
# Analyzing Capabilities and Methods of Deep Learning Models in Medical Diagnostics

im Rahmen der Prüfung zum

**Bachelor of Science (B.Sc.)**

des Studienganges Informatik

an der Dualen Hochschule Baden-Württemberg Karlsruhe

von

**Theresa Geber**

und

**Niklas Waibel**

| | |
|---|---|
| Abgabedatum: | 19. Mai 2025 |
| Bearbeitungszeitraum: | 30.09.2024 - 19.05.2025 |
| Matrikelnummer, Kurs: | 0000000, TINF22B1 |
| Matrikelnummer, Kurs: | 0000000, TINF22B1 |
| Betreuer der Studienarbeit: | Prof. Dr. Markus Reischl (KIT) |

# Eidesstattliche Erklärung

Wir versichern hiermit, dass wir unsere Studienarbeit (T3_3100 & T3_3200) mit dem Thema:

*Analyzing Capabilities and Methods of Deep Learning Models in Medical Diagnostics*

gemäß § 5 der "Studien- und Prüfungsordnung DHBW Technik" vom 29. September 2017 selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Wir versichern zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt.

Karlsruhe, den March 26, 2025

Datum fixen

Geber, Theresa

Waibel, Niklas

# Contents

# List of Abbriviations

| | |
|---|---|
| **Adam** | Adaptive Moment Estimation |
| **AI** | Artificial Intelligence |
| **ANN** | Artifical Neural Network |
| **AP** | Anteroposterior |
| **AUC** | Area Under Curve |
| **CAV** | Concept Activation Vector |
| **CNN** | Convolutional Neural Network |
| **CPU** | Central Processing Unit |
| **CV** | Computer Vision |
| **DL** | Deep Learning |
| **DNN** | Deep Neural Network |
| **FCN** | Fully Convolutional Network |
| **GAN** | Generative Adversarial Networks |
| **GAM** | Generalized Additive Model |
| **GPU** | Graphical Processing Unit |
| **Grad-CAM** | Gradient-weighted Class Activation Mapping |
| **IG** | Integrated Gradients |
| **LIME** | Local Interpretable Model-Agnostic Explanations |
| **ML** | Machine Learning |
| **NLP** | Natural Language Processing |
| **OCR** | Optical Character Recognition |
| **PA** | Posteroanterior |
| **PNG** | Portable Network Graphics |

| | |
|---|---|
| **RL** | Reinforcement Learning |
| **RNN** | Recurrent Neural Networks |
| **RMSProp** | Root Mean Square Propagation |
| **SGD** | Stochastic Gradient Decent |
| **SHAP** | Shapley Additive Explanations |
| **SSD** | Single Shot MultiBox Detector |
| **TCAV** | Testing with Concept Activation Vectors |
| **XAI** | Explainable AI |
| **YOLO** | You Only Look Once |

# List of Figures

# List of Tables

# List of Listings

# 1. Foundations and State of the Art

The goal of this chapter is to explain the technical foundations this work is based on. This includes an overview about Deep Learning (DL) and an insight into its architecture. Afterwards, we introduce the topic of Computer Vision (CV) and provide information about the topic of explainability.

## 1.1. Introduction to Deep Learning

In the rapidly advancing domain of Artificial Intelligence (AI), it is imperative to delineate the distinct branches of study, notably Machine Learning (ML) and its more specialized subset, DL. This section aims to establish a foundation by exploring fundamental terminologies and their interrelationships.

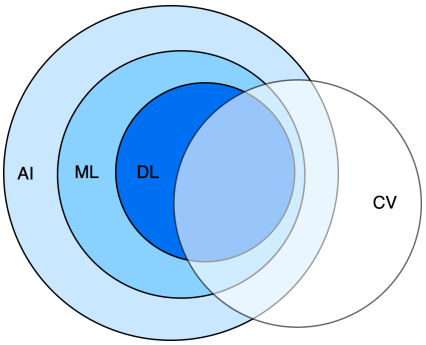


Figure 1.1.: An attempt on modeling the relationship between the different areas of AI; Source: based on [1, 2]

ML is a subset of AI that allows computers to learn from data without "needing to be explicitly programmed". Overall, recent progress in ML algorithms and methodologies has led to the development of intelligence systems that demonstrate cognitive capabilities comparable to certain human cognitive functions [3].

While traditional ML techniques have enabled algorithms to perform tasks such as classification, clustering, and data generation [4], they often rely on manual feature

extraction and domain-specific knowledge for transforming raw data into a usable format [5]. This dependency often requires significant human intervention and expertise.

DL, however, represents a paradigm shift within ML [6], enabling models to automatically learn complex representations of data at multiple levels of abstraction [7]. Employing the same learning paradigms as ML (supervised, unsupervised, and Reinforcement Learning (RL)), DL leverages Deep Neural Networks (DNNs) to process raw data directly, minimizing the need for manual feature extraction [8]. This capability allows DL models to excel in tasks that involve high-dimensional, unstructured data [7], such as image recognition, Natural Language Processing (NLP), and game playing, going beyond the performance limitations of traditional ML algorithms.

## 1.2. Deep Learning Architectures

At its core, DL utilizes neural networks, specifically DNNs, which are composed of multiple layers of interconnected nodes. This structure is designed to recognize patterns and make decisions based on input data, amplifying to simulate and copy how human brains process information.

The central component of a neural network is the neuron (or node), which mimics the functionality of biological neurons found in brains [9]. Each neuron in a neural network receives one or more inputs, processes them by performing a weighted sum followed by a non-linear activation function, and then produces an output [10]. This output can either be used directly as the final result or passed on to additional neurons in subsequent layers.

A typical DNN consists of three distinct layer types: (a) one input layer, which accepts data from external sources, (b) one or more hidden layers, which perform computations based on the provided functions, and (c) an output layer, which generates the final output determined by the given input [9].

Historically, the conceptual framework of neural networks was first established by Warren McCulloch and Walter Pitts in the 1940s, who introduced the concept of a neuron as a binary threshold unit [11]. However, the successful training of deep neural architectures remained a formidable challenge until the development of the backpropagation algorithm

by Rumelhart, Hinton, and Williams in the 1980s [12]. The renaissance of neural networks in the early 2000s propelled by the advancements in computational power and the availability of large-scale datasets, has resulted in the widespread application of DL techniques [13].

### 1.2.1. Deep Neural Networks and Their Structure

DNNs are an extension of the basic neural network framework. They include multiple layers of neurons, each layer transforming the input data into a slightly more abstract and composite representation, allowing them to learn complex patterns. The "depth" of a neural network refers to the number of layers it has. A network is considered a deep neural network if it has multiple, at least three, hidden layers. Otherwise, it can be considered a shallow Artifical Neural Network (ANN) [14]. Each neuron in a layer is typically connected to every neuron in the next layer, making it a fully connected network.

In addition to the architectural structure of layers and neurons, several fundamental components are integral to the functionality and performance of neural networks. These key elements include weights and biases, activation functions, and optimization algorithms.

**Weights and Biases** are essential for their functionality and learning capabilities, influencing how well they generalize to unseen data. Weights are numerical values that quantify the strength and direction of connections between neurons, thereby determining the influence exerted by one neuron's output on the input to another neuron. During training, weights are adjusted using optimization algorithms like backpropagation to minimize the difference between predicted and actual outputs [15]. Biases are additional parameters added to the weighted sum of inputs prior to the application of the activation function, thereby enabling the model to adjust the activation threshold to achieve a more precise fit to the underlying data distribution [5].

**Activation functions** are mathematical transformations applied to the output of each neuron that allow the model to learn and represent complex, non-linear relationships between the input data and the output [15]. Common activation functions include:

1. Sigmoid: $\sigma(z) = \frac{1}{1+e^{-z}}$, squashes inputs to a range between 0 and 1 [5].
2. Tanh: $\tanh(z) = \frac{e^{z}-e^{-z}}{e^{z}+e^{-z}}$, squashes inputs to a range between -1 and 1 [5].
3. ReLU: $g(z) = \max(0, z)$ , retains positive inputs and discards negatives, aiding in mitigating the vanishing gradient problem [5].
4. Leaky ReLU: $g(z) = \begin{cases} z & \text{if } z > 0 \\ \alpha z & \text{otherwise} \end{cases}$ , allows small values for negative inputs and passes positive inputs through, addressing the dying ReLU issue [15].
5. Softmax: $g(z_i) = \frac{e^{z_j}}{1+\sum_{c=1}^{C} e^{z_c}}$ converts logits into probabilities summing to 1, facilitating multi-class classification by representing class probabilities [15].

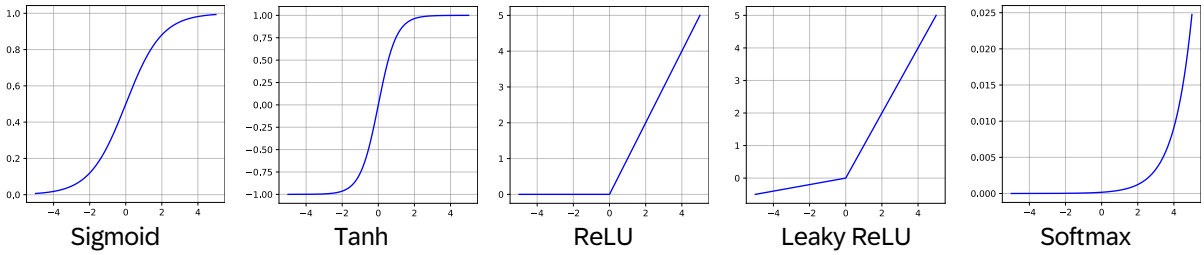


Figure 1.2.: Common Activation Functions visually displayed

**Loss Functions** measure the deviation of the predicted output and the actual output. The objective during the training phase is to minimize this loss function [16, 17].

1. Mean Squared Error: $MSE = \frac{1}{n}\sum_{i=1} n(y_i - \hat{y}_i)^2$, primarily used for regression tasks.
2. Cross-Entropy Loss: $CCE = -\sum_{i=1}^{C} y_i \log(\hat{y}_i)$, (or other cross entropy variations) are primarily used for (image) classification tasks, measuring the difference between two probability distributions.

**Backpropagation** is a fundamental algorithm for training neural networks, consisting of two main steps: the forward pass and the backward pass. In the forward pass, input data traverses through the network to produce predictions, which are subsequently compared to actual target values, resulting in the computation of a loss function to quantify error [15]. In the backward pass, backpropagation leverages the chain rule of calculus to efficiently compute the gradients of the loss function with respect to each weight in the network. This process is accomplished by propagating the error backward through the network,

one layer at a time, enabling the determination of each weight's contribution to the total error [18]. These gradients are then utilized in conjunction with optimization algorithms.

**Optimizers** play a crucial role in training neural networks by updating the model's weights and biases to minimize the loss function based on the gradients computed during backpropagation. The choice of optimizer can significantly impact the convergence speed, generalization performance, and overall effectiveness of the training process.

1. Stochastic Gradient Decent (SGD): Improves upon standard gradient descent by using a single sample or a mini-batch to compute gradients at each iteration. This approach introduces noise into the optimization process, which can help escape local minima. However, it may also lead to higher variance in updates and oscillations around the minimum, potentially slowing convergence [19].
2. Root Mean Square Propagation (RMSProp): Adjusts the learning rate according to the magnitude of recent gradients. This allows RMSProp to adjust the learning rate dynamically, giving more weight to recent gradients while effectively disregarding older ones. This promotes to a more stable and balanced convergence while also addressing the diminishing learning rate problem [20].
3. Adaptive Moment Estimation (Adam): An adaptive optimization algorithm that integrates the strengths of RMSProp. Adam computes individual adaptive learning rates for different parameters by estimating the mean of gradients and the uncentered variance of gradients of the gradients, dynamically adjusting the step size for each parameter [19].

**Regularization and Normalization Techniques** improve the model's generalization performance by reducing overfitting. Overfitting occurs when a model learns the noise in the training data rather than the underlying patterns, resulting in poor performance on unseen data. To address this challenge, a variety of effective regularization methods can be employed. One dynamic approach is *Dropout* where a fraction of neurons is stochastically set to zero during each training iteration, effectively "dropping out" these neurons from the network. This process prevents the network from developing an overreliance on specific neurons, promoting redundancy and robustness in the learned representation [5].

Another technique, *Batch Normalization*, is designed to stabilize and accelerate the training process of neural networks by normalizing the output at each layer. This is achieved by applying a transformation that scales and shifts the output within each mini-batch, resulting in a normalized distribution with zero mean and unit variance [5].

