UNIVERSIDADE D
COIMBRA

José Pedro Pereira Amorim

Deep Learning Interpretability Methods for Cancer Medical Imaging

PhD Thesis in Informatics Engineering, Intelligent Systems, supervised by Professor Pedro Manuel Henriques da Cunha Abreu and Professor João António Miranda dos Santos, and presented to the Department of Informatics Engineering of the Faculty of Sciences and Technology of the University of Coimbra

May 2023

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE D COIMBRA

PhD Thesis in Informatics Engineering
Intelligent Systems

Deep Learning Interpretability Methods for Cancer Medical Imaging

Author:
José Pedro Pereira Amorim
jpamorim@dei.uc.pt

Advisors:
Professor **Pedro Henriques Abreu**, PhD
Professor **João Santos**, PhD

Coimbra, 2023

This page is intentionally left blank.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisor, Prof. Pedro Henriques Abreu, for his invaluable guidance from the beginning of this journey. His insightful feedback and constant encouragement were pivotal to the completion of this thesis. Thank you for your constant support.

I would like to express my thanks to the great professors I had the pleasure to work with. Mauricio Reyes and Henning Müller for their careful reading of my work and constructive feedback.

I further wish to extend my appreciation to Dr. João Santos and the IPO Porto Research Centre, and the Fundação para a Ciência e a Tecnologia, for financially supporting my research throughout this doctoral program. To the Department of Informatics Engineering of the University of Coimbra, especially the Cognitive and Media Systems Group of the Centre for Informatics and Systems, and Prof. Penousal Machado, I appreciate the opportunity to conduct my research in your lab.

I am immensely grateful to my friends for their unwavering support throughout these academic years. Miriam, Ricardo, Inês, and Joana who went through the same experience as me, were a source of inspiration and encouragement. I wanted to express my thanks to my closest friends for bringing happiness to my life and helping me clear my mind in the most challenging times.

I would also like to thank all the understanding and love I received from my family. They have been my pillars of strength, always there to provide me with comfort and strength.

To my girlfriend Inês, thank you for keeping me grounded, and motivated and helping me to see the bigger picture.

This page is intentionally left blank.

Abstract

Artificial Intelligence (AI) techniques have proven to be successful in a variety of contexts. In particular, Deep Learning techniques have achieved human-level results in medical imaging tasks, but their use in real-world contexts has been limited due to their inherent lack of interpretability.

Enclosed in the context of eXplainable AI, interpretability is understood as a set of techniques or model properties that make the output generation process of the system explainable and understandable to humans.

Despite several progress developed in the area, there are still many challenges to be tackled, including the following: (1) lack of standard terminology of interpretability AI, (2) increase of complexity of the models for better performance sacrificing their interpretability, (3) unreliability of some interpretability techniques and variability of results between each other, and (4) lack of interpretability metrics and ground-truths.

In this thesis, we attempted to address the challenge (1) by developing a global taxonomy of interpretable AI that could be used by multiple stakeholders such as developers, physicians, and lawyers.

Concerning challenge (2) we developed an approach to transfer the knowledge of a complex network to a simpler model maintaining interpretability and increasing performance; we also studied the effect of regularization of the quality of explanations and show that overall smaller regularization values produce better explanations.

With regard to challenge (3) we evaluate different interpretability techniques based on the robustness to natural noise and found that some techniques were more robust than others.

As for challenge (4) we developed an evaluation approach and proposed various interpretability metrics, moreover we propose an approach to obtain a ground-truth based on interpretability techniques.

Resumo

As técnicas de Inteligência Artificial (IA) provaram ser bem-sucedidas numa variedade de contextos. Em particular, as técnicas de aprendizagem profunda alcançaram resultados de nível humano em tarefas de imagens médicas, mas o seu uso em contextos reais é limitado devido à sua inerente falta de interpretabilidade.

Incluída no contexto de inteligência artificial explicável, interpretabilidade é entendida como um conjunto de técnicas ou propriedades de modelo que tornam o processo de geração de saída do sistema explicável e compreensível para humanos.

Apesar do progresso, ainda há muitos desafios a serem enfrentados, incluindo os seguintes: (1) falta de terminologia padrão em relação à IA interpretável (2) aumento da complexidade dos modelos para melhor desempenho sacrificando a sua interpretabilidade, (3) falta de confiabilidade de algumas técnicas de interpretabilidade e variabilidade de resultados entre si, e (4) falta de métricas de interpretabilidade e verdades fundamentais.

Nesta tese, tentamos enfrentar o desafio (1) ao desenvolver uma taxonomia global de IA interpretável que possa ser usada por múltiplas partes interessados, como desenvolvedores, médicos e advogados.

Relativamente ao desafio (2) desenvolvemos uma abordagem para transferir o conhecimento de uma rede complexa para um modelo mais simples, aumentando a interpretabilidade e mantendo o desempenho; também estudamos o efeito da regularização na qualidade das explicações e demonstramos que valores que menor regularização em geral produzem melhores explicações.

Em relação ao desafio (3) avaliamos diferentes técnicas de interpretabilidade com base na robustez ao ruído natural e descobrimos que algumas técnicas eram mais robustas do que outras.

Quanto ao desafio (4) desenvolvemos uma abordagem de avaliação e propusemos várias métricas de interpretabilidade, além disso, propomos uma abordagem para obter uma verdade fundamental com base em técnicas de interpretabilidade.

Contents

List of Figures

This page is intentionally left blank.

List of Tables

Chapter 1

Introduction

Deep Learning (DL) has emerged as a prominent technology in recent years, thanks to its ability to automatically extract and learn complex features from data, reducing the need for manual feature engineering [132]. However, this also implies that these features may be poorly understood. Additionally, with the increase in complexity of the models, their capacity to learn the mappings between input data and output has been greatly enhanced. The high complexity of DL models has also raised concerns about their interpretability and safety, which must be addressed to ensure that they can be effectively deployed in critical real-world scenarios like in healthcare.

In this context, based on the need for interpretability - a set of techniques or model properties that make the output generation process of the system explainable and understandable to humans - several challenges arise. Namely, the sacrifice of performance for interpretability, the variability between the explanations of different interpretability techniques, and the lack of interpretability metrics and ground-truths. Increasingly, these problems are not only related to physicians but also to patients who seek more knowledge about their illness and its treatment.

1.1 Motivation

Artificial Intelligence (AI) has the potential to greatly improve the treatment and diagnosis of patients as well as assist physicians, reducing their workload and bringing new insights. Machine learning (ML) is a field of AI that consists in applying typically model-based approaches capable of finding the underlying relationships among data. Deep Learning, a subset of ML, is a group of powerful algorithms based on artificial neural networks to recognize patterns in complex data, such as medical images. The power of their predictions lies in their capacity to autonomously acquire high-level data representations, where the parameters of the higher layers are able to express sophisticated representations using the

simpler representations learned by the lower layers [87].

Convolutional Neural Networks (CNNs) are the cornerstone of most state-of-the-art decision-making systems in medical imaging. They have reached human-level performance in several tasks such as melanoma detection from dermoscopic images [40], or lymph node metastases detection from histopathological images [32]. The automatic detection of diseases is especially important in the case of Tumor Lymph Node Metastasis (TNM) as it requires a highly skilled pathologist and is time-consuming and error-prone [170].

The benefits of decision-making systems can be better understood using digital pathology as an example. With the advent of the digital era, the Whole Slide Image (WSI) format emerged, which typically exhibits a size of 35.000 x 46.000 pixels or 1.6 gigapixels [22] and requires a highly skilled pathologist while being extremely time-consuming and error-prone [170]. Due to that, it is not surprising that between 2014 and 2022 the US Food and Drug Administration approved 521 Artificial Intelligence and Machine Learning (AI/ML)-enabled medical devices [76] for clinical purposes encompassing different areas, turning the application of ML in healthcare context a reality. The rate of approved devices is accelerating, with 206 machines having been approved since 2021. The majority of these devices fall in the category of clinical decision-support systems [76], having the role of assisting physicians in the diagnosis of patients. Nevertheless, due to the impact that clinical decisions can have on the life of the patient, decision-support systems cannot be allowed to make decisions independently [78].

In healthcare scenarios, the physician has responsibility in the decision [189] and should understand the rationale behind the systems' predictions. The black-box nature of DL systems does not allow any understanding of the mechanism behind their predictions, which invalidates their use in a clinical setting. This situation can become more critical due to new European regulations, not only the physicians but also the patients are entitled to an explanation [166]. This follows the High-Level Expert Group on AI (AI HLEG) which was set up by the European Commission that lists explicability as one of the ethical principles that must be respected in order to ensure that AI systems are developed, deployed, and used in a trustworthy manner [5]. And while it does not set the requirement for an interpretable representation of a mathematical model it should provide an explanation that the decision improves the explainee's understanding of the decision generation process [36].

Interpretability is a difficult concept to define and thus many definitions have emerged in the literature. The main dividing point is concerning two main concepts: interpretability and explainability. While several researchers use these two terms interchangeably [1, 20, 146], other works suggest that there is a difference between these terms [135, 154, 193]. These disparities motivated a collaboration with researchers from different fields to propose an overarching terminology of interpretability of AI systems that can be accepted by ML researchers but also by the social sciences community which is explored in Chapter 3. In this thesis work, interpretability will be assumed as a set of techniques or model properties

that make the output generation process of the system explainable and understandable to humans. This can be achieved by introducing interpretability by design, which we call intrinsic interpretability (i.e. before training the model parameters), or by generating post-hoc explanations that do not affect the training of the model parameters. An AI system is interpretable if it is possible to translate its working principles and outcomes in human-understandable language without affecting the validity of the system. Explainability is to illustrate what features or high-level concepts were used by the ML system to generate predictions for one or multiple inputs.

While some traditional ML models such as decision trees are simple and small enough that allows any user to inspect them and understand their rationale, DL models are too complex to understand without external tools. Due to the increasing computational power and efficiency of machines, it became possible to train DL models with a higher number of parameters, materializing in the use of dozens of layers and millions of parameters. For example, the VGG16 network, a popular CNN, has 16 layers and 134.7 million trainable parameters [210]. Clearly, if interpretability is a prerequisite for clinical use, DL models cannot be used in their present condition.

Even though interpretability is essential for healthcare professionals, choosing traditional ML models based on their interpretability entails a significant sacrifice in accuracy. This is more noticeable in the medical imaging context where most of the state-of-the-art approaches are based on DL [136]. However, interpretability is not the only issue in this context. The relationship between it and the performance of the complexity of the methods is typically neglected in research studies

Intrinsic interpretability leverages the performance interpretability trade-off by adding constraints to the model to make it's behavior more understandable to humans. The idea is that the complexity of the network is inversely proportional to its interpretability and reducing the complexity of the network while maintaining performance should be pursued.

Post-hoc explanations avoid this trade-off by allowing the DL model to be opaque and explaining the decisions after training. However, some concerns have risen in the research community about the limitations of the existing post-hoc methods in terms of consistency and reliability [2, 51].

Even though many interpretability methods have been proposed, it is important to carefully evaluate and select the most appropriate method for a given task. Additionally, it is important to consider the intended audience for the explanations - either the physician or patient - to present the results in a way that is understandable and actionable for that user. For example, while a healthcare professional may require more detailed and technical explanations of the reasoning behind an AI system's decision, a patient may need a simpler and more intuitive explanation.

The demand for interpretability also arises due to a mismatch between the objectives of

the model and of the users. Although DL techniques have reached human performance in many tasks, they were optimized based only on the minimization of classification error and compared based only on performance metrics such as accuracy. In addition to the high accuracy of ML algorithms, users have additional desiderata such as fairness, privacy, reliability/robustness, causality, and trust [70]. The problem is that these objectives are hard to define, and even harder to evaluate. Evaluation of interpretability is a difficult task as there is no ground-truth of what it should look like. Three different evaluation approaches for interpretability have been proposed in the literature: application-grounded, human-grounded, and functionally-grounded [233]. Application-grounded evaluation involves conducting user studies with an expert within a real application - such as a pathologist detecting tumor cells on a WSI. Human-grounded evaluation involves conducting user studies with non-domain experts without a specific application in mind. Functionally-grounded evaluation requires no user study and instead uses a proxy metric following a formal definition of interpretability. This evaluation approach is by far less time and cost-consuming.

1.2 Research Questions

Based on the issues previously illustrated, three research questions were identified:

RQ-1: Is there a connection between the complexity of the model and its interpretability?

RQ-2: Can interpretability methods help understand how deep learning models produce their decisions?

RQ-3: Do deep learning models rely on relevant clinical information when classifying medical images?

1.3 Contributions

As part of the progress of this work, the following publications were submitted:

C1. **Amorim, J. P., Abreu, P.H., Fernández, A., Reyes, M., Santos, J. & Abreu, M. H. (2022). Interpreting deep machine learning models: An easy guide for oncologists.** ***IEEE Reviews in Biomedical Engineering*****, 16(1), 192-207.** `[Biomedical Engineering (Q1)]`.

C2. **Graziani, M., Dutkiewicz, L., Calvaresi, D., Amorim, J. P., Yordanova, K., Vered, M., Nair, R., Abreu, P. H., Blanke, T., Pulignano, V., Prior, J. O., Lauwaert, L.,Reijers, W.,Depeursinge, A.,Andrearczyk, V. & Müller,**

H. (2022). A global taxonomy of interpretable ai: unifying the terminology for the technical and social sciences. ***Artificial Intelligence Review*****, 56, 3473–3504.** [Artificial Intelligence (Q1); Linguistics and Language (Q1)].

C3. **Amorim, J. P., Domingues, I., Abreu, P.H., Reyes, M. & Santos, J. (2018). Interpreting deep learning models for ordinal problems. In** ***European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*** **(pp. 373–377). i6doc.** [CORE2018 Ranking B].

C4. **Amorim, J. P., Abreu, P.H., Reyes, M. & Santos, J. (2020). Interpretability vs. Complexity: The Friction in Deep Neural Networks. In** ***International Joint Conference on Neural Networks (IJCNN)*** **(pp. 373–377). IEEE Xplore.** [CORE2020 Ranking A].

C5. **Amorim, J. P., Abreu, P.H., Santos, J., Cortes M. & Vila, V. (2023). Evaluating the faithfulness of saliency maps in explaining deep learning models using realistic perturbations.** ***Information Processing & Management*****, 60(2), 103225.** [Computer Science Applications (Q1)].

C6. **Amorim, J. P., Abreu, P.H., Santos, J. & Henning, M. (2023). Evaluation of similarity between post-hoc and intrinsic interpretability for histopathological imaging.** ***ArXiv e-prints*****, Submitted to** ***Decision Support Systems*** [Information Systems (Q1)].

In sum, the work developed during the course of this doctoral program resulted in the following research contributions: **6 research papers: 3 published in Q1 journals, (plus 1 submitted), 1 conference paper published in an A conference, and 1 paper published in a B conference**.

From the literature review, we were able to identify current and future trends in explainable AI in the context of oncology about what researchers and clinicians focus on in the following categories:

- Interpretability strategies and explanations;
- Deep learning architectures and techniques;
- Types of cancers and predictive tasks;
- Data modalities and exams.

As a result of our review, we identified three main issues: (1) limitations on the applications of interpretability methods; (2) lack of reliability of some interpretability methods; and (3)

lack of evaluation metrics for interpretability methods. While the review showed that the main interpretability strategy is saliency maps and feature importance, it also uncovered the reliability issues of those methods and a lack of evaluation metrics to assess their correctness and faithfulness to the underlying prediction they are trying to explain. These issues motivated subsequent works (Chapter 4, Chapter 5, and Chapter 6) of evaluating the reliability of methods.

As detailed in the Research Questions section three research questions were explored during this work. In the following section, we describe the main contributions of our work with respect to these research questions individually.

RQ-1: Is there a connection between the complexity of the model and its interpretability?

Intrinsic interpretable models can be understood by users because the capacity of the model is reduced by the addition of constraints that reduces the complexity of the model. While some traditional machine learning models, such as decision trees, are not as predictively powerful as DL models, they are more interpretable.

Chapter 4 proposes an approach to transfer the knowledge of deep neural networks to intrinsic interpretable models without losing performance. Showing that reducing complexity can have a positive impact on interoperability. Also, we extended our study of complexity and interpretability to post-hoc interpretability. We conducted an ablation study where the impact that regularization, a popular way to reduce the complexity and capacity of the network, has on the quality of post-hoc interpretability.

RQ-2: Can interpretability methods help understand how deep learning models produce their decisions?

During the experiments in Chapters 4, 5 and 6 different intrinsic interpretability methods and post-hoc interpretability methods were explored and applied in the context of oncology.

In Chapter 4 we proposed an approach based on knowledge distillation [108] to transfer the knowledge obtained by deep learning models to intrinsically interpretable models (e.g. decision tree). This approach allows the user to understand the basis of the classifications of the DL model based on the behavior of the interpretable model which mimics it.

In Chapter 6 we adapted Prototypical Part Network for digital pathology which contains interpretable components, namely the similarity score and attribution map which not only allow the user to understand how the network makes the decision but also is used by the model to formulate its decision. Together, the different components give us a complete understanding of the decision.

RQ-3: Do deep learning models rely on relevant clinical information when classifying medical images?

In Chapter 5 we proposed an evaluation approach based on the introduction of realistic perturbations to evaluate the impact that the introduction of medical evidence, or the removal of it has on post-hoc explanations, namely saliency map methods. This evaluation approach solves some problems present in current evaluation approaches where uniform or random noise is introduced, which can produce unrealistic input images and result in unreliable evaluation results. It was also proposed new interpretability metrics, based on saliency models, that can measure the correlation between two saliency maps.

We have proposed an evaluation approach that combines intrinsic interpretability in the form of a prototypical part network (ProtoPNet) and post-hoc explanation in the form of saliency maps for the measure of overlap between methods. Overall the overlap found was substantial but not total. Two methods presented statistically a bigger overlap with ProtoPNet, SmoothGrad and Occlusion. Deconvolution and Lime have shown consistently lower overlap.

1.4 Outline

The thesis is structured as follows. Chapter 1 introduces the scope and objectives of the thesis. In Chapter 2 can be found the main concepts of deep learning, an overview of interpretability methods, and presents a literature review of interpretability methods in the context of oncological diseases. Chapter 3 presents a comprehensive and multi-disciplinary taxonomy of interpretable AI.

Chapter 4 addresses the first research question, which is to determine if there is a relationship between the complexity of a model and interpretability. It proposes an approach to transfer the knowledge of complex networks to interpretable networks without losing performance. We continue our study of complexity and present a study where the purpose is to evaluate the impact that complexity has on the faithfulness of post-hoc explanations, namely saliency map methods.

Starting with Chapter 5 we tackled an open problem identified in the literature review in Chapter 2 concerning the lack of evaluation metrics for interpretability methods. We propose an approach to evaluate the faithfulness of the saliency maps by introducing natural perturbations. This entails conducting experiments with an extensive set of saliency map methods. The following chapter, therefore, proceeds to find alternatives for saliency maps.

In Chapter 6 we focus on the development of an intrinsical interpretable model and following the evaluation metrics developed in the previous Chapter 5 we start to compare the explanation made by these models and the saliency maps produced by black-box models.

Finally, Chapter 7, which ends the thesis, summarising its main conclusions and presenting future work directions.

Chapter 2

Interpretable Deep Learning in Oncology

Today, in healthcare scenarios, we are living in a digital era where physical patient records are mapped to digital formats. This has opened the possibility to improve the efficiency and quality of treatment provided to patients by building decision-support systems.

The majority of ML algorithms are supervised which means that in these scenarios, they need a help of a physician to label the data before the mining process starts. As an example, in the overall survival prediction of breast cancer patients, it is necessary that a physician labels the set of patient data that will be used in the training process with the target variable. When this variable is discrete we are presented with a classification problem (benign or malignant), or a regression problem in case the variable is continuous (overall survival - measured in months).

Among different ML paradigms that are used in medical contexts, the Artificial Neural Network (ANN) is a popular supervised algorithm inspired by biological neurons, and began to be used in healthcare in the early 90s [175]. The ANN is an analogy used by computer scientists to emulate the behavior of the human brain and is composed of an input, an output, and intermediate layers, which are also called hidden layers. Similarly to biological neurons, each artificial neuron, or perceptron [191], receives a set of inputs, either from the input layer or other neurons, performs a linear combination based on its weights and make a non-linear decision whether to activate the neuron and fires it.

Due to the increasing computational power, the complexity of these networks has substantially grown, materializing in the use of dozens of layers and millions of neurons. In this context, Deep Learning (DL) techniques - a subset of ANN techniques - emerged as the state of the art for many real-world problems, surpassing other ML techniques, and reaching human-level performance in several tasks such as in the classification of melanoma from dermoscopic images [40], or the detection of lymph node metastases in breast cancers

from pathology images [32].

Despite its vast potential DL suffers from several disadvantages. First is the dependency on large amounts of data and computational power. Also, the black-box nature of DL makes it difficult to interpret their decisions and prevents their dissemination in clinical practice.

This chapter presents an overview of how DL techniques make decisions and illustrates the strategies that can be used in the oncological field to explain them, as it is an essential step towards the integration of DL in the workflow of physicians in the field of oncology. To better illustrate these strategies to the reader that can include ML researchers, oncologists, and other healthcare agents we give self-explanatory oncological examples. The reader can refer to Reyes *et al.* [184] for a review of more specific areas such as radiology [184], or Liu *et al.* [140] for work more related to traditional machine learning techniques.

From the works found in the literature, more than 60% are related to breast, skin, or brain cancers and the majority focused on explaining the importance of tumor characteristics (e.g. dimension, shape) in the disease behavior prediction. Among the DL techniques used in the oncology field which were interpreted, the majority are multilayer perceptrons and convolutional neural networks. We also have found that the majority of works focus on medical imaging (e.g. mammogram, histological images, and dermoscopic images) applied to breast and skin cancer. Existing explanations focus on the most prevalent diseases as well as well-curated datasets and challenges targeted at those diseases. Overall, most works focus on the validation of the knowledge acquired by the DL model for the diagnosis of malignancy or detection of a cancer disease.

Despite being successfully applied in different cancer scenarios, endowing deep learning techniques with the ability to explain their predictions, while maintaining their exceptional performance, will continue to be one of the greatest challenges faced by artificial intelligence. Future work includes the extension of interpretability methods for debugging model misbehavior and acquiring new knowledge about the disease, as well as largely overlooked cancer tasks such as tumor segmentation and image registration. Also, the evaluation of interpretability methods so that they can be compared and validated.

Throughout the next two overview sections, we will talk about several ANN techniques illustrating their internal architectures and learning processes using a self-explanatory oncological example, that consists of the classification of a breast tumor based on handcrafted features such as mass density (fat-containing - 0, low - 1, equal - 2, high - 3), shape (round - 0, oval - 1, irregular - 2) and the breast side that it was found (left - 0 or right - 1) as well as the raw mammogram. Using such features as an input, the goal of the different types of ANNs will predict an output related to the malignancy of the tumor (benign - 0 or malignant - 1).

2.1 Artificial Neural Networks Techniques Overview

Artificial Neural Networks (ANN) are a set of algorithms, inspired by the human brain, that are sometimes called "universal approximators", because they can learn to approximate mappings between any input x and an output y, assuming they are correlated. ANNs are composed of layers of neurons, which combine input from the data with a set of coefficients, or weights, assigning significance to inputs with regard to the output label.

Perceptron - The Perceptron [194] is the the precursor to the ANN techniques. In this binary classification algorithm, the linear predictor chooses to "fire" based function combining a set of weights with the input vector.

Training process - As seen in Figure 2.1, after receiving a set of variables as input (x_1, x_2, ..., x_n), the perceptron will attribute weights for each variable (w_1, w_2, ..., w_n) and afterward will use a mathematical function also known as activation function that will use the weighted sum of the input variables to produce a desired output (y). For each set of input variables, the output (y) is compared to the label corresponding to the expected output, also known as the target. During training, the weights are continuously changed to move the output of the perceptron and the target closer together.

In the example provided in Figure 2.1, the perceptron is given the breast cancer tumor variables density, shape, and side and given the weights obtained during training (0.8, 0.7, and 0 respectively), predicting the tumor to be malignant.

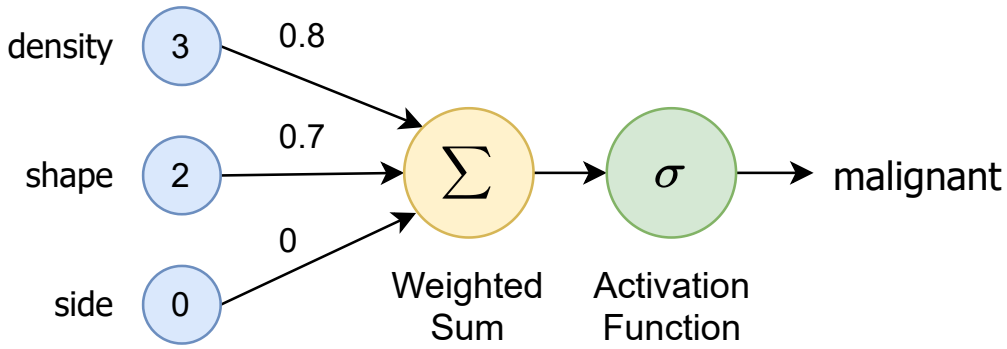


Figure 2.1: The Perceptron computes the weighted sum of the breast cancer tumor input variables, and an activation function turns the output into a binary prediction of malignancy.

Multilayer Perceptron - The Multilayer Perceptron (MLP) [194] is the natural extension of the perceptron to solve more complex problems. Rather than having a single unit, or neuron, the MLP has multiple layers with multiple neurons each, as can be seen in Figure 2.2. Also, the linear activation function of the perceptron is replaced by a non-linear activation function which helps to solve non-linear problems. Due to its multiple-layered structure, the MLP can be seen as a deep neural network.

Training process - After receiving a set of variables as input (x_0, x_1 ..., x_n), each intermediate neuron present in the hidden layers acts like a perceptron, performing the weighted combination of its inputs and applying a non-linear activation function. The output of the activation function of each neuron, also known as activation, acts as input for the neurons of the next layer. The combination of activation of the last intermediate layer produces a desired output (y).

MLPs have been explored on multiple public datasets for breast cancer diagnosis based on tumor characteristics such as density, shape, and side with high accuracy (>97%). Figure 2.2 illustrates the approach used in [204] based on the public Wisconsin Breast Cancer dataset. In the example, given the tumor variables (density, shape, and side) the model learns optimal weight values during training, to predict the malignancy.

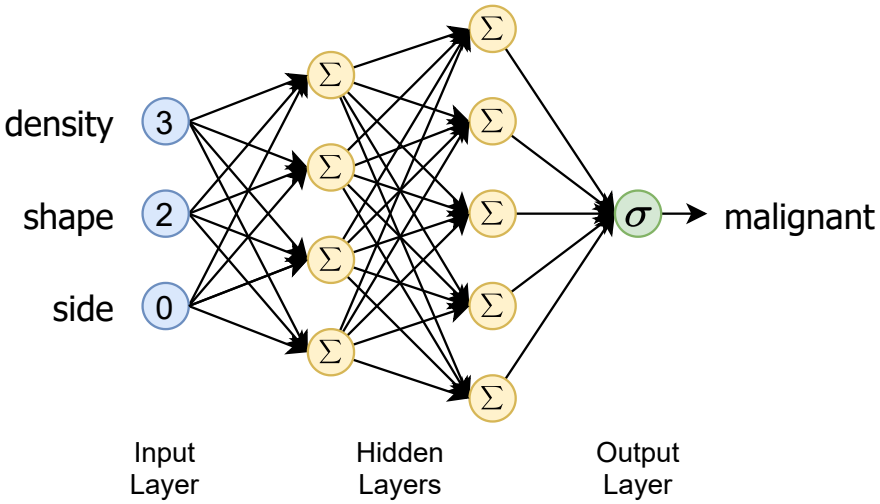


Figure 2.2: A Multilayer Perceptron (MLP) is composed of an input layer, an output layer, and two hidden layers similar to perceptrons that predict malignancy based on breast cancer tumor variables.

Due to their nature, MLPs do not scale well to images. As an example, for an image with a width and height of 100 pixels, the MLP would require 10,000 neurons just in the first layer and this number would grow exponentially with each layer.

Convolutional Neural Networks - Convolutional Neural Networks (CNN) [33, 133] techniques emerged as a solution to address the previous computational problem.

Training process - CNNs treat the image as a matrix (Figure 2.3), extracting features using a mathematical operation called convolution which helps preserve the spatial relationship between neighboring pixels. The convolution slides a small matrix, called a filter, over the original image, and for every position, it computes the element-wise multiplication between the two matrices, and the resulting value forms a single element of the output matrix, called a feature map. The filter is composed of weights (w) that are learned during training.

During feature extraction, each convolutional layer is composed of n filters resulting in n feature maps. The values of the feature maps of the last convolutional layer are concatenated into a single vector and used as an input for a MLP which makes the prediction y. During training, the values of the filter matrices and of the MLP are continuously changed to move the output closer to the expected targets.

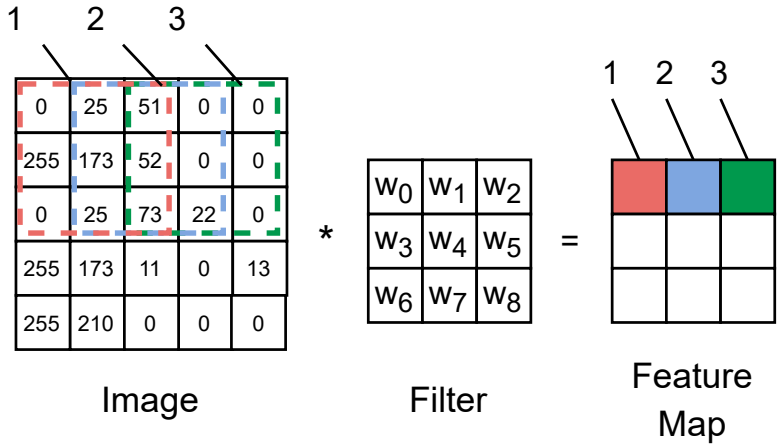


Figure 2.3: The convolution operation produces a feature map, where each element is the result of the element-wise multiplication between the region of the image and the filter (shown in the same shade).

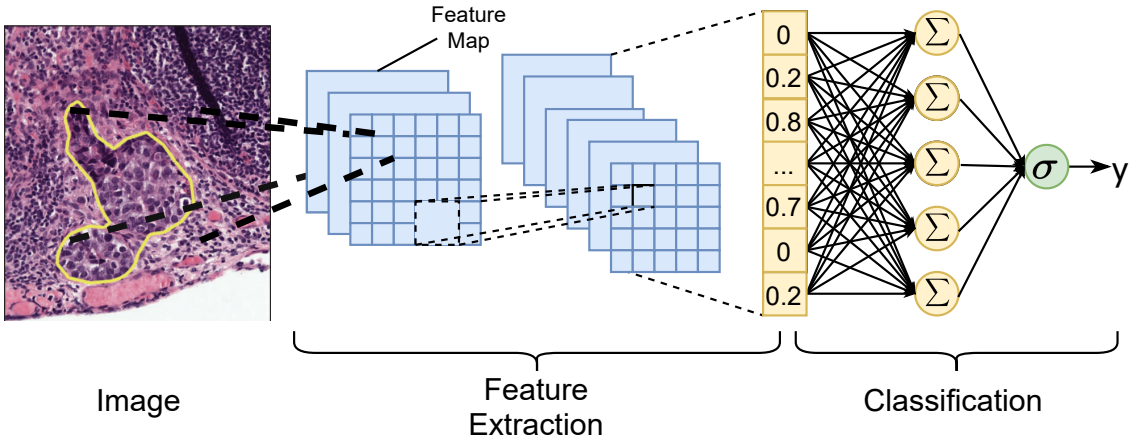


Figure 2.4: Diagram of Convolutional Neural Network (CNN) used in [32] for the detection of lymph node metastases of breast cancer in histopathological images. First, each convolutional layer produces feature maps using the convolution operator across the previous layers' output. The output of the feature extraction is concatenated into a feature vector which serves as input for the classification MLP which predicts the presence of metastases.

CNNs were used for example in the context of detection of lymph node metastases of breast cancer based on whole-slide images of digitally scanned tissue sections of over two hundred patients [32]. Figure 2.4 illustrates the approach which led to performance comparable with an expert pathologist interpreting the slides. The CNN learns the weights of the filters, and during the feature extraction is able to extract features which may include

the color and shape of the nuclei. The features are used to make the classification, which predicts the tissue to be malignant.

Although CNNs are able to take advantage of the spatial relationships between pixels, they struggle with large sequence data such as text.

Recurrent Neural Networks - Recurrent Neural Networks (RNN) techniques solve this issue by having a small network looped for each element of the sequence, allowing information to persist. A simple RNN contains a hidden state, h_t, at time t which depends on the input of the current step t and the state of the previous step.

Training process - RNNs are usually composed of only a layer of neurons, which takes an input (x_i) and predicts the output (o_i) in a recurrent way (Figure 2.5a).

This refers to the fact that its processing unit (P) is looped n times, where n represents the number of elements of the sequence. During training, the weights of the RNN are continuously changed to minimize the difference between the target sequenced, and the predicted one. As represented by the self-arrow in Figure 2.5b, the processing unit shares information among steps allowing the context and information from each slice to be passed on until a final diagnosis is given (y_t) [105].

In the example provided in Figure 2.5a, the RNN is presented in an unfolded version, where the processing unit is repeated for each step in the sequence. It corresponds to an approach for the treatment prognosis of patients with lung cancer based on Computerized Tomography (CT) of four different stages (pre-treatment, 1-month follow-up, 3-month follow-up, and 6-month follow-up) [243]. Outcomes such as survival and metastases were predicted using a RNN based on the set of features extracted from the CT using a CNN. At each step, and based on the context that is passed from the previous step, it learned to extract and memorize useful context and pass it to subsequent steps until a final prognosis was made.

Autoencoder - The autoencoder [26] is an unsupervised algorithm, which means that, unlike the previous supervised algorithms, it does not require labeled data in the training process. The goal of autoencoders is to learn a compressed representation (code) of the input data by reconstructing it as the output of the network. By restricting the size of the code, the technique can discover the interesting structures of the data, and in the case of denoising autoencoder, even reconstruct noisy images. Depending on the characteristics of the input, the encoder and decoder can have different architectures, some based on multilayer perceptrons and others on convolutional neural networks.

