1). Depending on the estimated urgency level, images

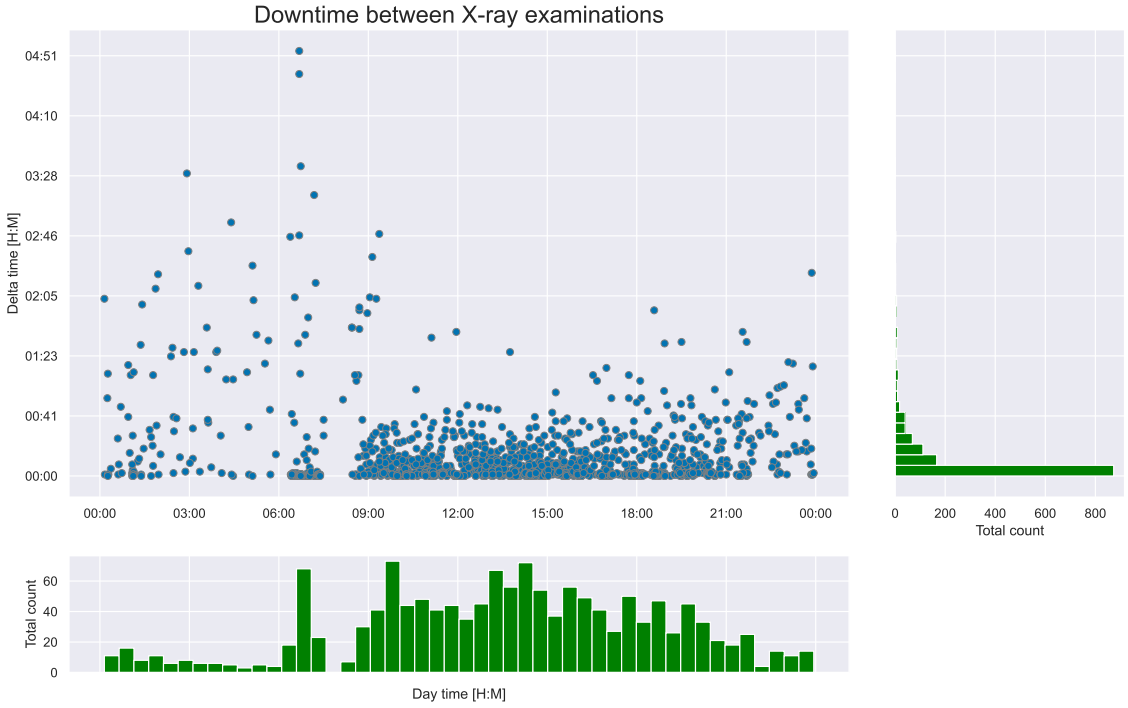


Figure 7.3: Discrete distribution of chest X-ray generation speed. The x-axis shows the time in 24-hour format, while the y-axis shows the calculated time deltas. The histogram (in the x- and y- directions) is shown in green.

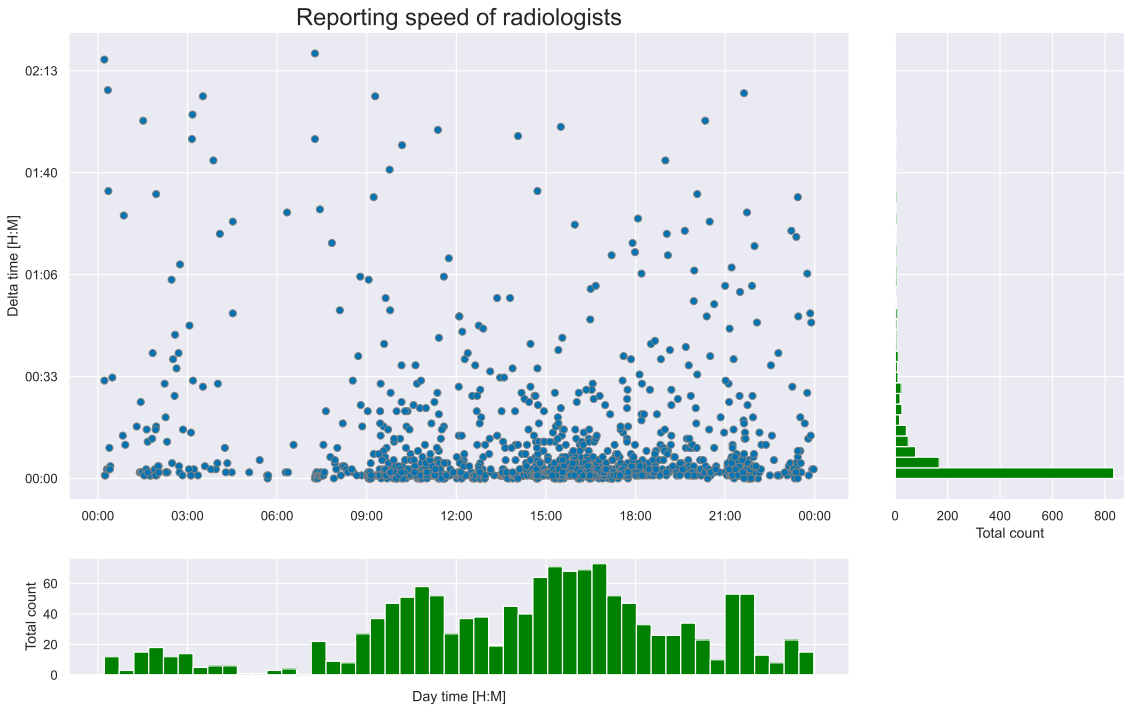


Figure 7.4: Discrete distribution of chest X-ray reporting times by radiologists. The x-axis shows the time in 24-hour format, while the y-axis presents the calculated time deltas between two chest X-ray reports. The histogram (in the x- and y-directions) is shown in green.

were inserted into an existing worklist while considering images with a similar or higher urgency level. The rearranged worklist was processed by the same modeled reporting process used in the FIFO scenario.

To counteract the problem of false negative predictions (i.e., sorting positive examinations to the end of the worklist), the maximum waiting time was restricted to a specific limit. If an examination on the worklist had a waiting time longer than the maximum waiting time, it was assigned with the highest urgency level and moved to the top of the worklist. While this should help to reduce the problem caused by false negative predictions (i.e., dangerously long maximum RTATs), it should also be counterproductive to the original goal of reducing the average RTAT for critical findings.

7.2 Experiments and results

All methods were tested using a Monte Carlo simulation over 11,000 days with 24 hours of clinical routine, covering the generation of approximately 1,000,000 chest X-rays. Furthermore, the worklist was completely finalized to zero once every 24 hours in all simulations. In our evaluation, we compared the average and maximum RTATs of the simulations.

7.2.1 Pathology distribution

The analysis of pathology distribution at the UKE was performed by manually annotating (i.e., expert radiologist) 600 chest X-ray reports from two weeks—one from August 2016 and one from February 2019. Both weeks were randomly selected and used to approximate the pathology distribution. The chest X-rays included all study types and degrees of disease manifestation.

Since the stationary patient collective was from a hospital of maximum care (i.e., a larger institution with more than 1000 beds), the proportion of chest X-rays without pathological findings was only 31 %. The prevalence of the most critical finding, “pneumothorax”, was 3.8 %. The results for pathology distributions are presented in Table 7.1.

Table 7.1: Finding prevalence in chest X-rays at the University Medical Center Hamburg-Eppendorf (approximation by 600 samples from August 2016 and February 2019). The table is ordered by finding urgency, as defined by two expert radiologists.

Finding	**Total count**	**Prevalence [%]** $N = 600$
Pneumothorax	23	3.8
Congestion	124	20.7
Pleural effusion	236	39.3
Infiltrate	100	16.7
Atelectasis	124	20.7
Cardiomegaly	117	19.5
Mass/nodule	38	6.3
Foreign object	298	49.7
Normal	186	31.0

7.2.2 Chest X-ray generation and reporting time analysis

The metadata of 1,408 examinations—including all types of chest X-ray studies—were used to determine a discrete distribution of chest X-ray generation and radiologist reporting speed. The examinations were from two randomly selected and non-consecutive weeks from Monday 00:00 AM until Sunday 00:00 AM. To model the acquisition process, the creation timestamps of two consecutive chest X-rays were used to calculate the delta between their creation. The same method was employed for reporting speed. Here, the report finalization timestamp was used to determine the delta between two chest X-rays. Thereafter, all deltas greater than 2 h 30 min were removed to ensure that outliers were only found in the discrete distribution of chest X-ray generation and not in the discrete distribution of reporting speed. Such outliers exist because no examination may be acquired over a long period of time ($>$ 2 h 30 min).

7.2.3 Hospital's report turnaround time analysis

The average RTAT for a chest X-ray—measured over two randomly selected and non-consecutive weeks (1,408 examinations)—was 80 min with a range between 1 min and 1041 min. Assuming that a chest X-ray report by an experienced radiologist will only take several minutes, this range for reporting times can be explained by different

external influences, such as night shifts, change of shifts, working breaks, or backlogs.

7.2.4 Operation point selection

Before the convolutional neural network (trained for multilabel classification) can be used for smart worklist ordering, an operating point must be defined. A threshold for every pathology must be selected to derive a binary classification (i.e., the finding is present or not) from the continuous response of a convolutional neural network (see Section 3.3). This corresponds to the selection of an operation point on the ROC curve. While an exhaustive evaluation of different threshold combinations for all pathologies is computationally prohibitive, the focus of this thesis was on pneumothorax only (the most critical finding in this setting). Here, the average RTAT was estimated for different operating points by sampling the ROC curve at different false positive rates (FPRs).

As shown in Figure 7.5, higher FPRs reduce the effect of smart worklist prioritization to almost zero (i.e., almost all examinations are rated as urgent). Also, the other extreme (i.e., low FPR) has no effect if nearly all images are rated as non-urgent. Figure 7.5 also shows that the optimal operation point to reduce the average RTAT is at an FPR of 0.05.

For the optimal operation point at FPR $= 0.05$, the corresponding true positive, false negative, and true negative rates are shown in Table 7.2. Table 7.2 also shows that the optimal operation point to reduce the average RTAT still has a moderate FNR of approximately 0.20 for most findings. The higher the FNR, the more likely it is that individual examinations with critical findings will be sorted to the end of the worklist. Hence, a second operation point was selected with a low FNR of 0.05 to determine whether this can help to reduce the maximum RTAT. Table 7.2 presents the corresponding false positive, true negative, and true positive rates.