Additionally, *Early Stopping* is a regularization technique that terminates training when the performance on a validation set stops improving, preventing the model from overfitting to the training data [15]. However, numerous additional regularization techniques exist that are not covered within the scope of this paper.

**Hyperparameters** represent a class of parameters that are predetermined manually before the initiation of the training process, as opposed to being learned by the neural network itself. Notable hyperparameters include the learning rate, the loss function, the number of hidden layers, the number of neurons within each layer and the batch size [17]. More comprehensive and specific hyperparameters will be covered in the subsequent section, *Convolutional Neural Networks and Their Structure*.

## 1.2.2. Convolutional Neural Networks and Their Structure

Convolutional Neural Networks (CNNs) represent a specialized architecture within the realm of DL, specifically designed to process and analyze structured grid-like data, notably images. The unique structure of CNNs leverages spatial hierarchies and local patterns, making them particularly effective for tasks in CV, encompassing image classification, object detection, and segmentation, addressed later in this paper [5, 15].

### 1.2.2.1. Layers of a Convolutional Neural Network

CNN architectures comprise multiple layer types, each designed to fulfill a particular role in the processing of input data. These layers include convolutional layers, pooling layers, and fully connected layers.

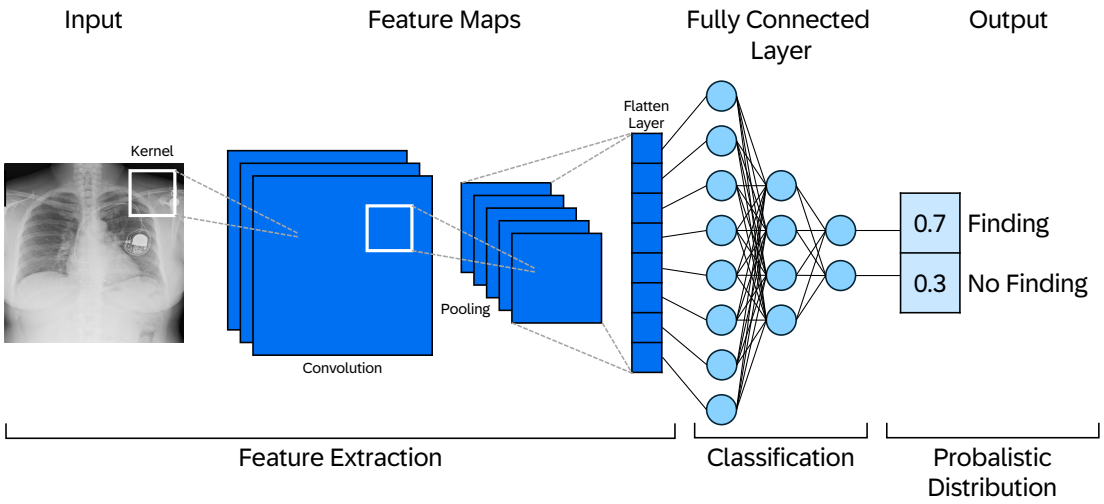


Figure 1.3.: The general structure of a CNN

**Convolution Layers** are the cornerstone of CNNs. They perform a convolution operation, which involves sliding a set of learnable filters (kernels) over the input data (tensor) to produce feature maps. Each kernel, a small, spatially-localized weight matrices, detects specific local patterns in the input, such as edges or textures [21]. The size of these filters (e.g., $(3 \times 3)$ or $(5 \times 5)$) determines the scope of feature detection, while the stride, specifying the step size at which the filter advances across the input data, determines the spatial dimensions of the output feature map. A larger stride value results in a smaller output feature map, as it bypasses more elements of the input during the convolution operation [5, 22].

The convolution operation can be mathematically expressed as:

$$(f * g)(x, y) = \sum_m \sum_n f(m, n) g(x - m, y - n) \tag{1.1}$$

where $f$ is the filter, and $g$ is the input feature map. The resultant output feature map emphasizes regions of the input that match the patterns encoded by the filters, effectively highlighting features of the input data that the filters were trained to recognize. However, the convolution operation itself may alter the spatial dimensions of the input data, depending on the size of the filters and the stride value used [23].

To maintain the spatial dimensions of the input data after the convolution operation, padding is often applied by adding extra rows and columns around the input. Common

padding strategies include "valid" (no padding) and "same" (padding added to ensure the output has the same dimensions as the input) [21].

**Pooling Layers** reduce the spatial dimensions of the feature maps, consequently reducing the number of parameters and computational complexity within the CNN simultaneously acting as a safeguard against overfitting. *Max Pooling* and *Average Pooling*, two frequently used pooling operations, capture the maximum or average value in each patch of the feature map [22]. Mathematically, they can be expressed with

$$y_{\text{max\_pooling}} = \max(x_1, x_2, \ldots, x_n) \tag{1.2}$$

$$y_{\text{average\_pooling}} = \text{avg}(x_1, x_2, \ldots, x_n) \tag{1.3}$$

.

where $x_i$ are the elements within the pooling window. This operation preserves important features while reducing the dimensionality [24].

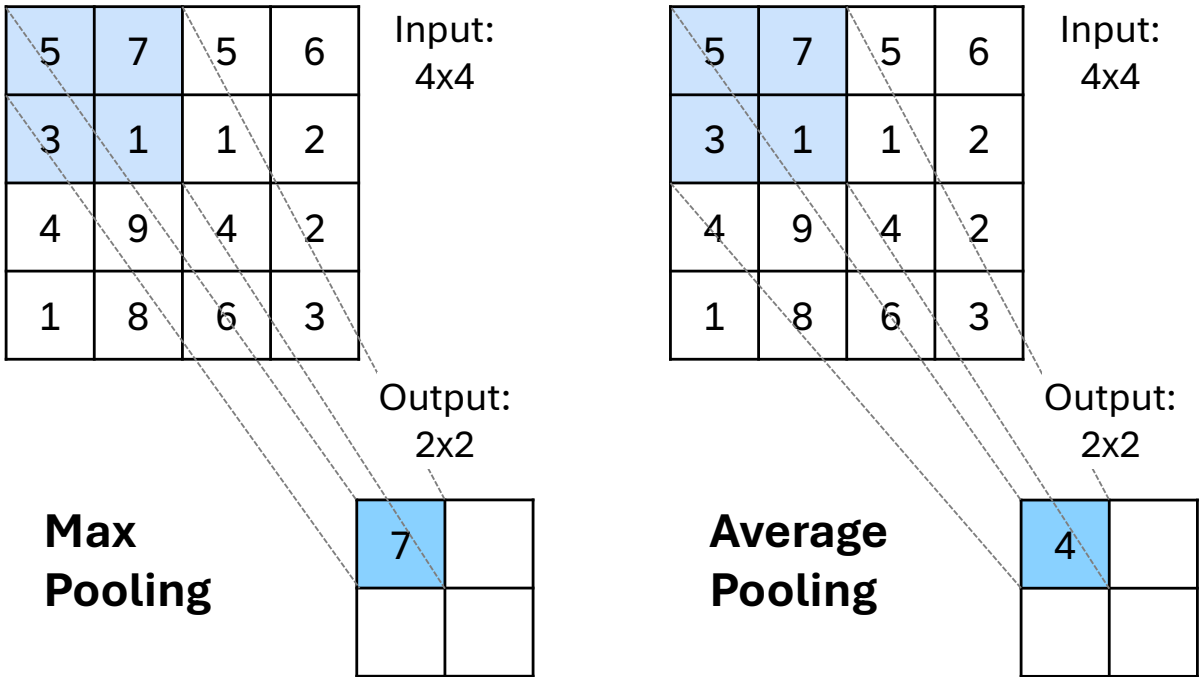


Figure 1.4.: Max and Average Pooling visually explained

**Fully Connected (Dense) Layers** are characterized by every neuron being connected to all neurons in the previous layer, allowing for comprehensive integration of features extracted from earlier layers. Usually positioned towards the end of the network, these layers operate as classifiers, interpreting the high-level features produced by convolutional and pooling layers [25]. Despite their effectiveness in recognizing global patterns, the dense connectivity of these layers can result in significant computational expense and a higher risk of overfitting, necessitating careful management through techniques such as regularization [26].

#### 1.2.2.2. Transfer learning and Common CNN Architectures

Transfer learning is a technique that allows knowledge gained from one task to be applied to another, often related, task. This is particularly beneficial in scenarios where labeled data is scarce. By utilizing pre-trained CNNs, which have learned general features from large datasets like ImageNet, one can significantly reduce training time and improve model performance on specific tasks. The process typically involves freezing the early layers of a pre-trained model and fine-tuning the later layers to adapt to the new dataset.

The emergence of transfer learning has substantially augmented the capabilities of CNN architectures, facilitating the use of pre-trained models for new tasks with limited data [27]. After the introduction to the principles of transfer learning, this chapter will provide a brief summary of the various architectures.

**AlexNet** (developed in 2012), a deeper architecture with five convolutional layers followed by three fully connected layers, revolutionized CV by significantly outperforming traditional methods in image classification challenges. It introduced ReLU activations and extensive use of data augmentation and dropout [28].

**GoogLeNet (Inception)** (developed in 2014) features the Inception module, which allows for parallel convolutional operations with multiple filter sizes ($1\times1$, $3\times3$, and $5\times5$) within the same layer, enabling effective multi-scale feature extraction. The architecture employs global average pooling instead of fully connected layers, reducing parameters and enhancing performance. Its ability to capture multi-scale features makes it effective for complex medical images, allowing for improved sensitivity and specificity in tasks like tumor detection and classification [28, 29].

**ResNet** (developed in 2015) introduced the concept of residual blocks, which allow the network to learn residual functions with reference to the input of a layer, thus mitigating the vanishing gradient problem in very deep networks. This architecture enables the training of extremely deep networks with hundreds or even thousands of layers, making it ideal for medical image processing where depth can enhance feature extraction [22, 27].

**DenseNet** (developed in 2017) enhances feature propagation by connecting each layer to all previous layers, promoting feature reuse and improving gradient flow. This architecture reduces the number of parameters needed compared to traditional networks while maintaining high performance in image classification tasks, making it a strong candidate for transfer learning in medical applications [27, 30]

**MobileNetV2** (developed in 2018) introduces the inverted residual structure, improving both accuracy and efficiency of DNNs by reducing the number of channels and increasing it after the computions. Additionally, depthwise separable convolutions and linear bootleneck layers make MobileNetV2 a lightweight model and hence, particularly suitable for real-time applications in mobile and edge computing environments [31].

**EfficientNet** (developed in 2019) optimizes CNN architecture through a compound scaling method that balances depth, width, and resolution. This allows EfficientNet models to achieve high accuracy with significantly fewer parameters than previous architectures. Its efficiency and performance have made it a popular choice for transfer learning across various domains, particularly in resource-constrained environments [28].

**U-Net** (developed in 2015) is widely recognized for its effectiveness in biomedical image segmentation. Its encoder-decoder structure is specifically designed for pixel-wise classification, making it particularly suitable for tasks such as tumor segmentation in histopathology images. U-Net can easily leverage transfer learning by fine-tuning on specific datasets [29, 32].

If not used in this paper, rather remove it

### 1.2.3. Further Deep Learning Architectures

To ensure a comprehensive discussion, it is crucial to recognize the existence of various other architectures within the landscape of DL. These architectures, including Multi-Layer Perceptrons (MLPs), Recurrent Neural Networkss (RNNs), and Generative Adversarial Networkss (GANs), among others, have undeniably made significant contributions to

the field. However, they are primarily optimized for applications other than image classification [8]. GANs, for instance, are more geared towards creating new images rather than accurately classifying existing ones [33]. Similarly, RNNs are specialized for handling sequential data and are predominantly used in natural language processing tasks rather than image analysis [34].

## 1.3. Computer Vision - An Application of Neural Networks

CV is an interdisciplinary domain that focuses on developing algorithms, techniques, and systems that allow computers to process, analyze, and comprehend visual data from the environment, mimicking the complexity of human visual perception. This domain encompasses a broad range of tasks, including image classification, object detection, image segmentation or Optical Character Recognition (OCR) [35]. DL architectures, notably CNNs, have transformed the field of CV by significantly enhancing the capability of machines to perform complex visual tasks with high accuracy and efficiency [36].