Training process - The denoising autoencoder (Figure 2.6) contains an encoder that receives the noisy input, compresses into a small representation, called code, and is reconstructed by a decoder into the original noiseless input. Due to the small size of the code, the autoencoder learns the distinctive features of the image and learns to ignore random

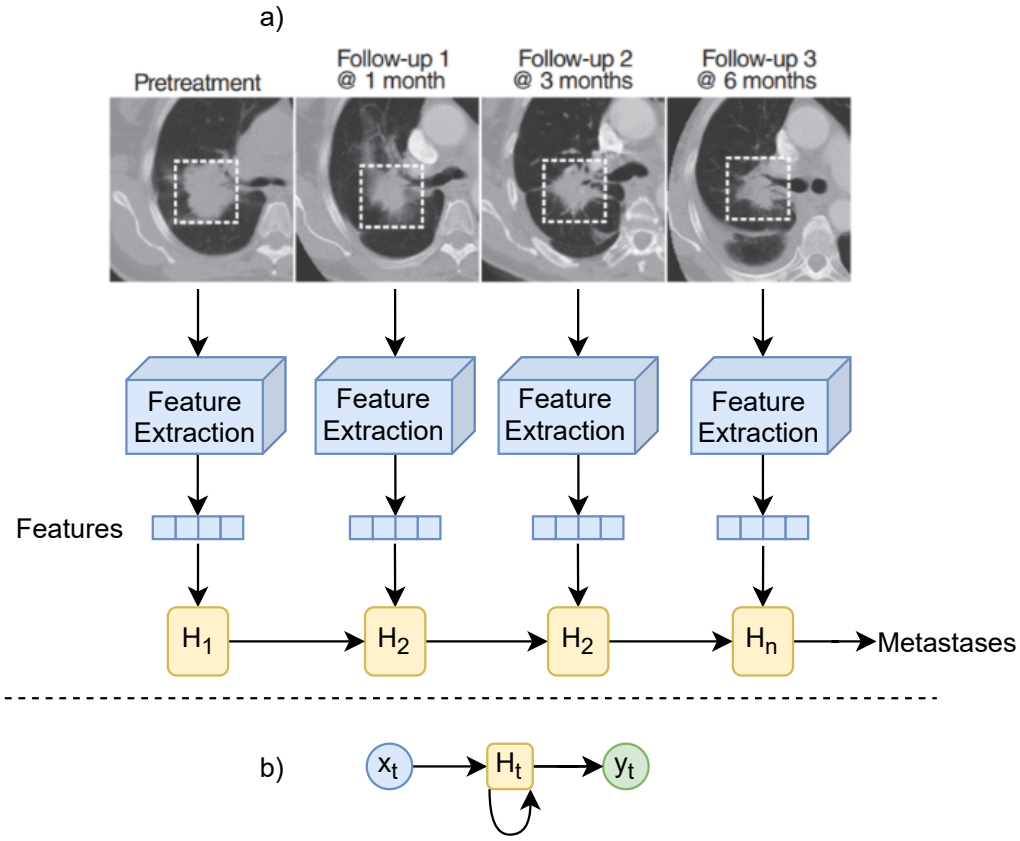


Figure 2.5: a) The Recurrent Neural Network (RNN) first extracts a set of visual features from CT slides from multiple stages using a CNN. The hidden units optimize their weights to learn useful information from the features and pass stage-specific context sequentially until a final metastases prediction is made. b) Hidden unit (H_t) shared between steps (t) and receives the context of previous CT scan (x_t) and predicts the prognosis (y_t).

noise. During training, the weights of the neurons present in the encoder and decoder are continuously updated to reduce the difference between the original input and the output, called reconstruction error, to find useful patterns in the data.

One frequent use of denoising autoencoders is the extraction and compression of relevant features for the detection of genes correlated with the ER status of patients with breast cancer [219]. Figure 2.6 illustrates how the autoencoder is given a set of gene expression data with some noise with the task of compressing the data into an relevant representation (code).

2.2 Interpretability Concepts Overview: Desiderata, Dimensions and Strategies

The significance of interpretability when developing ML solutions is well-known in academia and corporations. However, there is no consensus on the definition of interpretability [97].

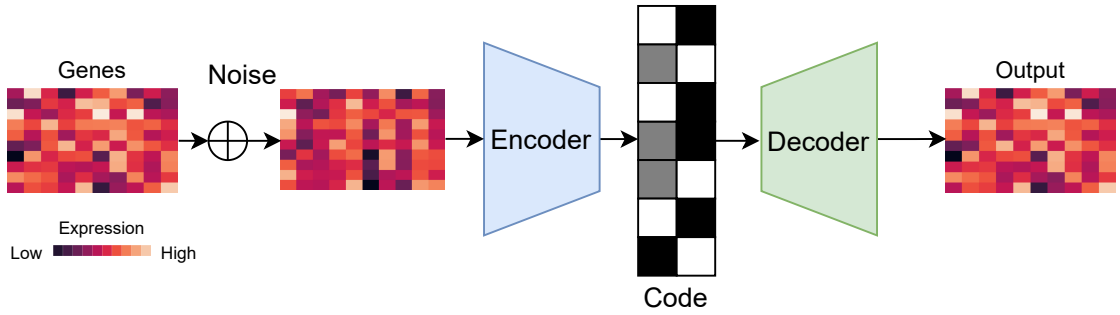


Figure 2.6: In a Denoising Autoencoder an encoder transforms a noisy gene expression data into a compressed representation (code) and the decoder transforms the code back into a denoised version of the original data.

One of the most used definitions was presented by [70] which defined interpretability as the "ability to explain or to present in understandable terms to a human". The definition used in this work for interpretability is a set of techniques or model properties that make the output generation process of the system explainable and understandable to humans.

2.2.1 Desiderata of Interpretability

The demand for interpretability arises due to a mismatch between the objectives of the model and of the users - clinicians and patients. Although DL techniques have reached human performance in melanoma diagnosis from dermoscopic images [40], or the detection of lymph node metastases in breast cancers from pathology images [32], the need to interpret them emerges, especially in healthcare contexts.

In addition to the high accuracy of ML algorithms, users have additional desiderata. Doshi-Velez and Kim [70] specified five main desiderata for interpretability:

- Fairness: Assure that protected groups (e.g. gender, ethnicity) are not somehow discriminated against (explicit or implicit);
- Privacy: Assure that sensitive information is protected;
- Reliability/Robustness: Assure high algorithmic performance despite variation of parameter or input;
- Causality: Assure that the predicted change in output due to a perturbation will occur in the real system;
- Trust: Allow users to trust a system capable of explaining its decisions rather than a black box that just outputs the decision itself.

2.2.2 Dimensions of Interpretability

Interpretability methods can be characterized by a set of dimensions [151]: global and local interpretability, intrinsic and post-hoc interpretability, and model-specific and model-agnostic interpretability. These will be described in what follows.

Global and Local Interpretability This dimension reflects the scope of interpretability of a model and depicts the portion of predictions that the model can explain. To perform a classification task an ML algorithm first creates a data-driven model based on a set of input features (e.g. age and sex) during the training phase. The objective of this phase is to allow neurons to select important features and learn the relationships between them and the target output. Global interpretability aims to analyze this model, to understand the common patterns in the overall data that help make decisions, by studying the model's parameters (i.e. weights), and the learned relationships. Local interpretability aims to understand the relationship between the set of input features of a specific case and the model decision.

In our MLP example (Figure 2.2), based on the instances provided, the network learned relationships that help predict the tumor malignancy, based on its density, shape, and breast side. As the breast side (left or right) where the tumor appears is not indicative of the level of malignancy, the network should have learned to discard this input feature.

Global interpretability could help understand which relationships the network learned, and the example of breast side confirms that it was not used. Global interpretability can also help us know if non-random sources of noise which have not been removed have affected the model's learning (e.g. artifacts). Local interpretability could help understand the importance of the input features in the malignancy prediction of a particular patient.

Intrinsic and Post-hoc Interpretability While the increase in complexity of ANNs (i.e. number of neurons), helps solve complex problems, it increases the difficulty to interpret them. Intrinsic interpretability refers to models which due to their simplicity are interpretable by themselves, such as decision trees or sparse linear models [151]. Complex models can increase their intrinsic interpretability by constraining their complexity or simplifying their behavior. Examples of these constraints are sparsity, monotonicity, adding domain knowledge, or even constraints on the complexity of the network by limiting the number of neurons or layers.

Post-hoc interpretability refers to the application of interpretability methods after the model's training [151]. Post-hoc methods help elucidate how the model works without constraining it.

In our MLP example, we could instead use a short decision tree or a small sparse MLP

to achieve intrinsic interpretability or choose to maintain the complexity of the MLP and use a post-hoc method such as feature importance to understand the importance of the input features.

Model-specific and Model-agnostic Another way to classify interpretability methods is based on the dependency the method has on the type of model which it tries to explain. Model-agnostic methods can be applied to different types of models, while model-specific methods are only applicable to a specific type of model [151].

In our example, while a model-agnostic method could extract the importance of the density and shape from a model trained from any ML algorithm, a model-specific method would only be able to do the same for similar models.

2.2.3 Interpretability Strategies

During the training phase, DL algorithms create data-driven models that can be interpreted using different strategies producing different types of explanations. Namely feature importance, saliency map, model visualization, surrogate model, domain knowledge and example-based explanations, which will be introduced next.

Feature Importance

One of the more explored explanations is feature importance, which gives the importance or contribution of an input feature on the prediction of an example. Two main approaches are used for computing feature importance: sensitivity analysis [25] and decomposition [24, 31].

Sensitivity analysis computes the effects of the variation in the input variables in the model's output and help us answer the question "What change would make the instance more or less like a specific category?".

Decomposition approaches successively decompose the importance of the output of a layer into previous layers, until the contribution that the input features have on the output is found. It helps us answer the question "What was the feature's influence on the model's output?".

If we extract the feature importance of a decision of our example, it can have different meanings depending on the type of method used. High sensitivity values for density and shape mean that their growth would also increase the prediction of malignancy. While high contribution values of density and shape mean that the prediction of malignancy was highly influenced by the value of these features.

Saliency Map

When dealing with images, saliency maps [70] (or heatmaps) can be used to visually illustrate variations in the importance of different features, using color to convey the weight of pixels in a given prediction.

Similarly to feature importance, the pixel values of saliency maps can be obtained following two main approaches: Back-propagation methods compute the relevance of a pixel by propagating a signal from the output neuron backward through the layers to the input image in a single pass [24]. Perturbation methods compute pixel relevance by making small changes in the pixel value of the input image and compute how the changes affect the prediction [209].

An example of a saliency map, extracted from a CNN trained to predict the malignancy based on mammogram patches is seen in Figure 2.7. The red and yellow regions correspond to the most important regions of the image. The method correctly focuses on the mass, supporting our confidence in the model's decisions.

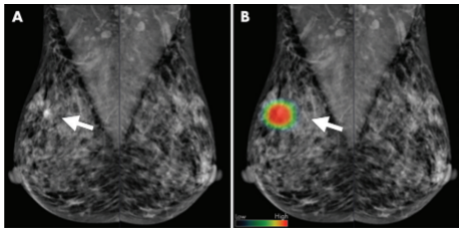


Figure 2.7: Example of a saliency map depicting the important pixels for malignancy prediction based on mammograms. Left: ground-truth expert segmentation. Right: saliency map, where the pixel intensity indicates the importance of the pixel in the classification.) [85].

Model Visualization

The ML algorithm receives an example with a set of input features, and in its internal process creates a combination of its features also called internal features. Some strategies help to visualize patterns detected in an image [248], whereas others help to visualize the feature distribution in the dataset [95, 144]. Also, whereas some strategies may choose real images from the data set that contains a pattern detected by the network [241], others artificially create images that accentuate the same patterns [160, 167].

In Figure 2.8 we can see regions of mammograms that contain patterns detected by individual filters of the CNN trained to diagnose the tumor malignancy.

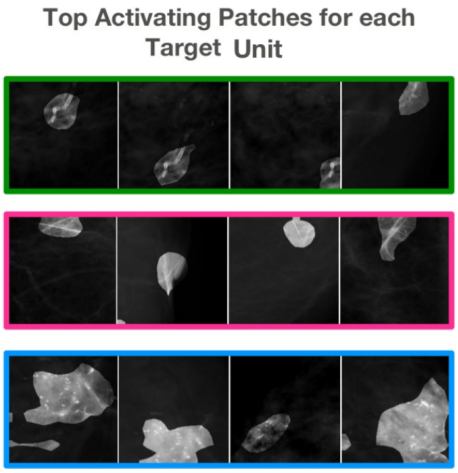


Figure 2.8: Illustration of the internal behavior of a network unit by visualizing regions of mammograms with patterns detected by individual units of the network [241].

Surrogate model

A surrogate model is an interpretable model that is trained to explain predictions of a black-box model. In the example of oncology, a rule list [186] can be extracted from a network allowing the clinician to understand the knowledge produced by the algorithm. Each rule specifies a condition that when evaluated as true produces one result (benign/malignant in malignancy diagnosis). One way of doing this is by creating a new dataset where each example of the dataset used to train the DL model is combined with its prediction and the task of the surrogate model is to predict these values.

While global surrogate models approximate the model in all the input space, local surrogate models approximate single predictions, which makes them more accurate and faithful to the model explained.

To better understand what a surrogate model is, let's consider the example in Figure 2.9, where we can see a rule list extracted from a MLP that demonstrate its decisions. This surrogate model was built by iterating through the MLP neurons and inspecting the connections between the input features and the output label so that they can be represented by rules. A decision tree is another appropriate type of surrogate model. This method could be seen as an unordered rule list where each leaf is a separate rule where the condition is the label of the path from the root to the leaf.

Domain Knowledge

Although DL algorithms extract internal features (combination of input features) automatically during the training phase, the domain knowledge of the medical field which physicians have can be used to validate the decision of the network.

Rule 1: IF (density = 'high' or 'equal') and (shape = 'irregular')
THEN **malignant**

Rule 2: IF (density = 'high')
THEN **malignant**

Rule 3: IF (density = 'fat-containing') and (shape = 'irregular')
THEN **benign**

Rule 4: OTHERWISE **benign**

Figure 2.9: Rule list extracted from a MLP trained to predict the malignancy of a breast tumor using a surrogate model strategy.

The introduction of domain knowledge from medical doctors on training can help produce models that resemble how medical doctors diagnose or focus on the features or areas they pay particular attention to [242].

In the case of malignancy diagnosis, domain knowledge can be introduced directly as an input feature, for example, a discrete value indicating the shape of the tumor. Domain knowledge can also be used as an additional target variable (e.g. shape, density), besides malignancy, allowing us to evaluate of how well the model predicts both target variables similarly to how clinicians also take those variables into account.

Example-based explanation

Example-based explanation methods select examples of the dataset that explain the behavior of the network [151]. This behavior is usually explained using the internal features (combination of input features) extracted from the examples by the network.

Similar examples are instances of the dataset that have similar values on the internal features and produce the same prediction as the example whose prediction we are explaining [47].

Counterfactual explanations can be used to explain predictions of examples by finding small changes in the example that cause the network to change its prediction.

Usually, examples of a dataset can be grouped together based on existing patterns. A prototype is a particular example of the dataset representative of its group.

Table 2.1 associates the interpretability strategies previously introduced with the dimensions of interpretability, namely scope and intrinsic vs. post-hoc. The dimensions of model specificity vs. agnostic were omitted as it depends on the actual algorithms used and not on the broader interpretability strategy.

	Dimensions	
Strategy	**Scope**	**Intrinsic vs. Post-hoc**
Feature Importance	Local	Post-hoc
Saliency Map	Local	Post-hoc
Model Visualization	Global	Post-hoc
Surrogate Model	Local/Global	Post-hoc
Domain Knowledge	Global	Intrinsic
Example-based	Global	Post-hoc

Table 2.1: Association between interpretability strategies and dimensions of interpretability.

2.3 Interpreting Deep Learning in Oncology

The use of DL techniques has become widespread in the oncology area, covering different pathologies, but their interpretation remains an unexplored field [43, 230]. In this section, an overview of interpretability strategies applied to oncological diseases will be presented. The section will be divided into different diseases, namely breast cancer, skin cancer, lung cancer, brain cancer, and others. This division was chosen to promote the best understanding of the area by the main target audience of this paper - oncologists, clinicians, and other practitioners.

We conducted a search of papers in the PubMed database published between January 2014 and September 2020 with individual and combination of search terms such as "interpretability", "deep learning", "oncology", "cancer" and "decision support systems", and compiled the results in Table 2.2. In total, 44 works were found, where the majority target breast cancer (30%), skin cancer (23%), lung cancer (9%), and brain cancer (11%). The most common interpretability strategies were saliency maps (32%) and feature importance (20%) and among the prediction tasks, most works focused on the diagnosis of malignancy (45%) and of different pathologies (27%).

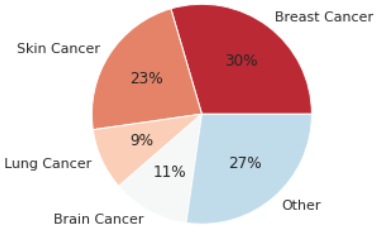
Figure 2.10 helps visualize the distribution of papers based on different classifications present in Table 2.2, namely the target disease and task as well as the interpretability strategy (explanation) and ANN technique (architecture).

2.3.1 Breast Cancer

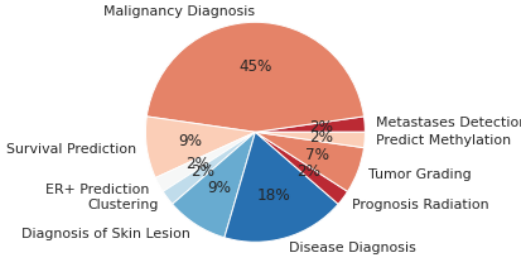
Prediction of breast cancer malignancy has been one the most successful applications of deep learning in oncology, achieving 87% sensitivity and 96% specificity when diagnosing mammograms [205]. It also is the main task in interpretability work (69% of breast cancer studies). Due to the availability of well-curated public datasets on breast cancer, mainly mammograms and hematoxylin and eosin (H&E) stained histological images, research in

this area has taken a step forward.

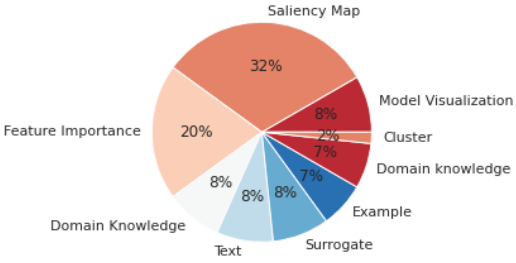
When dealing with imaging data, researchers found it important to visualize the patterns detected by the networks either through model visualization techniques or with saliency maps, please refer to section 2.2.3. These patterns were then either validated by experts or correlated with medical concepts. For other types of data (e.g. gene expression, hand-crafted features), researchers mainly focused on computing feature importance or extracting surrogate models (i.e. rule lists). In what follows, we analyze in detail some of the main selected works on the topic.



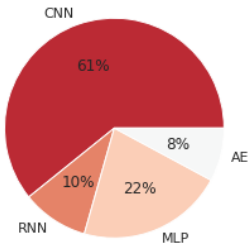
(a) Disease



(b) Task



(c) Interpretability strategy



(d) ANN techniques

Figure 2.10: Distribution of papers reviewed based on characteristics of Table 2.2

Graziani *et al.* [94] visualized the patterns of a metastases detection CNN for WSI H&E images by synthesizing images that increase the network's confidence on the prediction (Activation Maximization [160, 167]) and by extracting saliency maps [202]. They found that the network detected nuclei-resembling shapes and regions of nuclei with marked variations in size and irregular shapes. Hsieh *et al.* [241] used Network Dissection method [30] to visualize the patterns of individual filters of a malignancy classifier based on mammograms and developed a web-based tool for expert annotation. Figure 2.11 illustrates a pattern labeled as 'Calcified Vessels'. Also, other BI-RADS [114] medical concepts (e.g. mass margin) were found to overlap with patterns detected by the network.

Table 2.2: Summary of reviewed articles.

Ref	Disease	Task	Modality	Explanation	Architecture	Dataset
[94]	Breast Cancer	Metastases Detection	WSI H&E	Model Visualization, Saliency Map	CNN	Public
[241]	Breast Cancer	Malignancy Diagnosis	Mammogram	Model Visualization	CNN	Public
[90, 93]	Breast Cancer	Malignancy Diagnosis	WSI H&E	Feature Importance, Domain Knowledge	CNN	Public
[120]	Breast Cancer	Malignancy Diagnosis	Mammogram	Domain Knowledge, Saliency Map	CNN	Public
[17]	Breast Cancer	Malignancy Diagnosis	Mammogram, Ultrasound, MRI	Domain Knowledge	CNN	Public
[134]	Breast Cancer	Malignancy Diagnosis	Mammogram	Saliency Map, Text	CNN + RNN	Public
[11]	Breast Cancer	Malignancy Diagnosis	Hand-crafted	Feature Importance	CNN	Public
[13]	Breast Cancer	Malignancy Diagnosis	Hand-crafted from H&E	Surrogate	MLP	Private
[38]	Breast Cancer	Malignancy Diagnosis	Hand-crafted from H&E	Surrogate	MLP	Public
[112]	Breast Cancer	Survival Prediction	Gene expression, Biomarkers	Feature Importance	MLP	Public
[7]	Breast Cancer	ER+ Prediction	Metabolomics Data	Feature Importance	AE + MLP	Public
[139]	Breast Cancer	Clustering	Gene expression, CNA data	Model Visualization	AE	Public
[228]	Skin Cancer	Malignancy Diagnosis	Dermoscopic images	Model Visualization	CNN	Public
[61]	Skin Cancer	Malignancy Diagnosis	WSI H&E	Saliency Map	CNN	Private
[181]	Skin Cancer	Malignancy Diagnosis	Dermoscopic images	Saliency Map	CNN	Public
[72]	Skin Cancer	Diagnosis of Skin Lesion	WSI H&E	Saliency Map	CNN	Public
[86]	Skin Cancer	Diagnosis of Skin Lesion	Dermoscopic images	Saliency Map	CNN	Public
[62]	Skin Cancer	Malignancy Diagnosis	WSI H&E	Saliency Map	CNN	Public
[196]	Skin Cancer	Diagnosis of Skin Lesion	Dermoscopic images	Example	CNN	Public
[207, 208]	Skin Cancer	Malignancy Diagnosis	Dermoscopic images	Feature Importance, Example, Surrogate	MLP	Public
[58]	Skin Cancer	Diagnosis of Skin Lesion	Dermoscopic images	Example, Saliency Map	CNN	Public
[182]	Lung Cancer	Disease Diagnosis	Chest Radiograph	Saliency Map	CNN	Public
[174]	Lung Cancer	Malignancy Diagnosis	CT	Domain knowledge	CNN	Public
[206]	Lung Cancer	Malignancy Diagnosis	CT	Domain knowledge	CNN	Public
[63]	Lung Cancer	Prognosis Radiation	Biomarker, clinical data	Domain knowledge	AE + MLP	Private
[176]	Brain Cancer	Tumor Grading	MRI	Saliency Map	CNN	Public
[177]	Brain Cancer	Tumor Grading	MRI	Feature Importance, Saliency Map	MLP	Public
[101]	Brain Cancer	Predict Methylation State	MRI	Model Visualization	CNN + RNN	Public
[130]	Brain Cancer	Survival Prediction	MRI	Feature Importance	CNN	Public
[150]	Brain Cancer	Survival Prediction	WSI H&E, Biomarkers	Saliency Map	CNN	Public
[4]	Other	Malignancy Diagnosis	Gene expression	Feature Importance	MLP	Public
[249]	Other	Survival Prediction	Gene and protein expression	Feature Importance	MLP	Public
[171]	Other	Disease Diagnosis	RNA-seq expression, SVN data	Feature Importance, Surrogate	MLP	Private
[79]	Other	Disease Diagnosis	Volumetric Laser Endomicroscopy	Saliency Map	CNN	Private
[75]	Other	Disease Diagnosis	Endoscopic images	Saliency Map	CNN	Public
[125]	Other	Disease Diagnosis	WSI H&E	Saliency Map	CNN	Private
[113]	Other	Disease Diagnosis	DESI	Cluster	AE	Private
[253]	Other	Disease Diagnosis	Ophtalmic images	Domain Knowledge	CNN	Private
[254]	Other	Malignancy Diagnosis	Ultrasound	Domain knowledge	CNN	Private
[258]	Other	Malignancy Diagnosis	WSI H&E	Text, Saliency Map	CNN + RNN	Public
[236]	Other	Disease Diagnosis	Chest Radiograph	Text, Saliency Map, Text	CNN + RNN	Public
[259]	Other	Tumor Grading	WSI H&E	Text, Saliency Map	CNN + RNN	Private

Rather than being validated by experts, Graziani *et al.* [90, 93] introduced Regression Concept Vectors (an extension of Concept Activation Vectors [118]) which let them detect the importance of medical concepts (i.e. area, perimeter, and contrast) on the decisions of a breast cancer malignancy classifier based on WSI H&E network, even though they were not present in the training dataset. The contrast was found to be positively correlated with malignancy, while the correlation was negatively correlated. Kim *et al.* [120] used medical concepts during training, computing their importance alongside saliency maps to help explain the malignancy diagnosis of mammograms.

Antropova *et al.* [17] visualized the values of both deep features and hand-crafted features from different image modalities (i.e. Mammogram, Ultrasound, DCE-MRI) and found that their fusion improved malignancy diagnosis performance, most likely due to the low agreement between deep and handcrafted features.

Lee *et al.* [134] trained a malignancy diagnosis network able to justify its decisions both visually and textually. It trained a language model that composes text description [120, 121, 236, 259] from mammograms. Although the descriptions are still not sufficiently good (i.e. "There are sharp lines on some part of the complexly formed mass."), they show that this interpretability strategy has great potential.

When dealing with hand-crafted features relating to tumor size and shape, researchers found it important to simplify the network to behave linearly [11] making it easier to compute the feature importance or extract simpler classifiers that could present physicians with simple rules (i.e. decision rules [13] and symbolic rules [38]) increasing interpretability.

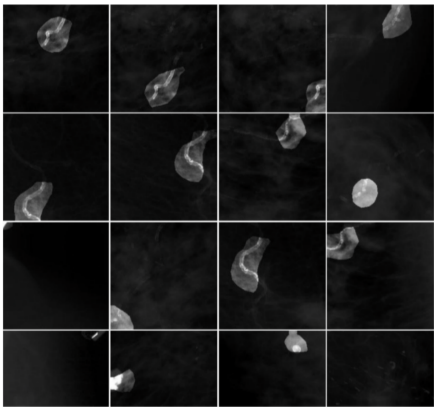


Figure 2.11: Example of pattern detected by the network and labeled by an expert as 'Calcified Vessels' in the web-based labeling tool [241].

Feature importance was the focus of most works dealing with gene expression data. For example, SALMON [112] predicted the survival risk of patients with breast cancer, and feature importance of eigengene's modules and other clinical information, they confirmed that age, progesterone receptor status, and other five mRNA sequence data co-expression modules play pivotal roles in patient prognosis. Similar methods, using the H2O [99] library, were used to detect the important features in the detection of estrogen-receptor-positive (ER+) patients based on the classification of the Estrogen Receptor Status of breast cancer patients based on metabolomics data [7]. They found eight commonly enriched significant metabolomics pathways: isoleucine, putrescine, glycerol, 5'-deoxy-5'-methylthioadenosine, ornithine, tocopherol beta, phenylalanine, and arachidonic acid. Finally, Liu *et al.* [139] used an autoencoder to find clusters of breast cancer patients based on their gene expression and copy number alteration data and visualized them using heatmaps. They found that the cluster of patients with ER-negative breast cancer patients usually has a poor prognosis.

2.3.2 Skin Cancer

Works in skin cancer are almost evenly divided on the malignancy diagnosis and diagnosis of multiple skin diseases. The modality used was also divided between two types, dermoscopic images (70%) and H&E stained histopathological images (30%). Similarly to breast cancer detection, DL has also achieved great results in skin cancer detection based on medical imaging [74]. Interpretability methods for these pathologies ranged from saliency maps, model visualization, rule extraction, text explanations and example-based explanations.

A simple visualization method was used to visualize the activation of neurons of a CNN trained to predict the malignancy of dermoscopic images [228]. Inspection of activations led to finding neurons related to medical concepts such as borders, lesions, and skin type, as well as different image artifacts such as hairs.

Cruz-Roa *et al.* [61] proposed a DL technique for the malignancy diagnosis using histological images and visualized the most salient patterns in that task which when validated by pathologists were found to be related to large-dark nuclei. Researchers also tried to improve the quality of saliency maps by making changes to the architecture of the network when diagnosis malignancy based on dermoscopic images [181] and diagnosis of skin diseases based on WSI H&E images [72]. PatchNet [181] found a trade-off between interpretability and performance, as smaller patch sizes provided saliency maps with better visual interpretability at the expense of worse generalization capabilities. Paschali *et al.* [72] also found that smaller convolutional filters resulted in more fine-grained saliency maps. Gonzalez-Diaz *et al.* [86] incorporated segmentation of lesion areas based on high-level dermoscopic features and used these segmentations to diagnose skin lesions and show relevant regions.

Example-based explanations are also useful interpretability strategies in skin cancer, as shown by Sadeghi *et al.* [196] which conducted a study that revealed that similar examples provided by DL techniques help users in classifying skin lesions from dermoscopic images. In the study, accuracy increased from 51% to 61% when the 15 most similar cases were provided to the users. Silva *et al.* [207, 208] unified complementary explanations to explain skin lesion predictions from dermoscopic images. The method extracted rules and presented them as text sentences alongside positive and counter-factual examples for every decision. Also on the same task, Codella *et al.* [58] explained the decision with similar examples using k-nearest neighbors on the deep features and highlighted the most salient regions of the image.

2.3.3 Lung Cancer

Interpretability research on the diagnosis of lung cancer focused mainly on two modalities, Chest Radiography (X-Ray) or Computed Tomography (CT). Similarly, to breast and skin cancer, DT techniques have been shown to be able to reach human-level performance. In the diagnosis of 14 different pathologies from chest radiographs, a CNN achieved radiologist-level performance [182]. Radiologists confirmed, by inspecting saliency maps, that the network localizes accurately the lung masses [260].

Other works focused on the integration between hand-crafted features related to medical concepts and deep features. Paul *et al.* [174] developed a model for the malignancy diagnosis of lung cancer using CT images and interpreted their correlation with medical features used by physicians by iteratively replacing deep features and evaluating the drop

in confidence. Although deep features were not found to be perfectly correlated with medical features, they could represent 9 of the medical features with the deep features without losing performance. In the same task, Shen *et al.* [206] proposed to model that made high-level predictions for the tumor malignancy, and low-level predictions of medical features - calcification, subtlety, lobulation, sphericity, internal structure, margin, texture, and speculation. The approach achieved comparable or better results with state-of-the-art methods in the public Lung Image Database Consortium (LIDC).

Finally, Cui *et al.* [63] used a combination of hand-crafted features composed of clinical features and cancer biomarkers in a non-small cell lung cancer who received radiotherapy to predict the damage caused by the treatment. The results found that better performance was achieved by integrating the hand-crafted features with the deep features extracted from an autoencoder [143].

2.3.4 Brain Cancer

Unlike previous pathologies, brain cancer research deviates from the diagnosis of diseases and focuses on survival prediction (40%) and tumor grading (40%), almost entirely based on Magnetic Resonance Imaging (MRI) (83%).

When performing tumor grading - distinguishing lower grade gliomas from high-grade gliomas from MRI - researchers have focused on producing saliency maps from the 3D MRI scans or Region of Interest (ROI) annotated by experts. Pereira *et al.* [176] extended existing saliency map methods for three-dimensional inputs [202, 214]. The ROI classifier achieved better performance than the 3D scan 93% and 90% accuracy), but they were both able to locate the tumor. Pereira *et al.* [177] also used a feature importance method [185] to identify MRI sequences that were relevant for features extracted from the network, and then produce saliency maps. The sequences chosen were consistent with domain knowledge.

Han *et al.* [101] train a model to predict the methylation state of the MGMT regulatory regions using MRI of Glioblastoma Multiforme (GBM) patients, resulting in 62% accuracy. The MRI scans were extracted from the Cancer Imaging Archive (TCIA) [56] and the methylation data from the Cancer Genome Atlas (TCGA) [222]. The authors developed an online visualization tool that allows the user to load an MRI scan and visualize the activation of different filters. Through this, the model was found to classify lesions with ring enhancement with negative methylation status and tumors with less clearly defined borders and heterogeneous texture with positive methylation status.

Lao *et al.* [130] constructed a model for survival prediction of patients with GBM based on deep features and hand-crafted features extracted from MRI. To reduce the number of features used, feature selection was done using feature importance methods to find features that were robust to tumor segmentation uncertainty, highly predictive and non-redundant.

Survival prediction was also performed using histological samples and genomic data [150] with validation of produced saliency maps by expert pathologists.

2.3.5 Other Pathologies

Other oncological pathologies have been shown interested in interpretability using different modalities of data (not exclusively images). Researchers that applied DL techniques on data of multiple pathologies have sought to interpret them using feature importance. For example, Ahn *et al.* [4] trained a network for malignancy diagnosis based on gene-expression data from multiple tissues and by computing the feature importance of individual genes on the diagnosis found a sub-group suspected to be oncogene-addicted as an individual gene contribute extensively in the classification. Similarly, Yousefi *et al.* [249] proposed a model for the survival prediction based on clinical, gene-expression, and protein-expression data of multiple tissues and computed the sensitivity of each feature on the survival risk, identifying that TGF-Beta 1 signaling and epithelialmesenchymal transition (EMT) gene sets are associated with poor prognosis. Oni *et al.* [171] diagnosed eight different cancer types from RNA-seq expression and single nucleotide variation (SNV) data. To explain its decisions, a linear surrogate model [185] was extracted, where its coefficient's magnitude corresponded to the importance of the genes in the prediction. The location and variability of explanations were visualized using 2D embeddings of the RNA-seq input data. They found genes related to cell proliferation and tumor growth were important for the diagnosis.

In the diagnosis of early Barrett's Neoplasia using Volumetric Laser Endomicroscopy [79], saliency maps [260] focused on the glands located around the first layers of the esophagus in high-grade dysplasia cases, and on homogeneous esophagus layers in non-dysplastic Barrett's esophagus cases. Garcia-Peraza-Herrera *et al.* [75] extended the same saliency map method to interpret the diagnosis of esophageal cancer based on endoscopic images. By computing saliency maps of different resolutions they were able to detect unhealthy patterns and diseased tissue.