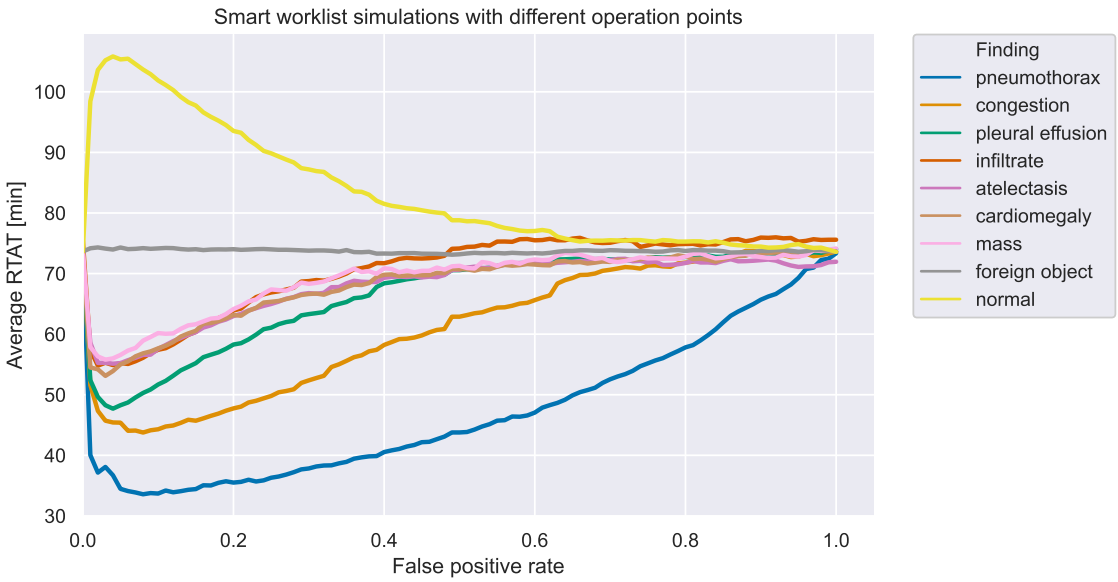


Figure 7.5: Investigation of different operation points for the neural network. To find the optimal operation point to reduce the average report turnaround time (RTAT) for critical findings, multiple simulations were run with false positive rates between 0 and 1.

Table 7.2: Different operation points for the convolutional neural network. The first column shows the true positive rate (TPR), false negative rate (FNR), and true negative rate (TNR) for the optimal operation point (having a false positive rate (FPR) of 0.05) with the best average report turnaround time (RTAT) reduction. The second column shows the operation point for a low FNR of 0.05 (i.e., reducing the likelihood of dangerously long RTATs for critical findings).

	FPR = 0.05			FNR = 0.05		
Finding	**TPR**	**FNR**	**TNR**	**TPR**	**FPR**	**TNR**
Pneumothorax	0.82	0.18	0.95	0.95	0.20	0.80
Congestion	0.71	0.29	0.95	0.95	0.24	0.76
Pleural effusion	0.86	0.14	0.95	0.95	0.21	0.79
Infiltrate	0.75	0.25	0.95	0.95	0.27	0.73
Atelectasis	0.61	0.39	0.95	0.95	0.39	0.61
Cardiomegaly	0.75	0.25	0.95	0.95	0.18	0.82
Mass/nodule	0.51	0.49	0.95	0.95	0.72	0.28
Foreign Object	0.51	0.49	0.95	0.95	0.78	0.22

7.2.5 Workflow simulations

Figure 7.6 summarizes the effects of all four simulations (i.e., FIFO, Prio-lowFNR, Prio-lowFPR, and Prio-MAXwaiting) on RTAT. For the simulations Prio-lowFPR and Prio-MAXwaiting, the optimal operation point (as shown in Table 7.2) to reduce average RTAT was used.

The average RTAT for critical findings was significantly reduced in the Prio-lowFPR simulation when compared to the FIFO simulation (e.g., pneumothorax: 37.5 min vs. 80.1 min, congestion: 46.6 min vs. 80.5 min, pleural effusion: 51.3 min vs. 80.5 min). As expected, increased average RTAT was only reported for normal examinations, with a significant increase from 80.2 min to 117.3 min. However, the maximum RTAT in the Prio-lowFPR simulation also increased compared to the FIFO simulation for all eight findings (e.g., pneumothorax: 1297 min vs. 890 min) since some examinations were predicted as false negatives and sorted to the end of the worklist. Notably, the low FNR of 0.05 in Prio-lowFNR did not help to reduce the maximum RTAT (e.g., pneumothorax: 1293 min vs. 1178 min).

In the Prio-MAXwaiting simulation, the false negative prediction problem was countered by using a maximum waiting time. As a result, the maximum RTAT was reduced for most findings (e.g., pneumothorax from 1297 min to 979 min). Notably, the average RTAT was only slightly higher than the Prio-lowFPR simulation (e.g., pneumothorax: 38.5 min vs. 37.5 min).

Finally, the last simulation was the upper limit for a smart worklist prioritization by virtually employing a perfect classification algorithm (Perfect) with a true positive and true negative rate of 1. Table 7.3 presents comparisons with the other four simulations. For pneumothorax, the Prio-MAXwaiting average RTAT was only 8.3 min slower than the Perfect-simulation, while FIFO was 49.8 min slower.

Statistical analysis The predictive performance of the convolutional neural network was assessed by using the AUROC. The AUROC results shown in Figure 7.1 were averaged over five-time random subsamples.

Welch's t-test was used to assess the significance of the smart worklist prioritization. First, a null distribution was simulated for the RTAT, where examinations were sorted

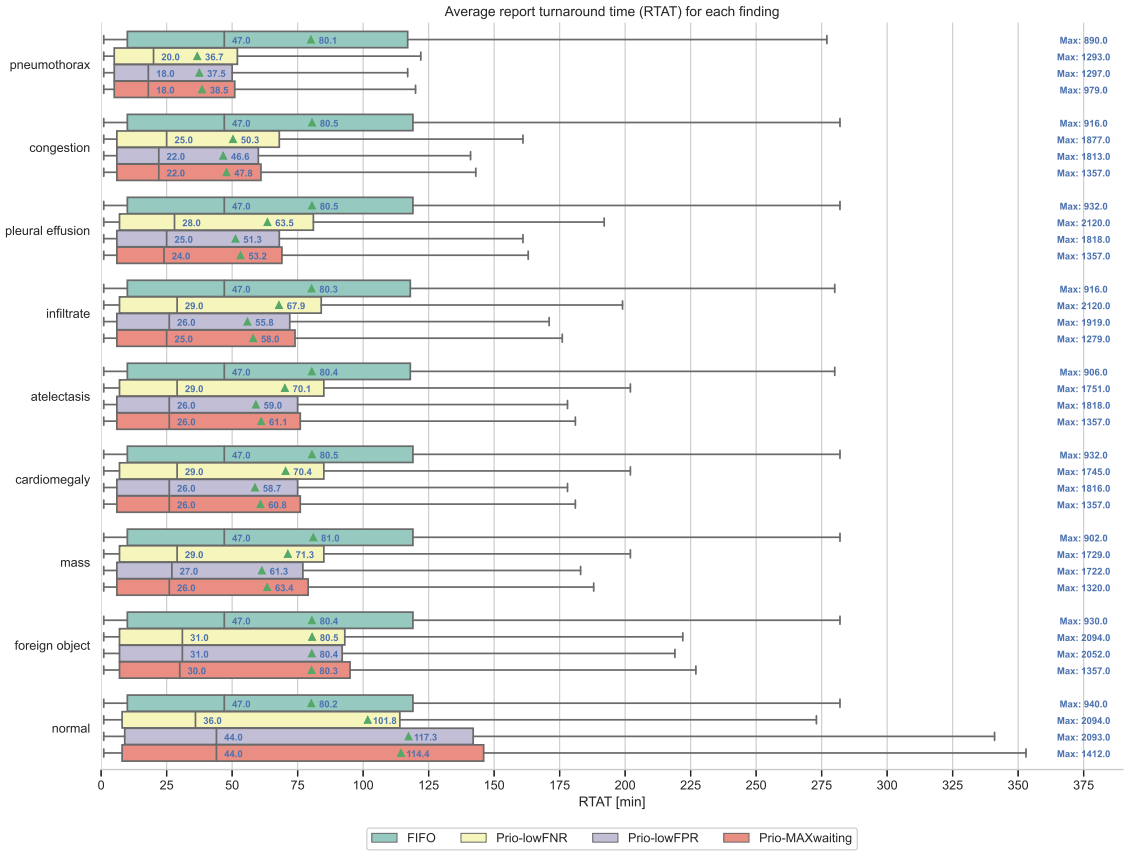


Figure 7.6: Report turnaround times (RTAT) for all eight pathological findings and for normal examinations on the basis of four different simulations: FIFO (green), Prio-lowFNR (yellow), Prio-lowFPR (purple), and Prio-MAXwaiting (red) with a maximum waiting time (light purple). Green triangles mark the average RTAT, while vertical lines mark the median RTAT. The maximum RTAT for each simulation and finding is shown on the right side.

based on the FIFO principle (i.e., random order). Secondly, an alternative distribution with worklist prioritization was simulated. Both distributions were then used to determine whether the average RTAT for each finding had changed significantly by calculating the p-value via Welch's t-test. Each distribution was simulated with a sample size of 1,000,000 examinations and the significant level was set to 0.05. For all findings except "foreign object", we calculated a $p < 0.0001$. This result supports the existence of a significant change to the average RTAT when smart workflow prioritization is used.

Table 7.3: Comparison of all four simulations (i.e., FIFO, Prio-lowFNR, Prio-lowFPR, and Prio-MAXwaiting) with a perfect classification algorithm simulation (i.e., Perfect). The table is ordered by finding urgency and the results are presented in the style (avg / max) [minutes] for each simulation.

Finding	FIFO	Prio-lowFNR	Prio-lowFPR	Prio-MAXwaiting	Perfect
Pneumothorax	80.1 / 890	36.7 / 1293	37.5 / 1297	38.5 / 979	30.3 / 320
Congestion	80.5 / 916	50.3 / 1877	46.6 / 1813	47.8 / 1357	35.2 / 510
Pleural effusion	80.5 / 932	63.5 / 2120	51.3 / 1818	53.2 / 1357	45.4 / 1016
Infiltrate	80.3 / 916	67.9 / 2120	55.8 / 1919	58.0 / 1279	49.8 / 1110
Atelectasis	80.4 / 906	70.1 / 1751	59.0 / 1818	61.1 / 1357	51.4 / 1361
Cardiomegaly	80.5 / 932	70.4 / 1745	58.7 / 1816	60.8 / 1357	52.2 / 1332
Mass/nodule	81.0 / 902	71.3 / 1729	61.3 / 1722	63.4 / 1320	52.8 / 1301
Foreign object	80.4 / 930	80.5 / 2094	80.4 / 2052	80.3 / 1357	80.7 / 2053
Normal	80.2 / 940	101.8 / 2094	117.3 / 2093	114.4 / 1412	131.5 / 2087

7.3 Discussion

The clinical workflow simulation demonstrated that a significant reduction in average RTAT for critical findings in chest X-rays can be achieved by a smart worklist prioritization using neural networks. Furthermore, it was shown that the problem of false negative predictions by a convolutional neural network can be significantly reduced by introducing a maximum waiting time.

This was proven in a realistic clinical scenario since all simulations were based on representative retrospective data from the University Medical Center Hamburg-Eppendorf. By extracting discrete distributions of chest X-ray acquisition rate as well as radiologist reporting speed, the temporal sequence of a working day could be precisely recreated.

As in other application areas, an important question related to what error rates we can ethically and legally tolerate before convolutional neural networks can be applied to patient care. Here, the legal requirements are likely to be lower and the ethical acceptability higher than for systems with automatic diagnosis. With intelligent worklist ordering, the final diagnosis for all exams is still made by a radiologist and all exams are thus seen by a radiologist.