### 1.3.1. Foundational Concepts and Tasks in Computer Vision

CNNs have firmly established themselves as the cornerstone of modern Computer Vision among the diverse array of DL architectures, primarily due to their unparalleled proficiency in handling image data [36]. As discussed in the previous section, the core components of CNNs (convolutional layers, pooling layers, and fully connected layers) work synergistically by utilizing local connections to concentrate on small regions of the input image and progressively combining these basic features to form more complex patterns at deeper layers. In the initial layers, CNNs identify low-level features such as edges and textures, whereas the deeper layers are capable of recognize more complex and high-level features, enabling the efficient capturing and understanding of visual information [37].

Consequently, CNNs demonstrate exceptional performance in a variety of CV tasks:

1. Image Classification: CNNs learn to categorize images into predefined classes by identifying distinguishing features from labeled datasets [38].

2. Object Detection: Techniques like Region-based CNNs (R-CNN), You Only Look Once (YOLO), and Single Shot MultiBox Detector (SSD) locate and identify objects within images [39, 40].
3. Image Segmentation: Architectures such as U-Net and Fully Convolutional Network (FCN) perform pixel-wise classification to delineate object boundaries, dividing images into segments according to those boundaries [41].
4. Image Generation: GANs, commonly incorporating CNN layers, exhibit proficiency to generate innovative and realistic images derived from learned probabilistic distributions [33].

### 1.3.2. Applications and Impact

The application of DL in the field of CV has transcended academic research, with profound implications across various industries. In the agriculture sector, precision farming benefits from CV through automated crop monitoring, pest detection, and yield prediction, optimizing resource utilization and improving crop quality [42]. Similarly, the development of autonomous vehicles has been propelled by advancements in CV systems. Self-driving cars can now perceive and navigate their environment with unprecedented accuracy, recognizing objects, pedestrians, and road signs in real-time, thus promoting safer and more efficient transportation [42]. Moreover, the retail industry has also undergone significant transformations thanks to the application of CV. CV-driven automated checkout systems, powered by advanced object recognition and tracking algorithms, have streamlined the shopping experience for customers while reducing labor costs for retailers [42]. Beyond retail, CV has also bolstered security and monitoring systems through enhanced facial recognition, anomaly detection, and behavior analysis in surveillance applications [42].

However, the focus of this paper lies on the medical applications of CV. In healthcare, CV has transformed healthcare by enhancing diagnostic accuracy, treatment planning, patient monitoring, and biomedical research. Automated medical image analysis using CV significantly improves disease diagnosis [43].

In oncology, CV algorithms identify cancerous cells in histopathological images, enabling early detection [43]. Radiology benefits from CV by facilitating the interpretation of

complex medical images such as MRIs and CT scans, aiding in the detection of tumors and fractures [43]. Surgical precision can be enhanced through real-time guidance and preoperative planning, highlighting critical anatomical structures [44]. In organ and tissue segmentation, CV provides high-precision delineation essential for radiation therapy and disease monitoring [44]. Patient monitoring systems utilize CV to detect falls and track vitals, improving clinical care and in biomedical research, CV aids in cell tracking and quantitative analysis, accelerating discoveries [43].

While CV offers significant benefits in healthcare and other domains, integrating it into practice presents challenges such as requiring large datasets, high computational resources, and ensuring interpretability and robustness. The next section will explore these issues in greater depth.

### 1.3.3. Challenges and Future Directions

Despite the remarkable advancements in CV facilitated by DL, several challenges persist:

- **Data Dependency**: High-quality, labeled datasets are essential for training CV models. However, acquiring such datasets is difficult due to privacy issues, the necessity of expert annotations, and inconsistencies in data quality and standards among institutions. Furthermore, rare medical conditions suffer from a lack of adequate training data, complicating model development [45].
- **Computational Demand**: The reliance on substantial computational resources, such as high-performance GPUs and TPUs, for developing sophisticated CV models not only limits their accessibility in resource-constrained environments but also raises concerns about the sustainability and cost-effectiveness of deploying these models widely [22].
- **Interpretability**: Due to the black-box nature, the lack of interpretability inherent in DL models presents significant challenges in understanding and explaining their decision-making processes, particularly in critical domains such as healthcare [46]. This topic will be further elaborated upon in the following section.
- **Robustness and Generalization**: Ensuring models generalize well to diverse, unseen data and are robust against adversarial attacks is a significant challenge.

Variations in imaging techniques and patient populations can affect the performance of models, necessitating robust algorithm development [47].

## 1.4. Explainable Artificial Intelligence

Explainability in Artificial Intelligence has emerged as a critical domain of research and application, addressing the transparency and interpretability of complex AI models. With the rise of AI systems in critical areas such as healthcare, the ability to understand and trust these models has become not just an academic concern but a societal imperative. Understanding the importance of this field requires a comprehensive exploration of what explainability means.

In a broader context, explainablity refers to the degree to which an observer can comprehend the underlying reasons for a decision made by an AI system. Unlike traditional software, where the decision-making logic is often explicitly coded and thus transparent, AI systems, particularly those employing DL techniques, operate as "black boxes" [48]. This chapter explores the multifaceted dimensions of explainability, including the distinct terminology, methods and challenges. Therefore, it is necessary to distinguish between the various Explainable AI (XAI)-principles.

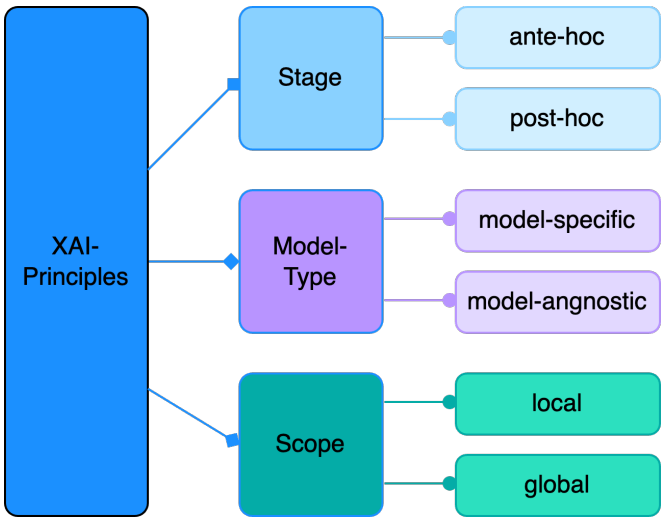


Figure 1.5.: Different XAI Principles; Source: based on [49, 50]

In the context of XAI, "ante-hoc" and "post-hoc" approaches refer to the stage in which explainability mechanisms are applied relative to the development and deployment of the

AI model. Ante-hoc explainability involves applying methods and techniques that are built into the AI model from the very beginning of the development process, ensuring that the model is inherently interpretable and transparent as it is being constructed [51]. Contrarily, post-hoc techniques are employed to provide insights and explanations after a model has been trained, which is primarily utilized in the context of black box models but also applicable for white box models [52]. Moreover, the model-type describes the dependency of explanation techniques on the internal structure of AI models and can be categorized into two main types, "model-specific" and "model-agnostic" methods. Thirdly, it is important to specify the scope of the interpretation, whether it is global or local. The global interpretation provides a high-level overview of the model's general effectiveness, while local interpretation explain the unique factors influencing one individual prediction [52].

### 1.4.1. Terminology: Interpretability and Explainability

Within the field of elucidating the decision-making processes of AI systems, the terms *Interpretability* and *Explainability* are often used interchangeably, but they do not uniformly convey identical meanings [53].

*Interpretability* pertains to models that can be intuitively understood by human users, providing immediate insight into the mapping of inputs to outputs and offering inherent transparency through a straightforward structure. Linear regression models, for example, are linear functions with coefficients, and the magnitude of these coefficients directly indicates the extent to which each feature influenced the decision [52]. The *Interpretability* of models decreases as their complexity increases, which can be attributed to factors such as correlated features or the employment of non-linear functions with millions of parameters to fit the data. As a result, it becomes progressively more difficult for humans to understand the decision-making process. When a model's complexity exceeds a certain threshold (although there is no clear static boundary), it is no longer considered to be *interpretable*, and the process of clarifying its internal mechanics and decision-making patterns is referred to as *Explainability* [53, 54].

*Explainability* encompasses an array of techniques and methods employed to clarify the intricate inner workings of complex black and white box AI models that are considered too elaborate for straightforward human understanding [55]. "Black box" models, such

as DNNs, are highly complex and opaque, making it challenging to understand how they map inputs to outputs. Conversely, "white box" models, like decision trees or rule-based systems, typically have a more transparent structure. However, they can still become difficult to interpret as they increase in complexity or size [56].

In summary, while *Interpretability* focuses on the model's inherent clarity (making it passive characteristic of a model), *Explainability* emphasizes the communication of the model's reasoning to users, making it a more interactive and user-centered concept.

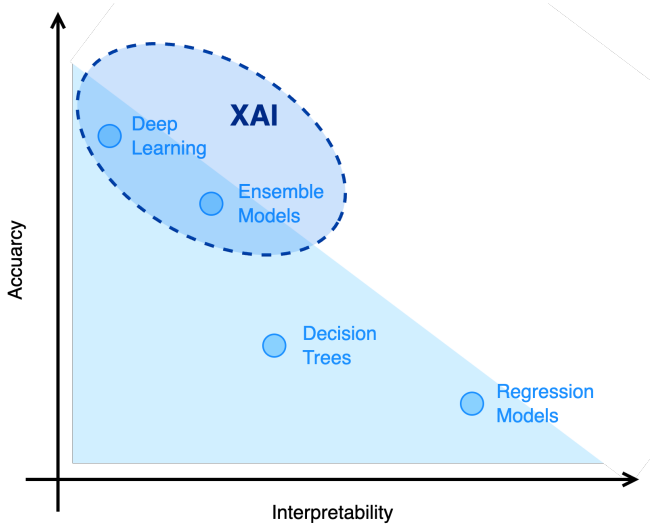


Figure 1.6.: Correlation between accuracy and interpretability of AI models and the models that are within the domain of explainability/XAI; Source: [52]

In the realm of explainability, various post-hoc techniques are employed. These methods can be broadly categorized into two primary methodological paradigms: (1) model-specific and (2) model-agnostic methods.

Model-agnostic methods, such as Local Interpretable Model-Agnostic Explanations (LIME) and Shapley Additive Explanations (SHAP), treat the model as a black box and aim to explain its decisions by perturbing the input features and observing the corresponding changes in the output [57]. This can be done from both local and global perspectives. In contrast to model-specific methods, model-agnostic approaches can be applied to any ML model, irrespective of its underlying architecture. Model-specific methods, on the other hand, are tailored or applicable to specific model architectures and exploit their internal structure to generate explanations. Examples include layer-wise

relevance propagation (LRP) for neural networks and decision tree visualizations for tree-based models [58] which will not be further addressed in this study.

## 1.4.2. Post-Hoc Explainability Methods

As mentioned in the past section post-hoc methods generate explanations for complex black-box models without altering their internal structures. Post-hoc explainability techniques, such as feature attribution methods or saliency maps, can be further categorized as model-specific or model-agnostic, depending on whether they are tied to a specific architecture or can be applied across different models. The following sections provide a detailed examination of various explainability techniques, forming the essential foundation for the subsequent comparison of post-hoc and intrinsically explainable approaches. This knowledge is crucial for assessing their respective strengths and limitations in the context of medical imaging and for determining which approach is better suited to the demands of this study.

### 1.4.2.1. Feature Attribution Methods

Feature attribution methods aim to quantify the contribution of each input feature to a model's prediction, by assign a measure of importance to each feature [59]. Given a trained model $f : X \to Y$ where $X$ represents the input feature space and $Y$ the output predictions, these methods estimate the influence of each feature $x_i \in X$ on the model's decision $f(X)$. Below, three widely-used feature attribution methods are examined: Local Interpretable Model-Agnostic Explanations (LIME), SHAP (SHapley Additive exPlanations), and Integrated Gradients (IG).

**Local Interpretable Model-Agnostic Explanations (LIME)** is a prominent model-agnostic designed to provide interpretable explanations for ML models by approximating them locally with simpler, interpretable models. The core idea behind LIME is to create a surrogate model that is easier to interpret while still being faithful to the predictions of the complex model in the vicinity of a specific instance [60].

The process begins by selecting an instance for which an explanation is desired. LIME generates a dataset of samples around this instance by making small modifications to its

features. For each sample, the complex model is queried to obtain predictions. These predictions are then used to train a simple interpretable model, such as a linear regression or decision tree, which approximates the behavior of the black box model locally [61].

One of LIME's strengths is its ability to provide insights into individual predictions, allowing users to understand which features contributed most significantly to the model's decision for that specific instance. This local interpretability is particularly valuable in domains such as healthcare [61]. However, because LIME relies on locally trained surrogate models, it may produce inconsistent results across different runs and lacks global interpretability [62].