Korbar *et al.* [125] interpreted the diagnosis of colorectal polyps based on histological images using saliency maps[202, 260] and found that adding a boundary box around them increased their similarity with pathologists' segmentations.

Inglese *et al.* [113] used DL techniques to find a high-level representation of mass spectrometry imaging data from colorectal adenocarcinoma biopsies. The features extracted from the network were visualized in two dimensions using t-SNE [226] unveiling clusters with different chemical and biological interactions occurring.

Zhang *et al.* [253] developed a diagnostic system of ophthalmic images that explained the diagnosis with sub-tasks. In addition to the diagnosis of disease, the network segmented important anatomical regions and detected other illnesses. The results show an accuracy

of 93% on the diagnosis, localization accuracy of the foci of 82% in normal lighted images, and 90% in fluorescein sodium eye drops.

Zhang *et al.* [254] proposed a system for diagnosing the malignancy of thyroid nodules on ultrasound with performance comparable with radiologists. The network provides predictions on medical concepts based on the TI-RADS lexicon.

The automatic generation of text reports based on medical imaging systems is also an active research area. Zhang *et al.* [258] presented a network trained on H&E patches for the malignancy diagnosis of bladder cancer and conditioned an RNN-based language model to generate text descriptions and visual attention (i.e. saliency maps) highlighting regions of the image relevant for specific parts of the text (Figure 2.12). MDNet [259] establishes a relationship between histological images of bladder cancer and diagnostic reports to generate text descriptions and provide visual attention to specific parts of the text.

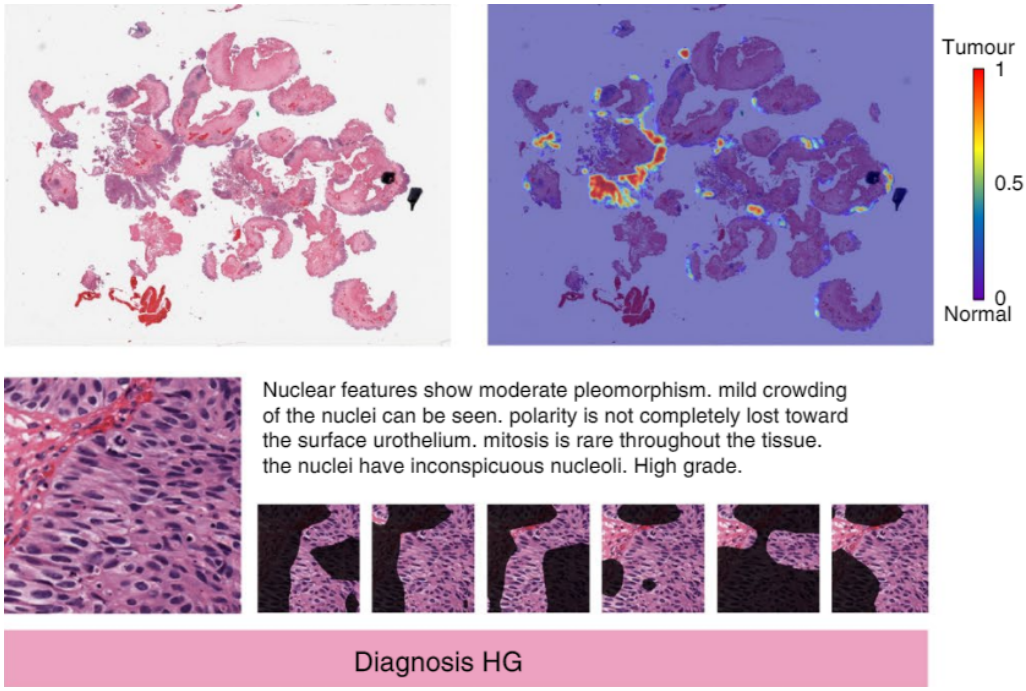


Figure 2.12: Left: H&E stained whole-slide tissue image. Right: saliency map generated [258]. Bottom: description generated for the image and feature-aware attention maps.

2.4 Open Issues and Promising Research Directions

As DL grows in popularity, so does the need for interpretability in the dichotomy between ML and medical practice. From this survey, it becomes clear that are four main issues that need more attention: (1) limitation on the applications of interpretability methods; (2) limitation on medical tasks explored; (3) lack of reliability of some interpretability

methods; and (4) lack of evaluation metrics for interpretability methods. Throughout this section, we will provide a discussion on the former four issues.

2.4.1 Limitation on the applications of interpretability methods

Du *et al.* [71] classified three major applications of interpretability strategies: model validation, model debugging, and knowledge discovery.

Model validation verifies that the model was able to learn useful knowledge and avoid learning biased information. The majority of works reviewed follow in this category, for example, works that explored the use of saliency maps mainly focused on verifying that the region highlighted corresponded to regions segmented by experts.

Other applications for the interpretation of deep learning models, such as model debugging and knowledge discovery, were overlooked by the current literature and constitute promising directions to further improve the diagnostic capabilities of models and discover new insights into the biology of different cancer diseases.

Model debugging aims at analyzing what leads to the misbehavior of models and erroneous predictions. Interpretability can help to uncover the reason for this misbehavior, by inspecting the examples that were misclassified by the model, examples that have artifacts from the data collection (e.g. metal tools in a CT scan, hairs in a dermoscopic image), in addition to difficult to diagnose cases. Model debugging is also extremely relevant when generalizing the model for other hospital data or for clinical use where the risk for misbehavior is much bigger. This application is still overlooked in current works in the field of oncology.

Carlini *et al.* [46] demonstrated that standard models can make perfect predictions in random training sets while performing poorly on the test set. This proves the model's ability to memorize the input data even if it is random which causes low generalization to unseen data. The lack of generalization of models which can be caused by overfitting to the training dataset must be an active concern of all ML practitioners, especially deep learning techniques as the high complexity of the models coupled with a low data size increases the risk of overfitting.

Another issue related to model debugging is adversarial attacks which consist of inputs that are intentionally crafted to force the model to make a mistake. Finlayson [77] demonstrated how an adversarial noise added to a dermoscopic image previously diagnosed as benign with over 99% confidence by a highly accurate model resulted in the model predicting malignant with 100% of confidence even though the difference is imperceptible to the human eye. Finlayson [77] also pointed at insurance claims approvals as a possible motivation for adversarial attacks.

Another problem with generalizability is discriminatory bias where models learn unin-

tended associations regarding minority subgroups due to bias in the data used to train the model [116]. An example is how malignancy diagnosis systems with accuracy similar to that of board-certified dermatologists under-performed on images of lesions in the skin of color due to the majority of training examples representing fair-skinned patients [116].

Discriminatory bias is not the only type of bias that can cause problems as there have been several instances where exceptional results have been obtained from the model learning to distinguish slides based on the hospital they came from or the clinicians that generated the ground truth rather than actual evidence in the slide [116]. For example, a system for the detection of pneumonia on chest x-rays was able to learn to associate the use of a portable x-ray machine with pneumonia [250].

Knowledge discovery allows physicians and researchers to obtain new insights into the physiology of the disease by interpreting the deep learning model and its decision process, such as finding that HER2 receptor over-expression is related to breast cancer. Knowledge discovery could lead to finding other receptors to help in the characterization of cancer diseases that are still unknown to this date. While some visualization methods have been used to discover clusters of patients with specific characteristics [113, 139], this direction of research is still mostly unexplored.

2.4.2 Limitation on medical tasks explored

Analysis of the results of the review (Figure 2.10) shows that 72% of works focus on some type of disease classification (45% malignancy diagnosis, 18% disease diagnosis, 9% diagnosis of skin lesion). This shows a great imbalance as there exists many more medical tasks in the oncology field with promising results but still lack interpretability. In the following sub-section relevant work on other medical tasks will be briefly reviewed. Those medical tasks are:

- Tumor or lesion segmentation: identify the set of voxels which make up the lesions or tumors present [103, 148];
- Organ and substructure segmentation: identify the set of voxels that make up either the contour or the interior of the objects of interest [119];
- Cancer prognosis: estimate the likely course and outcome of a disease [243, 263];
- Radiation treatment planning: determine location and dosage to deliver the most desirable dose distribution of radiotherapy [137];
- Image registration: seeks to determine a transformation that will map two volumes (source and reference) to the same coordinate system [83];
- Image generation and enhancement: includes many different tasks to improve the quality of the input from removing obstructing artifacts or noise in images to completing missing data [52, 244, 255].

Tumor and lesion segmentation is an important first step for numerous other tasks such as diagnosis and treatment planning, in order to evaluate the extent of the diseased tissue. DL techniques have achieved state-of-the-art results in brain tumor segmentation from MRI scans [103, 148]. The same type of networks have also been used in the segmentation of different lesions of the skin based on dermoscopic images [3, 129].

Segmentation of organs and substructures is also a critical step before radiotherapy in order to decide which regions to avoid targeting with radiation. One example of it is the segmentation of organs from abdominal CT scans [119].

Cancer prognosis is comprised of a large number of sub-tasks such as survival prediction and prediction of the likelihood of metastases. Zhu *et al.* [263] for example, reviewed a large number of studies that applied DL techniques to different cancer prognoses tasks such as cancer recurrence, progression, and survival prediction [263]. Other studies focused on sub-tasks that concern the progression of the disease after treatment, from the prediction of future distant metastases and local-regional recurrence using pre-treatment, post-treatment, and follow-up medical imaging scans [243].

Radiation treatment planning requires not only the segmentation of diseased tissue but also the dosage that should be used. A CNN-based model was used to MRI to accurately transfer contrast into CT images with clearly identified air, brain soft tissue, and bone highly similar to that of current methods based on CT and used in medical practice [137].

Image registration, also known as image fusion, is commonly used to combine two modalities - for example, PET-CT is obtained by combining two different modalities (PET and CT), but also multiple images of the same modalities. Fu *et al.* [83] reviews a large number of DL techniques proposed for the image registration of different modalities such as T1 and T2 MRIs and MRI and CT.

In addition, DL approaches also have seen success in restoring medical images corrupted with noise or artifacts, but the interpretation of the reasoning behind this process has also been pointed out as a challenge [252]. The extensive use of CT in medical analysis has raise some concerns due to the large dose of radiation that it delivers to the patient. Low-dose CT is a solution for this problem, but by using lower radiation amounts, noise, and artifacts become a problem. DL techniques have been proposed to reconstruct low-dose CT images and recover from noise and streaking artifacts caused by metal objects [52, 244, 255].

Even though DL techniques have helped the numerous problems pointed out above, they all face the same obstacle which prevents their use in clinical practice, the lack of interpretability. Future research efforts should then be targeted in the exploration of other applications of interpretability methods other than model validation and different cancer tasks than disease diagnosis. With the expansion of cancer applications, other interpretability strategies will emerge based on images (the most used modality) and other modalities that may be more associated with other problems.

2.4.3 Lack of reliability of some interpretability methods

Some post-hoc interpretability methods can present bias [2, 122] and might not be representative of the behavior of the model they are trying to explain [64]. This happens because although explanations should approximate as much as possible the actual behavior of the model, during the process of optimization (e.g. backpropagation) some inputs given to the network are outside the distribution of the training data and can trigger artifacts of the deep learning model.

As different interpretation methods sometimes focus on distinct aspects of the model [255], a promising direction to improve the reliability of the interpretations is to deploy an ensemble of complementary interpretability methods. Furthermore, interpretability methods should also be provided with imperfect data (i.e. noisy) to guarantee robustness to noise.

2.4.4 Lack of evaluation metrics for interpretability methods

To quantitatively evaluate an interpretability method without the validation of an expert requires a formal definition of interpretability and the use of a proxy metric describing the quality of the explanation [70]. The lack of ground-truth explanations, for example, the expert annotated tumor segmentations which indicated what the expected value of a saliency map should be, makes it difficult to make a quantitative analysis of the results and generalize the obtained results. One of the possible solutions to solve this issue is to conduct a comparative study between the interpretation produced by the deep model and one produced by a set of physicians. However, once again, this solution may not be generalizable, hence most studies conduct the evaluation by letting experts (e.g. pathologists) compare the explanations of a few selected examples and their domain knowledge.

Future research should help find interpretability metrics able to assess methods based on three factors. First, evaluate how faithful the explanations are to the actual model's behavior. Second, evaluate how easily the explanations are understood by the physician. Third, evaluate the usefulness of the explanation of its target application (i.e. model validation). Only by evaluating these factors can explanations extracted from deep learning models be truly trusted and applied in clinical practice.

2.5 Conclusions

Interpretability of deep learning is a growing field with mostly open problems and many opportunities for the field of medicine and oncology.

The lack of interpretability in deep learning has been pointed out as a major problem by many researchers that have studied the application of deep learning in various areas of

medicine and bioinformatics [43, 183, 230].

In this chapter, we presented an overview of various deep learning techniques and illustrated how the decisions of these could be interpreted with self-explanatory oncological cases to better illustrate. We also review the related research on the application of interpretability methods for cancer diseases, summarizing their main conclusions.

Overall, a high number of studies focused on breast, skin, and brain cancers (60%) and on the explanation of the importance of tumor characteristics like tumor dimensions and shape, in the prediction of decision systems. The majority of DL techniques interpreted were multilayer perceptrons and convolutional neural networks, often used to predict based on raw images or handcrafted features extracted from them.

As discussed in the previous section, three main issues were identified: (1) limitation on the applications of interpretability methods; (2) lack of reliability of some interpretability methods; and (3) lack of evaluation metrics for interpretability methods.

Future research should go beyond model validation and apply interpretability to understand how models misbehave, as well as discover new knowledge about different cancer diseases. Also, although DL has been successful in many cancer tasks (e.g. tumor segmentation, cancer prognosis, and image registration), works aim at interpreting models on these tasks remain unexplored. Lastly, future research in the design of evaluation metrics and frameworks is mandatory to assess the reliability of AI systems and for increasing the trust to be used in clinical practice.

These conclusions motivated the work presented in Chapter 5 and Chapter 6 on the development of evaluation metrics and evaluation approaches for interpretability methods.

Chapter 3

A global taxonomy of interpretable AI

The last decade saw a sharp increase in research articles concerning interpretability for Artificial Intelligence (AI), also referred to as eXplainable AI (XAI). In 2020, the number of articles containing "interpretable AI", "explainable AI", "XAI", "explainability", or "interpretability" has increased to more than three times that of 2010, following the trend shown in Figure 3.1.

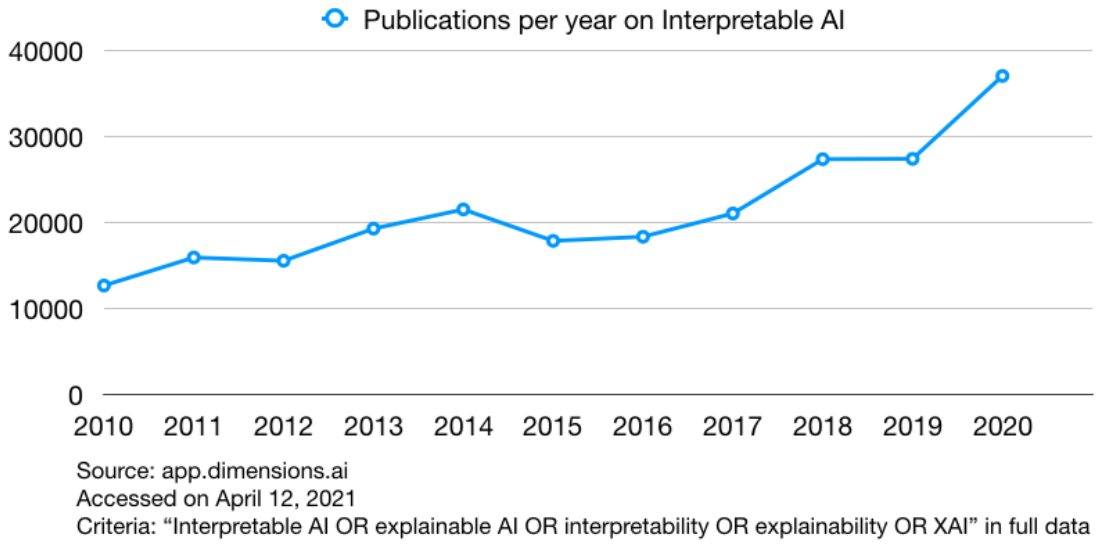


Figure 3.1: Trends of the publications containing "interpretable AI" or "explainable AI" as keywords

Being applied to an increasingly large number of applications and domains, AI solutions mostly divide into the two approaches illustrated in Figure 3.2. On the one side, we have *Symbolic AI*, symbolic reasoning on knowledge bases as an important element of automated intelligent agents, which reflect the humans' social constructs into the virtual

world [195]. To communicate intuitions and results, humans (henceforth agents) tend to construct and share rational explanations, which are means to match intuitive and analytical cognition [169]. On the other side, Machine Learning (ML) and Deep Learning (DL) techniques reach high performance by learning from the data and through experience. The complexity of the tasks in both approaches has increased over time, together with the complexity of the models being used and their opacity. A rising interest in interpretability came with the increasing opacity of the systems and with the frequent adoption of "black-box" techniques such as DL, as documented by multiple studies [1, 19, 20, 53, 135, 146, 149, 157, 193, 221].

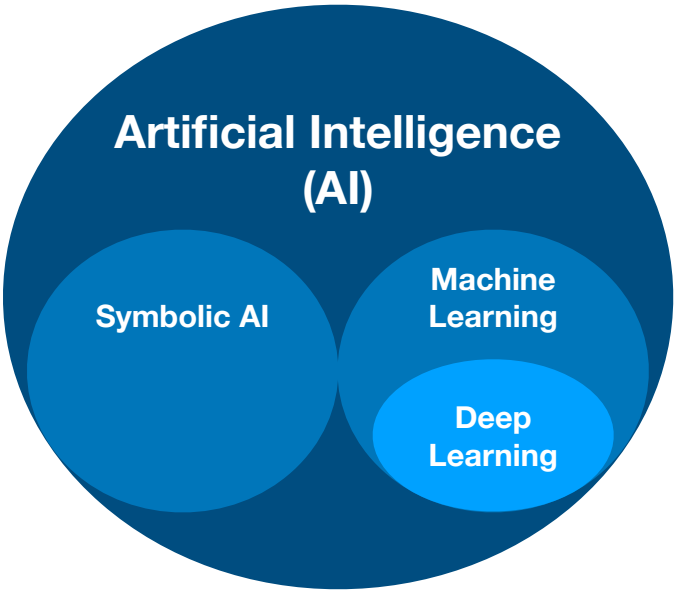


Figure 3.2: Graphical representation of Artificial Intelligence, Machine Learning, and Deep Learning adapted from `https://www.intel.com`.

A strong condition to ensure the reliable use of AI is improving the understanding of its internal mechanics, particularly when complex DL models are deployed. As the previous studies on interpretability point out, understanding the decision-making of an AI system is a non-trivial task that spans three areas, namely understanding the task, the performance metric used by the model, and the type of experience being used. With the intent of improving interpretability within these three areas, a large number of requirements, tools, and techniques have been developed in different application fields, leading to inconsistent use of the terminology. Interpretability is often confused with more abstract notions of fairness, privacy, and transparency [238]. These terms do not have a clear and unique definition and the understanding of these terms may differ depending on the domain and context. Similarly, the words interpretable and explainable have been used interchangeably in some articles [135, 146], while others use a strong distinction between the two terms [193]. Undoubtedly, there is a link between the act of interpreting and that of explaining, as shown by the etymology of the words themselves (that we report in Table 3.3). Interpretability has been presented as "explaining or presenting in under-

standable terms to a human", "providing explanations" to humans [146] and "assigning meaning to an explanation" [172]. For [193], however, there is a strong distinction between *interpreting* and *explaining* since models may be developed to directly encompass the ability to explain their decision-making. In this case, interpretability refers to meeting the transparency requirement at the task definition level, whereas explanation refers to a post-hoc (after training) evaluation of the model understandability.

The different perspectives about the technical terminology are discussed in several articles within the specific context of explainable AI and ML design, finding difficult integration within the other domains that are driving and shaping AI development. Policies for funding and regulating AI research also refer to concepts such as *transparency*, *explicability*, *reliability*, *informed consent*, *accountability*, and *auditability* of the systems [35, 36, 73]. Clarifying what these terms refer to and unifying the social and technical perspectives on these aspects is fundamental to determining directions for progress and encouraging cross-disciplinary discussion and interaction on AI developments. Fields that analyzed the impact of technologies over the centuries such as cognitive sciences, sociology, philosophy, and ethics constitute invaluable resources of knowledge from which it is possible to evaluate and understand how human trust evolves over time and how it can be built to motivate the adoption of new technologies. If the use of a global terminology is adopted by these disciplines, then a broader range of possibilities can open, encouraging the design of interpretability tools that are not only useful and understandable to ML developers but to a wider audience ranging from the final decision-maker to anyone affected by this decision [225].

In this chapter, we present a taxonomy and interdisciplinary definitions for interpretability and interpretable AI that can be used in multiple contexts which came to fruition from the collaboration of researchers from 8 disciplines in the social and technical sciences. Also, we propose the study of a use case in the medical field to demonstrate the relevance of unifying perspectives and adopting a common terminology.

3.1 Background

Several articles in the literature proposed a taxonomy of interpretable AI. Table 3.1 reviews in chronological order the numerous definitions that were given in the ML literature for *interpretable*, *explainable*, *transparent*, *decomposable* and *intelligible*. While trying to be as complete as possible, we clarify that this table is not exhaustive. We excluded from this review the articles that defined the taxonomy for developing a single technique. Discordance can be noticed on the meaning assigned to the terms by the articles in this collection, with major dividing points emerging on the words: (i) interpretable and explainable; (ii) transparency and decomposability ; (iii) intelligible and interpretable;

The terms interpretable and explainable are equated, for example, by several researchers [1, 20, 57, 146, 157, 231]. An even broader number of articles describes a clear distinction between these two terms [19, 37, 53, 135, 149, 154, 172, 193], suggesting that a distinction between these two terms is more popular among researchers. As for interpretability, multiple definitions exists also within the context of explainability, for which we refer the reader to the systematic review by [233]. The work by Arrieta et al. [19], for instance, distinguishes interpretability from explainability, which is defined as a human-understandable interface that exists between the user and the system. Transparency is used in multiple articles with the meaning described by Lipton in [135] of model decomposability [53, 57, 135]. In other articles, this term is used as a synonym for interpretability [19, 157] or for functional understanding of the model [149]. Rudin et al. [193] define transparency as models with particular properties such as monotonicity since these models are transparent in the way their variables are jointly related. Finally, the concept of intelligible model equated to that of an inherently interpretable model in [20], while it is used meaning the introduction of interpretability constraints in the model design in [57, 154].

None of the articles in Table 3.1 considers the taxonomy used by policymakers, regulators, philosophers, and sociologists discussing the impact of AI on society and on the research community. The perspectives in these articles are discussed by experts in AI development and familiarity with ML. As a consequence, different definitions are used in social sciences. We review the existing definitions and gather the perspectives from a multidisciplinary pool of experts to provide a taxonomy that can be used in multiple domains in a unique way that adapts to both the social and the technical sciences.

3.2 Methods

A round table public meeting was held online on April 29th, 2021 on "A Global Taxonomy for Interpretable AI"[1]. Endorsed by the AI4Media project within the European Union's Horizon 2020 for research and innovation plan, this event was organized to bring together researchers from multidisciplinary backgrounds to collaborate on a global definition of interpretability that may be used with high versatility in the documentation of social, cognitive, philosophical, ethical and legal concerns about AI. A total of 18 experts were invited to participate in the event. The selection of the experts was tailored to obtain the most representative consortium of the fields dealing with Interpretable AI at the moment. The final pool of experts involved in this work also depended on the experts' interests and their availability but the selection was by no means at all made in such a way to steer the discussion in the direction of a pre-agreed consensus. The experts were both internal members of the AI4media project and external non-affiliated members. The external experts were invited so as to obtain a balanced perspective on the topic that

[1] `https://taxonomyinterpretableai.wordpress.com/`, as of October 2021.

Table 3.1: Multiple Taxonomies - Part 1

Interpretable	**Explainable**	**Transparent**	**Intelligible**	**Ref.**
The system operations can be understood by a human, either through introspection or through a produced explanation.	To show the rationale behind each step in the decision. It is linked to justification and affects user acceptance and satisfaction.	Not mentioned.	Not mentioned, although they refer to introspective explanations.	[37]
Ability to explain or to present in understandable terms to a human.	Not mentioned.	Not mentioned.	Not mentioned.	[70]
A non-monolithic concept reflecting several distinct ideas.	Solely intended as post-hoc interpretability. Post-hoc explanations can be verbal, and visual.	Understanding the mechanism by which the model works. Related to simulatability and decomposability.	Understandable models are sometimes called transparent.	[135]
A mapping of an abstract concept into a domain that the human can make sense of.	Collection of features [...] that have contributed to producing a given decision.	Achievable by both interpreting and explaining ML outcomes	Post-hoc interpretability should be contrasted to incorporate interpretability into the structure of the model.	[154]
Used more frequently than "explainable" by the ML community, referring to a powerful tool for justifying AI-based decisions.	Not mentioned.	Not mentioned.	Understandability is characterized by no means of understanding the internal model functioning. Understandable is different from intelligible.	[1]
The level to which an agent gains and can make use of both the information embedded within explanations given by the system and the information provided by the system's transparency level.	The level to which a system can provide clarification for the cause of its decisions/outputs.	The level to which a system provides information about its internal workings or structure and the data it has been trained with.	Not mentioned.	[223]
Equated with "explainability", it defines the degree to which an observer can understand the cause of a decision."	Establishing an interaction between the explainer and the explainee (i.e. the subject on the receiving end of an explanation), that is contextual and selective, based on a small subset of causes.	Briefly mentioned as interlinked to trust.	Not mentioned.	[146]
Acknowledgment of multifaceted definitions from earlier studies.	Answering "why" and "why not" questions to improve the user's mental model of the system. In other cases, equated to interpretable.	Providing explanations on how the system works, clearly describing the model structure, equations, parameter values, and assumptions.	A system that is "clear enough to be understood". It is challenging to understand how an AI system should be defined in order to be "intelligible" since this would require the clarification of "complex computational processes to various types of users".	[57]
Broadly defined, referring to the extraction of relevant knowledge (visualization, language, or equation) about domain relationships contained in the data.	Used as a synonym of interpreting.	A feature engineering process to enhance the analysis of model interpretability.	Not mentioned.	[157]

Table 3.2: Multiple Taxonomies - Part 2

Interpretable	Explainable	Transparent	Intelligible	Ref.
Used interchangeably with explainable.	Post-hoc explanations involve an auxiliary method after a model is trained. Self-explaining models generate local explanations that may not be directly interpretable.	Not mentioned.	A "directly interpretable" model, namely intrinsically understandable by most consumers.	[20]
It is a domain-specific notion that does not allow a general-purpose definition. An interpretable ML model is constrained in model form so that it is either useful to someone, or obeys structural knowledge of the domain [...]	Possibly unreliable and misleading, explanations are not faithful to what the original model computes. Often, they do not make sense nor do they provide enough detail to understand what the black box is doing.	Fully transparent models are allowed to understand their variables and the related correlations.	Not mentioned.	[193]
It refers to the degree of human comprehensibility of a given black-box model or decision.	It refers to the numerous ways of exchanging information about a phenomenon (a model's functionality or the rationale and criteria for a decision) with multiple stakeholders.	A model is transparent if its functionality can be comprehended in its entirety by a person.	Not mentioned.	[149]
It is a passive characteristic of a model referring to the level at which it makes sense for a human observer (also referred to as transparency).	Any action or procedure to clarify the internal model functions.	As in Lipton, described by Simulability, Decomposability and Algorithmic Transparency.	Not mentioned. Understandable is different from intelligible.	[53]
It encompasses multiple concepts and definitions. Generally, it is associated with models with inherently interpretable behavior.	It is intended as the generation of post-hoc explanations for black-box models.	It is intended as an explanation of how the system works.	Not mentioned.	[19]
Assigning meaning to an explanation.	Process of describing one or more facts, facilitating the understanding of said facts by a human consumer.	Not mentioned.	Not mentioned.	[172]
Assigning a subjective meaning to a model, object, or variable that is possible to be interpreted by the explainee.	The activity of producing more interpretable objects manipulating symbolic information.	Providing a clear representation of the black-box dynamics.	Concerning the explainee, it is intended a successful consumption of an explanation.	[55]

went beyond the purpose of the project itself. For each of the discussed disciplines, at least one external expert was included in the discussion. The selection was done based on the previous publication records on interpretable AI and on the reported interest and availability to participate in the study.

The workshop was organized in two sessions, consisting of a round table discussion and a panel session with a question and answer format. The first session consisted of seven short talks of 12 minutes followed by 3 minutes for questions. The second session involved a panel of five experts discussing questions from the audience concerning the role and implications of AI and transparency. The workshop was streamed on YouTube[2] and spectators were able to interact with the audience through a live chat.

The round table resulted in a solid basis for this work and steered further discussion and proposed future research directions. We hope that this work may constitute a first solid step towards finding a global consensus on the taxonomy for interpretable AI for both the social and the technical sciences.

3.3 Results

3.3.1 Etymology and existing definitions

Table 3.3 analyzes the etymology of frequently used words in the context of interpretable AI. Looking at the historical formation and the original meaning of a word can shed light on its roots and history, deepening the understanding of its meaning and the context in which it should be used. The word clue, for example, gains meaning from its intrinsic referral to Greek mythology. It originates from the Germanic word clew which indicates a ball of thread or yarn. Theseus used a clue of thread to find the exit of the Labyrinth. When people say "give me a clue", they refer to some helpful information and not the ball of yarn itself. Understanding the etymology of the words in the AI interpretability terminology can help in a similar way to better understand the meaning of each term and why one word is more appropriate than another in specific contexts.

Figure 3.3 illustrates how some of the terms defined in Table 3.3 (such as intelligible, transparent, explainable, accountable, auditable and reliable) slightly change their meaning depending on the context, acquiring multiple shades and connotations as they interact with the different domains. This analysis, based on the cross-disciplinary knowledge of the people participating in the initiative, gives insights into how each domain envisions these concepts. Some conflicts in the definitions are shown as the words are used in one or another discipline. The attention towards one or more concepts is mostly heterogeneous, with some disciplines focusing more on one aspect than others. While heterogeneity in

[2] `https://www.youtube.com/watch?v=aVLCDORsqmo`, as of February 2022.

Table 3.3: Analysis of the etymology of the terms related to interpretability.

ID	Word	Etymology		ML Definition
1	Interpretability, Interpretable	From late Latin interpretabilitis from Latin interprĕtor, interprĕtāri (to interpret).	To interpret, comment, explain, expose, illustrate, to translate.	To translate, expose, and comment on the generation process of one or multiple ML systems outcomes, making the overall process understandable by a human.
2	Explainability, Explainable	From 1600 use of explain + -able adapted from Latin explāno, explānāre	To explain, clarify, expose, illustrate, state clearly	To indicate with precision, to illustrate what features or high-level concepts were used by the ML system to generate predictions for one or multiple inputs. In intelligent agent systems: possibly iterative process of symbolic knowledge manipulation to make it interpretable.
3	Transparency, Transparent	Medieval Latin adaptation of the words trans (on the other side) and pārĕo, pārēre (to appear, to show).	To see through.	A *transparent* ML system has a non-opaque output-generation process where the role of the individual components, the learned paradigms, and the overall behavior of the model are known and can be simulated by a human user.
4	Intelligibility, Intelligible	From Latin intellegibilis, intellegibilis, II class adjective.	To understand, comprehend, decipher.	An intelligible ML system is an understandable system with inherent interpretability
5	Accountability, Accountable.	From 1770 use of accountable + -ity, adapted from Old French acont derived from Latin compŭto, compŭtāre, which has multiple meanings including to count, to estimate, to judge and to believe.	Used from the 1610s with the sense of "rendering an account", meaning providing a statement answering for conduct.	An accountable ML system is expected to justify its outcomes and behavior
6	Reliability, Reliable	From Scottish of the 1560s "raliabill", derived from Old French relier a derivation of the Latin rĕlĭgo, rĕlĭgāre (meaning to tie, to bind).	From the 1570s used with the sense of to depend, to trust, typically used in the expression "to rely on something/someone".	To be consistently good and be worthy of trust
7	Auditability, Auditable	From Latin noun auditŭs, auditūs.	The sense of hearing, the act of hearing, audition. Used in the sense of official audience, judicial hearing or examination.	An "auditable" ML system should provide information on how to perform an official audience of the model. For example, this can be done by providing extra documentation and functionalities.
8	Liability, liable	From Anglo-French liable, derived from Latin lĭgo, lĭgāre (to tie, to bind).	Legal responsibility for acts.	Legal liability of a product implementing ML, particularly in the case where something goes wrong.
9	Robustness, Robust	From French robuste, derived from Latin robustus, robustum.	The literal meaning is oaken, made of oak. Used in the figurative sense of strong, vigorous and resistant.	Robust ML systems are resistant, secure and reliable. Providing consistent results also in case of adversarial attacks, variations in the dataset, domain shifts, and outliers.

the attention to the words is legitimate and given by the intrinsic nature of each discipline, the strong changes in the meaning assigned to the same word by different disciplines may inhibit understanding and collaboration among different fields. The word *transparent* has been interpreted as "providing meaningful information about the underlying logic" in the EU legislation, whereas by technical developers this is often understood as a certain degree of understanding of the system mechanics, decomposability and simulability. In other words, if technicians and legislators were to think of the degrees of transparency of a vehicle, they would see different aspects. The former would think of pistons, fusible and the combination of these elements to the final engine. The latter would think of the degree of information available to the user about the working principles of the vehicle: starting the engine, stopping it from running, changing the direction and so on.