For smart worklist prioritization, the simulations have shown that average RTAT can easily be reduced at the expense of individual cases that are classified as false negatives and thus reported much later than the current FIFO principle. While it was question-

able whether this overall improvement outweighed the risk of delayed reporting for individual cases, the Prio-MAXwaiting simulation showed that the definition of a maximum waiting time—after which all examinations are assigned the highest priority—solves this problem. For the most critical finding (i.e., pneumothorax), maximum RTAT was reduced to the current standard while preserving the significant reduction of average RTAT.

The comparison in Table 7.3 shows that state-of-the-art convolutional neural networks can nearly reach the upper limit of a smart worklist prioritization for the average RTAT. On the other hand, for the maximum RTAT, it again reveals the problem of false negative predictions. Ideally, a perfect classification algorithm could reduce the maximum RTAT to 320 min for pneumothorax, which is a substantial improvement over the standard maximum of 890 min.

The predictions of this neural network could not only be used for smart worklist ordering but also for second reader or guidance applications. The second reader application could directly compare the diagnostic results from a radiologist with the prediction of the neural network. If a difference is detected, the system could provide instant feedback to the radiologist, who must resolve the difference before he can finalize the report. The opposite of this is represented by the guidance application. In this application, the predictions of the neural network could be presented to a radiologist as additional information for examinations. Notably, both applications carry the risk of radiologists becoming inert and always relying on the predictions of the neural network.

In addition to the use of a convolutional neural network possibly improving the diagnostic workflow, it should be noted that only the timely and reliable communication of any discovered findings by a radiologist to a referring clinician ensures that patients receive the clinical treatment they require.

Unlike previous publications [Gaskin et al., 2016], the present study included inpatients as well as outpatients. This is because the daily reporting routine at the University Medical Center Hamburg-Eppendorf involves all chest X-rays being sorted into a single worklist. Furthermore, substantially shorter (compared to published data from the United Kingdom) backlogs of unreported examinations were observed.

In healthcare systems where patients and referring physicians wait for days or even

weeks for reports—or have limited access to expert radiologists—the benefits of smart worklist prioritization could obviously be greater than in countries with a well-developed health system. The longer the reporting backlogs, the more likely it is that referring physicians will attempt to rule out critical findings in chest X-rays themselves. This poses the risks of subtle findings with potentially large clinical impacts (e.g., pneumothorax) being overlooked or important discoveries by radiologists being postponed for a negligently long time.

One limitation of the present study is that the OpenI dataset that the convolutional neural network was trained on mainly included outpatients, which contrasts with the predominantly stationary patient collective of the hospital. Therefore, the performance of the algorithm, which is already strong compared to other publications [Baltruschat et al., 2019c], cannot be directly transferred to the hospital-specific patient collective and will most likely decrease. However, it is important to note that the priority-based scheduling algorithm developed in this work is generic and can use any convolutional neural network that classifies chest X-ray pathologies. If the convolutional neural network classifier is improved, the scheduling algorithm will directly benefit.

7.4 Summary

Overall, the application of smart worklist prioritization by a convolutional neural network shows great potential to optimize clinical workflows and can significantly improve patient safety in the future. The clinical workflow simulations suggest that triaging tools should be customized based on local clinical circumstances and needs.

In the future, it will be important to include more pathologies and different degrees of manifestation to further improve the benefits of smart worklist prioritization. While this study only focused on the eight most common findings in a chest X-ray at a university hospital and ranked them accordingly, severe atelectasis (for example) can place patients´ health at greater risk than a small pleural effusion.

8 Conclusion and future perspective

This thesis has presented several improvements for multilabel disease classification in chest X-rays using deep learning. Chapter 2 laid the cornerstones for our proposed changes, Chapters 3 and 4 described the general methodology of neural networks, Chapter 5 presented new changes to the model architecture, Chapter 6 proposed advanced preprocessing to assist model training and discussed our annotation for the OpenI dataset, and Chapter 7 translated the findings into a clinical application and showed the significant impact that a neural network for chest X-ray analysis can have on smart worklist ordering.

In this thesis, four types of problems were addressed by employing convolutional neural networks to classify chest X-ray diseases, and several major contributions to this field were made. In Chapter 5, it was shown that the novel model architecture—which incorporates non-image features such as gender, age, and VP and has a larger input size—is superior to other architectures without such modifications. By analyzing the converged models with Grad-CAM, it was found that the models trained only with noisy labels learned false image features for classification. For example, the model sometimes used a medical tube to classify a pneumothorax. These findings motivated the introduction of annotations with minimal noise to the OpenI dataset [Demner-Fushman et al., 2016] in Chapter 6. In total, 3,125 chest X-rays were annotated by two radiologists. This dataset was used to demonstrate the beneficial effects of two advanced preprocessing methods for deep learning and determine how to leverage all of the relevant information simultaneously. Bone suppression and lung field cropping substantially increased the classification results and assisted the model training by normalizing the appearance of chest X-rays. Furthermore, the ensemble—combining models trained on different preprocessed image types—achieved the highest overall AUROC. Finally, the gap between research and clinical applications was closed in

Chapter 7. Notably, an important possible application for deep learning algorithms was identified: worklist prioritization based on urgency levels. To highlight the impact of deep learning algorithms for worklist prioritization, a novel simulation framework was developed to simulate a clinical workday based on empirical data from a radiology department. In a concluding experiment, it was shown that current deep learning algorithms can reduce the average RTAT for critical findings almost by a factor of two.

8.1 Future perspective

Throughout this thesis, three research directions were identified that have great potential to improve the contributions presented in this thesis. Two methodological extensions to improve disease classification in the radiological field were also explained, while another clinical application that could have a significant positive impact on reducing workload in a radiology department was also presented.

8.1.1 Multitask learning

It has been shown that multitask learning can have multiple positive effects for individual tasks that are related to each other [Ruder, 2017]. As shown and discussed in Section 5.3, global labels for supervised training can be problematic for the classification of chest X-ray diseases, especially when training is performed with labels generated by natural language processing (usually the only labels available for chest X-ray images). To overcome the problem of false feature learning by global labels, multitask learning can help to focus the attention of models by also learning the auxiliary task of disease segmentation or detection.

It can be beneficial for neural network models to learn the representation of anatomical structures using a segmentation task. Guendel et al. [2019] was the first to present results related to this idea by combining lung and heart segmentation with disease classification. Segmentation can force a model to learn a representation using the anatomical structure of a chest X-ray image—similar to a radiologist, who also learned the anatomy of the human thorax before diagnosing chest X-rays. While Guendel et

al. [2019] only used heart and lung segmentation mask, other anatomical structures (e.g., the ribs, clavicles, scapulae, and diaphragm) could also be very important to further improve the results.

As a second extension, the lateral view of a chest X-ray (the frontal and lateral view are often available) can easily be integrated into such a multitask learning framework. For example, the lateral view can be very important for pleural effusion classification.

8.1.2 Decomposition of a chest X-ray into pseudo-CT

The interpretation of two-dimensional medical projection images is a notoriously challenging task for radiologists since a two-dimensional projection image is a linear superposition of contributions from different depth layers of the imaged anatomical structure. On the other hand, two-dimensional medical projection images are—beyond their diagnostic value—omnipresent and widely used due to their low radiation dose, simplicity of acquisition, and low cost. Therefore, providing assistance and guidance to radiologists in the interpretation of images is a crucial ingredient for improving clinical workflows.

As described in [Baltruschat et al., 2018a], radiologists must mentally disassemble two-dimensional projection images to detect anomalies or perform quantitative analyses. In other words, an important step in the image analysis workflow involves the separation of the visible two-dimensional projection image into its different constituents (i.e., anatomical structures at different depths). This task involves the identification of structures and subtraction of these structures for subsequent analyses. For example, in the context of nodule detection in X-ray imaging, radiologists must consider that the appearance of a nodule in an image could be influenced by the ribs, spine, vasculature, and other anatomical structures. Particular aspects of this task have already been partially automated by technologies such as automatic rib cage removal algorithms, which require a fair amount of manual work and engineering to produce a clinically acceptable result. There is currently no general-purpose method available that allows the automation of this task.

While some initial work for this problem was performed by Albarqouni et al. [2017], Li et al. [2019], it remains an open question whether the use of a second view (e.g., the lateral view) can improve the decomposition of a chest X-ray into a pseudo-CT

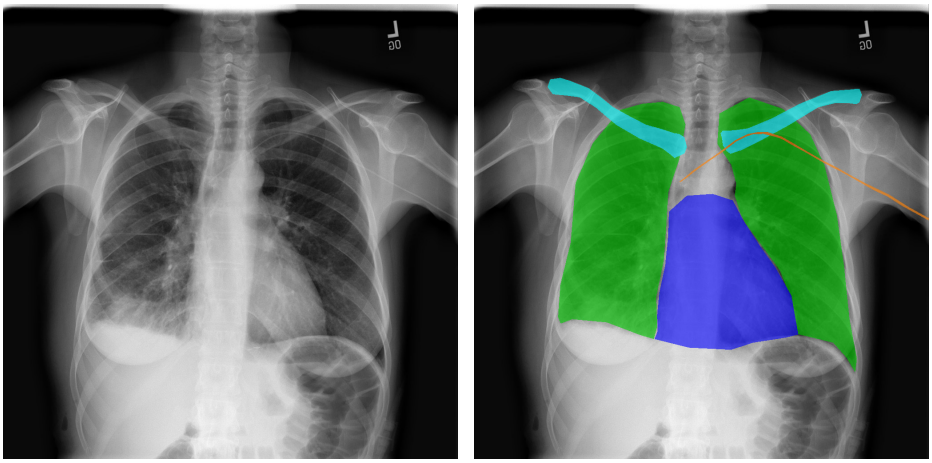
by using a neural network. At present, one project is investigating whether such additional information helps to reconstruct a complete pseudo-CT image and whether this pseudo-CT could help radiologists. Furthermore, such pseudo-CT could also be beneficial for disease classification with a convolutional neural network.

8.1.3 Malposition detection of central venous catheters in chest X-rays

While generic disease classification or triaging has received tremendous attention, it is notoriously challenging (as discussed in Sections 2.4.3 and 2.4.4). On the other hand, it is common to have a specific clinical question for a chest X-ray, which requires an assessment of the image that extends beyond simple classification. For example, the insertion of a central venous catheter is notoriously difficult and could result in a pneumothorax.

Notably, Pikwer et al. [2008] reported that the incidence of central venous catheter malposition ranges from 3.6 % to 14 %. Therefore, after most central venous catheter insertions, a chest X-ray is taken to verify that catheters are correctly positioned and determine whether or not a pneumothorax is present. At the University Medical Center Hamburg-Eppendorf, this medical question currently accounts for 20 % of all chest X-ray images. While some initial work [Yi et al., 2020] has dealt with the segmentation or detection of catheters in chest X-rays, no studies have dealt with the clinical application of malposition detection to date.

To address this open problem, I began to create a dataset with global classification labels and segmentation masks for central venous catheters as well as the lung, heart, and clavicles. Figure 8.1 presents initial segmentation masks without a pneumothorax segmentation. The additional anatomical structure was annotated because the anatomical landmarks (e.g., the carina and heart) can be used in the algorithm to determine the correct position of the catheter tip. The segmentation of the anatomical structure and the catheters can be performed using a fully convolutional network with a decoder [Long et al., 2015]. An alternative to catheter segmentation could also involve pathfinding via reinforcement learning [Sartoretti et al., 2019; Song et al., 2018].