**Shapley Additive explanations (SHAP)** is another widely used model-agnostic method that leverages concepts from cooperative game theory to provide explanations for individual predictions made by ML models. The foundation of SHAP lies in Shapley values, which assign a value to each feature based on its contribution to the prediction outcome [58].

SHAP calculates the contribution of each feature by considering all possible combinations of features and their corresponding predictions. Specifically, it measures how much each feature contributes to the prediction when included versus when excluded from a coalition of features. This approach ensures that SHAP values are fair and consistent across different feature sets [63].

One of SHAP's key advantages is its ability to provide both local and global interpretability. For individual predictions, SHAP values highlight which features pushed the prediction higher or lower compared to a baseline (usually the expected value). On a broader scale, aggregating SHAP values across multiple predictions allows researchers and practitioners to gain insights into feature importance across the entire dataset [57, 64]. Furthermore, SHAP offers different implementations tailored for various types of models, including DL models and tree-based models like XGBoost or LightGBM [63]. However, computing exact Shapley values is computationally expensive, especially for high-dimensional datasets, requiring approximations such as Kernel SHAP or Tree SHAP [65]. Despite this computational burden, SHAP has become a widely used technique [66, 67, 68, 69].

#### 1.4.2.2. Saliency Map Methods

Saliency map methods provide visual explanations by highlighting the most influential regions of an input (e.g., pixels in an image) that contribute to a model's prediction [70]. These methods are particularly useful for deep learning models in image classification with CNNs, where they help identify which parts of an image drive the model's decision [71].

**Gradient-weighted Class Activation Mapping (Grad-CAM)** Grad-CAM is a widely used model-specific technique that generates heatmaps to localize important regions in an image. The method computes the gradient of the model's output for a given class with respect to the feature maps of the final convolutional layer. These gradients are then pooled and used as weights to generate a class-discriminative heatmap, highlighting regions that most influenced the prediction [72]. Grad-CAM is particularly valuable in domains like medical imaging, where model interpretability is crucial. However, Grad-CAM struggles to accurately localize multiple occurrences of the same class in an image [73]. Furthermore, its effectiveness may be limited when applied to non-CNN architectures, as it inherently assumes the presence of convolutional layers. To leverage Grad-CAM with models lacking these layers, modifications may be necessary.

**Guided Backpropagation** Guided Backpropagation is a model-specific saliency method that modifies standard backpropagation. Unlike conventional backpropagation, which propagates all gradients, Guided Backpropagation applies a modified ReLU function that blocks negative gradients during both the forward and backward passes [74, 75]. This ensures that only positively contributing features influence the saliency map, highlighting relevant structures while suppressing noise. This selective approach is particularly effective for deep learning-based computer vision models [76]. Guided Backpropagation is often combined with other techniques, such as Grad-CAM, to generate sharper and more interpretable explanations [77]. However, its reliance on ReLU activations limits its generalization to architectures using alternative activation functions.

### 1.4.3. Intrinsic Explainability Approaches

In contrast to post-hoc explanation methods, which explain the behavior of existing models [52], intrinsically interpretable models are designed to provide transparency and interpretability a priori. Such models do not require supplementary explainability techniques, as their decision-making processes and computations are inherently understandable and directly accessible [78]. The following sections present an overview of various inherently explainable architectures and approaches, once again with the aim of establishing a solid foundation for the subsequent assessment of their suitability.

#### 1.4.3.1. Transparent Model Architectures

Some models are inherently interpretable due to their structure:

- Decision Trees and Rule-Based Models explicitly outline decision processes in an easily understandable format [79].
- Linear and Logistic Regression offer simple, interpretable relationships between input features and predictions [80].
- Generalized Additive Models (GAMs) extend linear models by allowing flexible, interpretable transformations of features while maintaining additive effects[81].

While these inherently interpretable models offer transparency, they often struggle to match the predictive performance of deep learning approaches on complex, high-dimensional tasks. As a result, they are less suitable for medical deep learning tasks.

#### 1.4.3.2. Feature Importance and Contribution

Some models, despite their relative complexity, incorporate mechanisms that enhance intrinsic interpretability:

- Attention Mechanisms in Neural Networks: In models like attention-based transformers, attention scores highlight which input features are most relevant for a given prediction.[82]

- Monotonic Constraints in Gradient Boosting Models: Enforcing monotonicity ensures that an increase in a particular feature value will always lead to a specific directional change in predictions, making the model behavior more predictable and interpretable [83].
- Sparse Models: Models that enforce sparsity (e.g., Lasso regression) select only the most relevant features [84], improving human interpretability.

#### 1.4.3.3. Concept-Based Learning

Concept-based learning enhances model interpretability by incorporating human-understandable concepts into the reasoning process [85]. Unlike traditional feature attribution methods, these techniques analyze how abstract, high-level concepts influence model decisions, making explanations more aligned with human intuition.

**Testing with Concept Activation Vectors (TCAV)** TCAV quantifies the influence of predefined human concepts (e.g., 'tumor presence') on model predictions by analyzing how changes along the corresponding Concept Activation Vector (CAV) in the model's internal feature space affect the output. This enables users to assess whether a model's decisions are influenced by meaningful concepts rather than uninterpretable patterns.[86]

**Disentangled Representation Learning** Disentangled representation learning is a technique that supports concept-based learning and helps models break down complex data into its individual components [87]. This process separates latent features into distinct, independent parts, making it easier to understand and analyze the data. By ensuring that each component corresponds to a real-world concept (e.g., shape, color, or texture in an image), these methods enhance transparency and facilitate domain-specific analysis[88]. For instance, in medical imaging, disentangled representations might separate features like bone density and tissue health, allowing for clearer insights into how models make diagnoses.

Concept-based learning can also be applied to black-box models, although it is less commonly used for post-hoc methods.

#### 1.4.3.4. Example-Based Explanations

In contrast to approximation approaches, where a new data point is sampled and explained (e.g., LIME), example-based methods interpret predictions by identifying similar cases from the training data. Instead of attributing importance to specific features, these methods justify decisions by referencing past instances, making them particularly intuitive for human understanding. [89].

**Counterfactual Explanations** Counterfactual explanations identify the smallest modifications needed in an input to change the model's prediction [89]. For instance, in a medical diagnosis model, a counterfactual explanation might show how a slight change in a patient's test results would lead to a different diagnosis, offering actionable insights.

**Prototype-Based Explanations** Prototype-based methods compare new inputs to representative examples (prototypes) learned by the model [90]. By showing how an input resembles or differs from these prototypes, this approach provides a structured, case-based justification for decisions. Prototype-based explanations can be viewed from two perspectives: as post-hoc explanations by comparing new inputs to learned prototypes after the model has been trained and as inherently explainable models.

### 1.4.4. Challenges

Despite the advancements in explainabilty methods like LIME and SHAP, several challenges persist in the field of XAI. One major challenge is the trade-off between interpretability and accuracy. Simplifying a model to enhance its interpretability often comes at the cost of its predictive performance [91]. Moreover, model-agnostic methods can be computationally expensive, particularly when dealing with large datasets or complex models, which can limit their practicality for real-time applications [92].

Another challenge is that both LIME and SHAP assume independence among features when generating explanations. In practice, features may be correlated, and this correlation can distort interpretations, leading to misleading conclusions about feature importance if not appropriately accounted for [93, 94].

Furthermore, while local interpretability helps in understanding specific predictions, it may not always align with the model's overall behavior. An explanation for one instance might not represent the model's general performance, leading to potential misunderstandings about how the model operates on a broader scale. Insights from individual predictions might not indicate the model's behavior across different datasets, potentially presenting an incomplete or misleading picture [64].

Beyond the formerly outlined difficulties, there are complementary aspects to consider that largely associated with model-agnostic explainability methodologies: There is no universal metric to assess the quality of explanations, making it difficult to objectively compare different methods. Lastly, the explanations provided by XAI methods need to be not only accurate but also understandable and actionable for end-users, which necessitates a careful balance between technical detail and simplicity [95].

In summary, while model-agnostic and model-specific interpretability methods provide valuable tools for understanding and explaining ML models, several challenges remain that require ongoing research and refinement to ensure they are both effective and practical in diverse applications.

# 2. Different Explainability Methods

As deep learning models continue to advance, their increasing complexity and opacity present significant challenges in the field of medical diagnostics [96]. The necessity for explainability in AI-driven decision-making has prompted the development of various methods aimed at providing insights into the reasoning behind model predictions. In the foundational chapter 1.4, two primary categories of explainability methods were introduced: post-hoc approaches, which generate explanations after model training, and intrinsically explainable models, which are designed to offer transparency by design. Each of these paradigms comes with inherent trade-offs between interpretability, performance, and computational efficiency, particularly relevant in the domain of medical imaging, where both accuracy and explainability are critical. Building on the comparison of these two paradigms, this chapter introduces a hybrid approach that combines the strengths of both. The following sections explore the structure, rationale, and specific techniques considered for this explainability layer, ultimately motivating the selection of Self-Correcting Attention as the most suitable method for this work that integrate explainability directly into the learning process while preserving the predictive capabilities of deep neural networks.

## 2.1. Comparison and Suitability Assessment of Post-hoc and Intrinsically Explainable Methods

Having examined both post-hoc methods and inherently explainable architectures, this section will now compare these two approaches and assess their respective strengths and weaknesses, focusing on computational efficiency, performance trade-offs, as well as practical implications in the context of medical imaging. This comparative analysis will provide insights into the suitability of each approach for various applications in the field.

Both post-hoc and inherently interpretable models have advantages and trade-offs in terms of computational efficiency. Post-hoc methods enable deep learning models to

maintain high predictive performance while providing explanations after training, but they come at a significant computational cost. Techniques such as SHAP, LIME, and Grad-CAM require additional inferences, gradient calculations, or perturbations, making them computationally demanding for real-time or large-scale medical imaging applications.

Inherently interpretable models, on the other hand, eliminate the need for additional processing during inference, improving efficiency. However, their training costs can be higher, and they may not achieve the same level of performance as deep learning black-box models. Attention-based architectures and prototype-based learning provide a balance between efficiency and interpretability but may still impose memory or processing constraints.

| **Aspect** | **Post-Hoc Methods** | **Inherently Explainable Models** |
|---|---|---|
| Computational Expense | High (requires additional computations per prediction) | Low to moderate (interpretable by design) |
| Training Cost | Typically low (can be added after training) | Can be high, depending on model complexity |
| Inference Speed | Slower due to multiple inferences and backpropagation | Faster due to direct interpretability |
| Scalability to Large Datasets | Limited due to high inference time | More scalable with efficient architectures |
| Interpretability | Requires additional processing | Built-in transparency |
| Suitability for Medical Imaging | Well-suited for CNN-based models but computationally demanding | Works well for efficiency-focused applications |

Table 2.1.: Comparison post-hoc methods vs. inherently explainable models

Overall, building a fully inherently explainable model from scratch for medical imaging is deemed impractical for this study for several reasons.

First, developing and training a new inherently interpretable model from the ground up to achieve a reasonable performance would necessitate extensive time and computational resources, surpassing the boundaries set for this research. Moreover, in the field of medical imaging, precision and accuracy are paramount, as misclassifications can result in severe

consequences. Lastly, existing models have been extensively validated and optimized, providing a robust foundation for further research and clinical applications.

The inherent trade-off between explainability and performance makes it difficult to justify the use of fully transparent models for complex image classification and segmentation tasks. While inherently interpretable models provide efficiency and transparency, they often fail to match the performance of deep learning-based black-box models, which excel in recognizing intricate patterns in high-dimensional data. In contrast, post-hoc methods enable high-performing deep learning models to provide explanations, they come with computational overhead and potential inconsistencies.For medical imaging tasks requiring high predictive accuracy, post-hoc methods like Grad-CAM and SHAP remain indispensable despite their computational demands. However, inherently interpretable models like GAMs and prototype-based learning offer promising alternatives for applications where efficiency is paramount.

In conclusion, a post-hoc approach might be the more practical choice for a medical imaging application, as it allows models to maintain high accuracy while still offering interpretability. Both post-hoc explanations and inherently explainable models have their limitations, making neither approach entirely ideal. In light of these limitations, this study aims to introduce a new approach that combines the core concepts of both approaches. This proposed approach will be elaborated upon in the following section.