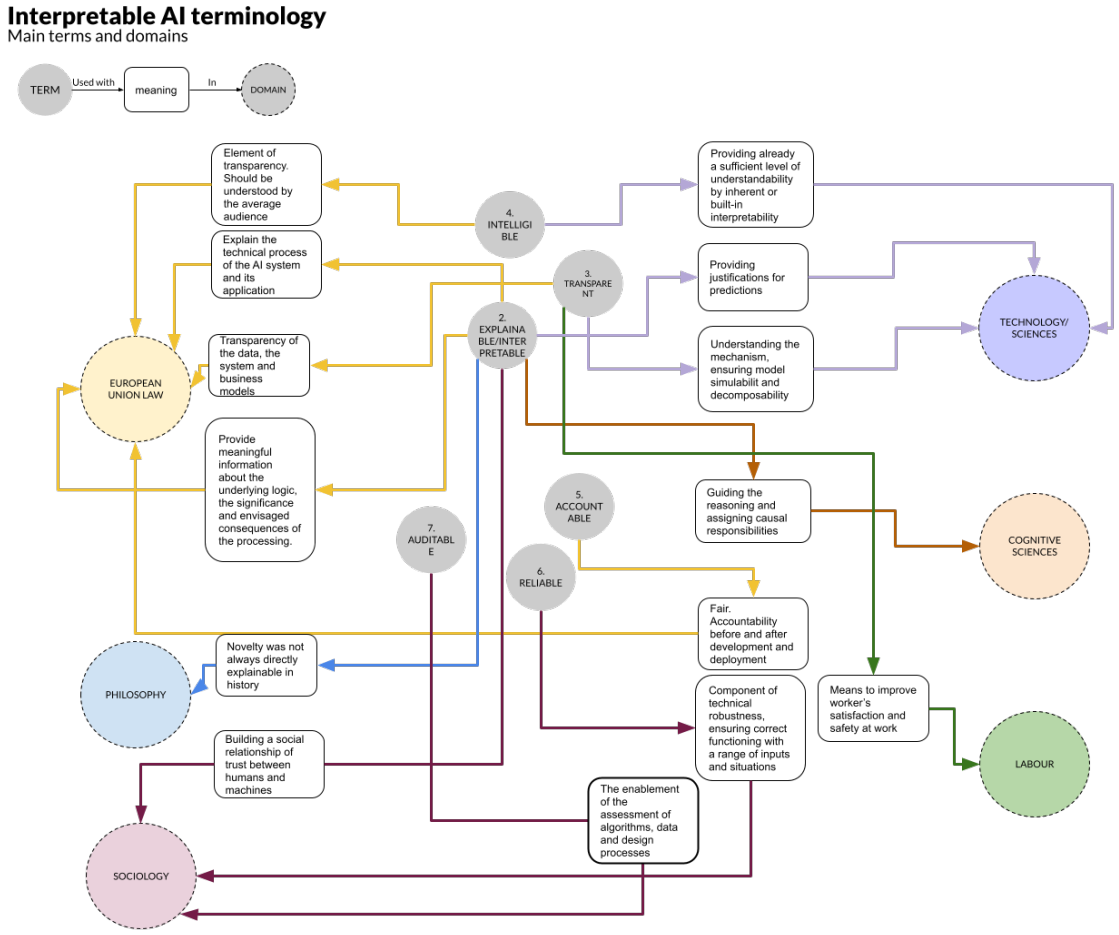


Figure 3.3: Differences of definitions in other domains than ML development. In this diagram, interpretable is equated to explainable since most of the social domains equate the two terms for simplicity.

3.3.2 A global definition of Interpretable AI

As an important contribution of this work, we derive a multidisciplinary definition of interpretable AI that may be adopted in both the social and the legal sciences.

In daily language, an instance, or an object of interest, is defined as interpretable if it is possible to find its interpretation, hence if we can find its meaning [211]. Interpretability can thus be conceived as the capability to characterize something as interpretable. A formal definition of interpretability exists in the field of mathematical logic, and it can be summarized as the possibility of interpreting, or translating, one formal theory into another while preserving the validity of each theorem in the original theory during the translation [220]. The translated theory as such assigns meaning to the original theory and it is an interpretation of it. The translation may be needed, for instance, to move into a simplified space where the original theory is easier to understand and can be presented in a different language.

From these explicit definitions, we can derive a multidisciplinary definition of interpretability that embraces both technical and social aspects: "Interpretability is the capability of assigning meaning to an instance by a translation that does not change its original validity". The definition of interpretable AI can then be derived by clarifying what should be translated: "**An AI system is interpretable if it is possible to translate its working principles and outcomes in human-understandable language without affecting the validity of the system**". This definition represents the shared goal that several technical approaches aim to obtain when applied to AI. In some cases, as we discuss in Sec. 4.4, the definition is relaxed to include approximations of the AI system that maintain its validity as much as possible. Interpretability is needed to make the output generation process of an AI system explainable and understandable to humans and it is often obtained as a translation process. Such a process may be introduced directly at the design stage as an additional task of the system. If not available by design, interpretability may be obtained by post-hoc explanations that aim at improving the understandability of how the outcome was generated. Interpretability can thus be sought through iterations and in multiple forms (e.g. graphical visualizations, natural language, or tabular data) which can be adapted to the receiver. This fosters the auditability and accountability of the system.

3.3.3 A global taxonomy

In what follows we present a global taxonomy for interpretable AI, and summarize the multiple viewpoints and perspectives gathered in this work. Table 3.4 presents the taxonomy with further detail on domain-specific definitions used in each of the eight fields studied in this work, namely law, ethics, cognitive psychology, machine learning, symbolic AI, sociology, labor rights, and healthcare research. Brackets specify the domain in which

Table 3.4: Taxonomy of Interpretable AI for the social and technical sciences. Brackets specify the domain in which each definition applies. Global marks a definition common to both the social and technical sciences.

Terminology	**Definition in AI**	**Family of AI systems (technical)**
Interpretability	(global) AI interpretability defines those AI systems for which it is possible to translate the working principles and outcomes in human-understandable language without affecting the validity of the system	Three families of AI systems may be identified by interpretable AI. These are (i) AI systems with built-in interpretability (ii) AI systems that are inherently interpretable (iii) AI systems that were explained by post-hoc methods. More details on these families in Table 3.5
	(EU law) AI interpretability defines the supply of meaningful information about the underlying logic, significance, and envisaged consequences of the AI system	-
	(symbolic AI) AI interpretability includes explanations of the symbolic AI systems in symbolic language	-
	(sociology) AI interpretability must define a social relationship of trust between the human and the machine	-
Interpretability by design	(global) The translation of the system's working principles and outcomes into human-understandable language is provided directly by the AI-system itself, interpretability being one of the tasks of the system	Two families of systems may be identified, namely (i) systems with a transparent design (e.g. introducing parameter sparsity, implementing monotonic functions [159]) (ii) systems with a self-explanatory objective that generate explanations for the model predictions (e.g. interpretable decision sets [128]).
Post-hoc interpretability	(global) The AI system is neither inherently interpretable nor interpretable by-design, rather additional analyses are performed to generate explanations without re-training the model parameters	Six families of post-hoc interpretability methods can be identified based on the form of the generated explanations into (i) feature attribution (ii) feature visualization (iii) concept attribution (iv) surrogate explanations (v) case-based explanations and (vi) textual explanations. For further details on these categories, we refer the reader to [19] and [91]
Local interpretability	(technical) Local interpretability is provided when interpretability analysis is performed on the system's outcome for a single input	The family of feature attribution methods contains several approaches that provide local interpretability [131, 142, 153, 185, 202, 209, 216, 261]
Global interpretability	(technical) Global interpretability is provided when interpretability analysis is performed to explain the system behavior for a set of inputs corresponding to an entire class or multiple classes	Post-hoc interpretability methods may provide global interpretability, such as distillation techniques [82] and the extraction of rule lists[49]
Explainability	(global) Explainable AI, also denoted as XAI, defines the branch of AI research that focuses on generating explanations for complex AI systems	The six families of post-hoc interpretability methods known as feature attribution, feature visualization, concept attribution, surrogate, case-based and textual explanations are addressed as explainable AI.
Transparency	(global) Transparency is used in AI to characterize those systems for which the role of internal components, paradigms and overall behavior is known and can be simulated	The family of linear regression models and decision trees in low dimension are transparent and can be simulated

each definition applies. If a term applies to both social and technical experts it is provided first and marked by the (global) identifier. Otherwise, it is marked as the domain-specific identified, i.e. EU law, sociology, etc. This table may be resorted to by practitioners in any of the above-mentioned fields to obtain a common definition for each term in the taxonomy and to inspect all the exceptions and variations of the same term in the literature. Our objective is not to impose one taxonomy above another, rather to raise awareness on the multiple definitions of each word in each domain, and to create a common terminology that researchers may refer to in order to reduce misinterpretations.

The following subsections explain how the proposed taxonomy adapts to the fields with their respective needs, challenges, and goals in terms of ML interpretability.

3.3.4 Use of the proposed terminology to classify interpretability techniques

In this section, we show how the terminology in Table 3.3 can be used to classify ML interpretability techniques. To do so, we group popular interpretability techniques into the families shown in Table 3.5. On the basis of this, Table 3.6 summarizes how each family of techniques can provide the properties described in Table 3.3. In the following, we give more insights concerning the classifications provided in Tables 3.5 and 3.6.

Due to their low complexity, models such as decision trees and sparse linear models have inherent interpretability, meaning they can be interpreted without the use of additional interpretability techniques [152]. These methods are intelligible, according to the definition in Table 3.3 ID 4. Black-box models, such as deep learning models, have surpassed the performance of traditional systems over complex problems such as image classification. However, due to their high complexity, they require techniques to interpret their decisions and behavior. These techniques often involve considering a close approximation of the model behavior that may be true in the locality of an instance (i.e. local interpretability) or for the entire set of inputs (i.e. global interpretability). They can be grouped according to the following criteria: (1) scope, (2) model-agnostic, and (3) result of explanation.

The *scope* of the technique shows the granularity of the decisions that are allowed as explanations, either global or local. *Global* interpretability techniques explain the behavior of the system as a whole, answering the question "How does the model make predictions?", while *local* interpretability techniques explain an individual or group of predictions, answering the question "How did the model make a certain prediction or a group of predictions?" [135].

Model-agnostic techniques can be applied to any model class to extract explanations, unlike model-specific techniques that are restricted to a specific model class. Interpretability techniques can also be roughly divided by their result or the type of explanation they produce, creating multiple families of techniques. It is important to note that some types of

Table 3.5: Definitions of families of interpretability techniques

Scope	Family	Definition
Inherent Interpretability	Interpretable Model	Models that are considered interpretable due to their low complexity and simple structure.
	Black-box Model	Models that are considered hard to interpret due to their high complexity and complicated structure.
Global Interpretability	Feature Visualization [161, 168]	Synthetization of new instances that help visualize features learned by the model or a specific part of the model.
	Prototype, Criticism [117]	A prototype is a data instance that is representative of all the data. A criticism is a data instance that is not well represented by the set of prototypes.
	Influential Instances [123]	Data instances of which the removal has a strong effect on the trained model.
	Dependency Plot	Depicts the functional relationship between a small number of input variables and predictions.
	Global Surrogate [108]	Interpretable model that is trained to approximate the predictions of a black-box model.
	Concept Attribution [92, 118]	Explain the model's behavior based on user-friendly concepts.
	Feature Importance [142]	Assigns a score to input features based on how useful they are at predicting a target variable.
Local Interpretability	Local Surrogate [185]	Local surrogate models are interpretable models that are used to explain individual predictions of black-box models.
	Saliency Map [131, 202]	Highlight the pixels that were relevant for a certain image prediction.
	Counterfactual Example [235]	A counterfactual explanation of a prediction describes the smallest change to the feature values that changes the prediction to a predefined output.
	Adversarial Example [88]	An adversarial example is an instance with small, intentional feature perturbations that cause a ML model to make a false prediction.

explanations are strongly preferred, as half the studies using interpretability techniques in the oncological field use either saliency maps or feature importance [12]. These techniques can produce data points that explain the behavior of the model [117, 131], visualizations of internal features [168] or produce simpler models that approximate the model [128, 142, 185]. It is important to choose the right technique based on its scope and family to reach the desired objective. Table 3.5 presents the families of techniques, their definitions, and important references [152].

Based on Tables 3.1, 3.2 and 3.4 we present Table 3.6 where we group families of interpretability techniques based on their scope and classify them based on their suitability to achieve each of the objectives mentioned in Tables 3.1 and 3.2. To achieve interpretability as intended in Table 3.3 (ID 1), local techniques are preferable since they allow users to interpret the outcomes of a system and thus increase its interpretability. Global techniques can be rather inaccurate at a local level, although they are more adequate to expose the mechanisms of a system in general. The decision-making process can become more transparent (ID 3) at the local or global level, depending on the scope of the interpretability techniques. Intelligibility (ID 4) is a characteristic of inherently interpretable models. It can be achieved for more complex models by approximating the decision function either

Table 3.6: Classification of families of interpretability techniques

Scope	Family	Interpretability	Explainability	Transparency	Intelligibility	Accountability	Auditability	Robustness
Inherent Interpretability	Interpretable Models	x	x	x	x	x	x	x
	Black-box Models	-	-	-	-	-	-	-
Global Interpretability	Feature Visualization	x	-	x	-	x	x	-
	Prototypes and Criticisms	x	-	x	-	x	x	x
	Influential Instances	x	x	x	-	x	x	x
	Dependency Plot	-	x	x	-	x	x	-
	Global Surrogate	x	x	x	x	x	x	-
	Concept Attribution	x	x	x	-	x	x	-
	Feature Importance	-	x	x	-	x	x	-
Local Interpretability	Local Surrogate	-	x	x	x	x	x	-
	Saliency Map	-	x	x	x	-	x	-
	Counterfactual Example	-	x	x	-	x	x	-
	Adversarial Example	-	-	-	-	-	x	x

locally or globally with an inherent interpretable model. It is also important to point out that even with the model being inherently interpretable, sometimes the features being used to train the models can be hard to understand, particularly for non-experts in feature engineering.

As for accountability, systems would need to justify their outcomes and behavior to be accountable, and thus the techniques that offer any interpretability or explainability can help to achieve this. Similarly, these techniques can also be used to examine the global behavior or reasoning of local decisions and provide auditability (ID 7). Finally, Robustness (ID 9) is not achievable by only understanding the behavior of the model. It would rather require finding or producing instances that make the model misbehave, limitations of the model, or data points that are outside the training data distribution.

At this point, we remark that interpretability techniques come with inherent risks. A desired property of interpretability is to help the end-user with creating the right mental model of an AI system. However, if one considers AI models to be lossy compression of data, then interpretability outcomes are a lossy compression of the model and are severely *underspecified*. In other words, it is possible to generate several different interpretations for the same observations. If used improperly, interpretability techniques can open new sources of risk. In some settings, interpretability outcomes can be arbitrarily changed. For example, [6] demonstrate a case of "fair washing", where fair rules can be obtained that represent an underlying unfair model. It is also possible for an AI system that predicts grades to be gamed if the underlying logic is fully transparent. Model explanations can demonstrate an AI model criterion to be illegal or provide grounds for appeals [238].

Finally, transparency also conveys trade-offs involved in decisions in an explicit manner that may otherwise be hidden [60].

From these considerations, it follows that interpretability requires a context-based scientific evaluation. Two standard approaches for such evaluations are (a) to establish baselines based on domain insights to evaluate the quality of explanations, and (b) to leverage end-user studies to determine effectiveness. For instance, user experiments have been used for trust calibration (knowing when and when not to trust AI outputs) in joint decision-making [256]. In another interesting approach, [128] measured the teaching performance of end-users in establishing how effective explanations are in communicating model behavior with good teaching performance indicating better model understanding.

Several quantitative measures to assess explanation risks have also been proposed in the literature. A common measure using surrogates involves approximating a complex model with a simpler interpretable one. Properties of the simpler model can then help address questions on the extent of interpretability of the original model. Common measures include *fidelity*, the fraction of time the simpler model agrees with the complex one, or *complexity*, the number of elements in the simpler model a user needs to parse to understand an outcome. *Faithfulness* metrics measure the correlation between feature importance as deemed by an AI model versus deemed by an explanation. *Sensitivity* measures [246] the degree to which explanations are impacted by non-trivial perturbations.

3.3.5 Terminology in the cognitive sciences

From the point of view of the cognitive sciences, interpretability (as defined in line 1 of Table 3.3), is considered part of the social interaction between an AI system and a user [107]. As the definition underlines, the concept of interpretability is strictly connected to the human ability to understand information. The process of understanding is defined in cognitive psychology as the ability of the human brain to infer or make predictions in semantic memory. Semantic memory is wired by connections of neurons that are created and consolidated by positive enforcement. A high-level model of such neural connections identifies areas that are specialized for reacting to specific stimuli (e.g. numbers, words, shapes, colors, actions, sounds). Depending on what kind of information is being understood, these areas may be used individually or share functions [237]. The understandability of something is thus the property of an object, may this be a model or the outcome of interpretability methods, to be understood by a human. Because the wiring of the neurons constituting the areas in the semantic memory is a result of individual experiences, understandability incorporates some degree of subjectivity and variability, e.g. what is understandable to someone may not be understandable to someone else. Users may vary greatly, so may their background and understanding of explanations. Thus to be widely applicable and useful to a variety of users, understandability shall not require any prior training of the addressees concerning the feature extraction, hyper-parameter

selection and training of AI systems.

Some aspects of human explanation generation (i.e. explainability as in ID 2 Table 3.3) do not coincide directly with what is intuitively thought about as transparency (ID 3 in Table 3.3). The first difference is that explanations are selected by humans. The selection is generally biased to reflect the mental model of the explainee. Even having a complete set of causal relations, people are more likely to rely on a few causes that may explain certain key aspects of the event [106]. It may at this point be noted that explainability should thus be intended differently from transparency, that is rather the unbiased provision of insights about the internal mechanics of an AI system.

3.3.6 Social and working environment

To develop a social relationship between humans and machines, interpretability needs to act as a social contract of trust between these two parties. Trust in the system leads to reliability (as intended in ID 6 of Table 3.3) and this can only be built through sustained understanding. Using understanding to build trust is a well-understood social science research problem, complicated by the fact that humans accept explanations first and foremost in a highly biased manner [141]. The fact that bias is part of every human understanding, however, should not limit the potential success of explainable AI. For this reason, AI explainability (ID 2 in Table 3.3) should be seen as a social translation, as investigated in recent studies in HCI like [115]. If only computer scientists are considered within the project ideation and development, however, there is the main risk, discussed by T. Miller in [147], of having the helpless being led by the clueless[3], namely having ML engineers building explainability mostly for other ML engineers. Social scientists and workers should be introduced in the analyses proposed by ML researchers, as the actual addressee and users of the algorithms. Collaborations should be built to develop types of human-computer interactions in ML that are more understandable to non-ML experts. If interpretability is not developed with the help of the social sciences, the risk of creating AI systems mainly for other researchers is high and it would undermine the efforts in building reliable and trustworthy automated systems.

AI may not be developed with the only intent of prioritizing the reduction of human input, as this may lead to the perception of AI as "inhuman" intelligence [69]. New algorithms should prioritize the creation of a relationship of trust above the desire to automate and reduce human input.

Within the realm of employment relations, work, and labor markets, the concept of "democracy at work" is generating into the discussion of the criteria for AI transparency (as defined in Table 3.3 ID 3). Of particular importance are the employees' rights of participation and consultation if AI algorithms are employed to make decisions at the workplace.

[3]In the original article, this problem is formulated as that of "the inmates running the asylum".

Employees should be guaranteed the possibility to get involved in management decisions about the organization of work and of working conditions. Democracy is thus essential to let the employees create optimal conditions for work and it translates into the need of transparency if AI systems are used to manage the working personnel. In particular, the workers' autonomy (the right of a worker to intervene), skill grading, and the ruling of organization and production processes should be regulated by transparent AI decisions. Transparency is thus desired to decide whether an algorithm is performing non-democratic practices, such as discrimination. It is thus intended in the sense of a means to improve the worker's satisfaction and safety at work (see Figure 3.3). Even further, it may help to identify the workplace conditions enabling discrimination in the first place.

3.3.7 The EU law on interpretability

In law, there is no precise definition of AI explainability. The High-Level Expert Group on AI (AI HLEG) set up by the European Commission lists *explicability*[4] as one of the ethical principles that must be respected in order to ensure that AI systems are developed, deployed and used in a trustworthy manner. The principle of explicability encompasses both the terms of transparency and explainability as defined in Table 3.3. From a legal point of view, explainability is seen as collecting meaningful insights on how a particular decision is made [36]. According to [36], it does not set the requirement for an interpretable representation of a mathematical model. Most important is that the explanation should assign meaning to the decision, i.e. so that the decision improves the explainee's understanding[5] of the decision generation process. It follows from the AI HLEG Guidelines that explainability should be adapted to the level of expertise and understanding of the individual concerned. [35] argue that in private decision-making, the legal requirements relate to the following four levels of ML explainability concepts: (i) providing the main features used for a decision, (ii) providing all features used for a decision, (iii) providing explanation on the way the features are combined to make the decision, and (iv) providing an understandable representation of the whole model. [234] propose the following categorization of what one may mean by an explanation of automated decision-making. Two kinds of explanations are possible, depending on whether one refers to: system functionality, i.e. the logic, significance, envisaged consequences, and general functionality of an automated decision-making system, e.g. the system's requirements specification, decision trees, pre-defined models, criteria, and classification structures; or to specific decisions, i.e. the rationale, reasons, and individual circumstances of a specific automated decision, e.g. the weighting of features, machine-defined case-specific decision rules, information about reference or profile groups. Furthermore, one can also distinguish between an ex-ante explanation (i.e. prior to the automated decision-making taking place) and an ex-post explanation (i.e. after the

[4] `https://www.europarl.europa.eu/cmsdata/196377/AI%20HLEG_Ethics%20Guidelines%20for%20Trustworthy%20AI.pdf`, as of February 2022.

[5] Intended in the scientific sense used in cognitive psychology (see Section 3.3.5).

automated decision has taken place) [234]. The focus of many legal scholars has been on the meaning of explainability from the data protection law point of view. The core debate has primarily focused on whether or not the General Data Protection Regulation 2016/679 (GDPR) creates a right to an explanation of an algorithmic decision, as argued by [89] and further discussed by [234]. The latter, in particular, argue that a non-existing "right to explanation" of a specific automated decision should not be mistaken with other GDPR provisions. The actual GDPR rather forms a "right to be informed" by claiming: (i) the right not to be subject to automated decision-making and safeguards enacted thereof (Article 22 and Recital 71); (ii) notification duties of data controllers (Articles 13–14 and Recitals 60–62); and (iii) the right to access (Article 15 and Recital 63). Others, like [201], point out that whether one uses the phrase "right to explanation" or not, data controllers need to provide the data subject with the "meaningful information about the logic involved, as well as the significance and the envisaged consequences of such processing for the data subject" (Article 13(2), 14(2), 15(1) of the GDPR). Such information must be meaningful to an individual confronted with a decision [201]. The test for whether the information is meaningful should therefore be functional - explanations are a means to help a data subject act rather than merely understand the mathematical processes behind decisions [73]. This is also in line with some of the claims done in the applicative domain at high-stakes, e.g. clinical decision-making [225].

Some scholars have studied how the legal requirements on explainability could be interpreted and applied to ML [36][100] used a COVID-19 use case scenario to assess the feasibility of legal requirements on algorithmic explanations. They concluded that the use of complex deep learning models in AI applications makes it hard to reconcile with the existing EU data protection law requirements, especially with regard to human legibility of explanations for non-expert data subjects. Similarly, [73] note that the legal concept of explanations as "meaningful information about the logic of processing" may not be provided by the kind of ML "explanations" computer scientists have developed. This further motivates the need to resort to a common ground where the objectives regarding interpretability can be discussed among the disciplines involved, for example on the basis of the taxonomy provided here. It is possible that in some cases transparency or explanation rights may be overrated or even irrelevant – the problem that is often referred to as *transparency fallacy*. In many cases what the data subject wants is not an explanation—but rather for the disclosure, decision, or action simply not to have occurred [73]. In high-risk AI systems, however, the recently proposed draft Regulation on AI (the AI Act) envisions transparency as one of the obligations of the operators. Article 13 of the draft AI Act requires high-risk AI systems to be "designed and developed in such a way to ensure that their operation is sufficiently transparent to enable users to interpret the system's output and use it appropriately." The obvious difference here, in comparison with the AI HLEG Guidelines, is that the transparency is addressed towards the users of the AI systems, that are not necessarily familiar with ML theory. This aligns with the requirement of person-

alized explanations discussed in Section 3.3.5 and contrasts with the current definition of transparency in the ML community where this property is rather intended as an objective peek inside the AI algorithm.

For AI systems that interact with natural persons, e.g. an emotion recognition system or a biometric categorization system and AI systems that generate deep fakes, the draft AI Act prescribes an obligation to inform or disclose the fact that they interact or are exposed to such systems. It is interesting that even though the draft AI Act does use the very term transparency, it does not refer to the explainability and the traceability dimension that were part of the concept according to the AI HLEG Guidelines. This shows the inconsistency of the terminology from a legal point of view. One obvious solution would be to amend the text of the regulation; if not, it would be subject to interpretation by the Court of Justice of the European Union that is likely to rely on other branches of science to complement the legal gaps, which shows the clear necessity of unified taxonomy.

3.3.8 An ethical point of view

The requirement of interpretability is often made on the basis of an analogy with human decision-making [59]. We expect bankers to explain why they reject a loan, physicians to explain why they discontinue treatment, and politicians to explain why they want to implement a certain policy. This requirement is often based on the idea of transparency: that seeing how a phenomenon happens generates accountability and the possibility of change [15]. The interpretation of phenomena in this sense derives from the epistemological concerns being debated since antiquity in philosophy. In the historical sense (in Table 3.3), interpreting has to do with understanding a particular course of action or decision-making and ethical concerns have to do with providing reasons for moral choices. Even prior to that, interpretation has been primarily a religious issue, namely concerning the interpretation of the holy scripture, which was supposed to transmit the word of God, in a way such that the true meaning of the text would be preserved.

Unlike other technologies, interpretation is one of the primary ethical concerns that are raised with the application of AI. While other technologies are also able to replace human functions (e.g., a walking stick takes over the function of a leg), AI is arguably the first technology that has the capacity to make decisions. And this raises both the epistemological question of *why* certain decisions were made by an AI system, as well as the ethical question of whether *good reasons* can be given for this decision, in case it is ethical significance.

What sets the ethical discussion apart from the technical perspective in Section 3.3.4, is its primary focus on the ethical value of an explanation, rather than in its epistemic value [189]. That is, a causal chain leading to the damage needs to be provided if an AI-generated decision may affect a human being.

As scholars have argued, however, human beings often do not need complete causal chains of explanation [59]. This opens up some new ethical issues and problems such as the intentional concealing of information, which may be obtained even by simply providing explanations of which the understandability is limited by the requirement of prior expert knowledge [15]. A patient might not be helped by a full causal explanation of a diagnosis but rather by a trustworthy account of understandable reasons expressed in clear and simple language.

From this perspective, we may raise three overarching ethical concerns of interpretable AI. First, there is the concern of "sacrifice". Because interpretation is always situated between the system and the user, it generates the inevitable risk of omission during interpretation. This can be due to either oversimplification (simplifying the model dynamics missing out on important technical details) or to over complexify (providing too technical explanations most users cannot grasp) [163]. Interpretation therefore inevitably sacrifices meaning. Second, we should be concerned about "hospitality", here intended as a common ground of understanding between strangers that aims to remedy the potential of conflict. Interpretation requires building bridges between different world visions, for instance between a physician and a patient, or a civil servant and a citizen. Third, interpretation raises the question of professional virtues. It is often part of a particular profession (a notary, a physician, a school teacher) to uphold certain standards of excellence in providing interpretability, for instance under the heading of the virtue of "fidelity". Importantly, what these standards mean in practice can differ significantly between different professional contexts.

In light of the above three (and other) ethical challenges, researchers have to consider how the ethical interpretability of AI systems should be realized in practice. Often, this requires finding ways in which humans and AI systems are able to work together in providing interpretations that are related to practices, sensitive to context, and provide good reasons for making ethical choices if required.

3.3.9 Not only humans: XAI in intelligent autonomous systems

Virtual agents are the most common embodiment of symbolic AI [195]. They can operate singularly, in a cooperative or adversarial fashion (within Multi-Agent Systems – MAS). The agents composing intelligent autonomous systems (MAS) are hardware/software-based computer systems characterized by any or all of the following: (i) autonomy (no direct intervention or human control), (ii) social ability (free to interact with other agents and humans), (iii) reactivity (perception of their environment and according to reactions), and (iv) pro-activeness (being goal-directed, they can take the initiative) [81]. MAS have increasingly become part of modern society and as such incorporated into an increasing number of everyday tasks [45].

Beyond their symbolic nature, modern agents can also leverage sub-symbolic algorithms (i.e., ML and DL), integrating them into their reasoning processes [199]. While symbolic agents are explainable by design (being mainly rule-based), the behavior of sub-symbolic or hybrid agents can result in being opaque for both human users and other agents. Such opacity harms the reputation of the single agents and the trust in the overall intelligent system [16, 55]. In the last decades, the majority of the articles in explainable agents focused on making intelligent systems understandable primarily to humans [16, 98, 192]. Bridging symbolic and sub-symbolic approaches is called neuro-symbolic integration [198, 215]. For example, [65] proposed to adopt neuro-symbolic and probabilistic approaches, [188] to adopt neuro-argumentative techniques, and [34] proposed two paths to achieve such integration. Nevertheless, current research indicates that the forthcoming decades will focus on the full development of conversational informatics [44, 162]. MAS are modeled after human societies and within MAS agents communicate with each other, sharing syntax and ontology. They interact via the Agent Communication Languages (ACL) standard [212] shaped around Searle's theory of human communication based on speech acts [200]. Therefore, multi-agent interpretability and explainability require multi-disciplinary efforts to capture all the diverse dimensions and nuances of human conversational acts, transposing such skills to conversational agents [54, 55]. Equipping virtual entities with explanation capabilities (either directed to humans or other virtual agents) fits into the view of socio-technical systems, where both humans and artificial components play the role of system components [239]. Ongoing international projects revolve around these concepts. For example, they are tackling intra- and inter-agent explainability (EXPECTATION), actualizing explainable assistive robots (COHERENT), countering information manipulation with knowledge graphs and semantics (CIMPLE), and relating action to effect via causal models of the environment (CausalXRL) [6]. Explainable agents can leverage symbolic AI techniques to provide a rational and shareable representation of their own specific cognitive processes and results. Being able to manipulate such a representation allows building one or more personalized explanations to meet the explainee (human and virtual) background and boost the success of the explanation process and overall interaction.

3.4 A case study: The medical domain

In this Section, we present a case study in a medical scenario. We show how each of the perspectives from the multiple domains (i.e. from the legislation, cognitive, social, ethical, philosophical, rights at work, ML, and symbolic AI) comes into play in a possible use case. As argued by [28, 225], the application of ML to clinical settings represents a relevant use case for interpretability, motivated by the high stakes, the complexity of the modeling task, and the need for reliability. From the legal perspective, clinicians are the

[6] Projects within the CHIST-ERA pathfinder programme for research on future and emerging information and communication technologies `https://www.chistera.eu/projects`

sole people legally accountable for any diagnosis and decision-making, hence accepting ML suggestions is seen as taking an acknowledged risk that may affect the survival and life quality of the patient. As the cognitive sciences suggest, clinicians should be able to revise their mental model of the AI system to be able to understand the principles applied by the systems' decision-making, ensuring the reliability of the systems. It is only through time and sustained use that a social relationship of trust between the physician and the automated system can be installed. Interpretability is to be sought in the medical application not only for the sake of the philosophical and epistemic value of explanations per se, but also as an ethical requirement to provide a factual, direct, and clear explanation of the decision-making process, especially in the event of unwanted consequences" [78, 189]. An AI-generated decision arguably needs to be interpretable if it can affect a human being. Given the high cost of making a mistake, the ML application cannot be allowed to take decisions independently, differently from other contexts where ML tools are used lightly, e.g. recommendation systems. This sets a major requirement to ensure the well-being of the physicians in the workplace, making sure that their confidence in the tools may increase over time and provide them with sufficient transparency to take the decisions on whether to rely or not on the AI system. To satisfy the requirements set by this analysis from the social sciences, the ML and symbolic-AI tools deployed for clinical use should interact with the experts for which technical solutions must be developed.

The interaction between humans and ML systems is a non-trivial task. Human reasoning is mostly based on high-level concepts that interact with each other to form a semantic representation. These interactions with semantic meaning are not necessarily represented by ML models that mostly operate on numeric features such as input pixel values, internal activations and model weights [118]. When the features used by the model are expressed in clinical terms, the interaction of the clinicians with the system is enhanced and can lead to successful cooperation. An example is the case described in [48]. Despite its high performance, the model for pneumonia risk detection had a hidden flaw. Cases of pneumonia with concurring asthma were assigned a lower risk of death than those without, despite the presence of this condition being known to worsen the severity of the cases. A correct prediction would have been the opposite diagnosis given the high risk of death. The misleading correlation (i.e. presence of asthma thus low risk of death from pneumonia) was rather a consequence of the effective care given to these patients by healthcare specialists that were promptly reacting to reduce the risk of death, and as a consequence lowering the recorded risk for these patients. The misleading feature "presence of asthma" was captured by the interpretability analysis and it was promptly understood by physicians since it was expressed as a clinical feature.

It is now worth pointing out that, as described by Asan *et al.*, "maximizing the user's trust does not necessarily yield the best decisions from a human-AI collaboration" and that the optimal trust level can be achieved when the user knows when the model makes errors. After recalling that the role of humans in the practical applications of AI has

been overlooked [21], they suggest that achieving such an understanding of both strengths and weaknesses of the models requires a combination of three main elements: (i) increasing transparency, (ii) ensuring robustness [39] and (iii) encouraging fairness. Concerning (i), XAI was mentioned as the most promising approach to alleviate the black-box effects [156, 184]. In addition, we believe that current AI model lifecycles are often too short for the user to acquire sufficiently high confidence, where novel approaches or even retrained versions of the same algorithm are constantly released, sometimes with only little quantitative performance improvement. This can be compared to a situation where drivers must flawlessly master their vehicle while the latter is continuously changing shape and characteristics. One must therefore foster patience to achieve an adequate level of trust, which involves an intimate relationship between the end-user and a particular instance of the model to seize the situations where the model is working well and where it does not. This was *de facto* encouraged by the U.S. Food and Drug Administration (FDA), which as of June 2021 only approved static algorithms. However, as pointed out by Pianykh *et al.* the performance of static AI algorithms tends to degrade over time, owing to the naturally occurring changes in local data and the environment [178]. Furthermore, the access to a large collection of well-curated, expert-labeled data from a source that has high relevance to the studied population and the question asked is also a severe barrier to widespread adoption in the clinics [240]. We can conclude that an optimal model lifecycle has yet to be discovered to balance between model performance and robustness as well as adequate user trust and data access to optimally train AI models.