(a) Original image **(b)** Segmentation overlay

Figure 8.1: Example image from the central venous catheter dataset. The original image (a) is shown on the left and the corresponding segmentation makes (c) are shown as an overlay on the right. The color blue indicates the heart, while green indicates the lungs, cyan represents the clavicles, and brown highlights the catheter.

A List of publications

Most of the material of this thesis is already published, only parts are currently under review for publication.

Some material from the other domains like magnetic partical imaging has been omitted because it does not thematically fit into this thesis. In particular, the approach to learn a matrix completion method from complex-valued system matrices to reduce calibration time in magnetic partical imaging [Baltruschat et al., 2020c] is not included.

Below you will find a list of research grants and awards as well as all journal and conference publications, patents and technical reports that have emerged from my time as a doctoral student.

A.1 Grants and awards

MICCAI Student Travel Award (2020), award

Honorable Mention Poster Award, SPIE Medical Imaging (2018), award

I3 Junior Project: Joint medical Image-reconstruction and processing for high-dimensional data using deep learning [link] (2019), funded by Technischen Universität Hamburg, research grant

Travel expenses to the 16th IEEE ISBI (2019), funded by University of Hamburg, travel grant

Symposium: Images and Networks of the Brain (2018), funded by Akademie der Wissenschaften in Hamburg (2018), grant

Travel expenses to the RSNA (2018), funded by DAAD - Deutscher Akademischer Austauschdienst, travel grant

Travel expenses to the 3rd MISS (2018), funded by University of Hamburg, travel grant

FMTHH project: DAISY – Development of a joint UKE-TUHH Deep Learning platform for biomedical image processing [link] (2017), funded by Forschungszentrum Medizintechnik Hamburg, research grant

Travel expenses to the 10th ICVSS (2017), funded by Leidenberger-Müller-Stiftung, University of Hamburg, travel grant

A.2 Journal publications

Baltruschat, I., Steinmeister, L., Nickisch, H., Saalbach, A., Grass, M., Adam, G., Knopp, T., & Ittrich, H. (2020a) Smart chest X-ray worklist prioritization using artificial intelligence: A clinical workflow simulation *European radiology*, 1–9 (see p. 105)

Baltruschat, I. M., Nickisch, H., Grass, M., Knopp, T., & Saalbach, A. (2019c) Comparison of deep learning approaches for multi-label chest X-ray classification *Scientific reports*, **9**: (1), 6381 (see pp. 3, 14, 63, 90, 107, 120)

Gessert, N., Sentker, T., Madesta, F., Schmitz, R., Kniep, H., Baltruschat, I., Werner, R., & Schlaefer, A. (2019) Skin lesion classification using CNNs with patch-based attention and diagnosis-guided loss weighting *IEEE Transactions on Biomedical Engineering*, **67**: (2), 495–503

A.3 Conference publications

Baltruschat, I. M., Grass, M., Saalbach, A., Nickisch, H., von Berg, J., Steinmeister, L., Ittrich, H., Knopp, T., & Adam, G. (2019a) Neuronale netze zur pathologiedetektion bei röntgenthoraxuntersuchungen: Verbesserung durch intelligente vorverarbeitung, In *Deutscher röntgenkongress (RöKo)* (see p. 87)

Baltruschat, I. M., Griese, F., Szwargulski, P., Werner, R., & Knopp, T. (2019b) Wrestling the devil of wasting time: MPI system matrix recovery by deep learning, In *International workshop on magnetic particle imaging (IWMPI)*

Baltruschat, I. M., Nickisch, H., Grass, M., Knopp, T., & Saalbach, A. (2018b) Patient data adapted deep learning for multi-label chest X-ray classification, In *Radiological society of north america (RSNA)* (see p. 63)

Baltruschat, I. M., Saalbach, A., Heinrich, M. P., Nickisch, H., & Jockel, S. (2018c) Orientation regression in hand radiographs: A transfer learning approach, In *Medical imaging 2018: Image processing* International Society for Optics and Photonics

Baltruschat, I. M., Steinmeister, L., Ittrich, H., Adam, G., Nickisch, H., Saalbach, A., von Berg, J., Grass, M., & Knopp, T. (2019d) Abstract: Does bone suppression and lung detection improve chest disease classication? In *Bildverarbeitung für die medizin (BVM)* (see p. 87)

Baltruschat, I. M., Steinmeister, L., Ittrich, H., Adam, G., Nickisch, H., Saalbach, A., von Berg, J., Grass, M., & Knopp, T. (2019e) When does bone suppression and lung field segmentation improve chest X-ray disease classification? In *IEEE international symposium on biomedical imaging (ISBI)* IEEE (see pp. 4, 14, 24, 87, 107)

Baltruschat, I. M., Steinmeister, L., Ittrich, H., Nickisch, H., Saalbach, A., von Berg, J., Grass, M., Knopp, T., & Adam, G. (2019f) Combining effects of advanced image processing for automatic chest disease classification by ensembling and deep learning, In *Radiological society of north america (RSNA)* (see p. 87)

Baltruschat, I. M., Steinmeister, L. A., Nickisch, H., Saalbach, A., Grass, M., Adam, G., Knopp, T., & Ittrich, H. (2020b) A clinical workflow simulator for intelligent chest X-ray worklist prioritization, In *International symposium on biomedical imaging (ISBI)* (see pp. 4, 105)

Baltruschat, I. M., Szwargulski, P., Griese, F., Grosser, M., Werner, R., & Knopp, T. (2020c) 3d-SMRnet: Achieving a new quality of MPI system matrix recovery by deep learning, In *International conference on medical image computing and computer-assisted intervention (MICCAI)* Springer (see p. 127)

Gooßen, A., Deshpande, H., Harder, T., Schwab, E., Baltruschat, I. M., Mabotuwana, T., Cross, N., & Saalbach, A. (2019a) Pneumothorax detection and localization in chest radiographs: A comparison of deep learning approaches, In *International conference on medical imaging with deep learning (MIDL), extended abstract*

Grass, M., Baltruschat, I. M., Saalbach, A., Nickisch, H., von Berg, J., Adam, G., Steinmeister, L., Ittrich, H., & Knopp, T. (2019) Effect of advanced image preprocessing for multi-label chest X-ray classification, In *European congress of radiology (ECR)* (see p. 87)

Ittrich, H., Baltruschat, I. M., Steinmeister, L. A., Grass, M., Saalbach, A., Knopp, T., Adam, G., & Nickisch, H. (2018) Effect of inter-observer variability on deep learning in chest X-rays, In *Radiological society of north america (RSNA)* (see pp. 4, 87)

Sirazitdinov, I., Lenga, M., Baltruschat, I. M., & Saalbach, D. V. D. A. (2021) HRnet powerd constellation models for central venous catheter malposition detection in chest X-ray

Steinmeister, L. A., Baltruschat, I. M., Ittrich, H., Nickisch, H., Saalbach, A., Knopp, T., Adam, G., & Grass, M. (2019) Leveraging interobserver variability for sensitive pathology detection in chest X-rays, In *Radiological society of north america (RSNA)* (see pp. 4, 87)

Steinmeister, L. A., Baltruschat, I. M., Nickisch, H., Saalbach, A., Grass, M., Adam, G., Knopp, T., & Ittrich, H. (2020) Intelligent chest X-ray worklist prioritization by deep learning, In *European congress of radiology (ECR)* (see p. 105)

A.4 Patents

Baltruschat, I. M., Knoop, T., Nickisch, H., & Saalbach, A. (2018a) *System and method for image decomposition of a projection image* Patent Published, EP3513730A1 (see p. 123)

Baltruschat, I. M., Nickisch, H., & Saalbach, A. (2017) *Training an image analysis system* Patent Published, EP 3432313

Baltruschat, I. M., Saalbach, A., Grass, M., Ittrich, H., & Steinmeister, L. (2020d) *Automatic verification of central venous catheter positioning in chest X-rays and reporting* Patent Pending

Groth, A., Saalbach, A., Baltruschat, I. M., von Berg, J., & Grass, M. (2019) *Multi-task deep learning method for a neural network for automatic pathology detection* Patent Pending

Saalbach, A., Brosch, T., Harder, T., Deshpande, H. N., Schwab, E., Baltruschat, I. M., & Wiemker, R. (2018) *Medical image device and operating method* Patent Pending

Saalbach, A., Schulz, H., Grass, M., & Baltruschat, I. M. (2019) *Screen capturing via mobile computing devices using screen capturing* Patent Pending

A.5 Technical reports

Gessert, N., Sentker, T., Madesta, F., Schmitz, R., Kniep, H., Baltruschat, I., Werner, R., & Schlaefer, A. (2018) Skin lesion diagnosis using ensembles, unscaled multi-crop evaluation and loss weighting *arXiv*

Gooßen, A., Deshpande, H., Harder, T., Schwab, E., Baltruschat, I., Mabotuwana, T., Cross, N., & Saalbach, A. (2019b) Deep learning for pneumothorax detection and localization in chest radiographs *arXiv* (see pp. 15, 107)

Bibliography

Albarqouni, S., Fotouhi, J., & Navab, N. (2017) X-ray in-depth decomposition: Revealing the latent structures, In *International conference on medical image computing and computer-assisted intervention (MICCAI)* Springer (see p. 123)

Albaum, M. N., Hill, L. C., Murphy, M., Li, Y.-H., Fuhrman, C. R., Britton, C. A., Kapoor, W. N., & Fine, M. J. (1996) Interobserver reliability of the chest radiograph in community-acquired pneumonia *Chest*, **110**: (2), 343–350 (see pp. 10, 23)

Annarumma, M., Withey, S. J., Bakewell, R. J., Pesce, E., Goh, V., & Montana, G. (2019) Automated triaging of adult chest radiographs with deep artificial neural networks *Radiology*, **291**: (1), 196–202 (see p. 106)

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016) Layer normalization *arXiv* (see p. 56)

Baltruschat, I., Steinmeister, L., Nickisch, H., Saalbach, A., Grass, M., Adam, G., Knopp, T., & Ittrich, H. (2020a) Smart chest X-ray worklist prioritization using artificial intelligence: A clinical workflow simulation *European radiology*, 1–9 (see p. 105)

Baltruschat, I. M., Grass, M., Saalbach, A., Nickisch, H., VON Berg, J., Steinmeister, L., Ittrich, H., Knopp, T., & Adam, G. (2019a) Neuronale netze zur pathologiedetektion bei röntgenthoraxuntersuchungen: Verbesserung durch intelligente vorverarbeitung, In *Deutscher röntgenkongress (RöKo)* (see p. 87)

Baltruschat, I. M., Griese, F., Szwargulski, P., Werner, R., & Knopp, T. (2019b) Wrestling the devil of wasting time: MPI system matrix recovery by deep learning, In *International workshop on magnetic particle imaging (IWMPI)*

Baltruschat, I. M., Knoop, T., Nickisch, H., & Saalbach, A. (2018a) *System and method for image decomposition of a projection image* Patent Published, EP3513730A1 (see p. 123)

Baltruschat, I. M., Nickisch, H., Grass, M., Knopp, T., & Saalbach, A. (2018b) Patient data adapted deep learning for multi-label chest X-ray classification, In *Radiological society of north america (RSNA)* (see p. 63)