## 2.2. A Hybrid Approach: Combining black-box model with inherently explainable features by creating an "Explainability Layer"

As outlined in the previous section, post-hoc explainability methods provide insights into complex models but often introduce inconsistencies and additional computational overhead. In contrast, inherently explainable models ensure transparency but may lack the predictive power required for high-dimensional tasks like medical imaging. Given these trade-offs, this study proposes an approach that has the potential to bridge the gap between explainability and performance by introducing an "Explainability Layer" which is a dedicated architectural component that enables a deep learning model to have high

accuracy while providing structured, interpretable reasoning. Instead of choosing between black-box models and inherently explainable architectures, this approach combines both in a hybrid concept:

1. **A Black-Box Model for High Performance:** A state-of-the-art deep CNN (e.g., ResNet, DenseNet) is used for feature extraction, leveraging its strong representation learning capabilities. Consequently, the lower layers (e.g., the initial 80% of the network) will be kept frozen to maintain the extraction of fundamental image features, including edges, textures, and patterns. Subsequently, the higher layers will be fine-tuned to adapt the model to the specific classification task at hand.
2. **An Explainability Layer for Transparent Decision-Making:** An additional module, designed to enforce interpretability, is inserted into the model pipeline. Instead of using a standard fully connected classifier, this layer(s) will refine predictions by aligning them with human-interpretable mechanisms, such as attention-based learning, prototype-based learning, or concept-based learning, which will be discussed in the following section.
3. **Self-Correction Mechanism:** Unlike standard explainability methods, which merely highlight features, the approach suggested in this study introduces a self-correcting mechanism that iteratively adjusts its focus, ensuring explanations are meaningful and decisions remain robust.

### 2.2.1. Potential Explainability Techniques Integrated During Fine-Tuning

This paper identifies three primary explainability-enhancing techniques that can be integrated during fine-tuning, which are subsequently explored regarding their feasibility, interpretability, and suitability for the NIH Chest X-ray dataset.

#### 2.2.1.1. Prototype-Based Learning (ProtoPNet)

This method replaces traditional black-box feature embeddings with a structured, case-based reasoning approach that enhances transparency and trustworthiness in deep learning models [97]. Integrating ProtoPNet within a CNN-based architecture for NIH Chest

X-ray dataset classification involves: (1) selecting and evaluating prototypes manually or via unsupervised clustering due to absent region-level annotations; (2) training the model to extract medically relevant prototypes representing key pathological features; (3) classifying new images during inference via similarity metrics in the latent space; and (4) visualizing relevant prototype patches for interpretable predictions.

The ProtoPNet could serve as an explainability layer that sits on top of the CNN's feature extraction pipeline. The CNN still learns hierarchical feature representations, but instead of relying on opaque embeddings, it passes extracted features to the prototype layer, which forces classification decisions to be aligned with learned visual patterns. This hybrid approach ensures that the base CNN retains its strong predictive capabilities while enforcing interpretability by grounding decisions in stored prototype examples. The final output thus would consist of both a prediction and a reference to the closest matching prototype, providing an additional layer of transparency and interpretability to the model's decision-making process. [98]

A Prototype Learning Approach ensures model decisions are justified through similar past X-ray cases, making it particularly suitable for medical imaging where interpretability is crucial. Its ability to provide visual, case-based explanations is beneficial for detecting and differentiating thoracic pathologies. Moreover, it facilitates clinician oversight, enabling radiologists to validate the model's decision-making process [99].

A primary challenge is ensuring that learned prototypes correspond to diagnostically meaningful regions [100] rather than spurious artifacts such as ribs or background noise. To mitigate this, prototype regularization can be employed to enforce attention on pathological structures. Additionally, attention heatmaps (e.g., Grad-CAM) can be used to verify prototype relevance [72], and manual expert validation may be necessary when dataset annotations are insufficient. Given the NIH dataset's lack of localized lesion annotations, an additional challenge is identifying reliable prototype samples. This may require semi-supervised learning or self-supervised feature extraction techniques to guide prototype selection. Furthermore, class imbalances in the NIH dataset could lead to biased prototype selection, necessitating data augmentation and resampling techniques to ensure diverse and clinically relevant prototype representations [101].

#### 2.2.1.2. Concept Alignment via Testing with Concept Activation Vectors (TCAV)

Testing with Concept Activation Vectors (TCAV) provide an alternative explainability technique by ensuring that model decisions align with high-level human-interpretable medical concepts rather than abstract feature embeddings. This approach assesses the influence of predefined medical concepts, such as "lung opacity" or "pleural effusion" on model predictions and incorporates these concepts into training. [102] To integrate TCAV into a CNN-based model trained on the NIH Chest X-ray dataset, the following steps are necessary: (1) Defining clinically relevant concepts where domain experts (radiologists) or medical literature must guide the selection of key imaging features, such as "lung opacity", "pleural thickening", or "air trapping". (2) Collecting concept reference images, since the NIH dataset does not provide explicit concept labels. A preprocessing step is required to obtain labeled concept examples. (3) Training the Concept Activation Vector (CAV) model to map CNN feature representations to the predefined medical concepts, creating CAVs that serve as interpretable feature directions in the model's latent space. (4) Incorporating concept influence into classification by initially detecting key medical concepts, then refining final predictions using learned concept influence scores, rather than merely predicting disease categories (e.g., atelectasis or effusion).

TCAV functions as a post-hoc explainability layer that analyzes the CNN's latent representations after training. The CNN still operates as a black-box classifier, but its feature space is retrospectively examined through the lens of predefined concept vectors. [103] These CAVs act as interpretability probes, identifying how strongly specific medical features contribute to the final prediction. Unlike feature attribution methods such as SHAP or LIME, TCAV explicitly encodes human-defined medical knowledge into the explanation process, making it particularly useful in clinical applications where transparency is essential. The integration of TCAV ensures that the CNN's decision-making process can be audited and interpreted without modifying its underlying training procedure.

CAV is particularly useful for medical AI applications where human experts require explicit alignment between model decisions and clinical diagnostic reasoning. It ensures that critical imaging characteristics known to be relevant for specific conditions are explicitly recognized by the model rather than emerging as uninterpretable latent features.

This approach is beneficial for detecting and analyzing thoracic pathologies where interpretability is a key requirement [104].

A significant challenge in applying TCAV to the NIH dataset is the lack of explicit concept labels, requiring an additional preprocessing step to generate labeled examples for each medical concept [105]. Automated concept labeling using models like CheXpert can partially address this issue, but manual validation by experts is still required as CheXpert doesn't currently support image data [106]. Another challenge is concept selection as choosing concepts that are both clinically relevant and sufficiently distinct for meaningful feature attribution is required. Additionally, TCAV requires sufficient concept representation in the dataset, meaning multiple high-quality example images must be available for each concept [103]. Although, data is limited, semi-supervised learning techniques can be employed to expand concept coverage.

The concept of an explainability layer is approached somewhat differently with TCAV. TCAV enhances explainability by adding a structured, concept-driven interpretation layer without fundamentally changing the CNN's architecture. This approach is addressed because of its conceptual similarity to the notion of an "explainability layer". However, it falls short of the core idea of employing an intrinsically explainable model.

#### 2.2.1.3. Attention Mechanisms for Feature Attribution

Attention mechanisms enhance explainability by directing the model's focus to clinically relevant regions within an input image rather than uniformly processing all pixels. Unlike post-hoc methods such as Grad-CAM, attention mechanisms are integrated directly into the model and influence decision-making during both training and inference.

To integrate Self-Attention mechanisms into a CNN-based model for classifying the NIH Chest X-ray dataset, the following steps are necessary: (1) Adding a Self-Attention layer by inserting a Self-Attention module after the final convolutional layers, thereby allowing the model to assign importance weights to different spatial regions of the feature map [107]. (2) Refining features via attention weights, where the Self-Attention mechanism modifies feature representations by emphasizing high-weighted regions corresponding to suspected pathology while suppressing irrelevant background details. (3) Conducting end-to-end training with a CNN backbone, ensuring that the attention-enhanced CNN

is trained in a manner that allows attention weights to dynamically adapt, optimizing classification performance.

By incorporating Self-Attention directly into the CNN's processing pipeline, the model retains the advantages of deep feature learning while integrating a structured mechanism that highlights relevant regions. Unlike post-hoc techniques such as Grad-CAM, which require additional computation after inference, Self-Attention mechanisms naturally produce interpretable attention maps as a byproduct of the model's prediction process. This makes them inherently explainable in the sense that decision-critical regions are explicitly weighted during classification rather than being analyzed after the fact [108].

Self-Attention Mechanisms could be particularly effective in medical imaging tasks where pathology is often localized and distinct, such as detecting opacities, pleural effusions, or pulmonary nodules. The dynamic adjustment of attention weights based on image content enables adaptability across diverse thoracic diseases. Moreover, Self-Attention's compatibility with any CNN backbone offers deployment flexibility without significant architectural changes.

Self-Attention requires addressing several challenges include ensuring medically meaningful Attention, computational overhead, and evaluating Attention maps without region-based labels. Validation using expert-annotated heatmaps or Grad-CAM, efficient implementations with channel-wise or spatial attention, and weakly supervised learning can help address these issues. Regularization techniques and hyperparameter tuning are also crucial for meaningful attention distributions and balancing performance with interpretability.

### 2.2.2. Evaluating Suitability: The Rationale for Selecting Self-Correcting Attention Over Concept- and Prototype-Based Explainability Methods

The selection of Self-Correcting Attention over Concept-Based Learning (TCAV) and Prototype-Based Learning (ProtoPNet) was primarily driven by tby considerations of **feasibility**, **dataset compatibility**, and **novelty**, ensuring that the approach could be implemented effectively within the available time frame.

From an implementation perspective, Self-Correcting Attention is more feasible than TCAV, which requires external concept labels, and ProtoPNet, which demands prototype validation by domain experts (as discussed in the previous subsection). Since no additional annotations are required for Self-Correcting Attention, it allows for a smooth integration into a standard CNN-based classification model while maintaining interpretability.

Regarding compatibility with the NIH Chest X-ray dataset, Concept-Based Learning relies on predefined concept labels (e.g., lung opacity, pleural effusion, infiltration), which are not explicitly available in the dataset. This makes TCAV difficult to implement without additional expert annotations. Similarly, ProtoPNet requires case-based prototypical representations, which are not inherently provided in the dataset, making it less suitable. Self-Correcting Attention, on the other hand, leverages attention maps derived from the original dataset, requiring no additional preprocessing or expert-defined prototypes. Concept-Based Learning [109, 86], Prototype-Based Learning [99, 110], and Self-Correcting Attention have received limited exploration but are gaining interest in medical AI, presenting promising research directions.

Finally, from an interpretability perspective, Self-Correcting Attention provides a direct, human-understandable explanation through heatmaps that highlight relevant lung regions in chest X-rays. While Concept-Based Learning allows for concept alignment, it is less intuitive when concepts are poorly defined. Prototype-Based Learning, although interpretable, requires human verification of selected prototypes, making it harder to validate without expert involvement. In contrast, Self-Correcting Attention automatically learns and adjusts its attention maps, ensuring that explanations remain consistent across training epochs. Considering these factors, **Self-Correcting Attention was selected** as the most suitable approach due to its appropriate implementation complexity, strong dataset compatibility, relatively unexplored nature, and direct interpretability.

| **Criteria** | **Self-Correcting Attention** | **Concept-Based Learning (TCAV)** | **Prototype-Based Learning (ProtoPNet)** |
|---|---|---|---|
| Feasibility | High | Low | Medium |
| NIH Chest X-ray Suitability | Works without modifications | Needs external concept annotations | Needs representative prototypes, which are not predefined |
| Novelty for medical imaging | High | High | High |
| Computational Cost | Moderate | High | High |
| Interpretability | Heatmaps provide direct visual explanations | Harder to define if concepts are unclear | Prototypes require domain expert validation |
| Stability of Explanations | Ensures stable attention maps across training | Model can still focus on irrelevant concepts | Prototype representations may not always be clinically meaningful |
| External Data Required | No | Yes | Yes |

Table 2.2.: Comparison of Suitability Self-Correcting Attention vs. Concept-Based Learning (TCAV) vs. Prototype-Based Learning (ProtoPNet)

## 2.3. Implementation of the Explainability Layer

### 2.3.1. Methodology for Integrating Self-Correcting Attention

**Self-Correcting Attention as an Explainability Layer** Unlike post-hoc attention visualization, self-correcting attention mechanisms inherently influence the decision-making process of the CNN. The attention module has to continuously refines its focus during training, reducing reliance on irrelevant features and improving generalization to unseen data. This hybrid approach should ensure that the CNN model remains powerful as a black-box classifier while embedding structured, interpretable attention maps as

part of its architecture. Additionally, by leveraging attention heatmaps as part of the model output, clinicians can visually confirm whether the model is focusing on medically appropriate regions, further improving trust and usability.