3.5 Conclusion

This work proposes an in-depth discussion of the terminology in interpretable AI, highlighting the risks of misunderstanding that exist if differing definitions are employed in the technical and social sciences. As noted by the experts, there are important gaps between how, for example, the legal legislation shows the notion of transparency and the meaning that is assigned to this word by ML experts and developers. While in the first case, transparency is intended as a subjective property that is influenced by the receiver's understanding and prior knowledge, in the technical sciences transparency is rather seen as an objective property that is not influenced by the receiver of the information. Similarly, the notion of interpretability is seen as the creation of a social contract of trust by social sciences, whereas this is yet too often intended as the explanation of the automated generation process of the AI system by most AI experts.

The taxonomy proposed in this chapter has the objective to harmonize the terminology used by lawyers, philosophers, developers, physicians, and sociologists, with the goal of building a solid basis for discussing the future of AI development in a multidisciplinary setting. We show how the proposed terminology is used in multiple domains and also its

versatility in social and technical discussions. By discussing these points on the concrete application of the medical domain we show that the need for a common terminology is real and that further reflection is needed to define how effective human-machine cooperation can be established. Without the help of the social sciences, it would not be possible to obtain a sustainable human-machine partnership and further research needs to be pursued at the frontier of the social and technical sciences.

Chapter 4

Study of the Interpretability Trade-offs

As previously mentioned DL techniques are a powerful tool for classification tasks such as image recognition [127], however, due to their black box models, the need for interpretability arose as one of the main topics in this area neglecting the performance and the complexity of such models. To address these issues, two works will be illustrated in this chapter. In section 4.1 we present a study of the trade-off between performance and interpretability by proposing a method for transferring the knowledge acquired by deep neural networks to interpretable models to produce more accurate interpretable models. In section 4.2 we study the relationship between the complexity of the network and the quality of its explanations by changing the regularization parameters of the network and measuring the impact on the faithfulness of saliency maps.

4.1 Study of the Trade-off between Performance and Interpretability

In spite of machines being powerful at classification tasks such as image recognition [127], the produced models can at times be complex and hard to interpret. Interpretable mimic learning [50] has drawn inspiration from model compression [41] to reduce this trade-off. Model compression consists of approximating a function learned by a slow and complex model with a faster and simpler model with comparable performance [41].

Ba and Caruana [23] demonstrated, using mimic learning, a variant of model compression, that shallow neural networks could, in principle, learn as accurate functions as the ones learned by deep nets. This was generalized by distillation [108], which works by using a transfer set to train the complex model with cross-entropy and softmax and using these soft predictions to train the distilled model.

While the motivation of the above model compression approaches was the reduction of the required storage and computational power at test time, by teaching interpretable models we can obtain another advantage, interpretability [50].

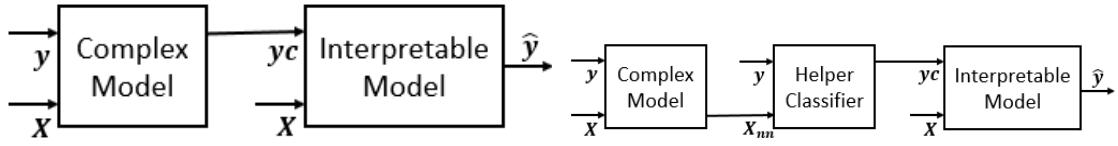
To this end, an ordinal mimic learning approach was proposed producing interpretable models which mimic the predictions of complex neural networks. The contribution of the present work is a new framework for ordinal mimic learning validated on 19 datasets.

4.1.1 Method

Our method, called Ordinal interpretable mimic learning extends interpretable mimic learning [50], generalizing the two pipelines for binary classification, to problems with ordinal classes. By combining the two training pipelines from [50] with two ordinal approaches, Multiclass and Frank&Hall [80], we can obtain interpretable models that mimic complex models.

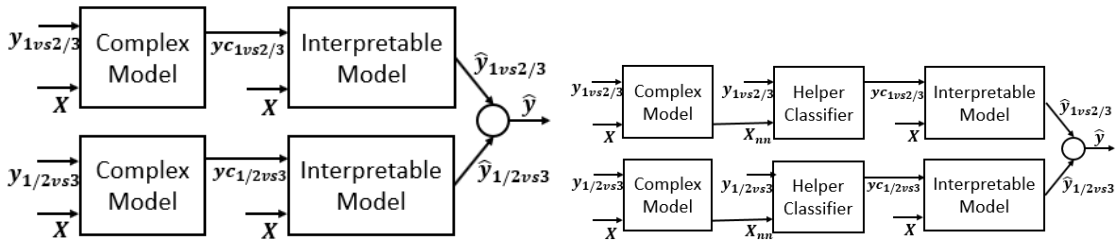
We propose four architectures that combine the pipelines in [50] and two ordinal approaches, Multiclass and Frank&Hall [80]. In Pipeline 1 (Figures 4.1a and 4.2a), we train the complex model(s) (e.g. feed-forward neural networks) using the training set $\{X, y\}$, composed of the original features X and the targets y, obtaining the soft predictions of the training set, yc. An interpretable model is then trained to mimic the complex model, using as input $\{X, yc\}$.

In Pipeline 2 (Figures 4.1b and 4.2b), the activations of the last hidden layer of the complex model(s), X_{nn}, are used in combination with the original targets y, to train Helper Classifier. We then take the soft predictions of the Helper Classifier, yc, and the original features, X, and train the interpretable model.



(a) Pipeline 1 (b) Pipeline 2

Figure 4.1: Illustration of the Multiclass Mimic Learning approach



(a) Pipeline 1 (b) Pipeline 2

Figure 4.2: Illustration of the Frank&Hall Mimic Learning approach for a 3-class problem

In both pipelines, at testing time, the classification of unseen samples is performed using only the mimic interpretable model(s). In the Multiclass approach (Figure 4.1), only one K-class classifier is trained and the soft predictions are weighted according with the class label. By weighting the soft predictions, the K class probabilities are combined into one numeric using the following equation:

$$yc = \sum_{k=1}^{K} [k * Pr(V = V_k)] \tag{4.1}$$

In the Frank&Hall architecture (Figure 4.2), the K-class classification problem is divided into $K-1$ classification problems. Each classifier i learns to differentiate classes $C_1, \ldots, C_i$ from classes $C_{i+1}, \ldots, C_K$. For each binary problem, we train a complex model, a Helper Classifier (in case of the pipeline 2), and an Interpretable Model. The predictions of each interpretable model are combined so that, for the case of a 3 class problem, if the two models agree with value -1 the result is class 1, if they agree with value 1 than the result is class 3, otherwise the result is class 2.

So far, we have considered as a complex model, a neural network with k neurons on the output layer. For pipeline 1, this can be generalized to any multi-class classifier capable of producing class probabilities[1]. In the case of pipeline 2, the use of the activations of the last hidden layer of the complex model, X_{nn}, restricts it to neural networks.

4.1.2 Experimental Setup

The ordinal datasets include the ones used in [84][2], which were used for benchmark different ordinal approaches, as well as two heathcare datasets described in [164, 165], where the use of the ordinal nature of the response to cancer treatment could improve its prediction.

Feature selection was made using Neighborhood Component Analysis [245] or ReliefF [190], which are filter methods (independent from the classification method) and suitable for multi-class classification.

We tested three different types of inputs for the mimic model, softmax, double and neighbor. The softmax is the one illustrated in Figures 4.1 and 4.2, where the input is $\{X, y_c\}$.

In the double, we train the interpretable model with the original dataset concatenated with the one with soft labels, $\{XX, yy_c\}$, represented on equation (4.2). For the neighbor, we discard the samples that the complex model classifies incorrectly, $\{XX', yy'_c\}$, represented on equation (4.3).

[1] When using multi-class classifiers with outputs in the range [1,k], the output can be used directly with no need to apply Equation 4.1.

[2] available at http://www.uco.es/grupos/ayrna/ucobigfiles/datasets-orreview.zip

$$\{XX, yy_c\} = \left[\begin{array}{cccc|c} x_{11} & x_{12} & \dots & x_{1m} & y_1 \\ x_{21} & x_{22} & \dots & x_{2m} & y_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} & y_n \\ \hline x_{11} & x_{12} & \dots & x_{1m} & y_{c_1} \\ x_{21} & x_{22} & \dots & x_{2m} & y_{c_2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} & y_{c_n} \end{array}\right] \tag{4.2}$$

$$\{XX', yy'_c\} = \left[\begin{array}{cccc|c} x_{11} & x_{12} & \dots & x_{1m} & y_1 \\ x_{21} & x_{22} & \dots & x_{2m} & y_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} & y_n \\ \hline x_{11} & x_{12} & \dots & x_{1m} & y'_{c_1} \\ x_{21} & x_{22} & \dots & x_{2m} & y'_{c_2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{n'1} & x_{n'2} & \dots & x_{n'm} & y'_{cn'} \end{array}\right], n' \leq n \tag{4.3}$$

Three interpretable supervised methods were selected to validate our approaches: Linear Regression, Regression Tree and Symbolic Regression; and a Feedfoward Neural Network (FNN) as our complex model.

All the above models were trained using MATLAB's (v. 9.3.0.713579) default parameters, excluding the FNN's hyperparameters which were tuned using grid search by exploring the number of hidden layers, $n_H \in \{1, 2, 3, 4, 5\}$, and the number of hidden units, $n_{HU} \in \{16, 32, 64, 128, 256, 512\}$.

4.1.3 Results

The algorithms were ranked by the average MAE, obtained using leave-one-out cross-fold validation on the healthcare datasets and over 2-folds on the other datasets, and the datasets were grouped according to the number of features, as shown in Table 4.1.

Table 4.1: Top 10 ranked algorithms for datasets grouped based on number of features and ordered by approach. FS, Pl, Classif, m and AR stand for feature selection, pipeline, classifier, number of features and average rank respectively. The top 5 algorithms' ranks based on the number of features are highlighted in bold.

Approach	Pl	Input	FS	Classif	m=1-4	m=5-25	AR
Multiclass	2	Double	none	RT	**5**	**1**	3.0
Multiclass	1	Softmax	none	RT	6	**3**	4.5
Multiclass	1	Softmax	NCAreg	RT	7	**4**	5.5
Multiclass	1	Neighbor	none	RT	10	**2**	6.0
Multiclass	1	Double	none	RT	9	**5**	7.0
Multiclass	2	Neighbor	none	RT	8	6	7.0
Frank&Hall	2	Softmax	NCAreg	SR	**2**	7	4.5
Frank&Hall	1	Softmax	none	SR	**3**	8	5.5
Frank&Hall	2	Softmax	none	SR	**1**	10	5.5
Frank&Hall	2	Softmax	none	RT	**4**	9	6.5

Based on Table 4.1 we observe that both Frank&Hall and Multiclass approaches reach the top 10 while no interpretable model that does not use the mimic approach did. We can also see that Frank&Hall methods did better on datasets with fewer features (< 5). We believe this happens because the models perform worse with fewer features, so by predicting a mid-class when different classifiers are disagreeing the Frank&Hall method avoids making bigger mistakes.

Table 4.1 shows that Multiclass with double and neighbor perform better with more than four features. This is in line with the general observation that "the optimal number of features increases with increasing sample size"[3] [110], since double and neighbor are trained with augmented datasets.

We also observe in Table 4.1 that every Frank&Hall approach reaching the top 10 is instantiated with softmax, indicating that, for Frank&Hall, softmax predictions do better than double and neighbor.

This may be because, by using hard predictions which can only take opposing values 0,1 rather than values in between [0,1], predictions of the different binary classifiers might disagree more.

Frank&Hall ($MAE = 1.3$, $std = 1.0$) algorithms show higher mean MAE and standard deviation than Multiclass ($MAE = 0.6$, $std = 0.4$), which indicates that this simplified ordinal classification approach may not capture correctly the ordinal nature of the classes.

4.1.4 Conclusions

In this work, an ordinal interpretable mimic learning framework was proposed to study the performance versus interpretability trade-off in the context of ordinal problems. Results show that the interpretable models trained to mimic complex models outperform the models trained directly on the original datasets. These results point to the possibility of leveraging the complexity of the model to obtain interpretability without sacrificing performance.

The main focus of our future work focused on the study of complexity as this early work indicates that while lowering complexity improves complexity when working with interpretable models, the same still cannot be said for DL models. It is also important to recognize that our conclusions are limited to the scope of the datasets and models used, therefore further work will help validate our approach to different problems.

[3] A warning should here be made to the fact that the optimal-feature-size relative to the sample size depends not only on the classifier but also the feature-label distribution [110].

4.2 Study of the Trade-off between Complexity and Interpretability

As neural networks grow in complexity [127] their capacity to learn mappings from the input data to the classification label increases. Explanations are provided to understand this mapping and the predictions made by the network. One of the proposed explanations is called saliency maps, which produce an estimation of each pixel's relevance in the overall prediction of the network for each input image. There are many saliency map methods [24, 155, 214, 251] which give different estimations of the relevance scores. To quantitatively evaluate if a given relevance score is suitable, we need to assess if the method truly discriminates between relevant and irrelevant pixels. Due to the non-existence of ground-truth relevance scores, the quantitative evaluation of saliency maps poses a demanding problem. Fidelity [18] is a concept that determines how well a relevance score agrees with how the model works.

Only increasing the capacity of the networks results in the overfitting of the model to the training data, resulting in a low test set performance. For this reason, regularization approaches are used to decrease the capacity of the network to fit the data and avoid overfitting. Some of these methods, such as L^2 regularization, reduce the complexity of the network, limiting the values of the network parameters.

However, it is not yet understood what impact certain changes in the network parameters have on the different saliency map methods and consequently in their interpretation.

4.2.1 Method

The problem which was here presented involves three aspects that will be described in this section, that includes the regularization of the network, the saliency map methods, and the interpretability metrics.

Regularization

Regularization is a modification to a learning algorithm intended to reduce its generalization error [87]. Many regularization approaches limit the capacity of models, and its complexity, by adding a parameter norm penalty.

L^2 Regularization commonly known as weight decay, it's a regularization strategy that drives the weights closer to zero, by adding a regularization term to the objective function:

Components of the weight vector corresponding to directions that do not contribute to reducing the objective function are decayed away through the use of the regularization

throughout training [87].

$$\Omega(\theta) = \frac{1}{2}\|w\|_2^2 \tag{4.4}$$

Early Stopping When training models with sufficient capacity to overfit the task, it is common to see that after a constant decrease in the training error and validation error, the validation error often starts to rise. Early stopping works by keeping a copy of the model parameters every time the validation error improves, to return the setting when the validation error was the lowest.

Saliency Map Methods

In our formal description, an *input* corresponds to an image and is represented by a tensor $x \in \mathbb{R}^d$. A model describes a function $f : \mathbb{R}^d \to \mathbb{R}^c$, which maps the d-dimensional images to a prediction vector where c corresponds to the number of classes of the classification problem.

Below, the explanation methods which produce saliency maps and which were used in this work will be briefly described.

Gradient The gradient quantifies how much a change in each feature would change the predictions $f(x)$ in a small neighborhood around the input [214].

$$Grad(x) = \frac{\partial f(x)}{\partial x} \tag{4.5}$$

DeConvNet The DeConvNet associates the architecture of the model, with a corresponding architecture that reverses the computations and produces an image as the output [251]. To do this, each layer is associated with a corresponding layer that reverses the computation.

Guided Backpropagation a combination of the previous two methods, guided backpropagation prevents the backward flow of negative gradients, corresponding to the neurons which decrease the activation of the units we are inspecting [214]. Negative gradients are set to zero while backpropagating.

Deep Taylor Decomposition DTD is obtained by propagating the model output through the network using redistribution rules, until the input features are reached [155].

The propagation rules are derived from a Taylor decomposition performed at each unit of the network.

Layer-wise Relevance Propagation Similar to DTD, LRP is obtained by propagating the model output through the network using redistribution rules [24]. LRP finds its mathematical foundations in Deep Taylor Decomposition [155]. The redistribution rules proportional decompose the relevance score of upper layers to obtain lower layer relevance scores, based on the forward mappings between layers.

Interpretability Metrics

Although a large number of saliency map methods have been proposed, relatively few metrics to evaluate their fidelity have been proposed. Fidelity is a concept that should capture how well the relevance given to each pixel represents the process of the model. We will now describe two interpretability metrics that can be considered a proxy for fidelity and evaluate the quality of the saliency maps.

Confidence drop tracks the decrease of confidence in the model's classification when removing a percentage of the most relevant pixels given by a saliency map. If the saliency maps present a high fidelity to the model, the confidence should drop faster than when the fidelity is low.

$$Drop(k) = f(x^{(0)}) - f(x^{(k)}) \tag{4.6}$$

In Equation 4.6 we can see that the confidence drop corresponds to the difference in confidence when perturbating the k higher relevant pixels.

Another interpretability metric is called Area Over the Perturbation Curve (AOPC) [18]. The AOPC tracks the decrease of confidence in the model's classification when iteratively removing the most relevant pixels given by a saliency map.

The AOPC equation is described in the following equation:

$$AOPC = \frac{1}{L+1}\langle\sum_{k=1}^{L} f(x^{(0)}) - f(x^{(k)})\rangle \tag{4.7}$$

In Equation 4.7, L is the number of pixel perturbation steps, $f(x)$ is the output value of the classifier for input image x (i.e. the confidence assigned to the class), $x(0)$ is the original image and $x(k)$ is the image after k perturbations.

4.2.2 Experimental Setup

The training set and test set were combined and 10-fold cross-validation was used to train and evaluate the model. The classifier is trained with different regularization values, To evaluate the interpretability of a model, we take the test samples and the trained classifier, and apply the saliency map method to the samples, resulting in saliency maps that are used to calculate the interpretability metric. These concepts are visually illustrated in Figure 4.3.

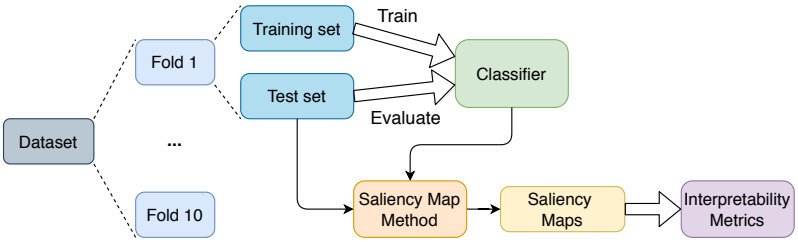


Figure 4.3: Architecture of experimental setup.

Dataset and Models

Experiments were performed on the CIFAR-10 dataset [126] as it is a well-known image classification dataset with suitable complexity. A description of the dataset is present in Table 4.2.

Table 4.2: Description of the dataset used in the study

name	samples	classes	width	height	channels
CIFAR10	60000	10	32	32	3

In this experiment, a standard convolutional neural network (CNN) containing three convolutional blocks and two fully-connected layers, was used. Each convolutional block is composed of two convolutional layers followed by a max-pooling layer. The classification is done using a softmax layer after the fully-connected layers. A figure representing the architecture of the network is presented in Figure 4.4.

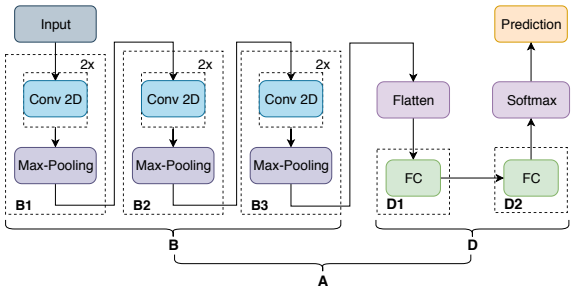


Figure 4.4: Architecture of the neural network.

During the training of the model, different levels of L^2 weigh decay were used (0, 0.0001, 0.001, 0.005, 0.01, 0.05) as well as early stopping to prevent overfitting, stopping training after 10 epochs of no improvement in the loss function. To understand the changes in interpretability that were caused by changing the complexity in different layers of the network, we separated regularization into different groups of layers. $B1$, $B2$, and $B3$ correspond to regularization on the first, second, or third convolutional block respectively; $D1$ and $D2$ correspond to regularization on the first or second fully-connected layer. Finally, B corresponds to regularization on all convolutional blocks, D corresponds to regularization on all fully-connected layers and A corresponds to regularization on all layers of the network.

Saliency Map Methods

In these experiments, five different saliency map methods were compared. The criteria for choosing these methods was based on their proven applicability in the literature as well as their properties. The methods chosen were gradient [214], DeConvNet [251], Guided Backpropagation [214], Deep Taylor Decomposition [155] and Layer-wise Relevance Propagation [24]. Some methods only estimate positive relevance while other methods estimate positive and negative relevance. For example, Guided Backpropagation and Deep Taylor Decomposition, produce only positive relevance. The implementation of the explanation methods was done using the iNNvestigate Toolbox v1.0.8 [8].

Saliency Metrics

We use two different perturbations, one by deleting the most relevant pixels given the relevance score provided by the saliency map, and the other by deleting a random pixel. In the random perturbation method, the value is a gray-scale value sampled from a uniform distribution in the case of gray-scale images and an RGB value in the case of colored images. This approach attempts to destroy the information contained in the pixel.

To measure the confidence drop caused by the perturbation, we have segmented different percentages of the most relevant pixels. We have chosen to group the pixels in the 5%, 10%, 20%, 30%, 40%, 50%, and 75% most relevant pixels.

To calculate the AOPC metric we used perturbation steps corresponding to 10% of the dataset.

4.2.3 Results

In this study, two experiments were conducted. In the first, regularization was applied in all layers of the network (convolutional and fully-connected). In the second experiment, we separated regularization by different groups of layers: regularization on only one layer

or block of layers ($B1$, $B2$, $B3$, $D1$, $D2$) and regularization applied to multiple blocks of layers (B, D, A).

How does the regularization of the deep neural network affect the quality of saliency maps?

The first experiment measures the interpretability of models regularized in all layers with different L^2 weight decay values.

Table 4.3: Results comparing interpretability (AOPC) of saliency map methods on models with different regularization values.

	L^2 **weight decay**					
Method	**0**	**0.0001**	**0.001**	**0.005**	**0.01**	**0.05**
DeConvNet	**0.314**	0.293	0.253	0.232	0.134	9.3E-12
Deep Taylor	0.218	**0.230**	0.229	0.201	0.114	9.3E-12
Gradient	**0.419**	0.413	0.417	0.356	0.226	9.3E-12
Guided Backprop	0.269	**0.281**	0.263	0.224	0.132	9.3E-12
LRP	0.423	0.421	**0.427**	0.362	0.231	9.3E-12

Table 4.3 is composed of the saliency map methods (first column), and the weight decay values (first row). The values of the table represent the mean AOPC value for the 10 folds. The highest interpretability values for each method are highlighted in bold.

The results in Table 4.3 show that the methods that display the highest values of interpretability and that produce saliency maps with more fidelity to the model's decision are the LRP and Gradient methods.

The results in Table 4.3 show a substantial difference between interpretability and saliency map methods. In general, the quality of such methods is higher when the network is trained with smaller regularization values, although the exact value is not consistent between methods.

To assess the statistical significance of the interpretability metric AOPC when the regularization values in all layers are changed, Friedman's test was applied with a significance level of 5%. We considered the different weight decays as well as the different saliency map methods. It was determined that the regularization does have a statistical significance on the interpretability metric. Statistical significance was detected between the lower regularization values (0, 0.0001, and sometimes 0.001) and the higher regularization values (0.01 and 0.05) consistently in all methods.

The same statistical significance tests were also conducted with the experiments using random perturbation which found no statistical difference between either of the regularization values in all saliency methods.

How does the layer regularized affect the interpretability of the network?

Another question we had was to learn what layers regularization is more appropriate to produce the saliency maps with better fidelity to the model. To answer this question we run an experiment training CNNs with regularization only on specific layers and we extracted the saliency maps o measure their fidelity using the AOPC metric.

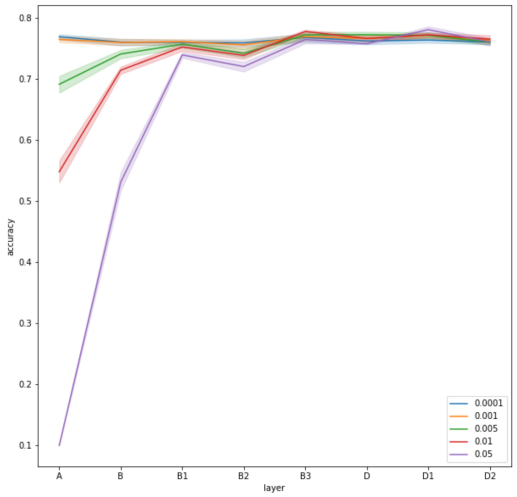


Figure 4.5: Accuracy of the different models based on layer and weight decay of regularization.

In Figure 4.5 it is possible to see the accuracy of the different models based on the layer which was regularized with the specified weight decay values. The models which were regularized in all layers have lower performance than the other models, especially with higher regularization values. From Figure 4.5 we can see that there is a performance benefit of using regularization on only some layers, and not in all of them.

Table 4.4: Mean AOPC of the DeConvNet method. The first column corresponds to the layer regularized, and the first row the L^2 weight decay. The values represent the mean AOPC, and the highest values for each regularization value are highlighted in bold.

	L^2 **weight decay**					
Layer	**0**	**0.0001**	**0.001**	**0.005**	**0.01**	**0.05**
B1	**0.3142**	0.3082	0.3004	0.2810	0.2772	0.2292
B2	**0.3142**	**0.3368**	0.3065	0.2882	0.2651	0.2328
B3	**0.3142**	0.3107	0.3135	0.3037	0.2710	0.3228
B	**0.3142**	0.2878	0.2651	0.2451	0.2603	0.1600
D1	**0.3142**	0.3253	0.3265	**0.3298**	**0.3425**	0.3666
D2	**0.3142**	0.3247	0.3238	0.3197	0.3249	0.3155
D	**0.3142**	0.3262	**0.3344**	**0.3298**	0.3395	**0.3503**
A	**0.3142**	0.2934	0.2526	0.2316	0.1343	0.0000

In Tables 4.4- 4.8 is visible that, for each saliency map method, the interpretability metric based on the L^2 weight decay used to regularize the specific layer of the model.

Regarding the presented results, we can see that interpretability appears to be higher with lower regularization values. Additionally, interpretability appears to be higher when

Table 4.5: Mean AOPC of the Deep Taylor method. The first column corresponds to the layer regularized, and the first row the L^2 weight decay. The values represent the mean AOPC, and the highest values for each regularization value are highlighted in bold.

	L^2 **weight decay**					
Layer	**0**	**0.0001**	**0.001**	**0.005**	**0.01**	**0.05**
B1	**0.2183**	0.2216	0.2252	0.2274	0.2211	0.2188
B2	**0.2183**	0.2283	0.2232	0.2226	0.2224	0.1988
B3	**0.2183**	**0.2368**	0.2310	0.2323	0.2245	0.2416
B	**0.2183**	0.2269	0.2312	0.2326	0.2236	0.1453
D1	**0.2183**	0.2279	**0.2465**	0.2427	**0.2499**	**0.2526**
D2	**0.2183**	0.2267	0.2302	0.2405	0.2235	0.2354
D	**0.2183**	0.2250	0.2348	**0.2452**	0.2473	0.2516
A	**0.2183**	0.2301	0.2292	0.2014	0.1138	0.0000

Table 4.6: Mean AOPC of the Gradient method. The first column corresponds to the layer regularized, and the first row the L^2 weight decay. The values represent the mean AOPC, and the highest values for each regularization value are highlighted in bold.

	L^2 **weight decay**					
Layer	**0**	**0.0001**	**0.001**	**0.005**	**0.01**	**0.05**
B1	**0.4192**	0.4212	0.4249	0.4167	0.4145	0.4153
B2	**0.4192**	0.4242	0.4244	0.4101	0.4153	0.3898
B3	**0.4192**	**0.4301**	0.4252	0.4295	0.4230	0.4279
B	**0.4192**	0.4279	0.4272	0.4149	0.3963	0.2309
D1	**0.4192**	0.4227	0.4252	0.4287	0.4226	**0.4358**
D2	**0.4192**	0.4181	**0.4291**	**0.4303**	**0.4273**	0.4341
D	**0.4192**	0.4211	0.4248	0.4289	0.4266	0.4201
A	**0.4192**	0.4127	0.4174	0.3564	0.2260	0.0000

Table 4.7: Mean AOPC of the Guided Backprop method. The first column corresponds to the layer regularized, and the first row the L^2 weight decay. The values represent the mean AOPC, and the highest values for each regularization value are highlighted in bold.

	L^2 **weight decay**					
Layer	**0**	**0.0001**	**0.001**	**0.005**	**0.01**	**0.05**
B1	**0.2690**	0.2722	0.2720	0.2698	0.2563	0.2454
B2	**0.2690**	**0.2884**	0.2696	0.2700	0.2561	0.2312
B3	**0.2690**	0.2860	0.2844	0.2825	0.2615	0.2881
B	**0.2690**	0.2777	0.2723	0.2611	0.2678	0.1629
D1	**0.2690**	0.2867	0.2852	0.2912	0.2903	0.2993
D2	**0.2690**	0.2792	0.2716	**0.2922**	0.2878	0.2755
D	**0.2690**	0.2788	**0.2919**	0.2867	**0.2951**	**0.3022**
A	**0.2690**	0.2813	0.2630	0.2243	0.1324	0.0000

regularization happens in higher convolutional layers or in fully-connected layers.

The methods that display the highest values of interpretability and that produce saliency maps with more fidelity to the model's decision are the LRP and Gradient methods.

Following further analysis of these results, we can plot the number of times that regularization in a specific layer has produced the best interpretability values for each method. This plot is presented in Figure 4.6, and as we can see, it once again shows that regularization is more effective in higher convolutional layers or in fully-connected layers.

Table 4.8: Mean AOPC of the LRP method. The first column corresponds to the layer regularized, and the first row the L^2 weight decay. The values represent the mean AOPC, and the highest values for each regularization value are highlighted in bold.

	L^2 **weight decay**					
Layer	**0**	**0.0001**	**0.001**	**0.005**	**0.01**	**0.05**
B1	**0.4226**	0.4274	0.4304	0.4202	0.4203	0.4196
B2	**0.4226**	0.4295	0.4289	0.4154	0.4184	0.3947
B3	**0.4226**	**0.4340**	0.4284	0.4344	0.4256	0.4302
B	**0.4226**	0.4309	0.4312	0.4206	0.4055	0.2336
D1	**0.4226**	0.4282	0.4278	**0.4351**	0.4288	**0.4439**
D2	**0.4226**	0.4227	**0.4333**	0.4337	**0.4351**	0.4387
D	**0.4226**	0.4273	0.4301	0.4334	0.4317	0.4281
A	**0.4226**	0.4211	0.4265	0.3622	0.2313	0.0000

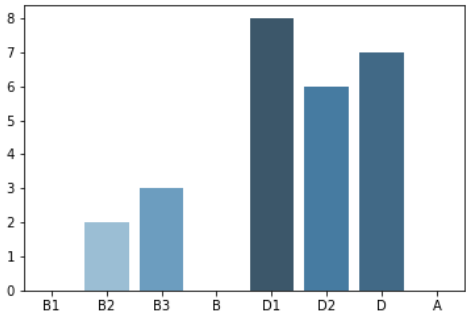


Figure 4.6: Plot showing the number of times the regularization in a layer produced the best interpretability value for each method.

4.2.4 Conclusions

In this work, we studied the relationship between regularization and interpretability in a CNN context. From the results obtained with the experimental data, the following main conclusions may be derived:

- The quality of saliency maps is higher when the network is trained with smaller regularization values;
- LRP and Gradient produce saliency maps with higher fidelity to the model's decision;
- Overall, to obtain higher interpretability, regularization should be applied on later convolutional layers or in fully-connected layers;
- Models in which all layers were regularized display lower interpretability than models in which only the fully-connected layers were regularized.

Future work directions should test other mechanisms of regularization such as dropout, as well as extend our work to other datasets. Also compare the saliency maps produced by the different methods to understand differences in their distributions.

Chapter 5

Evaluating the Faithfulness of Saliency Maps using Realistic Perturbations

Based on the advent of clinical data digitalization materialized for instance in Electronic Health Records (EHR), CNNs reached human-level performance in multiple medical fields, empowering the processing of such data for automatic decision-making. However, their inherent lack of interpretability due to their black-box nature - it's not possible to look at the weights of the model and why the classifier made the prediction - delays their adoption in clinical practice and minimizes their current usefulness and impact.

Also, following the General Data Protection Regulation 2016/679 (GDPR), data controllers need to provide the user meaningful information about the logic involved in the automatic decision, as well as the significance and the envisaged consequences of such processing user [96, 201]. Researchers have proposed different explanations for information access systems in different areas such as financial distress prediction [257], movie recommendation [138] or clinical decision making [173], but quantitative and systematic evaluation of such explanations is important before their integration in an information access system, which in the case of the clinical field can be a clinical decision support system (DSS).

Saliency maps are one of the most popular strategies for explaining convolutional neural networks. Saliency maps illustrate the importance of individual pixels of the input image on the overall prediction of a CNN. The color or intensity of each pixel in the saliency map corresponds to the weight that the same pixel in the input image had on the classification process.

In the oncological field, it was found that the majority of interpretability strategies employed fall on the saliency map category [12]. Despite the adhesion to saliency maps for

explaining deep learning models, the validation is purely qualitative - by inspecting a few individual cases - and requires domain experts. Due to the lack of tools and evaluation metrics for the systematic evaluation of saliency maps quantitatively [12]. Evaluating saliency maps remains an open challenge since the task of highlighting the most important pixels in the classification of an image has an inherent problem, the lack of ground-truths.

Previous attempts at evaluating saliency map methods [2] have chosen to introduce random or uniform noise [197] to the input images to study their faithfulness to classification or invariance to input change. However, the resulting images deviate too much from the distribution of images used to train the CNNs making their behavior possibly erroneous.

To address this issue, in this work, the main goal is to evaluate the robustness of the saliency maps in the addition of a nature perturbation environment. Specifically, we investigate the following research questions:

- What is the impact of the different perturbations on the model's predictions?
- Are saliency map methods sensitive to the introduction of natural noise?
- How are saliency maps methods compared to each other in terms of perturbation robustness?
- How does the perturbation region's size impact the saliency maps?