Baltruschat, I. M., Nickisch, H., Grass, M., Knopp, T., & Saalbach, A. (2019c) Comparison of deep learning approaches for multi-label chest X-ray classification *Scientific reports*, **9**: (1), 6381 (see pp. 3, 14, 63, 90, 107, 120)

Baltruschat, I. M., Nickisch, H., & Saalbach, A. (2017) *Training an image analysis system* Patent Published, EP 3432313

Baltruschat, I. M., Saalbach, A., Heinrich, M. P., Nickisch, H., & Jockel, S. (2018c) Orientation regression in hand radiographs: A transfer learning approach, In *Medical imaging 2018: Image processing* International Society for Optics and Photonics

Baltruschat, I. M., Steinmeister, L., Ittrich, H., Adam, G., Nickisch, H., Saalbach, A., VON Berg, J., Grass, M., & Knopp, T. (2019d) Abstract: Does bone suppression and lung detection improve chest disease classication? In *Bildverarbeitung für die medizin (BVM)* (see p. 87)

Baltruschat, I. M., Steinmeister, L., Ittrich, H., Adam, G., Nickisch, H., Saalbach, A., VON Berg, J., Grass, M., & Knopp, T. (2019e) When does bone suppression and lung field segmentation improve chest X-ray disease classification? In *IEEE international symposium on biomedical imaging (ISBI)* IEEE (see pp. 4, 14, 24, 87, 107)

Baltruschat, I. M., Steinmeister, L., Ittrich, H., Nickisch, H., Saalbach, A., VON Berg, J., Grass, M., Knopp, T., & Adam, G. (2019f) Combining effects of advanced image processing for automatic chest disease classification by ensembling and deep learning, In *Radiological society of north america (RSNA)* (see p. 87)

Baltruschat, I. M., Steinmeister, L. A., Nickisch, H., Saalbach, A., Grass, M., Adam, G., Knopp, T., & Ittrich, H. (2020b) A clinical workflow simulator for intelligent chest X-ray worklist prioritization, In *International symposium on biomedical imaging (ISBI)* (see pp. 4, 105)

Baltruschat, I. M., Szwargulski, P., Griese, F., Grosser, M., Werner, R., & Knopp, T. (2020c) 3d-SMRnet: Achieving a new quality of MPI system matrix recovery by deep learning, In *International conference on medical image computing and computer-assisted intervention (MICCAI)* Springer (see p. 127)

Baltruschat, I. M., Saalbach, A., Grass, M., Ittrich, H., & Steinmeister, L. (2020d) *Automatic verification of central venous catheter positioning in chest X-rays and reporting* Patent Pending

Bar, Y., Diamant, I., Wolf, L., Lieberman, S., Konen, E., & Greenspan, H. (2015) Chest pathology detection using deep learning with non-medical training *IEEE International Symposium on Biomedical Imaging (ISBI)*, 294–297 (see pp. 12, 14, 21, 63)

Beardmore, C., Woznitza, N., & Goodman, S. (2016) The radiography workforce: Current challenges and changing needs (see pp. 105, 106)

Becker, H., Nettleton, W., Meyers, P., Sweeney, J., & Nice, C. (1964) Digital computer determination of a medical diagnostic index directly from chest X-ray images *IEEE Transactions on Biomedical Engineering*, (3), 67–72 (see p. 20)

Berbaum, K., Franken, J. E., & Smith, W. (1985) The effect of comparison films upon resident interpretation of pediatric chest radiographs. *Investigative radiology*, **20**: (2), 124–128 (see p. 103)

Berlin, L. (2001) Statute of limitations and the continuum of care doctrine *American Journal of Roentgenology*, **177**: (5), 1011–1016 (see p. 106)

Bertrand, H., Hashir, M., & Cohen, J. P. (2019) Do lateral views help automated chest X-ray predictions? *arXiv* (see p. 14)

Beutel, J., Sonka, M., Kundel, H., Van Metter, R., Fitzpatrick, J., & OF Photo-optical Instrumentation Engineers, S. (2000) *Handbook of medical imaging: Medical image processing and analysis* Society of Photo Optical (see p. 11)

Bloomfield, F. H., Teele, R. L., Voss, M., Knight, D. B., & Harding, J. E. (1999) Inter-and intra-observer variability in the assessment of atelectasis and consolidation in neonatal chest radiographs *Pediatric radiology*, **29**: (6), 459–462 (see p. 23)

Brant, W., & Helms, C. (2007) *Fundamentals of diagnostic radiology* Lippincott, Williams & Wilkins (see p. 9)

Brosch, T., & Saalbach, A. (2018) Foveal fully convolutional nets for multi-organ segmentation, In *Medical imaging 2018: Image processing* International Society for Optics and Photonics (see p. 92)

Bundesamt für Strahlenschutz (2020) X-ray diagnostics: Frequency and radiation exposure [Available at https://www.bfs.de/EN/topics/ion/medicine/diagnostics/x-rays/frequency-exposure.html, Accessed 30 Oct. 2020] (see pp. 1, 7, 9, 26)

Burkov, A. (2019) *The hundred-page machine learning book* (Vol. 1) Andriy Burkov Quebec City, Can. (see p. 32)

Bustos, A., Pertusa, A., Salinas, J.-M., & DE LA Iglesia-Vayá, M. (2020) PadChest: A large chest X-ray image dataset with multi-label annotated reports *Medical image analysis*, **66**: 101797 (see pp. 17, 19)

Cai, J., Lu, L., Harrison, A. P., Shi, X., Chen, P., & Yang, L. (2018) Iterative attention mining for weakly supervised thoracic disease pattern localization in chest X-rays, In *International conference on medical image computing and computer-assisted intervention (MICCAI)* Springer (see pp. 15, 87)

Calli, E., Murphy, K., Sogancioglu, E., & VAN Ginneken, B. (2019) FRODO: Free rejection of out-of-distribution samples: Application to chest X-ray analysis *arXiv* (see p. 14)

Care Quality Commission (2017) Queen Alexandra hospital quality report [Available at https://www.cqc.org.uk/location/RHU03/reports, Accessed 5 Dec. 2019] (see pp. 1, 105)

Cauchy, A. (1847) Méthode générale pour la résolution des systemes d'équations simultanées *Comp. Rend. Sci. Paris*, **25**: (1847), 536–538 (see p. 38)

Chen, C., Dou, Q., Chen, H., & Heng, P.-A. (2018) Semantic-aware generative adversarial nets for unsupervised domain adaptation in chest X-ray segmentation, In *International workshop on machine learning in medical imaging (MLMI)* Springer (see p. 15)

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011) Natural language processing (almost) from scratch *Journal of machine learning research*, **12**: (ARTICLE), 2493–2537 (see p. 23)

Csurka, G., Dance, C. R., Fan, L., Willamowski, J., & Bray, C. (2004) Visual categorization with bags of keypoints, In *Workshop on statistical learning in computer vision at ECCV* (see p. 12)

Curry, H. B. (1944) The method of steepest descent for non-linear minimization problems *Quarterly of Applied Mathematics*, **2**: (3), 258–261 (see p. 38)

Cybenko, G. (1989) Approximation by superpositions of a sigmoidal function *Mathematics of control, signals and systems*, **2**: (4), 303–314 (see p. 30)

Daffner, R. (1999) *Clinical radiology: The essentials* (Second) Wolters Kluwer/Lippincott Williams & Wilkins (see p. 9)

Darby, M., Barron, D., & Hyland, R. (2012) *Oxford handbook of medical imaging* OUP Oxford (see p. 9)

Datta, S., Si, Y., Rodriguez, L., Shooshan, S. E., Demner-Fushman, D., & Roberts, K. (2020) Understanding spatial language in radiology: Representation framework, annotation, and spatial relation extraction from chest X-ray reports using deep learning *Journal of biomedical informatics*, **108**: 103473 (see p. 15)

Demner-Fushman, D., Kohli, M. D., Rosenman, M. B., Shooshan, S. E., Rodriguez, L., Antani, S., Thoma, G. R., & McDonald, C. J. (2016) Preparing a collection of radiology examinations for distribution and retrieval *Journal of the American Medical Informatics Association*, **23**: (2), 304–310 (see pp. 2, 3, 10, 16, 19, 22, 63, 88, 94, 107, 121)

Deng, L. (2012) The MNIST database of handwritten digit images for machine learning research [best of the web] *IEEE Signal Processing Magazine*, **29**: (6), 141–142 (see pp. 32, 33)

Dong, N., Kampffmeyer, M., Liang, X., Wang, Z., Dai, W., & Xing, E. (2018) Unsupervised domain adaptation for automatic estimation of cardiothoracic ratio, In *International conference on medical image computing and computer-assisted intervention (MICCAI)* Springer (see p. 15)

Duchi, J. C., Hazan, E., & Singer, Y. (2010) Adaptive subgradient methods for online learning and stochastic optimization *Journal of Machine Learning Research*, **12**: 2121–2159 (see p. 40)

Eldan, R., & Shamir, O. (2016) The power of depth for feedforward neural networks, In *Conference on learning theory (COLT)* (see p. 57)

Fortmann-Roe, S. (2012) Accurately measuring model prediction error [Available at http://scott.fortmann-roe.com/docs/MeasuringError.html, Accessed 18 Dec. 2020] (see p. 43)

Gasimova, A. (2019) *Automated enriched medical concept generation for chest X-ray images*. In *Interpretability of machine intelligence in medical image computing and multimodal learning for clinical decision support* (pp. 83–92) Springer (see p. 15)

Gaskin, C. M., Patrie, J. T., Hanshew, M. D., Boatman, D. M., & McWey, R. P. (2016) Impact of a reading priority scoring system on the prioritization of examination interpretations *American Journal of Roentgenology*, **206**: (5), 1031–1039 (see pp. 106, 119)

Ge, Z., Mahapatra, D., Sedai, S., Garnavi, R., & Chakravorty, R. (2018) Chest X-rays classification: A multi-label and fine-grained problem *arXiv* (see p. 14)

Gessert, N., Sentker, T., Madesta, F., Schmitz, R., Kniep, H., Baltruschat, I., Werner, R., & Schlaefer, A. (2018) Skin lesion diagnosis using ensembles, unscaled multi-crop evaluation and loss weighting *arXiv*

Gessert, N., Sentker, T., Madesta, F., Schmitz, R., Kniep, H., Baltruschat, I., Werner, R., & Schlaefer, A. (2019) Skin lesion classification using CNNs with patch-based attention and diagnosis-guided loss weighting *IEEE Transactions on Biomedical Engineering*, **67**: (2), 495–503

Glorot, X., Bordes, A., & Bengio, Y. (2011) Deep sparse rectifier neural networks, In *International conference on artificial intelligence and statistics (AISTATS)* (see p. 48)

Goodfellow, I., Bengio, Y., & Courville, A. (2016) *Deep learning* [http://www.deeplearningbook.org] MIT Press (see pp. 31, 32, 38, 39, 41, 42, 46, 55, 56)

Gooßen, A., Deshpande, H., Harder, T., Schwab, E., Baltruschat, I. M., Mabotuwana, T., Cross, N., & Saalbach, A. (2019a) Pneumothorax detection and localization in chest radiographs: A comparison of deep learning approaches, In *International conference on medical imaging with deep learning (MIDL), extended abstract*

Gooßen, A., Deshpande, H., Harder, T., Schwab, E., Baltruschat, I., Mabotuwana, T., Cross, N., & Saalbach, A. (2019b) Deep learning for pneumothorax detection and localization in chest radiographs *arXiv* (see pp. 15, 107)