**Self-Correction through Historical Attention Tracking** In conventional CNN models with attention layers, attention heatmaps are generated per image per epoch, but these are treated as independent outputs rather than iterative refinements. Our proposed self-correcting approach stores attention heatmaps from previous training epochs and compares them against the current epoch's attention distribution. This enables the model to assess whether its focus is shifting meaningfully or if it is displaying inconsistent or unstable behavior.

Let $A_t(x)$ represent the attention map for an input image $x$ at the epoch $t$, and let $\hat{A}(x)$ be the moving average of attention maps over the past $k$ epochs:

$$\hat{A}(x) = \frac{1}{k} \sum_{i=t-k}^{t} A_i(x) \tag{2.1}$$

If $A_t(x)$ deviates significantly from $\hat{A}(x)$ it suggests the model is inconsistently modifying its focus, leading to unstable and unreliable explanations.

**Self-Correction Loss for Attention Stability** To enforce stable learning, a Self-Correction Loss that penalizes drastic changes in attention distributions will be introduced:

$$L_{self-correct} = \|A_t(x) - \hat{A}(x)\|^2 \tag{2.2}$$

This term is integrated into the overall training objective:

$$L_{total} = L_{clasification} + \lambda_{attention} + \alpha L_{self-correct} \tag{2.3}$$

where:
$L_{classification}$ is the standard cross-entropy loss for disease classification.
$L_{attention}$ ensures that attention maps align with clinically relevant areas.
$L_{self-correct}$ stabilizes attention focus over training epochs.

The weighting factor $\alpha$ is adjusted dynamically, increasing for images where fluctuations are high, thereby progressively encouraging convergence towards stable and interpretable attention maps

#### Expected Benefits of Self-Correcting Attention

1. More Reliable Model Interpretability: By enforcing temporal consistency in attention, we ensure that the same image produces stable explanations over different epochs.
2. Reduced Sensitivity to Spurious Features: Since erratic attention fluctuations are penalized, the model avoids focusing on irrelevant image regions, such as artifacts or noise.
3. Improved Generalization: Regularizing attention behavior prevents overfitting to local dataset peculiarities and encourages generalizable feature learning.
4. Stronger Alignment with Clinical Intuition: Medical professionals expect consistency in AI-assisted diagnosis. A model whose focus shifts unpredictably is difficult to trust in real-world applications.

By integrating Self-Correcting Attention into the Explainability Layer, the robustness of attention mechanisms is enhanced, allowing deep learning models to produce clinically meaningful, stable, and verifiable explanations for medical imaging tasks. This represents a step beyond conventional attention-based methods, making AI decisions more transparent and reliable for disease classification in chest X-ray analysis.

### 2.3.2. Implementation of the "Explainability Layer"

# Literature Listing


[1] Kitaguchi, D. et al. *Artificial intelligence-based computer vision in surgery: Recent advances and future perspectives*. Vol. 6. 1. Wiley, 2021, pp. 29–36. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8786689/ (visited on 10/12/2024).

[2] Alzubaidi, L. et al. *Review of deep learning: concepts, CNN architectures, challenges, applications, future directions*. Vol. 8. 1. Springer Science+Business Media, 2021. URL: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00444-8#citeas (visited on 10/12/2024).

[3] Janiesch, C./ Zschech, P./ Heinrich, K. *Machine learning and deep learning*. Vol. 31. 3. Springer Science+Business Media, 2021, pp. 685–695. URL: https://link.springer.com/article/10.1007/s12525-021-00475-2 (visited on 10/10/2024).

[4] Mitchell, T. M. *Machine learning*. McGraw-hill New York, 1997. URL: https://www.cs.cmu.edu/~tom/files/MachineLearningTomMitchell.pdf (visited on 10/10/2024).

[5] Alzubaidi, L. et al. *Review of deep learning: concepts, CNN architectures, challenges, applications, future directions*. Vol. 8. 1. Springer Science+Business Media, 2021. URL: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00444-8#citeas (visited on 10/11/2024).

[6] Dhar, V. *The Paradigm Shifts in Artificial Intelligence*. 2023. arXiv: 2308.02558 [cs.AI]. URL: https://arxiv.org/abs/2308.02558 (visited on 10/12/2024).

[7] Sarker, I. H. *Machine Learning: Algorithms, Real-World Applications and Research Directions*. Vol. 2. 3. Springer Science and Business Media LLC, 2021. URL: https://link.springer.com/article/10.1007/s42979-021-00592-x (visited on 12/10/2024).

[8] Sarker, I. H. *Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions*. Vol. 2. 6. Springer Nature, 2021. URL: https://link.springer.com/article/10.1007/s42979-021-00815-1 (visited on 10/12/2024).

[9] Mehlig, B. *Machine Learning with Neural Networks: An Introduction for Scientists and Engineers*. Cambridge University Press, 2021. URL: http://dx.doi.org/10.1017/9781108860604 (visited on 10/26/2024).

[10] Ishthaq, A. K. "A Study on Neural Network Architectures". In: *Core.ac.uk* (2016). URL: https://core.ac.uk/outputs/234645196/?utm_source=pdf&utm_medium=banner&utm_campaign=pdf-decoration-v1 (visited on 10/25/2024).

[11] McCulloch, W. S./ Pitts, W. *A logical calculus of the ideas immanent in nervous activity*. Vol. 5. 4. Springer Science and Business Media LLC, 1943, pp. 115–133. URL: https://link.springer.com/article/10.1007/BF02478259#citeas (visited on 10/11/2024).

[12] LeCun, Y. et al. *Backpropagation Applied to Handwritten Zip Code Recognition*. Vol. 1. 4. 1989, pp. 541–551. URL: https://www.semanticscholar.org/paper/Backpropagation-Applied-to-Handwritten-Zip-Code-LeCun-Boser/a8e8f3c8d4418c8d62e306538c9c1292635e9d27 (visited on 10/11/2024).

[13] Schmidhuber, J. "Deep learning in neural networks: An overview". In: *Neural Networks* 61 (2015), pp. 85–117. URL: http://dx.doi.org/10.1016/j.neunet.2014.09.003 (visited on 10/11/2024).

[14] Lepakshi, V. A. *Chapter 18 - Machine Learning and Deep Learning based AI Tools for Development of Diagnostic Tools*. Ed. by Parihar, A. et al. Academic Press, 2022, pp. 399–420. URL: https://www.sciencedirect.com/science/article/pii/B978032391172600011X (visited on 10/26/2024).

[15] Montesinos López, O. A./ Montesinos López, A./ Crossa, J. "Fundamentals of Artificial Neural Networks and Deep Learning". In: *Multivariate Statistical Machine Learning Methods for Genomic Prediction*. Springer International Publishing, 2022, pp. 379–425. URL: https://doi.org/10.1007/978-3-030-89010-0_10 (visited on 10/26/2024).

[16] Bergmann, D./ Stryker, C. *What is a loss function?* 2024. URL: https://www.ibm.com/think/topics/loss-function (visited on 10/26/2024).

[17] Kerkhof, M. et al. *No (good) loss no gain: systematic evaluation of loss functions in deep learning-based side-channel analysis*. 3. Springer Science+Business Media, 2023, pp. 311–324. URL: https://link.springer.com/article/10.1007/s13389-023-00320-6#citeas (visited on 10/26/2024).

[18] Bengio, Y./ Léonard, N./ Courville, A. *Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation*. 2013. URL: https://arxiv.org/abs/1308.3432 (visited on 10/26/2024).

[19] Abdulkadirov, R./ Lyakhov, P./ Nagornov, N. *Survey of Optimization Algorithms in Modern Neural Networks*. Vol. 11. 11. 2023. URL: https://www.mdpi.com/2227-7390/11/11/2466 (visited on 10/26/2024).

[20] Ruder, S. *An overview of gradient descent optimization algorithms*. 2017. arXiv: 1609.04747 [cs.LG]. URL: https://arxiv.org/abs/1609.04747 (visited on 10/26/2024).

[21] Yamashita, R. et al. *Convolutional neural networks: an overview and application in radiology*. Vol. 9. 4. Springer Nature, 2018, pp. 611–629. URL: https://insightsimaging.springeropen.com/articles/10.1007/s13244-018-0639-9#citeas (visited on 10/27/2024).

[22] Zhao, X. et al. "A review of convolutional neural networks in computer vision". In: *Artificial Intelligence Review* 57.4 (2024). URL: https://link.springer.com/article/10.1007/s10462-024-10721-6#citeas (visited on 10/27/2024).

[23] Raiaan, M. A. K. et al. *A systematic review of hyperparameter optimization techniques in Convolutional Neural Networks*. Vol. 11. 2024. URL: https://www.sciencedirect.com/science/article/pii/S2772662224000742 (visited on 10/27/2024).

[24] Gholamalinezhad, H./ Khosravi, H. *Pooling Methods in Deep Neural Networks, a Review*. 2020. arXiv: 2009.07485 [cs.CV]. URL: https://arxiv.org/abs/2009.07485 (visited on 10/27/2024).

[25] Scabini, L. F. S./ Bruno, O. M. *Structure and Performance of Fully Connected Neural Networks: Emerging Complex Network Properties*. 2021. arXiv: 2107.14062 [cs.LG]. URL: https://arxiv.org/abs/2107.14062 (visited on 10/27/2024).

[26] Xu, Q. et al. *Overfitting remedy by sparsifying regularization on fully-connected layers of CNNs*. Vol. 328. Chinese Conference on Computer Vision 2017, 2019, pp. 69–74. URL: https://www.sciencedirect.com/science/article/pii/S0925231218309524 (visited on 10/27/2024).

[27] Basyal, G. P./ Zeng, D./ Rimal, B. P. "Development of CNN Architectures using Transfer Learning Methods for Medical Image Classification". In: (2024). URL: https://arxiv.org/abs/2410.16711 (visited on 10/27/2024).

[28] Khan, A. et al. *A survey of the recent architectures of deep convolutional neural networks*. Vol. 53. 8. Springer Science+Business Media, 2020, pp. 5455–5516. URL: https://link.springer.com/article/10.1007/s10462-020-09825-6 (visited on 10/27/2024).

[29] Sarvamangala, D. R./ Kulkarni, R. V. "Convolutional neural networks in medical image understanding: a survey". In: *Evolutionary Intelligence* 15 (2021), pp. 1–22. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC7778711/ (visited on 10/27/2024).

[30] Mortazi, A./ Bagci, U. *Automatically Designing CNN Architectures for Medical Image Segmentation*. 2018. URL: https://arxiv.org/abs/1807.07663 (visited on 10/27/2024).

[31] Sandler, M. et al. " MobileNetV2: Inverted Residuals and Linear Bottlenecks ". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, 06/2018, pp. 4510–4520. URL: https://doi.ieeecomputersociety.org/10.1109/CVPR.2018.00474.

[32] Ronneberger, O./ Fischer, P./ Brox, T. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by Navab, N. et al. Cham: Springer International Publishing, 2015, pp. 234–241. (Visited on 10/27/2024).

[33] Sharma, P. et al. *Generative adversarial networks (GANs): Introduction, Taxonomy, Variants, Limitations, and Applications*. Springer Science+Business Media, 2024. URL: https://link.springer.com/article/10.1007/s11042-024-18767-y (visited on 10/26/2024).

[34] Stryker, C. *Recurrent Neural Network (RNN)*. 2021. URL: https://www.ibm.com/topics/recurrent-neural-networks (visited on 10/26/2024).

[35] Laad, M./ Maurya, R./ Saiyed, N. *Unveiling the Vision: A Comprehensive Review of Computer Vision in AI and ML*. 2024, pp. 1–6. URL: https://ieeexplore.ieee.org/document/10533631 (visited on 10/30/2024).

[36] Srilakshmi, V. et al. "Evolving Convolutional Neural Networks with Meta-Heuristics for Transfer Learning in Computer Vision". In: *Procedia Computer Science* 230 (2023). 3rd International Conference on Evolutionary Computing and Mobile Sustainable Networks (ICECMSN 2023), pp. 658–668. URL: https:

//www.sciencedirect.com/science/article/pii/S1877050923021269 (visited on 10/30/2024).

[37] Khan, A./ Laghari, A./ Awan, S. "Machine Learning in Computer Vision: A Review". In: *ICST Transactions on Scalable Information Systems* (2018). URL: https://publications.eai.eu/index.php/sis/article/view/2055 (visited on 10/30/2024).