To achieve that, three CNN (VGG16, ResNet50, and InceptionV3) were trained on the PatchCamelyon dataset. The PatchCamelyon dataset [229] was derived from the Camelyon16 dataset[32] and contains patches of H&E stained histopathological images of sentinel lymph node sections. It is a balanced dataset where half the images represent normal cases (i.e. benign) and the other half represent tumor cases (i.e. malignant). The histopathological images were perturbed using a natural perturbation strategy based on the introduction of regions of different sizes (i.e. 8x8 pixels, 16x16 pixels, and 32x32 pixels).

To test the natural perturbation, three scenarios were implemented:

- NN: where a perturbation of normal tissue was added in an image of normal tissue;
- NT: where a perturbation of normal tissue was added in an image of tumor tissue;
- TN: where a perturbation of tumor tissue was added in an image of normal tissue.

By evaluating the impact of these three scenarios on the saliency maps produced by a method we are able to measure its robustness to perturbation.

Overall, in spite of some differences between methods, it is proven with this work that saliency maps can be a good strategy to interpret CNN models.

This chapter follows the following structure: Section 5.1 will briefly review related works in the literature. In Section 5.2 we present the different components of the framework for evaluating the faithfulness of saliency maps using realistic perturbations: data selection, (2) model training, 3) image perturbation, 4) saliency map extraction, and 5) saliency map evaluation. We present and discuss the results of the experiments in Section 5.3 and answer each research question in a separate subsection. Finally, in Section 5.5 we conclude with our final remarks and steps for future work.

5.1 Background

Automatic analysis of whole slide images, such as metastases detection, is an important application of artificial intelligence, allowing it to avoid a tedious and time-consuming examination process as well as helping to detect small metastases that can easily be missed [227]. In the automatic Tumor Lymph Node Metastasis (TNM) detection task, deep learning algorithms achieved better diagnostic performance than a panel of 11 pathologists [27]. This process requires highly skilled pathologists and is time-consuming and error-prone [170].

The Camelyon16 challenge [32] has the goal of evaluating algorithms on the task of automatic detection of breast cancer metastases in whole-slide images of hematoxylin and eosin (H&E) lymph node sections. Convolutional neural networks (CNNs) have been successful in this task, with approaches achieving an area under the receiver operating curve (AUC) of 0.925 and helping increase the pathologist's AUC to 0.995 when combining the pathologist diagnosis with the deep learning model's diagnosis [32]. This represents an almost 85 percent reduction in the human error rate.

The accomplished performance of these models in the automatic analysis of whole slide images poses a great opportunity for their integration in the clinical decision support system (DSS), however, it is necessary that the model is able to explain the decision to the patient and be audited when misclassification is made. Similar explanations have already been explored for clinical DSS so that the model can point to the feature of the clinical history of the patient that impacted the misclassification [173]. Also, on the prediction of corporate bankruptcies, not only feature importance but also counterfactual examples have been explored for explaining financial distress prediction models [257]. In a recommendation system based on movie cover and description of movie images, an explanation highlighting the important regions of the image and the words in the description has been developed [138].

5.1.1 Saliency map methods

Saliency map methods, often called feature attribution methods [152], are a popular interpretability strategy when input features of the classification task are pixels [221]. This

is can be especially seen in the oncological field [12] where half of the interpretability strategies employed to understand deep learning models are saliency map methods.

Saliency maps highlight individual pixels of the input image which are important for a given's models predictions. The pixel values of saliency maps can be obtained following two main approaches: Back-propagation methods compute the relevance of a pixel by propagating a signal from the output neuron backward through the layers to the input image in a single pass [24]. Sensitivity methods compute pixel relevance by making small changes in the pixel value of the input image and compute how the changes affect the prediction [209].

5.1.2 Saliency map metrics

Evaluating saliency maps remains an open challenge since the task of highlighting the most important pixels in the classification of an image has an inherent problem, the lack of ground-truths. Also, there are many different criteria that can be used when evaluating saliency maps.

Equally, there are different evaluation approaches for interpretability: application-grounded, human-grounded, and functionally-grounded [232].

Application-grounded evaluation involves conducting user studies within a real application - such as pathologist detecting tumor cells. This is by far the most used approach for saliency maps as most works validate only a small number of examples [12]. However this evaluation approach has some problems as it requires domain experts, is time and cost-consuming, and highly specific for the application so it can be generalized for other applications.

Human-grounded evaluation involves conducting user studies with non-domain experts without a specific application in mind. This approach includes studies with laypeople to find which visual type of saliency map is preferred and what confers the biggest trust in the model [9]. This evaluation approach allows a better generalization for other tasks but it is still time and cost-consuming.

Functionally-grounded evaluation requires no user study and instead uses a proxy metric following a formal definition of interpretability. This evaluation approach is by far less time and cost-consuming and can be reproduced in a larger variety of contexts allowing higher generalization. It also allows the quantitative evaluation of an interpretability method without the validation of an expert [232].

One example of a quantitative evaluation of interpretability is the area over the perturbation curve (AOPC), which measured the faithfulness of the saliency map given by the drop in the model's prediction when perturbing the most salient pixels given by a saliency map [197]. Using this image perturbation strategy, it was found that reducing the com-

plexity of the network had a positive impact on the faithfulness of the saliency maps [14]. A similar measure of faithfulness was proposed by taking into account the pixels individually during perturbation [10]. Both measures of faithfulness are able to evaluate how the pixels of a saliency map accurately reflects the classification process of the model. The main disadvantage of these metrics is that after perturbation the images contain either a uniform (i.e. black or the pixels' mean value) or random noise, producing images unfamiliar to the model and not well represented in the training set. The values of the metrics are also dependent on the perturbation method (i.e. uniform or random noise) [224].

In recent work, Adebayo *et al.* [2] proposed several basic sanity checks for saliency map methods. The idea of sanity checks is to quantitatively inspect new saliency map methods to find if they lack sensitivity to the model and the data.

Model randomization evaluated if the saliency map methods were invariant to randomization of the model's parameters (i.e. weights). While label randomization broke the relationship between the information in the image and the label, forcing the model to memorize the training labels rather than learning the original relationship of the data. The saliency map methods were then evaluated on the invariance of what the model has learned before and after randomization.

Adebayo *et al.* [2] main finding is that some gradient-based methods in the literature are invariant under model randomization.

Sanity checks rule out methods that provide compelling images while failing to be sensitive to the model or the data. They inspect how the saliency map methods respond to random and unnatural data. But, it is also important to inspect how the saliency map method responds to natural data and to provide a further comparison between saliency map methods, where saliency metrics are needed. Subsequent work of Yona *et al.* [247] also argues through a causal re-framing of their objective of sanity checks, that some of the conclusions cannot be fully established due to confounding introduced by the custom tasks.

Several approaches have targeted increasing the realism of perturbed images based on learning the distribution of the training set and proposing counterfactuals [68, 158, 179]. And while they have shown good results on tabular data or simple image datasets such as MNIST, they have not yet been validated on complex image datasets such as medical imaging datasets.

The main problem with using uniform values or random noise to evaluate the saliency maps is that the noisy images are not representative of the training set and as the perturbed images are outside the distribution of expected values we cannot predict how the classifier will behave. While when adding malignant evidence with a natural perturbation strategy we can assume that model would drop the confidence of the image being benign, with the addition of random noise we cannot make such an assumption. ROAR [109]

is a perturbation strategy that tries to solve this problem by retraining the model with perturbed training samples and aligning the train and test distributions, but calculating evaluation metrics this way can be prohibitively expensive to obtain. Rather than using random values, this work proposes the introduction of natural noise, in the form of regions of existing training samples and evaluates the saliency map methods on how well they respond to the introduction of the regions of the same class as the original image or regions from a different class. The advantage of using natural noise is that the regions being used to perturb are still in the training distribution.

5.2 Method

Having in mind the main goal of this work which consists in quantifying the sensitivity of saliency map methods attending to the addition of natural noise, a five-stage pipeline was defined in the experimental setup and is illustrated in Figure 5.1: (1) data selection, (2) model training, 3) image perturbation, 4) saliency map extraction and 5) saliency map evaluation.

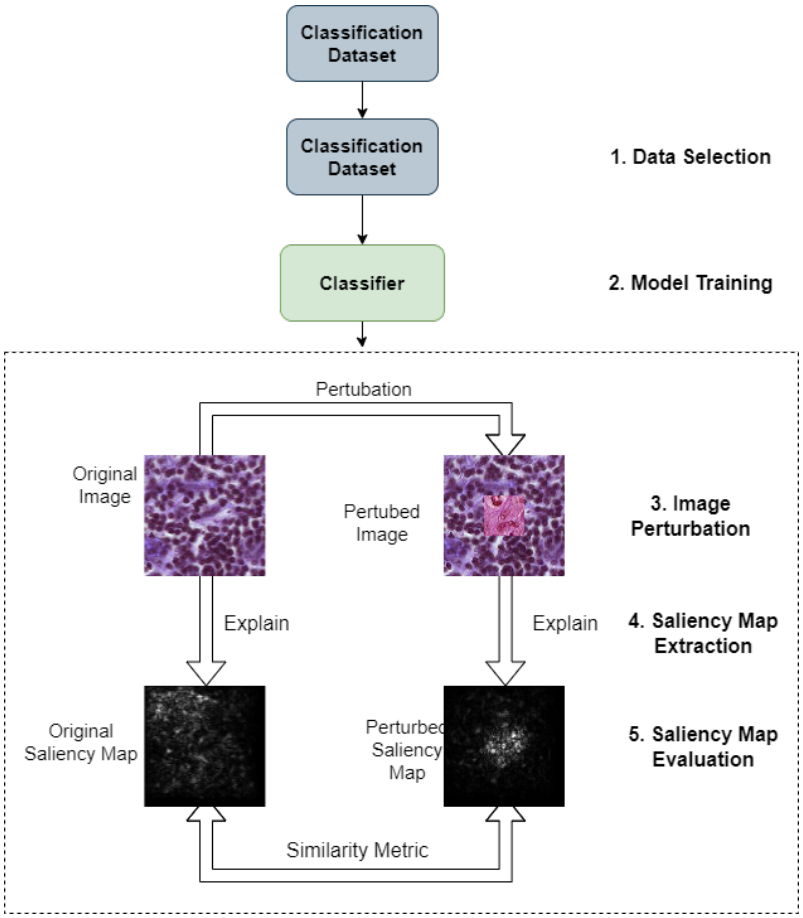


Figure 5.1: Automatic evaluation of saliency maps perturbation pipeline.

5.2.1 Dataset Selection

To evaluate the proposed approach, the PatchCamelyon dataset [229] derived from the Camelyon16 dataset[32] was used. The Camelyon16 dataset contains 400 H&E stained whole-slide images of sentinel lymph node sections split into 270 slides with pixel-level annotations for training and 130 unlabeled slides for testing. Based on the pixel-level annotations on the training set, indicating the location of benign and malignant tissue the PatchCamelyon dataset was created with a positive/negative balance of 50/50. PatchCamelyon dataset contains 327,680 patches with the size of 96 x 96 pixels extracted from Camelyon16 with a 10x magnification. The PatchCamelyon task is to classify the images into benign or malignant cases based on expert segmentations of malignant tissue.

PatchCamelyon dataset was chosen first because of the quality of the images which were curated and segmented by pathologists. Second, it is a large dataset with a high number of images allowing the training of a highly accurate model without overfitting. Third, unlike simple image datasets like MNIST, the images in PatchCamelyon are realistic, and in the task of malignancy detection, the introduction of a region with malignancy features on a benign image produces realistic counterfactuals.

Pre-processing The original PatchCamelyon dataset contains colored images of patches of Hematoxylin and Eosin (H&E) stained whole-slide images of sentinel lymph node sections lesions with a fixed size of 96x96 pixels. The patches were first normalized into a fixed range between 0 to 1 to improve the optimization process. Also to avoid overfitting, data augmentation was applied to increase the diversity of the images of the training set. Images were randomly flipped both vertically and horizontally, and random brightness augmentation was used.

K-fold cross-validation with 5 folds was used to better evaluate the model and the saliency map methods.

5.2.2 Model Training

Three CNN architectures were explored: VGG16 [210], ResNet50 [104] and InceptionV3 [218]. These CNN architectures were chosen as they represent state-of-the-art approaches for many medical imaging tasks and they achieved good results in the specific task of metastasis detection in the Camelyon16 challenge [32].

For the purpose of achieving the best performance on the medical imaging task of tumor detection of the PatchCamelyon dataset, the three CNNs were pre-trained on the ImageNet dataset [67] and fine-tuned on the medical dataset with a low learning rate.

VGG16 The last three fully-connected layers were replaced with a fully-connected layer with 128 neurons followed by a dropout layer with a 0.3 dropout rate and an output layer. An Adam optimizer with a 1e-4 learning rate was used.

ResNet50 The last fully-connected layer was removed and replaced with a fully-connected layer with 64 neurons followed by a dropout layer with a 0.2 dropout rate and an output layer. An SGD optimizer with a 5e-4 learning rate was used.

InceptionV3 The last fully-connected layer was replaced with two fully-connected layers with 64 neurons each followed by a dropout layer with a 0.2 dropout rate and an output layer. All the layers were trained first with an Adam optimizer with a learning rate of 1e-4.

All the models were trained with a batch size of 64 images, for 100 maximum epochs which were cut short by stooping the training early when the validation accuracy stops improving. The learning rate also was reduced by a factor of 0.2 when the validation accuracy also plateaus.

5.2.3 Image Perturbation

The perturbation strategy developed is targeted at evaluating the faithfulness of the saliency maps. It was developed on the assumption that if the saliency map highlights a region of the image as being important in the prediction of the tumor, replacing it with a region from a normal case will change the saliency map.

The PatchCamelyon dataset was produced based on the domain expert's annotations (i.e. tumor segmentations), producing positive cases if tumor pixels are present in its inner 32x32 square. Due to this mining schema by perturbing the inner region, we can have an informative perturbation mechanism. Intuitively, the larger the inner region is, the higher the difference between the saliency maps should be. We have selected three different degrees of perturbation, small (8x8 region), medium (16x16 region), and big large (32x32 region) to evaluate the saliency map methods.

By inspecting the correlation between the increase in the degree of perturbation and the change in the interpretability metrics, a ranking of saliency map methods can be found.

While this perturbation strategy is simple, it tackles a big problem that exists currently in the evaluation of interpretability methods, Out-of-Distribution perturbations. Common existing evaluation strategies use Gaussian noise on important pixels (high-valued pixels on saliency maps) and randomizing input images. The problem with using noise or uniform values (black or mean) to perturb the image is that the transformed image becomes to different from the images used to train the model, making its prediction and subsequent

interpretation too erroneous. By using an existing region rather than isolated pixels or non-natural noise we are making sure that the images remain inside of the distribution and that the interpretation is more trustworthy.

5.2.4 Saliency Map Extraction

We have selected a vast number of saliency map methods, for two major groups. This includes back-propagation methods and occlusion or perturbation methods [1].

Saliency Saliency [209], or gradient back-propagation, is a simple method where the pixel's sensitivity is given by the gradient of the loss function for the class we are interested in with respect to the input pixels. Each saliency map pixel's value represents how much a tiny change in the pixel would change the classification score for class c. The gradient method generates a highly noisy saliency map.

Deconvolution Deconvolution [251] provides a way to map the activation of intermediate layers back to the input layer. This mapping is performed by a Deconvolutional Network which attaches to the CNN layers and performs the opposite operation. For example, the unpooling layer does the inverse of the pooling layer.

Gradient Back-propagation Gradient Back-propagation [209], or VanillaGradient, also known as gradient back-propagation, is a simple method where the pixel's sensitivity is given by the gradient of the loss function for the class we are interested in with respect to the input pixels. Each saliency map pixel's value represents how much a tiny change in the pixel would change the classification score for class c. The gradient method generates a highly noisy saliency map.

GradientsInput GradientsInput is a variation Gradient Back-propagation [209] where the gradient of the pixel is multiplied by the value of the pixel. While the gradient indicates the importance of the pixel, the input value tells us how strongly this dimension is expressed.

SmoothGrad SmoothGrad [213] is a variant of Gradient Back-propagation where the saliency map is smoothed out by creating noisy copies of the input image and then averaging the gradient saliency maps of these noisy images. The resulting effect is a more sharp saliency map with less noise on irrelevant regions.

[1] The implementation of the saliency map methods was done using the tf-explain toolbox [145].

Grad-CAM Gradient-weighted Class Activation Map [203] Unlike gradient back-propagation, the gradient is not back-propagated all the way back to the image. Instead, they are back-propagating from the output neuron to the last convolutional layer to compute the weights for the feature maps. Resulting in a weight that represents the importance of the feature map for the prediction of the class.

Guided Grad-CAM Guided Gradient-weighted Class Activation Map [203] is variant of Grad-CAM with the goal of producing saliency maps with more fine-grained details at the pixel level. This is done by combining the saliency maps of Grad-CAM and Guided Back-propagation using element-wise multiplication.

Integraded Gradients Integrated Gradient [217] saliency map is computed by drawing a straight line in the network feature space from a baseline image and the input image and accumulating the gradients at all points along the path. The baseline image should ideally have no signal, so similarly to the original paper, we have also used a zero-based image (i.e. black image) as our baseline.

Occlusion Sensitivity Occlusion Sensitivity [251] computes the importance of regions of the image by inspecting if there is a drop in the confidence of the model in the predicted class when the region is occluded using a mask.

SHAP Shapley Additive explanations (SHAP) [142] requires the training of a distinct predictive model for each distinct combination of input features. By inspecting the gap between the predictions of two predictive models when a feature is added/subtracted, we can infer the importance of the feature in the prediction. Features whose presence or absence produced a large gap in predictions have large Shapley values and are deemed important.

LIME Local Interpretable Model-agnostic Explanations (LIME) [185] first produces an artificial dataset by occluding each feature of the original data points. Weights are assigned to the generated data points based on the closeness to the original point. Based on the generated weighted data a linear regression model is trained. The coefficients of the linear regression correspond to the importance of the input features to the model's predictions.

The methods explored in this work were Gradient Back-propagation (i.e. VanillaGradients), GradientsInputs, GradCAM, Guided-GradCAM, IntegratedGradients, OcclusionSensitivity, and SmoothGrad.

5.2.5 Saliency Map Evaluation

The metrics used to evaluate the saliency models were adapted for evaluating saliency maps [42] [2]. Saliency models have been frequently used in the literature to predict the position where an individual looks when viewing an image. The heatmap produced by the saliency model represents the probability of an individual looking at the pixel and is then compared to a ground-truth fixation map. We use the saliency metrics to compare the perturbed saliency map with the original saliency map which we have defined to be the ground-truth.

Following Riche *et al.* [187] we divided the metrics based on location-based or distribution-based and similarity or dissimilarity. This classification is summarized in Table 5.1.

Metrics	**Location-based**	**Distribution-based**
Similarity	jAUC, bAUC, sAUC, NSS, IG	SIM, CC
Dissimilarity	MSE, MAE	KL

Table 5.1: Saliency metrics divided by type.

Location-based metrics consider saliency map values as discrete locations at different threshold levels, while distribution-based metrics treat both saliency maps as continuous distributions.

Similarity metrics measure how similar two saliency maps are, while dissimilar metrics measure how dissimilar they are. Similarity should have higher values when we expect the saliency maps to not change (i.e. introduce evidence from the same class) while being lower when we expect a change (i.e. introduce evidence from a different class). The opposite should happen with the dissimilarity metrics.

Location-based metrics

Location-based metrics score saliency maps regarding how accurately they predict discrete pixel locations.

Area under ROC Curve (AUC) The Area under the ROC curve is the most widely used metric for evaluating saliency maps. When computing the AUC, the saliency map is treated as a binary classifier at various threshold values and the ROC curve represents the true and false positive rates for each threshold value.

Different AUC implementations differ in how true and false positives are calculated.

AUC-Judd (jAUC) uses a threshold level as a cut-off value to determine if pixel values in a saliency map are positive or negative.

[2]The saliency metrics were implemented based on the code provided in Bylinskii *et al.* [42].

AUC-Borji (bAUC) uses a uniform random sample of image pixels as negatives and defines the saliency map values above the threshold at these pixels as false positives.

Shuffled AUC (sAUC) penalizes center bias by sampling negative samples predominantly from the image center.

These saliency metrics were adapted by binarizing the ground truth saliency map (pre-perturbation) by setting a threshold and selecting the most salient pixels.

Normalized Scanpath Saliency (NSS) The Normalized Scanpath Saliency (NSS) is a similarity metric that measures the average normalized saliency map values of the locations of the ground truth saliency map.

Given a saliency map P and a binarized ground truth saliency map Q^B, NSS can be computed so:

$$NSS(P, Q^B) = \frac{1}{N} \sum_i \overline{P_i} \times Q_i^B \tag{5.1}$$

$$\text{where } N = \sum_i Q_i^B \text{ and } \overline{P} = \frac{P - \mu(P)}{\sigma(P)} \tag{5.2}$$

where i indexes the i-th pixel, and N is the total number of fixated pixels.

NSS is sensitive to false positives, as the metric is normalised over all the positive pixels on the binarized ground truth saliency map.

Similar to the AUC variants, the ground truth saliency map was binarized.

Mean Average Error (MAE) Mean Average Error (MAE) represents the average difference between the model's prediction and ground-truth (Equation 5.3) and can be used in regression problems.

$$MAE = \frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i) \tag{5.3}$$

Mean Squared Error (MSE) Mean Square Error (MSE) represents the average squared difference between the model's prediction and ground-truth (Equation 5.4).

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2 \tag{5.4}$$

Information Gain (IG) Information Gain (IG) is a similarity information-theoretic metric that measures saliency model performance compared to a baseline.

Given a binary map of pixels Q_B, a saliency map P, and a baseline map B, information gain is computed as:

$$IG(P, Q^B) = \frac{1}{N} \sum_i Q_i^B [log_2(e + P_i) - log_2(e + B_i)] \tag{5.5}$$

where i indexes the i-th pixel, N is the total number of fixated pixels, e is for regularization and information gain is measured in bits per fixation.

A score above zero indicates the saliency map is better than the baseline at predicting the fixated locations.

Similar to the AUC variants and Infogain, the ground truth saliency map was binarized.

Distribution-based metrics

Distributed-based metrics treat pixel values and locations of ground truth saliency maps as possible samples from an underlying distribution.

Similarity (SIM) The similarity metric (SIM) measures the similarity between two distributions, viewed as histograms. SIM is computed as the sum of the minimum values at each pixel, after normalizing the input maps. Given a saliency map P and a continuous fixation map Q^D:

$$SIM(P, Q^D) = \sum_i min(P_i, Q_i^D) \tag{5.6}$$

$$\text{where } \sum_i P_i = \sum_i Q_i^D = 1 \tag{5.7}$$

iterating over discrete pixel locations i.

A SIM of one indicates the distributions are the same, while a SIM of zero indicates no overlap.

Pearson's Correlation Coefficient (CC) Pearson's Correlation Coefficient (CC) is a statistical method for measuring how correlated or dependent two variables are. If we consider the distribution of pixels in the saliency map Q^D, and the saliency map P as random variables, we can measure their linear relationship:

$$CC(P, Q^D) = \frac{\sigma(P, Q^D)}{\sigma(P) \times \sigma(Q^D)} \tag{5.8}$$

where $\sigma(P, Q^D)$ is the covariance of P and Q^D. It is a similarity metric, which means that high positive CC values occur at locations where both the saliency map and ground truth saliency map have values of similar magnitudes.

Kullback-Leibler divergence (KL) The Kullback-Leibler divergence (KL) is a dissimilarity metric based on general information theory and it measures the difference between two probability distributions.

The KL metric takes as input a saliency map P and a ground truth saliency map Q^D, and evaluates the loss of information when P is used to approximate Q^D:

$$KL(P, Q^D) = \sum_i Q_i^D log(\frac{Q_i^D}{P_i}) \tag{5.9}$$

where e is a regularization constant.

One characteristic of KL is that it penalizes very sparse saliency maps.

Performance Metrics

One way to measure the impact that the perturbation strategy has on the underlying model is to measure the change in confidence in predictions when noise is introduced. We have selected a number of performance metrics for evaluating a classification model, namely accuracy, precision, recall, and Mean Squared Error (MSE). The metrics are not only used to compare with actual labels of the dataset but also among predictions before and after the perturbation.

True positives (TP) and true negatives (TN) represent the instances correctly classified by the model as being positive and negative, respectively. On the other-hand false positive (FP) and false negative (FN) represent the instances in which the model incorrectly classified as being positive and negative, respectively.

Accuracy represents the ratio of examples correctly classified (Equation 5.10).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.10}$$

Precision measures which proportion of the samples predicted as positive are actually positive:

$$Precision = \frac{TP}{TP + FP} \tag{5.11}$$

Recall on the other-hand measures which proportion of the actual positive samples were collectively predicted as positive:

$$Recall = \frac{TP}{TP + FN} \tag{5.12}$$

In this section, we have introduced an approach for the evaluation of saliency maps using realistic perturbations which avoid the problem of creating out-of-distribution images. We have also proposed the adaptation of saliency metrics used to evaluate saliency models for the comparison of saliency maps. The proposed approach was validated by evaluating 7 saliency map methods on a digital pathology dataset called PatchCamelyon.

5.3 Results

We have trained three CNNs on the PatchCamelyon dataset, VGG16, RestNetV3, and InceptionV3. Table 5.2 shows the classification results of the CNNs averaged across all folds. As can be seen, all the architectures are able to achieve very high performance with state-of-the-art results.

model	accuracy	auc	precision	recall
InceptionV3	0.965 (0.004)	**0.992 (0.001)**	0.967 (0.004)	0.964 (0.007)
ResNet50	0.965 (0.003)	**0.992 (0.001)**	0.97 (0.009)	0.962 (0.006)
VGG16	**0.971 (0.001)**	0.991 (0.002)	**0.974 (0.013)**	**0.968 (0.014)**

Table 5.2: Average performance metrics of the three CNN architectures.

To test the natural perturbation, three scenarios were implemented: NN - where a perturbation of normal tissue was added in an image of normal tissue, NT - where a perturbation of normal tissue was added in an image of tumor tissue, and TN - where a perturbation of tumor tissue was added in an area of image tissue.

To qualitatively evaluate the perturbation method and the saliency map methods we have selected one example for each of the NT and TN scenarios. As can be seen in Figure 5.2, when introducing a malignant region in a normal sample (i.e. top row) the saliency maps change to highlight the perturbed region. Likewise, when introducing a normal region in a malignant sample the perturbed region becomes less highlighted. This shows that our perturbation method is effective in changing the evidence present on the image and the model takes into account the perturbed regions and does not focus on artifacts. There is also

an apparent distinction between saliency maps produced by GradientInputs, Integrated-Gradients, SmoothGrad, and VanillaGradients which produce more fine-grained saliency maps when compared with GradCAM, Guided-GradCAM, and OcclusionSensitivity.

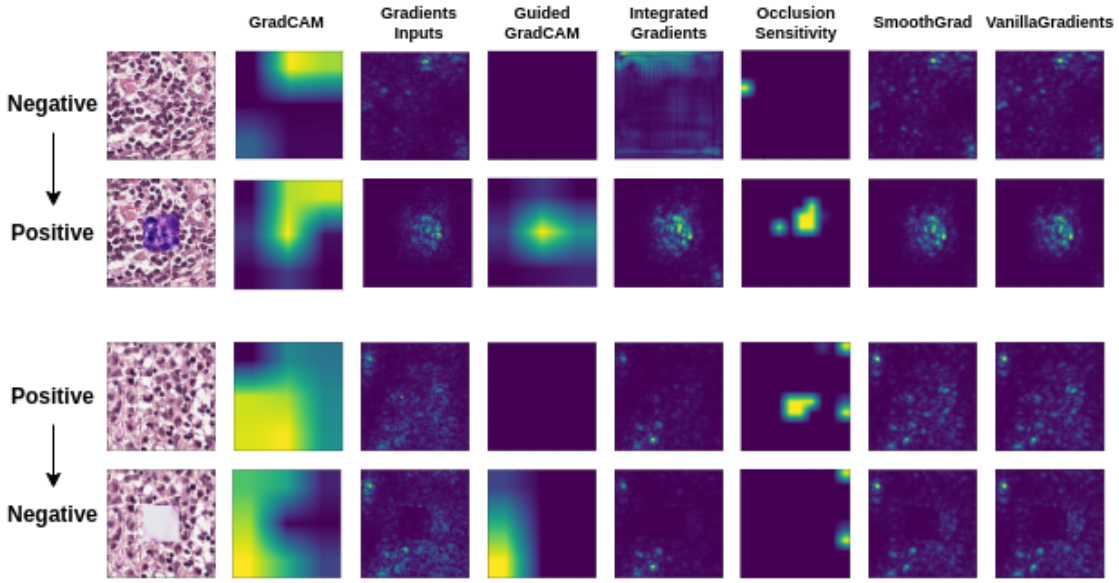
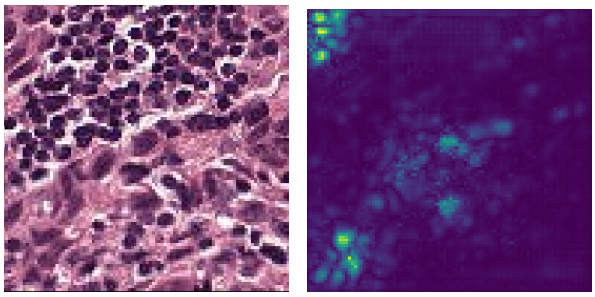


Figure 5.2: Qualitative analysis of saliency map methods before and after perturbation. Top row corresponds to a benign case perturbed with a malignant region (TN), while the bottom row corresponds to a malignant case perturbed with a benign region (NT). Yellow and green pixels represent malignancy evidence while blue pixels represent to normal pixels.



(a) Original tumor image (b) Saliency map produced by VanillaGradients

Figure 5.3: Example image with multiple regions depicting evidence of malignancy and the correspondent saliency map.

Nevertheless, there are a few cases, such as in Figure 5.3, where multiple regions depicting evidence of malignancy exist in the image. The optimal approach would be to perturb all of the regions with evidence. In Section 5.5 we propose a future extension of our approach for such cases.

Below, each of the four research questions stated in the introduction section will be analyzed separately.

5.3.1 What is the impact of the different perturbations on the model's predictions?

First, we will analyze the impact that perturbations have in the class that the CNN predicts. It is expected that if the perturbation introduces pixels from the same class as before, the predicted class should be the same and the accuracy remains high. However, if the pixels introduced are from a different class then the class predicted by the model should change and the accuracy should drop.

Table 5.3 compares the impact that the perturbation strategy has on the model's accuracy based on the class of the perturbed regions - whether it is from the same or different class than the original - and based on the level of perturbation - the size of the perturbed region (i.e. 8x8, 16x16 or 32x32). More specifically the table shows the average change in accuracy when comparing the predicted class before and after perturbation. Changes between saliency map methods appear due to the restriction that at least a pixel in the region should be highlighted. From the results we have validated that the proposed perturbation strategy is successful at introducing natural noise with evidence from a different class. There was a large drop in accuracy when comparing the perturbation from different classes with the same class.

	Same Class			Different Class		
method	8x8	16x16	32x32	8x8	16x16	32x32
GradCAM	0.989	0.974	0.946	0.988	0.927	0.679
GradientsInputs	0.996	0.974	0.963	0.992	0.928	0.699
Guided-GradCAM	0.990	0.974	0.947	0.988	0.925	0.678
IntegratedGradients	0.989	0.972	0.947	0.988	0.925	0.678
OcclusionSensitivity	0.990	0.974	0.948	0.988	0.926	0.676
SmoothGrad	0.988	0.972	0.944	0.988	0.926	0.684
VanillaGradients	0.994	0.981	0.976	0.989	0.922	0.689

Table 5.3: Average accuracy of predicted class before and after different levels of perturbation. Comparison of results after introducing regions from the same class or different classes.

5.3.2 Are saliency map methods sensitive to the introduction of natural noise?

We first extracted the saliency metrics for each scenario (NN, NT, TN), comparing the original and the perturbed saliency map. Then we calculated the change in metrics when the regions perturbed increased (8x8 to 32x32) and compared the three scenarios based on these differences.

In scenario NT it is expecting an increase of dissimilarity metrics (↑) and a decrease of similarity metrics (↓). As can be seen in Table 5.4, although most saliency map methods are sensitive to the introduction of natural noise. However, OcclusionSensitivity and

SmoothGrad were the exceptions presenting robust behavior.

method	↑ mse	↑ mae	↓ auc_judd	↓ auc_borji	↓ auc_shuff	↓ nss	↓ infogain	↓ sim	↓ cc	↑ kldiv
GradCAM	0.242	0.170	-0.100	-0.097	-0.081	-0.254	0.094	-0.036	-0.236	0.226
GradientsInputs	0.386	0.326	-0.047	-0.046	-0.025	-1.178	0.070	-0.007	-0.895	0.106
Guided-GradCAM	0.420	0.301	-0.046	-0.047	-0.037	-0.122	0.045	-0.016	-0.103	0.064
IntegratedGradients	0.152	0.116	-0.031	-0.029	-0.024	-0.379	0.019	-0.020	-0.368	0.038
OcclusionSensitivity	0.000	-0.083	-0.002	0.000	0.000	0.007	-0.079	-0.060	-0.015	-1.000
SmoothGrad	0.000	0.000	-0.006	-0.007	-0.006	-0.070	0.003	-0.003	-0.058	0.002
VanillaGradients	0.344	0.273	-0.027	-0.033	-0.023	-0.658	0.039	-0.005	-0.575	0.044

Table 5.4: Comparison of the saliency metrics' change when the perturbed region is increased between the scenarios NN and NT.

method	↑ mse	↑ mae	↓ auc_judd	↓ auc_borji	↓ auc_shuff	↓ nss	↓ infogain	↓ sim	↓ cc	↑ kldiv
GradCAM	0.106	0.092	0.005	-0.011	-0.010	0.033	-0.035	0.010	0.013	-0.336
GradientsInputs	-0.017	-0.038	0.108	0.094	0.113	1.496	-0.056	0.037	1.470	-0.097
Guided-GradCAM	-0.049	-0.011	0.015	0.066	0.092	0.051	-0.026	0.019	0.052	-0.077
IntegratedGradients	0.076	0.020	0.052	0.047	0.055	0.750	-0.028	0.012	0.615	-0.024
OcclusionSensitivity	0.167	0.000	0.009	0.008	0.000	0.035	-0.094	0.060	-0.014	-0.833
SmoothGrad	-0.041	-0.008	0.001	0.002	-0.004	0.041	-0.005	-0.003	-0.004	-0.002
VanillaGradients	-0.063	-0.051	0.099	0.084	0.069	1.283	-0.123	0.035	0.974	-0.280

Table 5.5: Comparison of the saliency metrics' change when the perturbed region is increased between the scenarios NN and TN.