Gordienko, Y., Gang, P., Hui, J., Zeng, W., Kochura, Y., Alienin, O., Rokovyi, O., & Stirenko, S. (2018) Deep learning with lung segmentation and bone shadow exclusion techniques for chest X-ray analysis of lung cancer, In *International conference on computer science, engineering and education applications (ICCSEEA)* (see p. 89)

Grass, M., Baltruschat, I. M., Saalbach, A., Nickisch, H., VON Berg, J., Adam, G., Steinmeister, L., Ittrich, H., & Knopp, T. (2019) Effect of advanced image preprocessing for multi-label chest X-ray classification, In *European congress of radiology (ECR)* (see p. 87)

Greenspan, H., Van Ginneken, B., & Summers, R. M. (2016) Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique *IEEE Transactions on Medical Imaging*, **35**: (5), 1153–1159 (see p. 21)

Groth, A., Saalbach, A., Baltruschat, I. M., VON Berg, J., & Grass, M. (2019) *Multi-task deep learning method for a neural network for automatic pathology detection* Patent Pending

Guan, Q., Huang, Y., Zhong, Z., Zheng, Z., Zheng, L., & Yang, Y. (2018) Diagnose like a radiologist: Attention guided convolutional neural network for thorax disease classification *arXiv* (see p. 14)

Guendel, S., Ghesu, F. C., Grbic, S., Gibson, E., Georgescu, B., Maier, A., & Comaniciu, D. (2019) Multi-task learning for chest X-ray abnormality classification on noisy labels *arXiv* (see p. 122)

Guendel, S., Grbic, S., Georgescu, B., Liu, S., Maier, A., & Comaniciu, D. (2018) Learning to recognize abnormalities in chest X-rays with location-aware dense networks, In *Iberoamerican congress on pattern recognition (CIARP)* Springer (see pp. 14, 64, 82, 84, 85)

Hanna, D., Griswold, P., Leape, L. L., & Bates, D. W. (2005) Communicating critical test results: Safe practice recommendations *The Joint Commission Journal on Quality and Patient Safety*, **31**: (2), 68–80 (see p. 106)

Hansen, L. K., & Salamon, P. (1990) Neural network ensembles *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**: 993–1001 (see p. 94)

Harzig, P., Chen, Y.-Y., Chen, F., & Lienhart, R. (2019) Addressing data bias problems for chest X-ray image report generation *arXiv*, **abs/1908.02123**: (see p. 15)

Hastie, T. J., Tibshirani, R., & Friedman, J. H. (2005) The elements of statistical learning: Data mining, inference, and prediction, In *Springer series in statistics* (see pp. 40–42, 44)

He, K., Zhang, X., Ren, S., & Sun, J. (2015a) Deep residual learning for image recognition *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778 (see pp. 12, 21, 57, 58, 62, 68)

He, K., Zhang, X., Ren, S., & Sun, J. (2015b) Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification *IEEE International Conference on Computer Vision (ICCV)*, 1026–1034 (see p. 48)

He, K., Zhang, X., Ren, S., & Sun, J. (2016) Identity mappings in deep residual networks, In *European conference on computer vision (ECCV)* (see pp. 2, 58, 63, 70, 74)

Heinrich, M. P. (2013) *Deformable lung registration for pulmonary image analysis of MRI and CT scans*. (Doctoral dissertation) Oxford University, UK (see p. 5)

Hopstaken, R., Witbraad, T., Van Engelshoven, J., & Dinant, G. (2004) Inter-observer variation in the interpretation of chest radiographs for pneumonia in community-acquired lower respiratory tract infections *Clinical radiology*, **59**: (8), 743–752 (see p. 23)

Hripcsak, G., Austin, J. H., Alderson, P. O., & Friedman, C. (2002) Use of natural language processing to translate clinical information from a database of 889,921 chest radiographic reports *Radiology*, **224**: (1), 157–163 (see p. 23)

Hripcsak, G., Kuperman, G. J., & Friedman, C. (1998) Extracting findings from narrative reports: Software transferability and sources of physician disagreement *Methods of information in medicine*, **37**: (01), 01–07 (see p. 23)

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017) Densely connected convolutional networks, In *IEEE conference on computer vision and pattern recognition (CVPR)* (see pp. 12, 64, 68)

Huda, W., & Abrahams, R. B. (2015) X-ray-based medical imaging and resolution *American Journal of Roentgenology*, **204**: (4), W393–W397 (see p. 20)

Hwang, S., & Park, S. (2017) *Accurate lung segmentation via network-wise training of convolutional networks*. In *Deep learning in medical image analysis and multimodal learning for clinical decision support* (pp. 92–99) Springer (see p. 15)

Imran, A.-A.-Z., & Terzopoulos, D. (2019) Semi-supervised multi-task learning with chest X-ray images, In *International workshop machine learning in medical imaging, MICCAI* Springer Nature (see p. 15)

Ioffe, S., & Szegedy, C. (2015) Batch normalization: Accelerating deep network training by reducing internal covariate shift, In *International conference on international conference on machine learning (ICML)* (see pp. 55, 56)

Irvin, J., Rajpurkar, P., Ko, M., Yu, Y., Ciurea-Ilcus, S., Chute, C., Marklund, H., Haghgoo, B., Ball, R. L., Shpanskaya, K. S., Seekins, J., Mong, D. A., Halabi, S. S., Sandberg, J. K., Jones, R., Larson, D. B., Langlotz, C. P., Patel, B. N., Lungren, M. P., & Ng, A. Y. (2019) CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison, In *Aaai conference on artificial intelligence* (see pp. 17, 19)

Islam, M. T., Aowal, M. A., Minhaz, A. T., & Ashraf, K. (2017) Abnormality detection and localization in chest X-rays using deep convolutional neural networks *arXiv* (see p. 15)

Ittrich, H., Baltruschat, I. M., Steinmeister, L. A., Grass, M., Saalbach, A., Knopp, T., Adam, G., & Nickisch, H. (2018) Effect of inter-observer variability on deep learning in chest X-rays, In *Radiological society of north america (RSNA)* (see pp. 4, 87)

Jaeger, S., Candemir, S., Antani, S., Wáng, Y.-X. J., Lu, P.-X., & Thoma, G. (2014) Two public chest X-ray datasets for computer-aided screening of pulmonary diseases *Quantitative imaging in medicine and surgery*, **4**: (6), 475 (see pp. 16, 19)

Johnson, A. E., Pollard, T. J., Berkowitz, S. J., Greenbaum, N. R., Lungren, M. P., Deng, C.-y., Mark, R. G., & Horng, S. (2019) MIMIC-CXR, a de-identified publicly available database of chest radiographs with free-text reports *Scientific Data*, **6**: (see pp. 17, 19)

Johnson, J., & Kline, J. A. (2010) Intraobserver and interobserver agreement of the interpretation of pediatric chest radiographs *Emergency radiology*, **17**: (4), 285–290 (see pp. 10, 23)

Karpathy, A. (2014) Stanford university cs231n: Convolutional neural networks for visual recognition (see p. 30)

Keller, J., Liu, D., & Fogel, D. (2016) *Fundamentals of computational intelligence: Neural networks, fuzzy systems, and evolutionary computation* Wiley (see p. 29)

Kesner, A., Laforest, R., Otazo, R., Jennifer, K., & Pan, T. (2018) Medical imaging data in the digital innovation age *Medical physics*, **45**: (4), e40–e52 (see p. 1)

Kiefer, J., Wolfowitz, J. et al. (1952) Stochastic estimation of the maximum of a regression function *The Annals of Mathematical Statistics*, **23**: (3), 462–466 (see p. 38)

Kim, H.-E., Kim, S., & Lee, J. (2018) Keep and learn: Continual learning by constraining the latent space for knowledge preservation in neural networks, In *International conference on medical image computing and computer-assisted intervention (MICCAI)* Springer (see pp. 13, 14)

Kingma, D. P., & Ba, J. (2015) Adam: A method for stochastic optimization, In *International conference on learning representations (ICLR)* (see pp. 39, 40, 74, 100)

Kohavi, R. et al. (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection, In *International joint conference on artificial intelligence (IJCAI)* Montreal, Canada (see pp. 41, 42, 44)

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012) *ImageNet classification with deep convolutional neural networks*. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (Eds.), *International conference on neural information processing systems (NeurIPS)* (pp. 1097–1105) (see pp. 2, 47, 55, 63, 68)

Krogh, A., & Hertz, J. A. (1992) A simple weight decay can improve generalization, In *International conference on neural information processing systems (NeurIPS)* (see p. 55)

Krogh, A., & Vedelsby, J. (1995) Neural network ensembles, cross validation, and active learning, In *International conference on neural information processing systems (NeurIPS)* (see p. 94)

Kukačka, J., Golkov, V., & Cremers, D. (2017) Regularization for deep learning: A taxonomy *arXiv* (see p. 55)

Landis, J. R., & Koch, G. G. (1977) The measurement of observer agreement for categorical data *biometrics*, 159–174 (see pp. 97, 98)

Lange, S., & Walsh, G. (2007) *Radiology of chest diseases* Thieme (see p. 9)

Laserson, J., Lantsman, C. D., Cohen-Sfady, M., Tamir, I., Goz, E., Brestel, C., Bar, S., Atar, M., & Elnekave, E. (2018) TextRay: Mining clinical reports to gain a broad understanding of chest X-rays, In *International conference on medical image computing and computer-assisted intervention (MICCAI)* Springer (see pp. 14, 26)

LeCun, Y., Boser, B. E., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. E., & Jackel, L. D. (1989) Backpropagation applied to handwritten zip code recognition *Neural Computation*, **1**: 541–551 (see p. 50)

LeCun, Y., Jackel, L. D., Bottou, L., Cortes, C., Denker, J. S., Drucker, H., Guyon, I., Muller, U. A., Sackinger, E., Simard, P., et al. (1995) Learning algorithms for classification: A comparison on handwritten digit recognition *Neural networks: the statistical mechanics perspective*, **261**: 276 (see pp. 32, 33)

Li, F., Engelmann, R., Pesce, L. L., Armato, S. G., & MacMahon, H. (2012) Improved detection of focal pneumonia by chest radiography with bone suppression imaging *European Radiology*, **22**: 2729–2735 (see p. 88)

Li, Z., Li, H., Han, H., Shi, G., Wang, J., & Zhou, S. K. (2019) Encoding CT anatomy knowledge for unpaired chest X-ray image decomposition, In *International conference on medical image computing and computer-assisted intervention (MICCAI)* Springer (see p. 123)

Li, Z., Wang, C., Han, M., Xue, Y., Wei, W., Li, L.-J., & Li, F. (2017) Thoracic disease identification and localization with limited supervision *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 8290–8299 (see pp. 64, 83, 87)

Lin, M., Chen, Q., & Yan, S. (2013) Network in network *arXiv* (see p. 59)

Liu, G., Hsu, T.-M. H., McDermott, M., Boag, W., Weng, W.-H., Szolovits, P., & Ghassemi, M. (2019) Clinically accurate chest X-ray report generation, In *Machine learning for healthcare conference (MLHC)* (see p. 15)

Long, J., Shelhamer, E., & Darrell, T. (2015) Fully convolutional networks for semantic segmentation, In *IEEE conference on computer vision and pattern recognition (CVPR)* (see p. 124)

Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013) Rectifier nonlinearities improve neural network acoustic models, In *International conference on machine learning (ICML)* (see p. 48)

MacMahon, H., Montner, S. M., Doi, K., & Liu, K. J. (1991) The nature and subtlety of abnormal findings in chest radiographs *Medical physics*, **18**: (2), 206–210 (see pp. 24, 25)

Mahapatra, D., Bozorgtabar, B., Thiran, J.-P., & Reyes, M. (2018) Efficient active learning for image classification and segmentation using a sample selection and conditional generative adversarial network, In *International conference on medical image computing and computer-assisted intervention (MICCAI)* Springer (see p. 15)

Majkowska, A., Mittal, S., Steiner, D. F., Reicher, J. J., McKinney, S. M., Duggan, G. E., Eswaran, K., Cameron Chen, P.-H., Liu, Y., Kalidindi, S. R., et al. (2020) Chest radiograph interpretation with deep learning models: Assessment with radiologist-adjudicated reference standards and population-adjusted evaluation *Radiology*, **294**: (2), 421–431 (see pp. 18, 19)

McCulloch, W. S., & Pitts, W. (1943) A logical calculus of the ideas immanent in nervous activity *The bulletin of mathematical biophysics*, **5**: (4), 115–133 https://doi.org/10.1007/BF02478259 (see pp. 30, 34)

Minsky, M., & Papert, S. (1969) *Perceptrons : An introduction to computational geometry* The MIT Press (see pp. 30, 35, 45)

Misra, D. (2019) Mish: A self regularized non-monotonic neural activation function *arXiv* (see p. 48)

Miyato, T., Kataoka, T., Koyama, M., & Yoshida, Y. (2018) Spectral normalization for generative adversarial networks *arXiv* (see p. 56)

Molinaro, A. M., Simon, R., & Pfeiffer, R. M. (2005) Prediction error estimation: A comparison of resampling methods *Bioinformatics*, **21**: (15), 3301–3307 (see pp. 44, 45)

Nair, V., & Hinton, G. E. (2010) Rectified linear units improve restricted boltzmann machines, In *International conference on machine learning (ICML)* (see pp. 47, 48)

Neuman, M. I., Lee, E. Y., Bixby, S., Diperna, S., Hellinger, J., Markowitz, R., Servaes, S., Monuteaux, M. C., & Shah, S. S. (2012) Variability in the interpretation of chest radiographs for the diagnosis of pneumonia in children *Journal of hospital medicine*, **7**: (4), 294–298 (see pp. 23, 103)

NHS England (2020) Diagnostic imaging dataset 2020-21 data [Available at https://www.england.nhs.uk/statistics/statistical-work-areas/diagnostic-imaging-dataset/, Accessed 30 Oct. 2020] (see pp. 1, 7, 9, 26)

Nishio, M., Fujimoto, K., & Togashi, K. (2019) Lung segmentation on chest X-ray images in patients with severe abnormal findings using deep learning *arXiv* (see p. 15)

Novack, V., Avnon, L. S., Smolyakov, A., Barnea, R., Jotkowitz, A., & Schlaeffer, F. (2006) Disagreement in the interpretation of chest radiographs among specialists and clinical outcomes of patients hospitalized with suspected pneumonia *European Journal of Internal Medicine*, **17**: (1), 43–47 (see p. 23)

Novikov, A. A., Lenis, D., Major, D., Hladuvka, J., Wimmer, M., & Bühler, K. (2018) Fully convolutional architectures for multiclass segmentation in chest radiographs *IEEE transactions on medical imaging*, **37**: (8), 1865–1876 (see p. 15)

Oakden-Rayner, L. (2017) Exploring the ChestXray14 dataset: Problems [Available at https://lukeoakdenrayner.wordpress.com/2017/12/18/, Accessed 5 Dec. 2019] (see pp. 23, 79, 85)

Oliva, A., & Torralba, A. (2001) Modeling the shape of the scene: A holistic representation of the spatial envelope *International journal of computer vision*, **42**: (3), 145–175 (see p. 12)

Ondategui-Parra, S., Bhagwat, J. G., Zou, K. H., Gogate, A., Intriere, L. A., Kelly, P., Seltzer, S. E., & Ros, P. R. (2004) Practice management performance indicators in academic radiology departments *Radiology*, **233**: (3), 716–722 (see p. 106)

Pan, S. J., & Yang, Q. (2010) A survey on transfer learning *IEEE Transactions on Knowledge and Data Engineering*, **22**: 1345–1359 (see p. 70)

Pesce, E., Withey, S. J., Ypsilantis, P.-P., Bakewell, R., Goh, V., & Montana, G. (2019) Learning to detect chest radiographs containing pulmonary lesions using visual attention networks *Medical image analysis*, **53**: 26–38 (see p. 15)

Philips Healthcare (2020) SkyPlate detector technical data [https://www.philips.de/healthcare/product/HCNOCTN343/skyplate-detektor-mobile-wlan-detektoren-24x3035x43/technische-daten, Accessed 8 Nov 2020] (see pp. 7, 9, 20)

Pikwer, A., Bååth, L., Davidson, B., Perstoft, I., & Åkeson, J. (2008) The incidence and risk of central venous catheter malpositioning: A prospective cohort study in 1619 patients *Anaesthesia and intensive care*, **36**: (1), 30–37 (see p. 124)

Potchen, E., Gard, J., Lazar, P., Lahaie, P., & Andary, M. (1979) Effect of clinical history data on chest film interpretation-direction or distraction, In *Investigative radiology* LIPPINCOTT-RAVEN PUBL 227 EAST WASHINGTON SQ, PHILADELPHIA, PA 19106 (see p. 103)

Putha, P., Tadepalli, M., Reddy, B., Raj, T., Chiramal, J. A., Govil, S., Sinha, N., KS, M., Reddivari, S., Jagirdar, A., et al. (2018) Can artificial intelligence reliably report chest X-rays?: Radiologist validation of an algorithm trained on 2.3 million X-rays *arXiv* (see pp. 14, 88)

Qian, N. (1999) On the momentum term in gradient descent learning algorithms. *Neural Networks*, **12**: (1), 145–151 (see p. 39)

Rachh, P., Levey, A. O., Lemmon, A., Marinescu, A., Auffermann, W. F., Haycook, D., & Berkowitz, E. A. (2018) Reducing STAT portable chest radiograph turnaround times: A pilot study *Current problems in diagnostic radiology*, **47**: (3), 156–160 (see p. 106)

Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., et al. (2017) CheXNet: Radiologist-level pneumonia detection on chest X-rays with deep learning *arXiv* (see pp. 12–14, 21, 64, 82, 83, 85)

Ramachandran, P., Zoph, B., & Le, Q. V. (2017) Searching for activation functions *arXiv* (see p. 48)

Rasamoelina, A. D., Adjailia, F., & Sinčák, P. (2020) A review of activation function for artificial neural network, In *IEEE world symposium on applied machine intelligence and informatics (sami)* IEEE (see p. 48)

Razavian, A. S., Azizpour, H., Sullivan, J., & Carlsson, S. (2014) CNN features off-the-shelf: An astounding baseline for recognition *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR)*, 512–519 (see p. 71)

Reiner, B. I. (2013) Innovation opportunities in critical results communication: Theoretical concepts *Journal of digital imaging*, **26**: (4), 605–609 (see p. 106)

Robbins, H., & Monro, S. (1951) A stochastic approximation method *The annals of mathematical statistics*, 400–407 (see p. 38)

Rosenblatt, F. F. (1962) *Principles of neurodynamics: Perceptrons and the theory of brain mechanisms* Spartan Books (see p. 34)

Royal College of Radiologists (2018) Clinical radiology UK workforce census report 2018 [Available at https://www.rcr.ac.uk/publication/clinical-radiology-uk-workforce-census-report-2018, Accessed 12 Dec. 2019] (see pp. 1, 105)

RSNA (2020) RSNA pneumonia detection challenge (2018) [Available at https://www.rsna.org/en/education/ai-resources-and-training/ai-image-challenge/RSNA-Pneumonia-Detection-Challenge-2018, Accessed 5 Dec. 2019] (see pp. 18, 19)

Rubin, J., Sanghavi, D., Zhao, C., Lee, K., Qadir, A., & Xu-Wilson, M. (2018) Large scale automated reading of frontal and lateral chest X-rays using dual convolutional neural networks *arXiv* (see p. 14)

Ruder, S. (2016) An overview of gradient descent optimization algorithms *arXiv* (see pp. 39, 40)

Ruder, S. (2017) An overview of multi-task learning in deep neural networks *arXiv* (see p. 122)

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986) Learning representations by back-propagating errors *Nature*, **323**: 533–536 (see pp. 2, 35, 36)

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M. S., Berg, A. C., & Li, F.-F. (2014) ImageNet large scale visual recognition challenge *International Journal of Computer Vision*, **115**: 211–252 (see pp. 12, 50, 71)

Saalbach, A., Brosch, T., Harder, T., Deshpande, H. N., Schwab, E., Baltruschat, I. M., & Wiemker, R. (2018) *Medical image device and operating method* Patent Pending

Saalbach, A., Schulz, H., Grass, M., & Baltruschat, I. M. (2019) *Screen capturing via mobile computing devices using screen capturing* Patent Pending

Salimans, T., & Kingma, D. P. (2016) Weight normalization: A simple reparameterization to accelerate training of deep neural networks, In *International conference on neural information processing systems (NeurIPS)* (see p. 56)

Santeramo, R., Withey, S., & Montana, G. (2018) *Longitudinal detection of radiological abnormalities with time-modulated LSTM*. In *Deep learning in medical image analysis and multimodal learning for clinical decision support (DLMIA)* (pp. 326–333) Springer (see p. 14)

Sartoretti, G., Kerr, J., Shi, Y., Wagner, G., Kumar, T. S., Koenig, S., & Choset, H. (2019) PRIMAL: Pathfinding via reinforcement and imitation multi-agent learning *IEEE Robotics and Automation Letters*, **4**: (3), 2378–2385 (see p. 124)

Sawant, A., Antonuk, L., & El-Mohri, Y. (2007) Slit design for efficient and accurate MTF measurement at megavoltage X-ray energies *Medical physics*, **34**: (5), 1535–1545 (see p. 20)

Schmidt, R. M., Schneider, F., & Hennig, P. (2020) Descending through a crowded valley–benchmarking deep learning optimizers *arXiv* (see p. 39)

scikit-learn (2020) Underfitting vs. overfitting — scikit-learn 0.24.0 documentation [Available at https://scikit-learn.org/stable/auto_examples/model_selection/plot_underfitting_overfitting.html, Accessed 15 Dec. 2020] (see p. 43)

Seide, F., & Agarwal, A. (2016) CNTK: Microsoft's open-source deep-learning toolkit, In *Acm international conference on knowledge discovery and data mining (KDD)* (see pp. 75, 100)

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017) Grad-CAM: Visual explanations from deep networks via gradient-based localization, In *IEEE international conference on computer vision (ICCV)* (see p. 78)

Shin, H.-C., Roberts, K., Lu, L., Demner-Fushman, D., Yao, J., & Summers, R. M. (2016) Learning to read chest X-rays: Recurrent neural cascade model for automated image annotation, In *IEEE conference on computer vision and pattern recognition (CVPR)* (see pp. 15, 87)