[38] Sultana, F./ Sufian, A./ Dutta, P. "Advancements in Image Classification using Convolutional Neural Network". In: *2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*. IEEE, 2018, pp. 122–129. URL: http://dx.doi.org/10.1109/ICRCICN.2018.8718718 (visited on 10/30/2024).

[39] Redmon, J. et al. *You Only Look Once: Unified, Real-Time Object Detection*. 2016. URL: https://arxiv.org/abs/1506.02640 (visited on 10/30/2024).

[40] Liu, W. et al. "SSD: Single Shot MultiBox Detector". In: *Computer Vision – ECCV 2016*. Springer International Publishing, 2016, pp. 21–37. URL: http://dx.doi.org/10.1007/978-3-319-46448-0_2 (visited on 10/30/2024).

[41] Öztürk, O./ Sariturk, B./ Seker, D. "Comparison of Fully Convolutional Networks (FCN) and U-Net for Road Segmentation from High Resolution Imageries". In: *International Journal of Environment and Geoinformatics* 7 (09/2020), pp. 272–279. URL: https://dergipark.org.tr/tr/download/article-file/1105454 (visited on 10/30/2024).

[42] Sinha, M./ Shukla, R./ Rastogi, E. *COMPUTER VISION APPLICATIONS AND CHALLENGES: A REVIEW*. Vol. 3404. URL: https://www.irjmets.com/uploadedfiles/paper//issue_12_december_2023/47895/final/fin_irjmets1704077132.pdf (visited on 10/30/2024).

[43] Javaid, M. et al. "Computer vision to enhance healthcare domain: An overview of features, implementation, and opportunities". In: *Intelligent Pharmacy* (2024). URL: https://www.sciencedirect.com/science/article/pii/S2949866X24000662 (visited on 10/30/2024).

[44] Gao, J. et al. "Computer Vision in Healthcare Applications". In: *Journal of Healthcare Engineering* (2018), pp. 1–4. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC5857319/ (visited on 10/31/2024).

[45] Munappy, A. R. et al. “Data management for production quality deep learning models: Challenges and solutions”. In: *Journal of Systems and Software* 191 (2022). URL: https://www.sciencedirect.com/science/article/pii/S0164121222000905 (visited on 10/31/2024).

[46] Dhar, T. et al. *Challenges of Deep Learning in Medical Image Analysis—Improving Explainability and Trust*. Vol. 4. 1. 2023, pp. 68–75. (Visited on 10/31/2024).

[47] Sanaat, A. et al. “Robust-Deep: A Method for Increasing Brain Imaging Datasets to Improve Deep Learning Models’ Performance and Robustness”. In: *Journal of Digital Imaging* 35 (2022), pp. 469–481. URL: https://pubmed.ncbi.nlm.nih.gov/35137305/ (visited on 10/31/2024).

[48] Samek, W./ Wiegand, T./ Müller, K.-R. *EXPLAINABLE ARTIFICIAL INTELLIGENCE: UNDERSTANDING, VISUALIZING AND INTERPRETING DEEP LEARNING MODELS*. Fraunhofer Heinrich-Hertz-Institut. URL: https://iphome.hhi.de/samek/pdf/SamITU18b.pdf (visited on 11/01/2024).

[49] Angelov, P. et al. “Explainable artificial intelligence: an analytical review”. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* (07/2021). (Visited on 11/01/2024).

[50] Adadi, A./ Berrada, M. “Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)”. In: *IEEE Access* 6 (2018). (Visited on 11/01/2024).

[51] Retzlaff, C. O. et al. “Post-hoc vs ante-hoc explanations: xAI design guidelines for data scientists”. In: *Cognitive Systems Research* 86 (2024), p. 101243. URL: https://www.sciencedirect.com/science/article/pii/S1389041724000378.

[52] Madathil, A. P. et al. “Intrinsic and post-hoc XAI approaches for fingerprint identification and response prediction in smart manufacturing processes”. In: *Journal of Intelligent Manufacturing* (2024). URL: https://link.springer.com/article/10.1007/s10845-023-02266-2#citeas (visited on 11/01/2024).

[53] Arrieta, A. B. et al. *Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI*. 2019. URL: https://arxiv.org/abs/1910.10045 (visited on 11/01/2024).

[54] Ali, S. et al. “Explainable Artificial Intelligence (XAI): What we know and what is left to attain Trustworthy Artificial Intelligence”. In: *Information Fusion* 99 (2023), p. 101805. URL: https://www.sciencedirect.com/science/article/pii/S1566253523001148 (visited on 11/01/2024).

[55] IBM. *Explainable AI*. 2023. URL: https://www.ibm.com/topics/explainable-ai (visited on 11/01/2024).

[56] Buhrmester, V./ Münch, D./ Arens, M. *Analysis of Explainers of Black Box Deep Neural Networks for Computer Vision: A Survey*. 2019. URL: https://arxiv.org/abs/1911.12116 (visited on 11/01/2024).

[57] Salih, A. M. et al. “A Perspective on Explainable Artificial Intelligence Methods: SHAP and LIME”. In: *Advanced Intelligent Systems* (2024). URL: http://dx.doi.org/10.1002/aisy.202400304 (visited on 11/03/2024).

[58] Molnar, C. *Chapter 6 Model-Agnostic Methods | Interpretable Machine Learning*. 2022. URL: https://christophm.github.io/interpretable-ml-book/agnostic.html (visited on 11/01/2024).

[59] *Introduction to AI Explanations for AI Platform*. 2025. URL: https://cloud.google.com/ai-platform/prediction/docs/ai-explanations/overview.

[60] Zafar, M. R./ Khan, N. “Deterministic Local Interpretable Model-Agnostic Explanations for Stable Explainability”. In: *Machine Learning and Knowledge Extraction* (2021), pp. 525–541. URL: https://www.mdpi.com/2504-4990/3/3/27 (visited on 11/03/2024).

[61] Ribeiro, M. T./ Singh, S./ Guestrin, C. *"Why Should I Trust You?": Explaining the Predictions of Any Classifier*. 2016. URL: https://arxiv.org/abs/1602.04938 (visited on 11/03/2024).

[62] Stadler, A./ Müller, W. G./ Harman, R. *Green LIME: Improving AI Explainability through Design of Experiments*. 2025. arXiv: 2502.12753 [stat.ML]. URL: https://arxiv.org/abs/2502.12753.

[63] Wang, H. et al. “Feature selection strategies: a comparative analysis of SHAP-value and importance-based methods”. In: *Journal of Big Data* 11.1 (2024). URL: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-024-00905-w#citeas (visited on 11/03/2024).

[64] Bell, A. et al. *It's Just Not That Simple: An Empirical Study of the Accuracy-Explainability Trade-off in Machine Learning for Public Policy*. Association for Computing Machinery, 2022, pp. 248–266. URL: https://doi.org/10.1145/3531146.3533090 (visited on 11/03/2024).

[65] Lundberg, S. M./ Lee, S.-I. "A unified approach to interpreting model predictions". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Long Beach, California, USA: Curran Associates Inc., 2017, pp. 4768–4777.

[66] Kwon, Y./ Zou, J. *WeightedSHAP: analyzing and improving Shapley based feature attributions*. 2022. arXiv: 2209.13429 [cs.LG]. URL: https://arxiv.org/abs/2209.13429.

[67] Salih, A. M. et al. "A Perspective on Explainable Artificial Intelligence Methods: SHAP and LIME". In: *Advanced Intelligent Systems* 7.1 (2024). URL: http://dx.doi.org/10.1002/aisy.202400304.

[68] Ponce-Bobadilla, A. V. et al. "Practical guide to SHAP analysis: Explaining supervised machine learning model predictions in drug development". In: *Clinical and Translational Science* 17.11 (2024). URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC11513550/.

[69] Bilodeau, B. et al. "Impossibility theorems for feature attribution". In: *Proceedings of the National Academy of Sciences* 121.2 (2024), e2304406120. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.2304406120. URL: https://www.pnas.org/doi/abs/10.1073/pnas.2304406120.

[70] Molnar, C. *10.2 Pixel Attribution (Saliency Maps) | Interpretable Machine Learning*. 2024. URL: https://christophm.github.io/interpretable-ml-book/pixel-attribution.html.

[71] Szczepankiewicz, K. et al. "Ground truth based comparison of saliency maps algorithms". In: *Scientific Reports* 13.1 (2023). URL: https://www.nature.com/articles/s41598-023-42946-w#citeas.

[72] Selvaraju, R. R. et al. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization". In: *International Journal of Computer Vision* 128.2 (10/2019), pp. 336–359. URL: http://dx.doi.org/10.1007/s11263-019-01228-7.

[73] Chattopadhay, A. et al. “Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks”. In: (2018), pp. 839–847.

[74] Springenberg, J. T. et al. *Striving for Simplicity: The All Convolutional Net*. 2015. arXiv: 1412.6806 [cs.LG]. URL: https://arxiv.org/abs/1412.6806.

[75] Mostafa, S. et al. “Leveraging Guided Backpropagation to Select Convolutional Neural Networks for Plant Classification”. In: *Frontiers in Artificial Intelligence* 5 (2022). URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC9132261/.

[76] Mostafa, S. et al. “Leveraging Guided Backpropagation to Select Convolutional Neural Networks for Plant Classification”. In: *Frontiers in Artificial Intelligence* 5 (2022). URL: https://www.frontiersin.org/journals/artificial-intelligence/articles/10.3389/frai.2022.871162/full?utm_source=chatgpt.com.

[77] Selvaraju, R. R. et al. *Grad-CAM: Why did you say that?* 2017. arXiv: 1611.07450 [stat.ML]. URL: https://arxiv.org/abs/1611.07450.

[78] *Interpretability techniques: state of the art*. MANAGEMENT SOLUTIONS. URL: https://www.managementsolutions.com/sites/default/files/minisite/static/22959b0f-b3da-47c8-9d5c-80ec3216552b/iax/pdf/explainable-artificial-intelligence-en-04.pdf.

[79] Quinlan, J. R. “Learning decision tree classifiers”. In: *ACM Comput. Surv.* 28.1 (03/1996), pp. 71–72. URL: https://doi.org/10.1145/234313.234346.

[80] Tripepi, G. et al. “Linear and logistic regression analysis”. In: *Kidney International* 73.7 (2008), pp. 806–810. URL: https://www.sciencedirect.com/science/article/pii/S0085253815530895.

[81] Caruana, R. et al. “Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission”. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD ’15. New York, NY, USA: Association for Computing Machinery, 2015, pp. 1721–1730. URL: https://doi.org/10.1145/2783258.2788613.

[82] Vaswani, A. et al. *Attention Is All You Need*. 2023. arXiv: 1706.03762 [cs.CL]. URL: https://arxiv.org/abs/1706.03762.

[83] Davis, J. et al. “Gradient boosting for quantitative finance”. In: *Journal of Computational Finance* (2021). URL: https://api.semanticscholar.org/CorpusID:233648116.

[84] Nardone, D./ Ciaramella, A./ Staiano, A. “A Sparse-Modeling Based Approach for Class Specific Feature Selection”. In: *PeerJ Computer Science* 5 (2019), e237–e237. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC7924712/.

[85] Ramaswamy, V. V. et al. *Overlooked factors in concept-based explanations: Dataset choice, concept learnability, and human capability.* 2023. arXiv: 2207.09615 [cs.CV]. URL: https://arxiv.org/abs/2207.09615.

[86] Kim, B. et al. *Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV).* 2018. arXiv: 1711.11279 [stat.ML]. URL: https://arxiv.org/abs/1711.11279.

[87] Wang, X. et al. *Disentangled Representation Learning.* 2024. arXiv: 2211.11695 [cs.LG]. URL: https://arxiv.org/abs/2211.11695.

[88] Wang, M. et al. “A Review of Disentangled Representation Learning for Remote Sensing Data”. In: *CAAI Artificial Intelligence Research* 1.2 (2022), pp. 172–190. URL: https://www.sciopen.com/article/10.26599/AIR.2022.9150012.

[89] Verma, S. et al. *Counterfactual Explanations and Algorithmic Recourses for Machine Learning: A Review.* 2022. arXiv: 2010.10596 [cs.LG]. URL: https://arxiv.org/abs/2010.10596.

[90] Biehl, M./ Hammer, B./ Villmann, T. “Prototype-based models in machine learning”. In: *Wiley Interdisciplinary Reviews Cognitive Science* 7.2 (2016), pp. 92–111. URL: https://pubmed.ncbi.nlm.nih.gov/26800334/.