On the other hand, with of the strategy TN (Table 5.5) it is expected a similar behavior for the saliency metrics, in the consequent increasing of the dissimilarity values and the decreasing of similarity values. Although the changes in this scenario were not quite as noticeable.

5.3.3 How are saliency maps methods compared to each other in terms of perturbation robustness?

To compare saliency map methods to each other in terms of perturbation robustness, we have extracted the saliency metrics for each saliency map method for the NT scenario at different levels of perturbation (i.e. 8x8, 16x16, 32x32). A comparison was made using the Friedman rank test and the mean of the 7 saliency map methods was compared with each other. The results were divided into three sets for each of the three levels of perturbation. The obtained ranks for the 8x8, 16x16 and the 32x32 perturbation levels are shown in Table 5.6, Table 5.7 and Table 5.8, respectively. Following the work of Denšar [66] with $N = 10$ (number of metrics) and $k = 7$ (number of saliency map methods) the 7 methods were compared among themselves for a 5% significance level using the two-tailed Nemenyi test [66]. It was possible to obtain CD = 2.849, where CD is the critical value for the difference of mean ranks between the 7 methods.

Next, we compared the 7 methods, this time without separating the different region sizes, with a two-tailed Nemenyi test, across the 30 evaluation metrics and a 5% significance level (Table 5.9). For this setup, we obtained a CD = 1.645. The CD is the critical value

method	mse	mae	auc_judd	auc_borji	auc_shuff	nss	infogain	sim	cc	kldiv	average
GradCAM	6.5	5	3	6.5	5	2.5	6.0	2	2.5	6	4.50
GradientsInputs	4	4	3	2	1	4	7	7	4	7	4.30
Guided-GradCAM	3	3	3	4.5	5	2.5	1	1	1	1	2.50
IntegratedGradients	2	2	1	1	2.5	1	3	4	2.5	3	2.20
OcclusionSensitivity	5	6	6	4.5	5	7	5	5	6	5	5.45
SmoothGrad	6.5	7	7	6.5	7	6	4	6	7	4	6.10
VanillaGradients	1	1	5	3	2.5	5	2	3	5	2	2.95

Table 5.6: Ranks of saliency map methods over the 10 saliency metrics with 8x8 region perturbation.

method	mse	mae	auc_judd	auc_borji	auc_shuff	nss	infogain	sim	cc	kldiv	average
GradCAM	5	5	1	1	1.5	1	4	2	1	3	2.45
GradientsInputs	2	2	4	4.5	5	5	3	7	4	5	4.15
Guided-GradCAM	4	4	2	3	3.5	2	1	1	2	1	2.35
IntegratedGradients	3	3	3	2	1.5	3	5	3	3	4	3.05
OcclusionSensitivity	6	6	6	7	6.5	7	7	5	6.5	7	6.40
SmoothGrad	7	7	7	6	6.5	6	6	6	6.5	6	6.40
VanillaGradients	1	1	5	4.5	3.5	4	2	4	5	2	3.20

Table 5.7: Ranks of saliency map methods over the 10 saliency metrics with 16x16 region perturbation.

method	mse	mae	auc_judd	auc_borji	auc_shuff	nss	infogain	sim	cc	kldiv	average
GradCAM	2.5	4	1	1	1	1	1	1	1	1	1.45
GradientsInputs	4	3	3	3	3	3	4	6.5	3	4	3.65
Guided-GradCAM	1	1	2	2	2	2	2	2	2	2	1.80
IntegratedGradients	5	5	4	4.5	4	5	5	3	5	5	4.55
OcclusionSensitivity	6	6	7	7	7	7	7	4	6	7	6.40
SmoothGrad	7	7	6	6	6	6	6	6	7	6	6.35
VanillaGradients	2.5	2	5	4.5	5	4	3	5	4	3	3.80

Table 5.8: Ranks of saliency map methods over the 10 saliency metrics with 32x32 region perturbation.

for the difference of mean ranks when taking into between the 7 methods among the 30 saliency metric values.

When taking into account the levels of perturbation separately and together, there is a significant statistical difference between Occlusion Sensitivity and SmoothGrad and the other saliency map methods.

5.3.4 How does the perturbation region's size impact the saliency maps?

Figure 2 shows what is the behaviour of the different saliency map methods when changing the size of the perturbed region on the TN scenario (based on Table 5.4). Three interesting patterns emerge when we analyze the plot. First, Grad-CAM and Guided-GradCAM show an increase in the rate of change between the first two region sizes and the last two. On the other hand, gradient-based methods such as GradientsInputs, IntegratedGradients, and VanillaGradients show an almost constant rate of change between region sizes. Lastly, as previously mentioned OcclusionSensitivity and SmoothGrad appear mostly invariant to

method	mean
GradCAM	2.74
GradientsInputs	3.89
Guided-GradCAM	2.31
IntegratedGradients	3.18
OcclusionSensitivity	6.06
SmoothGrad	6.39
VanillaGradients	3.43

Table 5.9: Mean rank of saliency map methods over the 30 saliency metrics.

perturbation.

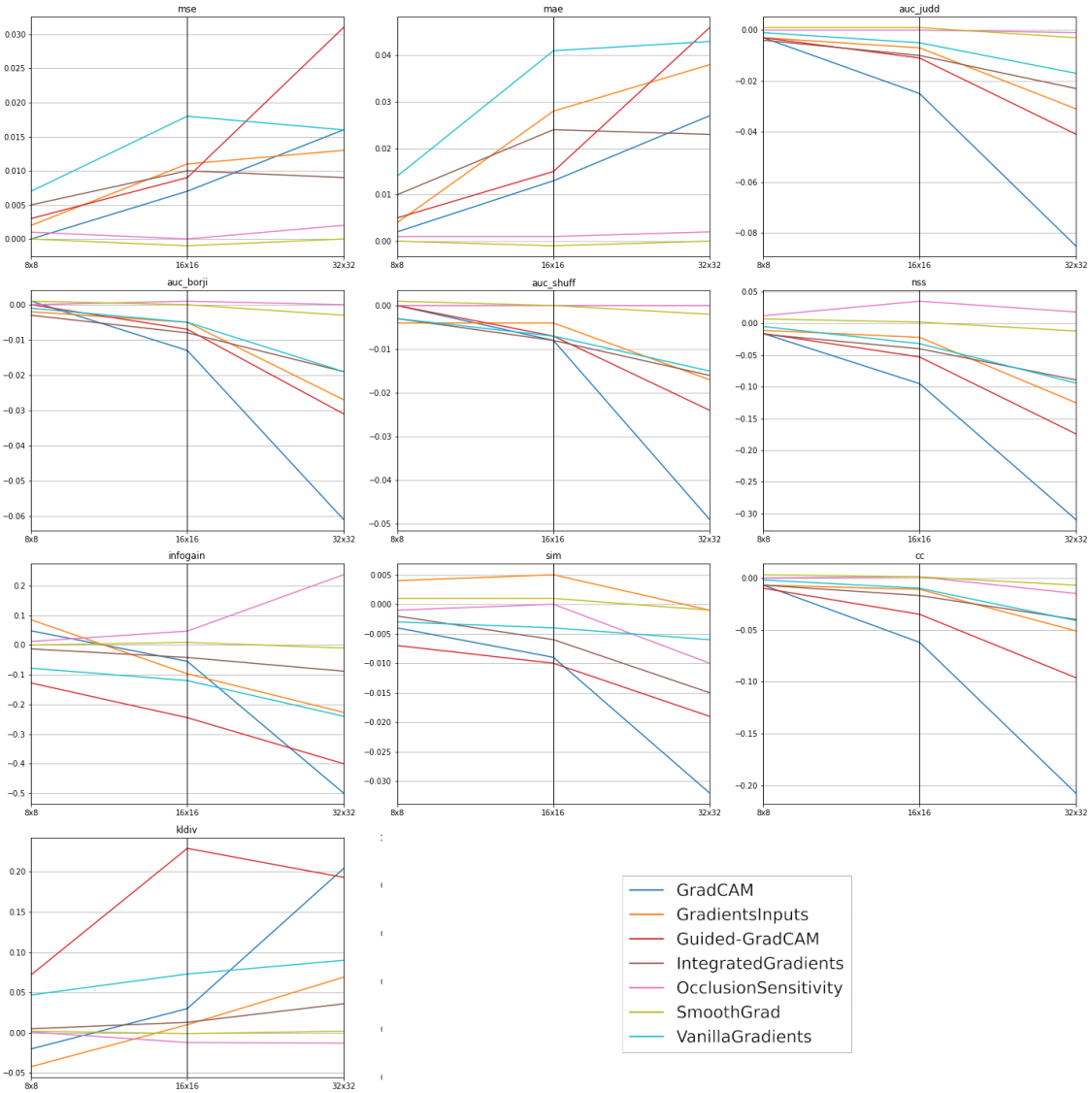


Figure 5.4: Plot showing metrics measuring the sensitivity of saliency map methods based on size of region perturbed.

5.4 Discussion

In this section, we first answer the research questions based on the results of our research. Then we discuss the theoretical and practical implications that our findings have on current and future research, proposing new future research directions.

5.4.1 Answering the Research Questions

What is the impact of the different perturbations on the model's predictions? The experiments' results showed a high drop in the model's confidence in the prediction when the addition of the noise is related to the opposite class in comparison to the one that exists in the same area before perturbation.

Are saliency map methods sensitive to the introduction of natural noise? OcclusionSensitivity and SmoothGrad methods have been found to be insensitive to natural noise when taking into account all the samples of the test set. Albeit for the other saliency maps methods, the behavior was different, especially in scenario NT. Finally, it is important to state that none of the saliency maps methods used present significant differences in performance.

How are saliency maps methods compared to each other in terms of perturbation robustness? OcclusionSensitivity and SmoothGrad have been shown to be more robust to perturbation presenting a significant statistical difference compared to the other methods. Grad-CAM and Guided-GradCAM have the overall best results but no significant statistical difference was found in comparison with the other methods.

How does the perturbation region's size impact the saliency maps? Grad-CAM and Guided-GradCAM present a specially high sensitivity to perturbations with a larger region (32x32) when compared with the other methods.

5.4.2 Implications on Saliency Map Evaluation

Typically, a saliency map sanity check [2, 197] has the following procedures:

- **Model Randomization:** randomize the weights of a model and evaluate the changes in saliency maps;
- **Label Randomization:** randomize the labels of the training samples, breaking the relationship between images and labels and forcing the model to memorize the

randomized labels without learning the original relationship of the data. The saliency maps should not be the same when the labels are randomized compared to when they are not;

- **Image Perturbation:** introduce Gaussian or uniform noise in regions highlighted by the saliency map and evaluate the performance loss of the model.

Our proposed approach complements the previous procedures by first introducing a perturbation method that is more consistent with data used to train the model - natural noise. It allows comparing how different saliency map methods' behavior changes with different types of regions - same class or different class. It also proposes the use of different saliency metrics that can further share light on the differences between saliency map methods.

5.5 Conclusions and Future Work

In this work, we have introduced a novel perturbation strategy for evaluating saliency maps. This strategy is based on the introduction of natural regions from other class cases and comparing the impact on the saliency maps based on both performance and saliency metrics - adapted from literature on saliency models.

Overall, in spite of some differences between methods, it is proven with this work that saliency maps can be a good strategy to interpret CNN models.

It is still uncertain what are the reasons for disparities in results. To move forward in this direction, the correlation between the characteristics of the saliency maps and the achieved results must be explored. For example, the sparsity of the salient pixels on the image, or the distribution of the salient pixel's values.

While the perturbation method focuses on a single region of the image, some of the saliency maps produced have in fact highlighted multiple regions of the image as having evidence of malignancy.

Based on these results future extensions of the current approach have to take into account the location of the salient regions to propose what regions of the image to perturb. Existing evaluation strategies [10, 109, 197] select the most salient pixels individually disregarding whether the pixels belong to the same region or not. Rather than looking at the pixels separately, we point to an extension of the proposed approach which takes the pixel values of the saliency map and finds clusters of pixels using the clustering algorithm K-means to find the optimal centroids to perturb with normal regions. The choice of saliency map methods can also be focused on the ones that have shown the best results, such as GradCAM and Guided-GradCAM.

Future work directions also include the validation of the results found in this study on a

larger set of datasets. More specifically datasets that possess domain expert annotations (e.g. segmentations of tumor regions) will be explored as these annotations can inform the perturbation mechanisms to target the correct pixels more deliberately.

Chapter 6

Validating Post-hoc Interpretability using Intrinsic Interpretability

Over the last few years, a vast number of interpretability methods [152] have been proposed for explaining CNNs' predictions, which can be divided in: intrinsic interpretability, referring to models that can be explained without further methods, usually by restricting the model's complexity; and post-hoc interpretability, referring to methods that explain the model decisions after training.

Saliency maps are one of the most popular post-hoc interpretability methods to explain CNN in the context of medical imaging [12]. This strategy illustrates the importance of individual pixels of the input image on the overall prediction of a CNN. The color or intensity of each pixel corresponds to the weight that the same pixel in the input image had on the classification process. Even though most methods have been shown continuously to be able to highlight regions with relevant medical evidence, the saliency maps generated by different methods exhibit a significant degree of variation, which is evidence of bias specific to each method that can not be overlooked. The qualitative analysis and evaluation of saliency map methods remain an open challenge.

A number of intrinsic interpretability strategies have also been adopted in medical imaging [12] with the objective of constraining the behavior of the classification model making it understandable to humans. Case-based reasoning closely approximates the reasoning process of physicians, which make decisions based on similar cases. Prototypes are special cases of case-based reasoning, where a small number of data points can represent the entire dataset [152]. To this end, ProtoPNet [51], a deep learning architecture based on a convolutional neural network, learns automatically the optimal prototypes and makes predictions based on the similarity of the instance to each prototype.

This network also generates an attribution map (i.e. heatmap) that highlights the location in the image which closely resembles the prototypes.

While both saliency map methods and ProtoPNet are capable of highlighting the importance of a region in the image for the prediction of the network, they do this through different mechanisms.

As intrinsic interpretability methods are considered to be more faithful to the underlying model's behavior [51] as they do not require an external method after training, it can become a good ground-truth of what regions of the image the saliency maps should focus. Therefore, we propose in this research work to use the intrinsic interpretable model's explanation as ground-truth and measure the overlap between it and the post-hoc explanations to validate them. Therefore, we propose an approach that automatically validates the results generated by post-hoc methods by comparing their results overlapped with the results achieved by an intrinsic interpretable method that is used as ground-truth. To this end, we adapted ProtoPNet for digital pathology and evaluated the overlap between it and 8 different saliency map methods using 10 saliency metrics.

In our experimental setup, we have trained three CNNs and three ProtoPNets based on three architectures (ResNet18, ResNet152 and DenseNet101) on the PatchCamelyon dataset (histopathologic scans of lymph node sections). Following the training of the networks, 8 different saliency map methods were used to extract saliency maps of the test set to be compared with attribution maps generated intrinsically by the ProtoPNets. To evaluate this approach, 10 metrics adapted for evaluating saliency maps [42] were used.

This chapter follows the following structure: Section 6.1 will briefly review related works in the literature. In Section 6.2 we present the different components of the study for evaluating the connection between saliency map methods and prototypes activation maps: 1) data selection, 2) model training, 3) prototypical parts extraction 4) saliency map extraction, and 5) saliency map evaluation. We present and discuss the results of the experiments in Section 6.3. Finally, in Section 6.4 we conclude with our final remarks and steps for future work.

6.1 Background

Interpretability is an important prerequisite for the adoption of computer-aided diagnosis systems. To this end, different computation competitions have emerged, and one of those is the Camelyon16 challenge [32] has the goal of evaluating algorithms on the task of automatic detection of breast cancer metastases in whole-slide images of hematoxylin and eosin (H&E) lymph node sections. Convolutional neural networks (CNNs) have been successful in this task, with approaches achieving area under the receiver operating curve (AUC) of 0.925, increasing to 0.995 when combined with pathologists' diagnosis (approximately

85 percent reduction in human error rate). But, although the potential is shown, without a clear understanding of their reasoning process, their application in clinical practice remains elusive.

There are two distinct approaches for this problem: intrinsic interpretability and post-hoc interpretability. Among post-hoc interpretability methods, the most adopted in medical imaging are saliency map methods. This can be especially seen in the oncological field [12] where half of the interpretability strategies employed to understand deep learning models are saliency map methods. These methods can be divided into two major groups: back-propagation methods and occlusion or sensitivity methods [152].

While not as popular, intrinsic interpretability strategies have also been adopted in medical imaging [12]. The first is the approximation of the network with an intrinsically interpretable model (i.e. decision rules) of similar performance but easier to understand [13]. Rather than generating saliency maps after training, it is also possible to incorporate in the network the heatmap generation through an attention mechanism or probability estimation (i.e. pixel-wise or patch-wise). The patch-wise heatmap has shown remarkable results in the diagnosis malignancy-based dermoscopic images [181]. Text explanations have become a reality with the advent of language models. An example is the training of a language model alongside the visual model for extracting text explanations while classifying the malignancy of mammograms Lee *et al.* [134].

Case-based reasoning is another intrinsic interpretability strategy that closely approximates the reasoning process of physicians, as they have to extract from their knowledge acquired from looking at similar cases. In this strategy, the classification of an instance is based on the classes of similar instances [29]. Prototypes are special cases of case-based reasoning, where a small number of data points are selected to represent all the data. Prediction of a data point can then be made by their similarity and dissimilarity with prototypes of either class. ProtoPNet [51] learns automatically the optimal prototypes in the data and makes predictions based on the similarity of the features extracted by a CNN and the features of the prototypes. With this added prototype layer, the network is capable of explaining the prediction based on a similarity score to each prototype and a heatmap denoting the location of each prototypical part. While ProtoPNet was not applied to medical imaging, an extension called IAIA-BL[29] which adds a component of fine annotation has shown great results in the classification of mammograms.

Although both saliency maps extracted via post-hoc interpretability methods (i.e. back-propagation) and attribution maps extracted via intrinsic interpretability methods (i.e. ProtoPNet) both highlight the important regions for the classification, they do this fundamentally through different mechanisms. Also, intrinsic interpretability methods are perceived as more trustworthy and more faithful to the behavior of the underlying model than post-hoc interpretability methods [51].

The evaluation of saliency maps is difficult because there is a lack of ground-truth on how the ideal saliency map should look. One strategy to evaluate a saliency map is by looking at the drop in confidence in the prediction when obscuring a region highlighted by the saliency map [10, 197]. This strategy demonstrated, in the digital pathology context, that reducing the complexity of the network had a positive impact on how much the saliency maps produced by the network reflected the model's reasoning [14].

Due to the fact of the higher trustworthiness of intrinsic interpretability methods, the attribution maps provided by ProtoPNet can be used as a ground-truth, allowing saliency map methods that produce saliency maps with a bigger overlap to the ground-truth to be considered more faithful to the model. Therefore, we developed an approach to evaluate saliency maps by overlapping the saliency map produced by a post-hoc method and the attribution map produced by an intrinsic methods

But, as we want to compare the heatmap generated by a saliency map method and prototypical part network, we can select the latter as our ground-truth and measure the overlap between them. Thus, we can use different metrics used for evaluating saliency models which produce heatmaps representing the probability of an individual looking at the pixel.

6.2 Method

Having in mind the main goal of this work which consists in comparing the saliency map methods and prototypical parts activation maps, a five-stage pipeline was defined in the experimental setup and is illustrated in Figure 6.1: 1) data selection, 2) model training, 3) prototypical parts extraction 4) saliency map extraction, and 5) saliency map evaluation.

6.2.1 Data Selection

To evaluate the proposed approach, the PatchCamelyon dataset [229] derived from the Camelyon16 dataset[32] was used. The Camelyon16 dataset contains 400 H&E stained whole-slide images of sentinel lymph node sections split into 270 slides with pixel-level annotations for training and 130 unlabeled slides for testing. PCam dataset contains 327,680 patches with size of 96 x 96 pixels extracted from Camelyon16 with a 10x magnification. The PatchCamelyon task is to classify the images into benign or malignant cases based on expert segmentations of malignant tissue.

PatchCamelyon dataset was chosen because of the quality of the images which were curated and segmented by pathologists and the large number of images.

The patches were first normalized into fixed range between 0 to 1 to improve the optimization process. Also to avoid overfitting, data augmentation was applied to increase

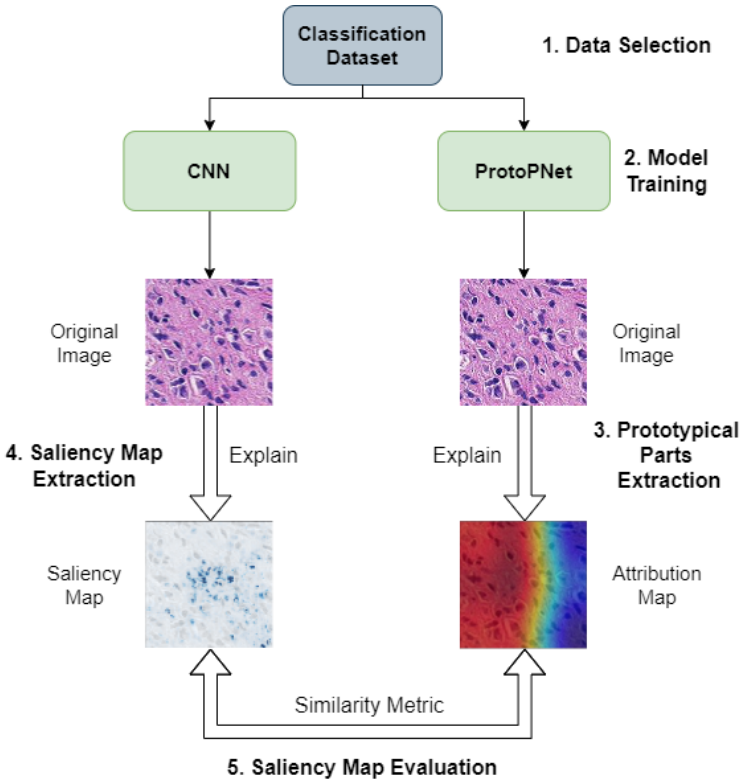


Figure 6.1: Automatic evaluation of saliency maps perturbation pipeline.

diversity of the images of the training set. Images were randomly flipped both vertically and horizontally, and random brightness augmentation was used.

6.2.2 Model Training

Three CNN architectures were explored: Resnet18 [104], Resnet152 [104] and DenseNet101 [111]. These CNN architectures were chosen as they represent state-of-the-art approaches for many medical imaging tasks and they achieved good results in metastasis detection in the Camelyon16 challenge [32].

For the purpose of achieving the best performance on the medical imaging task of tumor detection of the PatchCamelyon dataset, the three CNNs were pre-trained on the ImageNet dataset [67] and fine-tuned on the medical dataset with a low learning rate.

All the models were trained with a batch size of 64 images, for 100 maximum epochs which were cut short by stoping the training early when the validation accuracy stops improving. The learning rate also was reduced by a factor of 0.2 when the validation accuracy also plateaus.

Hyperparameter optimization through grid search was used to select thee optimal optimization algorithm and initial learning rate.

6.2.3 Prototypical Part Network

The ProtoPNet network [51] (Figure 6.2) is composed of a convolutional neural network that extracts features for the classification $z = f(x)$, followed by a prototype layer g, and a fully connected layer h. The CNN component is based on the three CNN architectures mentioned previously.

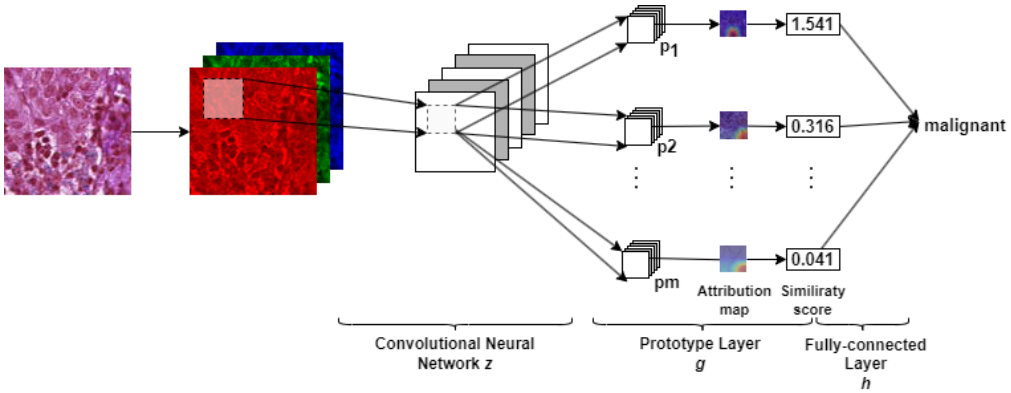


Figure 6.2: Architecture of ProtoPNet.

The prototype layer learns m prototypes which can represent the entire training set. Each prototype represents a different prototypical part or concept either of the malignant or benign class. After learning the prototypes, the layer computes the similarity scores between all patches of the image of the same size as the prototype using the L^2 distance function. The result is an attribution map for each prototype that indicates the regions of the image where it's most represented.

On the original ProtoPNet paper [51], the attribution map is created by aggregating the similarity scores using global max pooling. Rather than using max pooling, top-k average pooling was used [29] to use top 5% of the most activated convolutional patches that are closest to each prototype, instead of only the top most activated patch.

Finally, the similarity scores produced by the prototype layer are multiplied by the weight matrix of the fully connected layer to produce the output logits, which when passed through the softmax function produce the predicted probabilities for each class.

ProtoPNet training algorithm

Training of ProtoPNet is divided into three phases: (1) stochastic gradient descent (SGD) of layers before the last layer; (2) projection of prototypes; (3) convex optimization of the last layer.

In the first training stage, the convolutional layers' parameters and the prototype layer's parameters are optimized while keeping the last layer fixed.

During this phase, the loss function minimized is composed of the weighted sum of three losses: cross-entropy loss (CrsEnt), the cluster cost (Clst), and separation cost (Sep) (Equation 6.1):

$$\min \lambda_1 \text{CrsEnt} + \lambda_2 \text{Clst} + \lambda_3 \text{Sep} \tag{6.1}$$

The cross-entropy loss (CrsEnt) encourages the predicted classes to be the same as the target y_i in the training set composed of n instances (Equation 6.2):

$$\text{CrsEnt} = \frac{1}{n} \sum_{i=1}^{n} \text{CrsEnt}\left(h \circ g_p \circ f\left(x_i\right), y_i\right) \tag{6.2}$$

The cluster cost (Clst) encourages each training image to have some latent patch that is close to at least one prototype ($p_i \in P$) of its own class (Equation 6.3):

$$\text{Clst} = \frac{1}{n} \sum_{i=1}^{n} \min_{j:p_j \in P_{y_i}} \min_{z \in \text{patches}(f(x_i))} \|z - p_j\|_2^2 \tag{6.3}$$

The separation cost (Sep) encourages every latent patch of a training image to stay away from the prototypes not of its own class (Equation 6.4):

$$\text{Sep} = -\frac{1}{n} \sum_{i=1}^{n} \min_{j:p_j \notin P_{y_i}} \min_{z \in \text{patches}(f(x_i))} \|z - p_j\|_2^2 \tag{6.4}$$

During the second phase of the projection of prototypes, each prototype is projected onto the nearest training image patch from the same class as the prototype. This is done so that when interpreting the predictions made by the network, the prototypes represent actual patches of images in the training set.

Finally, in the last phase, the last layer is optimized using a convex optimization focusing on a sparsity property making the model rely more on positive evidence (i.e. predicting a class by using the prototypes of that class) and rely less on negative evidence (i.e. prototypes from the negative classes).

To find the optimal hyperparameters for each ProtoPNet architecture, grid search algorithm was used. The hyperparameters that were optimized were the number of prototypes, dimensions of prototypes, learning rates in each training phase, and weights for the three components of the loss functions - cross-entropy loss, cluster cost, and separation cost.

6.2.4 Saliency Map Methods

We have selected 8 popular saliency map methods of the two major groups: back-propagation methods and occlusion or sensitivity methods [1].

Back-propagation methods compute the relevance of a pixel by propagating a signal from the output neuron backward through the layers to the input image in a single pass [24]. Sensitivity methods compute pixel relevance by making small changes in the pixel value of the input image and compute how the changes affect the prediction [209].

Saliency Saliency [209], or gradient back-propagation, is a simple method where the pixel's sensitivity is given by the gradient of the loss function for the class we are interested in with respect to the input pixels. Each saliency map pixel's value represents how much a tiny change in the pixel would change the classification score for class c. The gradient method generates a highly noisy saliency map.

Deconvolution Deconvolution [251] provides a way to map the activation of intermediate layers back to the input layer. This mapping is performed by a Deconvolutional Network which attaches to the CNN layers and performs the opposite operation. For example, the unpooling layer does the inverse of the pooling layer.

GuidedBackprop GuidedBackprop [214] adds an additional guidance signal from the higher layers to the usual back-propagation. It combines deconvolution with back-propagation, by masking out negative values from either method.

SmoothGrad SmoothGrad [213] is a variant of Gradient Back-propagation where the saliency map is smooth out by creating noisy copies of the input image and then average the gradient saliency maps of these noisy images. The resulting effect is a more sharp saliency map with less noisy on irrelevant regions.

Integraded Gradients Integrated Gradient [217] saliency map is computed by drawing a straight line in the network feature space from a baseline image and the input image and accumulating the gradients at all points along the path. The baseline image should ideally have no signal, so similarly to the original paper, we have also used a zero-based image (i.e. black image) as our baseline.

Occlusion Sensitivity Occlusion Sensitivity [251] computes the importance of regions of the image by inspecting if there is a drop in the confidence of the model in the predicted

[1] The implementation of the saliency map methods was done using the pytorch captum toolbox [124].

class when the region is occluded using a mask.

SHAP Shapley Additive explanations (SHAP) [142] requires the training of a distinct predictive model for each distinct combination of input features. By inspecting the gap between the predictions of two predictive models when a feature is added/subtracted, we can infer the importance of the feature in the prediction. Features whose presence or absence produced a large gap in predictions have large Shapley values and are deemed important.

LIME Local Interpretable Model-agnostic Explanations (LIME) [185] first produces an artificial dataset by occlusing each feature of the original datapoints. Weights are assigned to the generated datapoints based on the closeness to the original point. Based on the generated weighted data a linear regression model is trained. The coefficients of the linear regression correspond to the importance of the input features to the model's predictions.

6.2.5 Saliency Map Evaluation

The metrics used to evaluate saliency models were adapted for evaluating saliency maps and attribution maps [42]. The main task of a saliency model is predicting eye movements made during image viewing. The saliency model produces a heatmap in which the pixel value represents the probability of an individual looking at the pixel. Evaluation of the saliency model consists of comparing the heatmaps to the ground-truth fixation map. We use the saliency metrics to compare the saliency map extracted from the CNNs and the attribution map of each prototype produced by the ProtoPNet.

Following Riche *et al.* [187] we divided the metrics based on location-based or distribution-based and similarity or dissimilarity. This classification is summarized in Table 6.1.

Metrics	**Location-based**	**Distribution-based**
Similarity	jAUC, bAUC, sAUC, NSS, IG	SIM, CC
Dissimilarity	MSE, MAE	KL

Table 6.1: Saliency metrics divided by type.

Location-based metrics consider saliency map values as discrete locations at different threshold levels, while distribution-based metrics treat both saliency maps as continuous distributions.

Similarity metrics measure how similar two saliency maps, while dissimilar metrics measure how dissimilar they are. Similarity should have higher values when we expect the saliency maps to not change (i.e. introduce evidence from the same class) while being lower when we expect a change (i.e. introduce evidence from a different class). The opposite should happen with the dissimilarity metrics.

Location-based metrics

Location-based metrics score saliency maps regarding how accurately they predict discrete pixel locations.

Area under ROC Curve (AUC) The Area under the ROC curve is the most widely used metric for evaluating saliency maps. When computing the AUC, the saliency map is treated as a binary classifier at various threshold values and the ROC curve represents the true and false positive rates for each threshold value.

Different AUC implementations differ in how true and false positives are calculated.

AUC-Judd (jAUC) use a threshold level as a cut-off value to determine if pixel values in a saliency map are positives or negative.

AUC-Borji (bAUC) uses uniform random sample of image pixels as negatives and defines the saliency map values above threshold at these pixels as false positives.

Shuffled AUC (sAUC) penalizes center bias by samplying negative samples predominantly from the image center.

These saliency metrics were adapted by binarizing the ground truth saliency map by setting a threshold and selecting the most salient pixels.

Normalized Scanpath Saliency (NSS) The Normalized Scanpath Saliency (NSS) is a similarity metric which measures the average normalized saliency map values of the locations of the ground truth saliency map.

Given a saliency map P and a binarized ground truth saliency map Q^B, NSS can be computed so:

$$NSS(P, Q^B) = \frac{1}{N} \sum_i \overline{P_i} \times Q_i^B \tag{6.5}$$

$$\text{where } N = \sum_i Q_i^B \text{ and } \overline{P} = \frac{P - \mu(P)}{\sigma(P)} \tag{6.6}$$

where i indexes the i-th pixel, and N is the total number of fixated pixels.

NSS is sensitive to false positives, as the metric is normalied over all the positive pixels on the binarized ground truth saliency map.

Similar to the AUC variants, the ground truth saliency map was binarized.

Mean Average Error (MAE) Mean Average Error (MAE) represents the average difference between the model's prediction and ground-truth (Equation 6.7) and can be used in regression problems.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i) \tag{6.7}$$

Mean Squared Error (MSE) Mean Square Error (MSE) represents the average squared difference between the model's prediction and ground-truth (Equation 6.8).

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 \tag{6.8}$$

Information Gain (IG) Information Gain (IG) is a similarity information theoretic metric that measures saliency model performance compared to a baseline.

Given a binary map of pixels Q_B, a saliency map P, and a baseline map B, information gain is computed as:

$$IG(P, Q^B) = \frac{1}{N}\sum_{i} Q_i^B[log_2(e + P_i) - log_2(e + B_i)] \tag{6.9}$$

where i indexes the i-th pixel, N is the total number of fixated pixels, e is for regularization, and information gain is measured in bits per fixation.

A score above zero indicates the saliency map is better than the baseline at predicting the fixated locations.

Similar to the AUC variants and Infogain, the ground truth saliency map was binarized.