Shiraishi, J., Katsuragawa, S., Ikezoe, J., Matsumoto, T., Kobayashi, T., Komatsu, K.-i., Matsui, M., Fujita, H., Kodera, Y., & Doi, K. (2000) Development of a digital image database for chest radiographs with and without a lung nodule: Receiver

operating characteristic analysis of radiologists' detection of pulmonary nodules *American Journal of Roentgenology*, **174**: (1), 71–74 (see pp. 16, 19)

SIIM (2019) The pneumothorax challenge - society for imaging informatics in medicine [Available at https://siim.org/page/pneumothorax_challenge, Accessed 5 Dec. 2019] (see pp. 18, 19)

Simonyan, K., & Zisserman, A. (2015) Very deep convolutional networks for large-scale image recognition, In *International conference on learning representations (ICLR)* (see pp. 2, 12, 62, 63)

Singh, H., Arora, H. S., Vij, M. S., Rao, R., Khan, M. M., & Petersen, L. A. (2007) Communication outcomes of critical imaging results in a computerized notification system *Journal of the American Medical Informatics Association*, **14**: (4), 459–466 (see p. 106)

Sirazitdinov, I., Lenga, M., Baltruschat, I. M., & Saalbach, D. V. D. A. (2021) HRnet powerd constellation models for central venous catheter malposition detection in chest X-ray

Sital, C., Brosch, T., Tio, D., Raaijmakers, A., & Weese, J. (2020) 3d medical image segmentation with labeled and unlabeled data using autoencoders at the example of liver segmentation in ct images *arXiv* (see p. 92)

Song, G., Myeong, H., & Mu Lee, K. (2018) SeedNet: Automatic seed generation with deep reinforcement learning for robust interactive segmentation, In *IEEE conference on computer vision and pattern recognition (CVPR)* (see p. 124)

Spearman, C. (1961) The proof and measurement of association between two things. (See p. 78)

Springenberg, J. T., Dosovitskiy, A., Brox, T., & Riedmiller, M. (2014) Striving for simplicity: The all convolutional net *arXiv* (see p. 55)

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014) Dropout: A simple way to prevent neural networks from overfitting *Journal of Machine Learning Research*, **15**: 1929–1958 (see p. 55)

Srivastava, R. K., Greff, K., & Schmidhuber, J. (2015) Training very deep networks *International Conference on Neural Information Processing Systems (NeurIPS)*, **28**: 2377–2385 (see p. 46)

Steinmeister, L. A., Baltruschat, I. M., Ittrich, H., Nickisch, H., Saalbach, A., Knopp, T., Adam, G., & Grass, M. (2019) Leveraging interobserver variability for sensitive pathology detection in chest X-rays, In *Radiological society of north america (RSNA)* (see pp. 4, 87)

Steinmeister, L. A., Baltruschat, I. M., Nickisch, H., Saalbach, A., Grass, M., Adam, G., Knopp, T., & Ittrich, H. (2020) Intelligent chest X-ray worklist prioritization by deep learning, In *European congress of radiology (ECR)* (see p. 105)

Sun, C., Shrivastava, A., Singh, S., & Gupta, A. (2017) Revisiting unreasonable effectiveness of data in deep learning era, In *Ieee international conference on computer vision (ICCV)* (see p. 16)

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2014) Going deeper with convolutions *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–9 (see pp. 2, 63, 99)

Tan, M., & Le, Q. (2019) EfficientNet: Rethinking model scaling for convolutional neural networks, In *International conference on machine learning (ICML)* (see p. 2)

Tang, Y., Wang, X., Harrison, A. P., Lu, L., Xiao, J., & Summers, R. M. (2018) Attention-guided curriculum learning for weakly supervised classification and localization of thoracic diseases on chest radiographs, In *International workshop on machine learning in medical imaging (MLMI)* Springer (see p. 15)

Team PLCO Project, " " Gohagan, J. K., Prorok, P. C., Hayes, R. B., & Kramer, B.-S. (2000) The prostate, lung, colorectal and ovarian (PLCO) cancer screening trial of the national cancer institute: History, organization, and status *Controlled clinical trials*, **21**: (6), 251S–272S (see pp. 16, 19, 82)

The Joint Commission (2020) 2020 national patient safety goals [Available at http://www.jointcommission.org/standards_information/npsgs.aspx, Accessed 5 Nov., 2019] (see p. 106)

Tieleman, T., & Hinton, G. (2012) *Lecture 6.5 - rmsprop, coursera: Neural networks for machine learning.* (tech. rep.) University of Toronto (see p. 40)

Tudor, G., Finlay, D., & Taub, N. (1997) An assessment of inter-observer agreement and accuracy when reporting plain radiographs *Clinical radiology*, **52**: (3), 235–238 (see p. 23)

Ulyanov, D., Vedaldi, A., & Lempitsky, V. (2016) Instance normalization: The missing ingredient for fast stylization *arXiv* (see p. 56)

United Nations DESA (2019) Growing at a slower pace, world population is expected to reach 9.7 billion in 2050 and could peak at nearly 11 billion around 2100 [Available at https://www.un.org/development/desa/en/news/population/world-population-prospects-2019.html, Accessed 15 Dec. 2020] (see p. 1)

Van Metter, R. L., Beutel, J., & Kundel, H. L. (Eds.) (2000) *Handbook of medical imaging, volume 1. Physics and psychophysics* SPIE https://doi.org/10.1117/3.832716 (see p. 9)

VAN Ginneken, B., Hogeweg, L., & Prokop, M. (2009) Computer-aided diagnosis in chest radiography: Beyond nodules *European Journal of Radiology*, **72**: (2), 226–230

VAN Ginneken, B., Stegmann, M. B., & Loog, M. (2006) Segmentation of anatomical structures in chest radiographs using supervised methods: A comparative study on a public database *Medical image analysis*, **10**: (1), 19–40 (see pp. 17, 19)

VAN Ginneken, B., TER Haar Romeny, B. M., & Viergever, M. A. (2001) Computer-aided diagnosis in chest radiography: A survey *IEEE Transactions on Medical Imaging*, **20**: 1228–1241 (see p. 11)

VON Berg, J., Levrier, C., Carolus, H., Young, S., Saalbach, A., Laurent, P., & Florent, R. (2016) Decomposing the bony thorax in X-ray images *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, 1068–1071 (see pp. 88, 90, 92)

VON Berg, J., Young, S., Carolus, H., Wolz, R., Saalbach, A., Hidalgo, A., Giménez, A., & Franquet, T. (2015) A novel bone suppression method that improves lung nodule detection *International Journal of Computer Assisted Radiology and Surgery*, **11**: 641–655 (see pp. 88, 90)

Wang, H., Jia, H., Lu, L., & Xia, Y. (2019) Thorax-Net: An attention regularized deep neural network for classification of thoracic diseases on chest radiography *IEEE journal of biomedical and health informatics* (see p. 14)

Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., & Summers, R. M. (2017) ChestX-Ray8: Hospital-scale chest X-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases, In *IEEE conference on computer vision and pattern recognition (CVPR)* (see pp. 2, 17, 19, 23, 63–65, 68, 74, 75, 79, 82–85, 87, 88, 107)

Wang, X., Peng, Y., Lu, L., Lu, Z., & Summers, R. M. (2018) TieNet: Text-image embedding network for common thorax disease classification and reporting in chest X-rays, In *IEEE conference on computer vision and pattern recognition (CVPR)* (see pp. 15, 88)

Wesp, W. (2006) Using stat properly *Radiol Manage*, **28**: (1), 26–30 (see p. 106)

Widrow, B. (1960) *An adaptive "ADALINE" neuron using chemical "memistors"* (tech. rep.) Solid-State Electronics Laboratory, Stanford Electronics Laboratories, Stanford University Stanford, California (see p. 34)

Widrow, B., & Hoff, M. E. (1960) *Adaptive switching circuits* (tech. rep.) Stanford Univ Ca Stanford Electronics Labs (see p. 34)

Wikimedia Commons (2018) File:structure of neuron.png - wikimedia commons [Available at https : / / commons . wikimedia . org / wiki / File : Structure _ of _ Neuron.png, Accessed 11 Dec. 2020] (see p. 30)

Wu, Y., & He, K. (2018) Group normalization, In *European conference on computer vision (ECCV)* (see p. 56)

Xing, Y., Ge, Z., Zeng, R., Mahapatra, D., Seah, J., Law, M., & Drummond, T. (2019) Adversarial pulmonary pathology translation for pairwise chest X-ray data

augmentation, In *International conference on medical image computing and computer-assisted intervention (MICCAI)* Springer (see p. 15)

Xu, B., Wang, N., Chen, T., & Li, M. (2015) Empirical evaluation of rectified activations in convolutional network *arXiv* (see p. 48)

Xu, Q.-S., Liang, Y.-Z., & Du, Y.-P. (2004) Monte carlo cross-validation for selecting a model and estimating the prediction error in multivariate calibration *Journal of Chemometrics*, **18**: (2), 112–120 (see p. 45)

Yan, C., Yao, J., Li, R., Xu, Z., & Huang, J. (2018) Weakly supervised deep learning for thoracic disease classification and localization on chest X-rays, In *ACM international conference on bioinformatics, computational biology, and health informatics (ACM-BCB)* ACM (see p. 14)

Yaniv, G., Kuperberg, A., & Walach, E. (2018) Deep learning algorithm for optimizing critical findings report turnaround time, In *Siim (society for imaging informatics in medicine) annual meeting* (see p. 106)

Yao, L., Poblenz, E., Dagunts, D., Covington, B., Bernard, D., & Lyman, K. (2017) Learning to diagnose from scratch by exploiting dependencies among labels *arXiv* (see pp. 14, 21, 64)

Yao, L., Prosky, J., Poblenz, E., Covington, B., & Lyman, K. (2018) Weakly supervised medical diagnosis and localization from multiple resolutions *arXiv* (see pp. 15, 64, 82–84)

Yi, X., Adams, S., Babyn, P., & Elnajmi, A. (2020) Automatic catheter and tube detection in pediatric X-ray images using a scale-recurrent network and synthetic data *Journal of digital imaging*, **33**: (1), 181–190 (see p. 124)

Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014) How transferable are features in deep neural networks? In *International conference on neural information processing systems (NeurIPS)* (see p. 71)

Youden, W. J. (1950) Index for rating diagnostic tests *Cancer*, **3**: (1), 32–35 (see p. 79)

Ypsilantis, P.-P., & Montana, G. (2017) Learning what to look in chest X-rays with a recurrent visual attention model *arXiv* (see p. 14)

Zech, J. R., Badgeley, M. A., Liu, M., Costa, A. B., Titano, J. J., & Oermann, E. K. (2018) Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study *PLoS medicine*, **15**: (11), e1002683 (see p. 14)

Zeiler, M. D., & Fergus, R. (2014) Visualizing and understanding convolutional networks, In *European conference on computer vision (ECCV)* Springer (see p. 50)

Zheng, S., Song, Y., Leung, T., & Goodfellow, I. J. (2016) Improving the robustness of deep neural networks via stability training *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4480–4488 (see p. 61)

Zhou, Y. T., & Chellappa, R. (1988) Computation of optical flow using a neural network *IEEE International Conference on Neural Networks (ICNN)*, 71–78 vol.2 (see p. 55)