[91] Gleicher, M. “A Framework for Considering Comprehensibility in Modeling”. In: *Big Data* 4.2 (2016), pp. 75–88. URL: https://doi.org/10.1089/big.2016.0007 (visited on 11/03/2024).

[92] Yang, W. et al. “Survey on Explainable AI: From Approaches, Limitations and Applications Aspects”. In: *Human-Centric Intelligent Systems* (2023), pp. 161–188. URL: https://link.springer.com/article/10.1007/s44230-023-00038-y#citeas (visited on 11/03/2024).

[93] Saeed, W./ Omlin, C. “Explainable AI (XAI): A systematic meta-survey of current challenges and future opportunities”. In: *Knowledge-Based Systems* 263 (2023), p. 110273. URL: https://www.sciencedirect.com/science/article/pii/S0950705123000230 (visited on 11/03/2024).

[94] Crook, B./ Schlüter, M./ Speith, T. *Revisiting the Performance-Explainability Trade-Off in Explainable Artificial Intelligence (XAI).* 2023. arXiv: 2307.14239 [cs.AI]. URL: https://arxiv.org/abs/2307.14239 (visited on 11/03/2024).

[95] Antoniadi, A. M. et al. "Current Challenges and Future Opportunities for XAI in Machine Learning-Based Clinical Decision Support Systems: A Systematic Review". In: *Applied Sciences* 11.11 (2021). URL: https://www.mdpi.com/2076-3417/11/11/5088 (visited on 11/03/2024).

[96] Caruana, R. et al. "Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* KDD '15. Sydney, NSW, Australia: Association for Computing Machinery, 2015, pp. 1721–1730. URL: https://doi.org/10.1145/2783258.2788613.

[97] Menis-Mastromichalakis, O. et al. *Semantic Prototypes: Enhancing Transparency Without Black Boxes.* 2024. arXiv: 2407.15871 [cs.LG]. URL: https://arxiv.org/abs/2407.15871.

[98] Fauvel, K./ Chen, F./ Rossi, D. "A Lightweight, Efficient and Explainable-by-Design Convolutional Neural Network for Internet Traffic Classification". In: *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.* KDD '23. Long Beach, CA, USA: Association for Computing Machinery, 2023, pp. 4013–4023. URL: https://doi.org/10.1145/3580305.3599762.

[99] Kim, E. et al. "XProtoNet: Diagnosis in Chest Radiography with Global and Local Explanations". In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 2021, pp. 15714–15723.

[100] Yan, B. et al. "FedRFQ: Prototype-Based Federated Learning With Reduced Redundancy, Minimal Failure, and Enhanced Quality". In: *IEEE Transactions on Computers* 73.4 (2024), pp. 1086–1098. URL: http://dx.doi.org/10.1109/TC.2024.3353455.

[101] Cui, Y. et al. "Class-Balanced Loss Based on Effective Number of Samples". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 2019, pp. 9260–9269.

[102] Chaddad, A. et al. "Survey of Explainable AI Techniques in Healthcare". In: *Sensors* 23.2 (2023). URL: https://www.mdpi.com/1424-8220/23/2/634.

[103] Santis, A. D. et al. *Visual-TCAV: Concept-based Attribution and Saliency Maps for Post-hoc Explainability in Image Classification*. 2024. arXiv: 2411.05698 [cs.CV]. URL: https://arxiv.org/abs/2411.05698.

[104] Sadashivaiah, V./ Yan, P./ Hendler, J. A. *Explaining Chest X-ray Pathology Models using Textual Concepts*. 2024. arXiv: 2407.00557 [cs.CV]. URL: https://arxiv.org/abs/2407.00557.

[105] Vats, A./ Pedersen, M./ Mohammed, A. "Concept-based reasoning in medical imaging". In: *International Journal of Computer Assisted Radiology and Surgery* 18.7 (2023), pp. 1335–1339. URL: https://link.springer.com/article/10.1007/s11548-023-02920-3#citeas.

[106] Wollek, A. et al. *German CheXpert Chest X-ray Radiology Report Labeler*. 2023. arXiv: 2306.02777 [cs.CL]. URL: https://arxiv.org/abs/2306.02777.

[107] Pan, X. et al. " On the Integration of Self-Attention and Convolution ". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, 2022, pp. 805–815. URL: https://doi.ieeecomputersociety.org/10.1109/CVPR52688.2022.00089.

[108] Mott, A. et al. *Towards Interpretable Reinforcement Learning Using Attention Augmented Agents*. 2019. arXiv: 1906.02500 [cs.LG]. URL: https://arxiv.org/abs/1906.02500.

[109] Nie, Y. et al. *ConceptCLIP: Towards Trustworthy Medical AI via Concept-Enhanced Contrastive Langauge-Image Pre-training*. 2025. arXiv: 2501.15579 [cs.CV]. URL: https://arxiv.org/abs/2501.15579.

[110] A. Santos, I. B. de/ Carvalho, A. C. P. L. F. de. *ProtoAL: Interpretable Deep Active Learning with prototypes for medical imaging*. 2024. arXiv: 2404.04736 [cs.CV]. URL: https://arxiv.org/abs/2404.04736.

[111] Mayo Clinic, ed. *Atelectasis*. https://www.mayoclinic.org/diseases-conditions/atelectasis/symptoms-causes/syc-20369684. 11/2023. (Visited on 11/21/2024).

[112] Vootiprux, W. *What is Cardiomegaly?* Ed. by MedPark Hospital. https://www.medparkhospital.com/en-US/disease-and-treatment/what-is-cardiomegaly. 10/2022. (Visited on 11/21/2024).

[113] Kampalath, R./ Jelic, S. *What is Consolidation?* Ed. by Verywell Health. https://www.verywellhealth.com/lung-consolidatio-5221270. 08/2024. (Visited on 11/21/2024).

[114] Brunner, S./ Johnson, T. *Treatment, causes, and symptoms of pulmonary edema (Fluid in the lungs).* Ed. by MedicalNewsToday. https://www.medicalnewstoday.com/articles/317218. 12/2023. (Visited on 11/21/2024).

[115] Brigham and Women's Hospital, ed. *Pleural Effusion.* https://www.brighamandwomens.org/lung-center/diseases-and-conditions/pleural-effusion. (Visited on 11/21/2024).

[116] Brigham and Women's Hospital, ed. *What are Chronic Obstructive Pulmonary Disease (COPD) and Emphysema?* https://www.brighamandwomens.org/lung-center/diseases-and-conditions/chronic-obstructive-pulmonary-disease-copd-and-emphysema. (Visited on 11/21/2024).

[117] Mayo Clinic, ed. *Pulmonary Fibrosis.* https://www.mayoclinic.org/diseases-conditions/pulmonary-fibrosis/symptoms-causes/syc-20353690. 02/2024. (Visited on 11/21/2024).

[118] Mayo Clinic, ed. *Hiatal hernia.* https://www.mayoclinic.org/diseases-conditions/hiatal-hernia/symptoms-causes/syc-20373379. 12/2023. (Visited on 11/21/2024).

[119] Weerakkody, Y./ Bell, D. J. *Pulmonary infiltrates.* Ed. by Radiopedia. https://radiopaedia.org/articles/pulmonary-infiltrates-1. 08/2020. (Visited on 11/21/2024).

[120] RWJ Barnabas Health, ed. *Lung Mass.* https://www.rwjbh.org/treatment-care/cancer/types-of-cancer/lung-thoracic-cancer/lung-mass/. (Visited on 11/21/2024).

[121] Brigham and Women's Hospital, ed. *What is a Lung Nodule?* https://www.brighamandwomens.org/lung-center/diseases-and-conditions/lung-nodules. (Visited on 11/21/2024).

[122] Schriber, A. D./ Gotwals, J./ Murray, M. *Pneumothorax.* Ed. by Brigham and Women's Hospital. https://www.msdmanuals.com/professional/pulmonary-disorders/mediastinal-and-pleural-disorders/pneumothorax. 08/2023. (Visited on 11/21/2024).

[123] Rahman, N. M. *Pneumonia*. Ed. by MSD Manual. https://healthlibrary.brighamandwomens.org/Search/85,P01321?pk_vid=11fde3189ab40e66173219821034fec5. 08/2023. (Visited on 11/21/2024).

[124] Moncivais, K./ Molinari, L. *Pleural Thickening*. Ed. by Mesothelioma. https://www.mesothelioma.com/pleural-thickening/. 11/2024. (Visited on 11/21/2024).

[125] Wang, X. et al. "ChestX-Ray8: Hospital-Scale Chest X-Ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 07/2017, pp. 3462–3471. URL: http://dx.doi.org/10.1109/CVPR.2017.369.

# A. Medical background

Although the primary focus of this work is on ML techniques, rather than on the detailed medical analysis of the diseases themselves, it is still crucial to understand the basic nature of these conditions. A foundational understanding of these diseases provides the necessary background for interpreting the dataset and the potential impact of some medical factors on the analysis. As such, the following section provides a brief overview of each of the diseases covered in the dataset.

- **Atelectasis** is the partial or complete collapse of a lung or a lobe, resulting breathing issues [111].
- **Cardiomegaly** refers to an abnormal growth of the heart, resulting in difficulty pumping blood [112].
- **Consolidation** occurs when pus, blood or other fluid flow inside the lung tissue, making it solid rather than flexible when filled with air [113].
- **Edema** happens when fluid collects in the lung, specifically in the air sacs, leading to difficulties when breathing [114].
- **Effusion**, also known as "water on the lungs" is the abnormal accumulation of fluid in the pleural space surrounding the lungs, which can impair breathing [115].
- **Emphysema** is a chronic lung condition where the alveoli, or air sacs, become damaged, leading to an obstruction of airflow [116]. The main cause for this disease is smoking.
- **Fibrosis** involves the thickening and scarring of lung tissue around and between the air sacs, hindering the oxygen flow into the bloodstream [117].
- **Hernia** occurs when parts of the stomach bulges through the diaphragm into the chest, leading to symptoms like heartburn or chest pain [118].
- **Infiltration**, as a context-dependent and non-specific term, refers to an abnormal substance thicker than air such as pus or blood accumulating within cells of the lung [119].

- **Mass** is an abnormal growth in the lungs that is either benign (non-cancerous) or malignant (cancerous) [120].
- **Nodule** is a small, solid growth embedded in the lung tissue indicating a variety of other diseases [121].
- **Pneumonia** is an infection of the lungs, specifically the alveoli, caused by bacteria, viruses such as COVID-19, or fungi that can lead to the air sacs be filled with pus and other liquid [122].
- **Pneumothorax** is the presence of air in the pleural space (space between chest wall and lung), causing the lung to collapse partially or completely [123].
- **Pleural Thickening** refers to the abnormal thickening of the pleura, the lining surrounding the lungs, due to scarring from infections or injury [124].

These overlaps, which make about a half of all images in this dataset containing a finding, can be seen in the following illustration. It shows the occurence and the connections of the 14 diseases of th Figure A.1 illustrates the exact distribution of the diseases in a pie chart. The center of the pie highlights the overlaps of the 14 diseases in this dataset.

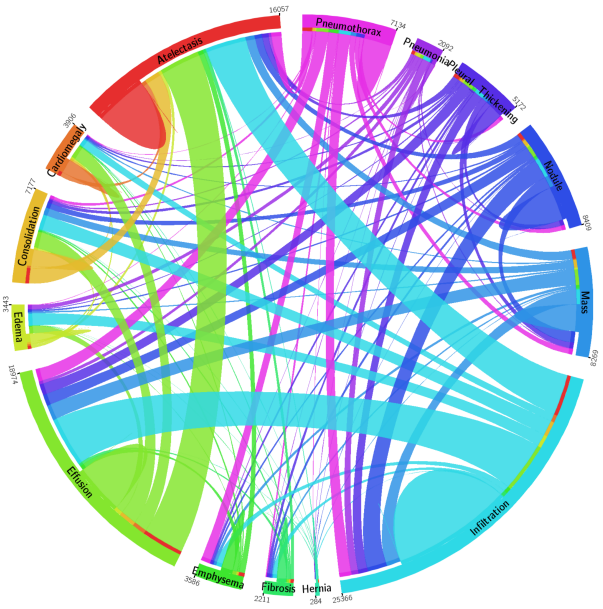


Figure A.1.: Proportions of images and ther overlapts by disease in the multi-label dataset; Source: [125]

Figure A.1 shows that there is a significant imbalance between various diseases. As an instance, Infiltration occurs approxiametely 90 times more frequently than Hernia. Figure A.1 implies that there are more images with multiple labels than with one, wheras actually about 60% of the images have only one label. This is due to images with more than two labels that are represented by more than one line in this graphic.