Distribution-based metrics

Distributed-based metrics treats pixel values and locations of ground truth saliency maps as possible samples from an underlying distribution.

Similarity (SIM) The similarity metric (SIM) measures the similarity between two distributions, viewed as histograms. SIM is computed as the sum of the minimum values at each pixel, after normalizing the input maps. Given a saliency map P and a continuous fixation map Q^D:

$$SIM(P, Q^D) = \sum_i min(P_i, Q_i^D) \tag{6.10}$$

$$\text{where } \sum_i P_i = \sum_i Q_i^D = 1 \tag{6.11}$$

iterating over discrete pixel locations i.

A SIM of one indicates the distributions are the same, while a SIM of zero indicates no overlap.

Pearson's Correlation Coefficient (CC) The Pearson's Correlation Coefficient (CC) is a statistical method for measuring how correlated or dependent two variables are. If we consider the distribution of pixels in the saliency map Q^D, and the saliency map P as random variables, we can measure their linear relationship:

$$CC(P, Q^D) = \frac{\sigma(P, Q^D)}{\sigma(P) \times \sigma(Q^D)} \tag{6.12}$$

where $\sigma(P, Q^D)$ is the covariance of P and Q^D. It is a similarity metric, which means that high positive CC values occur at locations where both the saliency map and ground truth saliency map have values of similar magnitudes.

Kullback-Leibler divergence (KL) The Kullback-Leibler divergence (KL) is a dissimilarity metric based on general information theory and it measures the difference between two probability distributions.

The KL metric takes as input a saliency map P and a ground truth saliency map Q^D, and evaluates the loss of information when P is used to approximate Q^D:

$$KL(P, Q^D) = \sum_i Q_i^D log(\frac{Q_i^D}{P_i}) \tag{6.13}$$

where e is a regularization constant.

One characteristic of KL is that it penalizes very sparse saliency maps.

Performance Metrics

One way to compare the predictions of the CNN and ProtoPNet is to measure the difference in confidence in the class from both models. We have selected a number of performance metrics for evaluating a classification model, namely AUC, accuracy, precision, recall, and

Mean Squared Error (MSE). These metrics are not only used to measure the performance of each model by comparing their predictions with actual labels but also to measure between predictions of CNNs and ProtoPNet with the same pre-trained base model.

True positives (TP) and true negatives (TN) represent the instances correctly classified by the model as being positive and negative, respectively. In the other-hand false positive (FP) and false negative (FN) represent the instances in which the model incorrectly classified as being positive and negative, respectively.

Accuracy represents the ratio of examples correctly classified (Equation 6.14).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6.14}$$

Precision measures which proportion of the samples predicted as positive are actually positive:

$$Precision = \frac{TP}{TP + FP} \tag{6.15}$$

Recall on the other-hand measures which proportion of the actual positive samples were collectively predicted as positive:

$$Recall = \frac{TP}{TP + FN} \tag{6.16}$$

In this section, we have introduced an approach for the evaluation of saliency maps using realistic perturbations which avoids the problem of creating out-of-distribution images. We have also proposed the adaptation of saliency metrics used to evaluate saliency models for the comparison of saliency maps. The proposed approach was validated by evaluating 8 saliency map methods on a digital pathology dataset called PatchCamelyon.

6.3 Results

In the experimental setup, three CNNs pre-trained on the ImageNet dataset have been selected: RestNet18, RestNet152, and DenseNet121. We trained the three CNNs on the PatchCamelyon dataset and also used them as base models for three ProtoPNets.

Table 6.2 shows the classification results of the models. Overall, both CNNs and ProtoPNets were able to achieve very high performance across all architectures. The most accurate model was the ProtoPNet based on the DenseNet121 architecture with an AUC of 0.981, the corresponding CNN having achieved an AUC of 0.975.

While the black-box CNNs perform better when using a ResNet18 and ResNet152 base

model, ProtoPNet with the DenseNet121 was the best model.

From the results, we can conclude that there was no clear trade-off of performance by adding the interpretability layer (i.e. prototypical layer) to achieve intrinsical interpretability.

model	base_model	auc	accuracy	precision	recall
PPNet	densenet121	**0.9814**	**0.9814**	0.9846	**0.9780**
CNN	densenet121	0.9750	0.9750	**0.9869**	0.9627
PPNet	resnet18	0.9703	0.9702	0.9667	0.9740
CNN	resnet18	0.9780	0.9780	0.9792	0.9768
PPNet	resnet152	0.9545	0.9545	0.9444	0.9658
CNN	resnet152	0.9628	0.9628	0.9647	0.9606

Table 6.2: Performance of models on malignancy detection.

To qualitatively evaluate the saliency map methods we have selected one example of malignancy and computed the saliency maps for each method for the network that achieved the best results (DenseNet121). Figure 6.3 presents side by side the result saliency maps overlapping the original image (A). Some of the methods (D and E) produce a saliency map with absolute values so the importance of each pixel is depicted by the darkness of the blue tone. Other methods (B, C, F, G, H, and I) produce both negative importance which is depicted in red, and positive importance depicted in green. Positive importance corresponds to evidence of malignancy while negative importance corresponds to evidence of benignity.

Deconvolution and Lime (B, I) highlight a vast region of the image but are able to somewhat focus on a vast number of nuclei. In comparison, methods such as SmoothGrad, Occlusion, and GuidedBackprop (C, E, H) appear more sparse in their activations.

To qualitatively evaluate the ProtoPNet's attribution maps we have selected the same test image as before and computed the attribution maps for 4 specific prototypes of the DenseNet121 base network. Figure 6.4 presents in the top row, for each prototype its most similar region extracted from the images of the training set. Below each image is shown the attribution map calculated for the original image used before.

When compared with the previous saliency map methods, the attribution maps generated by the ProtoPNet are very soft and smooth. This can be justified by the fact that the prototype dimensions that the hyperparameter optimization chose are small which can become a trade-off between increasing performance while disregarding fine-grained explanations.

We compared the predictions of the CNNs and ProtoPNets across all three architectures on the test set to understand if both models correlate with each other. Table 6.3 shows that AUC is high for the three architectures, suggesting that the CNN and ProtoPNet versions make similar predictions. DenseNet121, which was the most accurate architecture

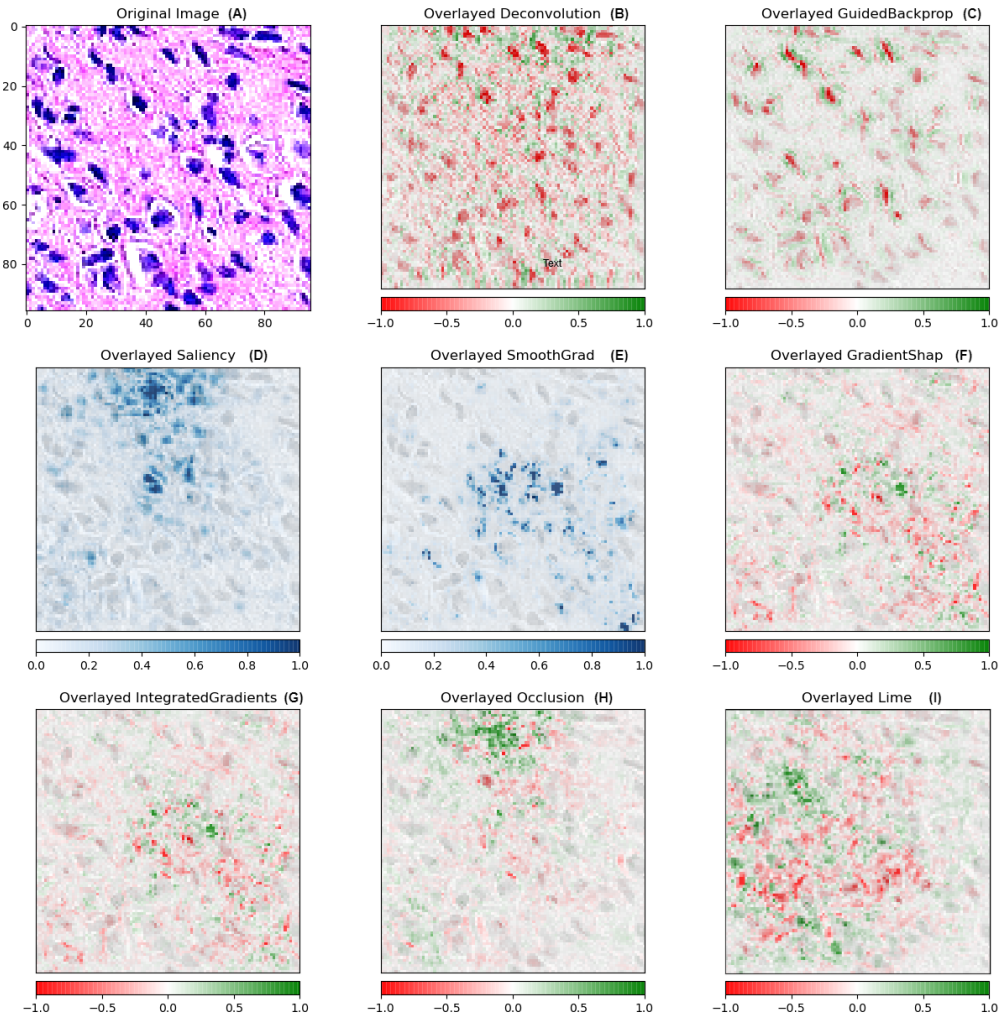


Figure 6.3: Qualitative analysis of saliency map methods for the DenseNet121 network. At the top left corner, we have the original image. The other images corresponding to the different saliency maps overlayed on top of the original image. Saliency maps with only absolute values represent positive evidence with blue tones, whereas darkness represents importance. Other saliency maps represent positive evidence of malignancy with green tones and negative evidence with red tones.

in the malignancy prediction task is also the architecture in which the predictions from the CNN and ProtoPNet versions are most related.

base_model	auc	accuracy	precision	recall
densenet121	**0.9749**	**0.9747**	**0.9656**	**0.9831**
resnet18	0.9703	0.9702	0.9654	0.9752
resnet152	0.9540	0.9539	0.9418	0.9672

Table 6.3: Comparing the predictions of CNN and ProtoPNet based on the same architecture.

To measure the overlap between CNN's saliency maps methods to ProtoPNet's attribution

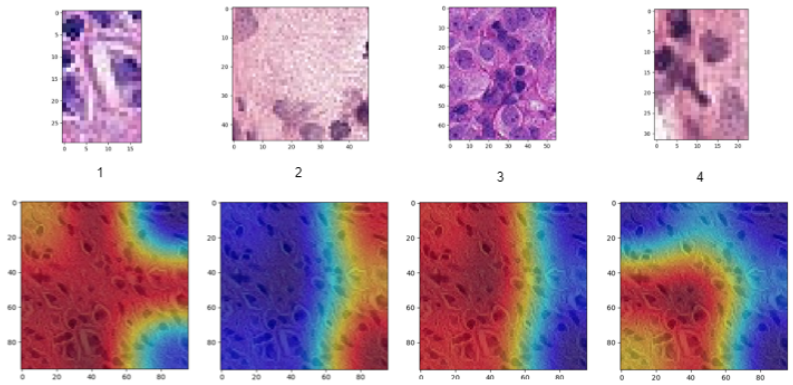


Figure 6.4: Qualitative analysis of prototypical parts of the ProtoPNet trained with the DenseNet121 architecture. At the top are the regions of the training images that are the most similar to each of the four prototypes. At the bottom are the attribution maps depicting the similarity of the original image to each prototype. The same original image as in Figure 6.3 was used.

maps we have extracted the explanations for each image in the test set and computed the saliency metrics (10) comparing each saliency map with the corresponding attribution map.

A statistical comparison was performed using the mean of each of the 8 saliency map methods and the Friedman rank test. The results were divided into three tables for each of the three architectures.

The averaged ranks across all metrics for the DenseNet121, ResNet152 and ResNet18 architectures are shown in Table 6.4. The methods that are shown to have the smallest overlap with ProtoPNet than other methods are highlighted.

	densenet121				resnet152				resnet18			
saliency_method	1	2	3	4	1	2	3	4	1	2	3	4
Deconvolution	**6.2**	**6.2**	**6.2**	**6.2**	3.6	4.1	4.1	4.5	4.5	4.5	3.7	4.2
GuidedBackprop	4.6	4.6	4.6	4.6	4.6	5.8	5.8	5.9	4.4	4.4	**5.8**	5.0
Saliency	3.9	3.9	3.5	3.5	3.3	4.3	4.3	4.4	4.7	4.3	4.7	4.8
GradientShap	5.7	5.7	5.0	5.0	4.3	5.4	5.4	5.5	**6.5**	**6.6**	5.6	6.0
IntegratedGradients	4.2	4.2	4.6	4.6	4.7	4.5	4.4	4.0	5.2	5.1	4.8	4.4
Lime	**5.6**	**5.6**	**6.2**	**6.2**	**6.1**	**7.2**	**7.2**	**7.0**	4.2	4.7	5.3	**5.9**
Occlusion	3.9	3.9	3.1	3.1	4.8	1.8	1.8	1.9	3.2	3.1	3.2	2.2
SmoothGrad	2.2	2.2	3.0	3.1	5.0	3.0	3.0	2.9	3.4	3.4	3.0	3.5

Table 6.4: Average rank of similarity between saliency map methods and ProtoPNet over the 10 saliency metrics.

Following the work of Denšar [66] with $N = 10$ (number of metrics) and $k = 8$ (number of saliency map methods), the 8 methods were compared among themselves for a 5% significance level using the two-tailed Nemenyi test [66] obtaining a CD (critical value for the difference of mean ranks between the 8 methods) of 2.949.

In the DenseNet121 architecture, a statistically significant difference was found between

SmoothGrad and both Deconvolution and Lime on all prototypes. When taking into account only the last two prototypes, Deconvolution and Lime have been found to have a smaller overlap than the majority of other methods. By analyzing the results, methods such as SmoothGrad, Occlusion, and Saliency continuously have a bigger overlap than methods such as Deconvolution, Lime, and GradientShap.

Figure 6.5 shows the average rank of the saliency map method depending on the prototype. Three interesting patterns emerge when we analyze the plot. First, depending on the architecture chosen the overlap of the methods can vary greatly. Second, despite this variation, some methods seem to produce saliency maps with a bigger overlap to ProtoPNet regardless of the architecture (i.e. SmoothGrad and Occlusion) while others show a smaller overlap (i.e. Deconvolution and Lime). Lastly, SmoothGrad saliency maps are less fine-grained than other back-propagation methods, so it makes sense because the attribution maps created by ProtoPnet have a low resolution and are upscaled to the input image size.

6.4 Conclusions

In this work, we proposed an approach to validate saliency map methods by measuring their overlap with the attribution maps produced by the intrinsic interpretable model ProtoPNet. As ProtoPNet does not use an external method to generate the attribution map it can be used as a ground-truth for the post-hoc methods. In our experimental setup, we compared 8 different popular post-hoc saliency map methods and the prototypical attribution maps generated by ProtoPnet. This was performed by looking at the closeness of predicted labels and measuring the overlap of saliency maps with ProtoPNet, with 10 different saliency metrics adapted from literature on saliency models.

ProtoPNet was not shown to trade-off performance in pursuit of interpretability, having achieved the most accurate model across all architectures. Also, the predictions of CNNs and ProtoPNets have been shown to correlate with each other.

While the saliency map methods produced a more fine-grained heatmap, ProtoPNet's attribution maps were soft and smooth. One possible justification relies on the prototype dimensions that the hyperparameter optimization chose, which were small. Also, we did not use fine annotation, as used in the IAIA-BL extension [29], as they were not available in the Camelyon16.

Overall, in spite of some differences in results depending on the architecture chosen, two methods have been found to have statistically a bigger overlap with ProtoPNet: SmoothGrad and Occlusion. Deconvolution and Lime have shown consistently lower overlap. One possible reason for these results is the fact that ProtoPNet produces smooth attribution maps. While not as smooth as ProtoPNet, SmoothGrad, and Occlusion are more sparse

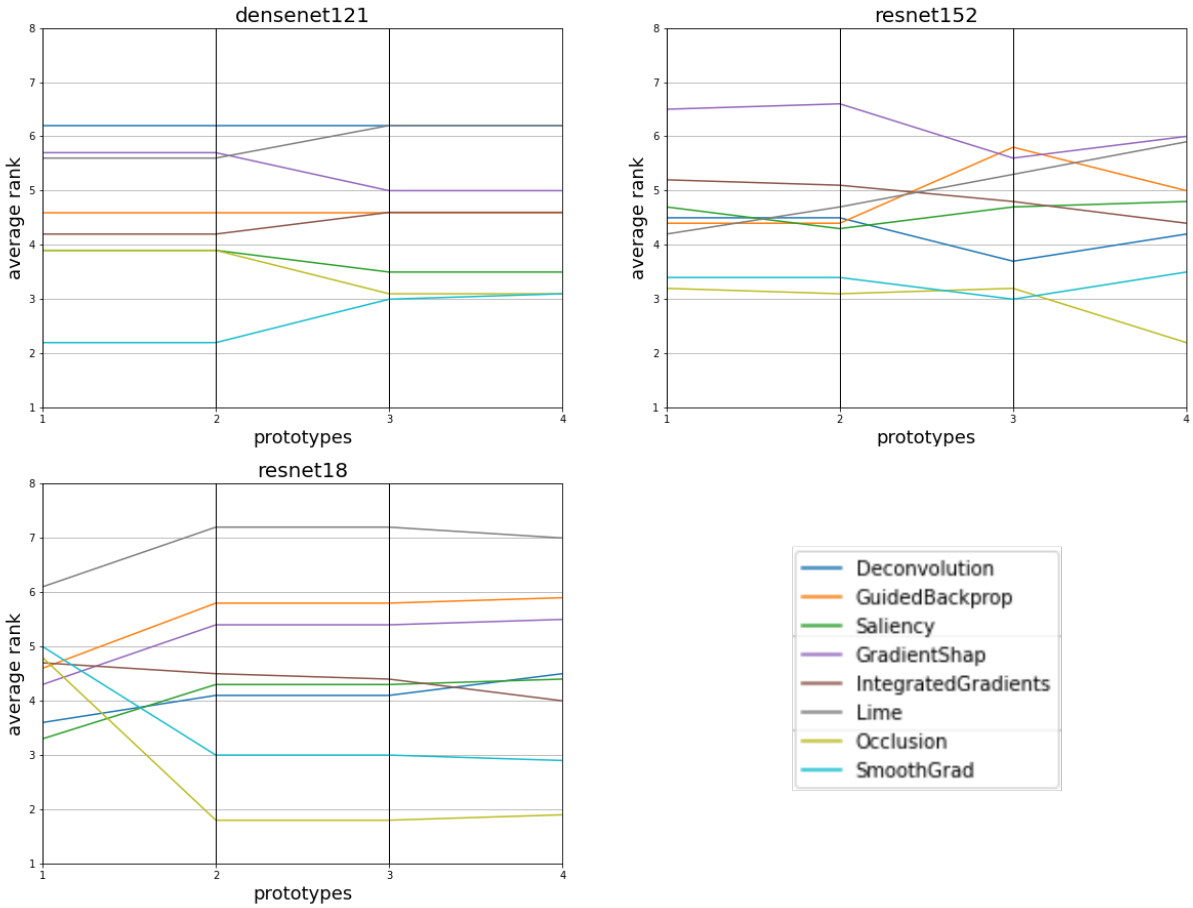


Figure 6.5: Plot showing the average rank of saliency map methods based on the four selected prototypes.

and conservative in the highlighting, while Deconvolution and Lime are more fine-grained but also more dispersed.

In this work, we focused on the saliency maps produced by the last fully-connected layer of the network. Further extensions must compare salience maps from intermediate layers. By doing so we can ascertain if the feature being detected by a filter or neuron of the network is related to the prototypes.

Future work directions also include the validation of the results on a dataset with fine annotation. An extension of ProtoPNet, called IAIA-BL[29], is capable of producing more fine-grained attribution maps by feeding the network fine annotation given by pathologists. While annotations are not always available, there are many public datasets where they are.

Chapter 7

Conclusions

In this chapter, we summarise our conclusions and insights regarding all of the research questions presented in Section 1.2. The following sections (Sections 7.1 to 7.3) discuss each research question individually. Finally, in Section 7.4 we provide our view on the next steps in Deep Learning interpretability for medical imaging in cancer.

7.1 Is there a connection between the complexity of the model and its interpretability? (RQ-1)

The experiments in this work (in particular Chapter 4) underline a strive to understand how the complexity of the model is related to interpretability and if it is possible to reduce the trade-off between performance and interpretability. As addressed in Chapter 4, a choice of an interpretable model would usually result in a substantial loss in performance. To reach that conclusion, we first developed an algorithm based on mimic learning to transfer the knowledge learned by complex deep neural networks to simpler interpretable ML models. The developed approach was validated on a medical dataset and showed that it was possible to increase the performance of these interpretable models by using the knowledge of deep neural networks as a teacher. The 10 most accurate interpretable models were not trained on the original data but on the augmented data from the deep neural networks. While models such as decision trees are intrinsically interpretable, the higher number of nodes and levels can turn these models incomprehensible. Knowledge distillation already has shown great results for training lightweight networks for medical image segmentation [108, 180], but we found potential in its use to also train intrinsically interpretable models by constricting the complexity of the network without losing performance.

Following the same direction, in chapter Chapter 4, we set our aim to look at a popular way to reduce the complexity and capacity of a network, regularization, and how it af-

fects the quality of the saliency maps produced by the network. A study was performed by adding regularization to different layers of the network at different values and measuring its impact on the faithfulness of the saliency maps produced by the regularized network. The faithfulness was measured with a metric called AOPC which evaluates the drop in confidence of the model when removing the pixels highlighted by the explanation as the most important. We were able to determine that regularization has an impact on interpretability, and networks with smaller regularization values produce more faithful explanations. However, regularization should not be performed in all layers, focusing only on later convolutional layers or in the fully-connected layers.

In Chapter 6 we have adapted the intrinsic interpretable model ProtoPNet for digital pathology. The network contains a prototypical layer that uses real images of the training set as prototypes representing each class and produces a similarity score as well as an attribution map (i.e. heatmap) which explains how close is the image being predicted to each prototype and which regions of the image are similar. This constraint in the network does not appear to have caused a drop in performance as mentioned in Chapter 6. Similar to the previous work, good results can be reached when the initial layers of the network, the feature extractor based on CNN, complexity is maintained while subsequent layers are constrained.

We can conclude that there is a connection between the complexity of the model and its interpretability and that generally, higher complexity of the model will result in higher interpretability as well as more faithful explanations.

7.2 Can interpretability methods help understand how deep learning models produce their decisions? (RQ-2)

During the experiments in Chapters 4, 5 and 6 different intrinsic interpretability methods and post-hoc interpretability methods were explored. The advantage of employing intrinsic interpretable models to make clinical decisions is that the behavior of the model can be understood without the use of external methods. While the models used in Section 4.1 (e.g. decision tree) are by themselves interpretable, the Prototypical Part Network used in Chapter 6 contains interpretable components, namely the similarity score and attribution map which not only allow the user to understand how the network makes the decision but also is used by the model to formulate its decision.

In Chapters 4, 5 and 6 we also explored the use of post-hoc interpretability methods in the form of saliency maps. Saliency maps highlight regions of the input image that are important for the decision made by a model.

Intrinsic interpretable models help understand the behavior of the deep learning models at a deep level. Models such as decision trees have components such as the nodes which

are easy to understand, so given an input the user can simulate the model's output. Intrinsic interpretable models based on deep learning such as ProtoPNet allows the same by introducing components such as prototypical layer which provides an explanation. For example, ProtoPNet presents the representative examples from the dataset, the similarity of the input image to these examples, and the impact that the presence of these prototypes has on the decision. Together, these different components transmit a complete understanding of the decision of the network.

Post-hoc interpretability methods on the other hand allowed us to understand how the model made decisions on a lower level compared to intrinsic interpretable models. The saliency maps showed what pixels were important in the classification, but the mechanisms behind that decision are still opaque.

7.3 Do deep learning models rely on relevant clinical information when classifying medical images? (RQ-3)

Recently many researchers have been concerned about using post-hoc explanation methods, especially saliency maps, as results are highly inconsistent, unreliable, and show invariance to model parameters [2, 51].

In Chapter 5 we proposed an approach to evaluate saliency map methods based on the introduction of realistic perturbation as opposed to the uniform and random perturbation which is prevalent in the literature. Doing so we could extract relevant salient metrics and evaluate quantitatively whether the saliency maps were robust (i.e. invariant) to the introduction of evidence of malignancy or the subtraction of it. From the results, we could conclude that while saliency map methods were helpful in understanding DL model decisions, they exhibit high levels of robustness to perturbation, especially Occlusion Sensitivity and SmoothGrad, which needs to be further investigated.

In Chapter 6 we introduced the attribution maps generated by ProtoPNet as a ground-truth for post-hoc interpretability to evaluate what was the overlap between the two maps. Overall, in spite of some differences in results depending on the CNN architecture chosen, two methods have been found to be statistically more similar to ProtoPNet: SmoothGrad and Occlusion, while Deconvolution and Lime were consistently the most dissimilar methods. Qualitatively it was possible to assess that the saliency maps focused on cell nuclei and disregarded background, pointing to actual clinical information.

7.4 Limitations and Further Directions

The interplay between the complexity of the model and the interpretability that can be achieved is still not fully understood. We studied the relationship between complexity and interpretability in the particular case of regularization and showed that it was possible to increase the performance of simpler interpretable models by transferring the knowledge learned by complex DL models. Future directions include the other techniques for reducing complexity such as model pruning [102] to reduce the number of connections in the network, or even apply knowledge distillation similar to chapter 4 to transfer the knowledge to smaller networks.

When investigating post-hoc interpretability we focused on the application and evaluation of saliency maps as they are one of the most popular techniques in the literature (Section 2.5). But, to get a more complete understanding of the model's behavior a combination of post-hoc methods should be used. For example, Olah *et al.* [168] used a combination of feature visualization, attribution, and dimensionality reduction to visualize individual neurons, and also the combination of neurons that fire at a given spatial location. Also, interactive tools can be used to inspect different elements in the network in real-time.

Innumerable interpretability methods have been proposed but there is a lack of evaluation metrics capable of comparing and validating them. Most evaluation metrics of post-hoc explanations such as saliency maps and feature importance depend on the introduction of random noise. In this thesis, we proposed an evaluation approach based on the introduction of natural noise and the study of the impact of the perturbation on the explanations. Also, the overlap between post-hoc interpretability and intrinsic interpretability was measured.

Evaluation of interpretability methods will remain a difficult task until high-quality ground truth from the target audience is not available. Collaboration with clinicians to define clear annotations to establish a ground truth is essential for evaluating interpretability methods. Similarly, patients will also become the receivers of explanations of DL systems and also will need to become part of studies to evaluate what are the optimal explanations for different medical modalities. The collaboration between ML experts, physicians, and patients is essential for not only the development and testing of new DL systems but also the maintenance and improvement of already deployed systems.

In chapter 3 we developed a global taxonomy of interpretable AI used by lawyers, philosophers, developers, physicians, and sociologists, with the goal of building a solid basis for discussing the future of AI. Despite initial efforts to establish a shared vocabulary, its adoption as a standard practice has yet to be achieved.

Many of the current interpretability methods and evaluation metrics for DL models rely

on the introduction of random noise of the input image and the study of its impact on the network output and explanation. Further work should move away from random modification and towards a natural modification. Rather than targeting specific pixels or regions of the image, researchers should focus on the introduction of evidence of a specific pathology or the slight modification of a feature or concept to study the importance of such concept on the output of the network and explanation. Synthetic datasets can be curated by physicians and ML experts by creating pairs of images that are only dissimilar in a specific characteristic that is being studied. Similarly, generative models can be used to synthesize the synthetic dataset [262].

While post-hoc explanations allow DL systems to not compromise their performance and predictive capability, they are considered by researchers more unreliable and inconsistent. Intrinsically interpretable models impose constraints in the model for their behavior to be more understandable by the user, instilling more trust. Future work should focus on closing the gap in performance between intrinsically interpretable models and black-box models.

The datasets and problems in this thesis are limited by the computational power and time that were available, so most works focused on one pathology, digital pathology. Nevertheless, they can arguably be applied to deal with other data modalities.

This page is intentionally left blank.

References

[1] Amina Adadi and Mohammed Berrada. Peeking inside the black-box: a survey on explainable artificial intelligence (xai). *IEEE access*, 6:52138–52160, 2018.

[2] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, pages 9505–9515, 2018.

[3] Adekanmi Adegun and Serestina Viriri. Deep learning model for skin lesion segmentation: Fully convolutional network. In *Image Analysis and Recognition*, pages 232–242, 2019.

[4] Taejin Ahn, Taewan Goo, Chan Hee Lee, Sungmin Kim, Kyullhee Han, Sangick Park, and Taesung Park. Deep Learning-based Identification of Cancer or Normal Tissue using Gene Expression Data. *IEEE International Conference on Bioinformatics and Biomedicine*, pages 1748–1752, 2019.

[5] AIHLEG. Ethics Guidelines for Trustworthy AI. `https://www.europarl.europa.eu/cmsdata/196377/AI%20HLEG_Ethics%20Guidelines%20for%20Trustworthy%20AI.pdf`. Accessed: 2023-01-01.

[6] Ulrich Aïvodji, Hiromi Arai, Olivier Fortineau, Sébastien Gambs, Satoshi Hara, and Alain Tapp. Fairwashing: the risk of rationalization. In *International Conference on Machine Learning*, pages 161–170, 2019.

[7] Fadhl M Alakwaa, Kumardeep Chaudhary, Lana X Garmire, and Bioengineering Graduate Program. Deep Learning Accurately Predicts Estrogen Receptor Status in Breast Cancer Metabolomics Data. *Journal of Proteome Research*, 17(1):337–347, 2018.

[8] Maximilian Alber, Sebastian Lapuschkin, Philipp Seegerer, Miriam Hägele, Kristof T. Schütt, Grégoire Montavon, Wojciech Samek, Klaus-Robert Müller, Sven Dähne, and Pieter-Jan Kindermans. innvestigate neural networks! *Journal of Machine Learning Research*, 20:1–8, 2019.

[9] Ahmed Alqaraawi, Martin Schuessler, Philipp Weiß, Enrico Costanza, and Nadia Berthouze. Evaluating saliency map explanations for convolutional neural net-

works: A user study. In *International Conference on Intelligent User Interfaces*, page 275–285, 2020.

[10] David Alvarez-Melis and Tommi Jaakkola. A causal framework for explaining the predictions of black-box sequence-to-sequence models. In *Conference on Empirical Methods in Natural Language Processing*, pages 412–421, 2017.

[11] David Alvarez-Melis and Tommi Jaakkola. Towards Robust Interpretability with Self-Explaining Neural Networks. In *Advances in Neural Information Processing Systems*, pages 7775–7784, 2018.

[12] J. P. Amorim, P.H. Abreu, Alberto Fernández, Mauricio Reyes, João Santos, and Miguel H. Abreu. Interpreting deep machine learning models: An easy guide for oncologists. *IEEE Reviews in Biomedical Engineering*, 16(1):192–207, 2021.

[13] José P. Amorim, Inês Domingues, Pedro Abreu, and João Santos. Interpreting deep learning models for ordinal problems. In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pages 373–378, 2018.

[14] José P. Amorim, Pedro H. Abreu, Mauricio Reyes, and João Santos. Interpretability vs. Complexity: The Friction in Deep Neural Networks. In *International Joint Conference on Neural Networks*, pages 1–7, 2020.

[15] Mike Ananny and Kate Crawford. Seeing without knowing: Limitations of the transparency ideal and its application to algorithmic accountability. *New Media & Society*, 20(3):973–989, 2018.

[16] Sule Anjomshoae, Amro Najjar, Davide Calvaresi, and Kary Främling. Explainable agents and robots: Results from a systematic literature review. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 1078–1088, 2019.

[17] Natalia Antropova, Benjamin Huynh, and Maryellen Giger. A deep feature fusion methodology for breast cancer diagnosis demonstrated on three imaging modality datasets. *Medical Physics*, 44(10):5162–5171, 2017.

[18] Leila Arras, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. Explaining Recurrent Neural Network Predictions in Sentiment Analysis. In *Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 159–168, 2017.

[19] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, Raja Chatila, and Francisco Herrera. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82–115, 2020.

[20] Vijay Arya, Rachel KE Bellamy, Pin-Yu Chen, Amit Dhurandhar, Michael Hind, Samuel C Hoffman, Stephanie Houde, Q Vera Liao, Ronny Luss, Aleksandra Mojsilović, et al. One explanation does not fit all: A toolkit and taxonomy of ai explainability techniques. *INFORMS Annual Meeting*, pages 1–18, 2021.

[21] Onur Asan, Alparslan Emrah Bayrak, and Avishek Choudhury. Artificial intelligence and human trust in healthcare: Focus on clinicians. *Journal of Medical Internet Research*, 22(6):1–7, 2020.

[22] Kimberly Ashman, Huimin Zhuge, Erin Shanley, Sharon Fox, Shams Halat, Andrew Sholl, Brian Summa, and J Quincy Brown. Whole slide image data utilization informed by digital diagnosis patterns. *Journal of Pathology Informatics*, 13:100113, 2022.

[23] L. J. Ba and R. Caruana. Do deep nets really need to be deep? In *International Conference on Neural Information Processing Systems*, volume 2, pages 2654–2662. MIT Press, 2014.

[24] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PLOS ONE*, 10(7):1–46, 2015.

[25] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. How to Explain Individual Classification Decisions. *Journal of Machine Learning Research*, 11:1803–1831, 2010.

[26] Pierre Baldi. Autoencoders, unsupervised learning and deep architectures. In *International Conference on Unsupervised and Transfer Learning Workshop*, page 37–50, 2011.

[27] Péter Bándi, Oscar Geessink, Quirine Manson, Marcory Van Dijk, Maschenka Balkenhol, Meyke Hermsen, Babak Ehteshami Bejnordi, Byungjae Lee, Kyunghyun Paeng, Aoxiao Zhong, Quanzheng Li, Farhad Ghazvinian Zanjani, Svitlana Zinger, Keisuke Fukuta, Daisuke Komura, Vlado Ovtcharov, Shenghua Cheng, Shaoqun Zeng, Jeppe Thagaard, Anders B. Dahl, Huangjing Lin, Hao Chen, Ludwig Jacobsson, Martin Hedlund, Melih Çetin, Eren Halici, Hunter Jackson, Richard Chen, Fabian Both, Jörg Franke, Heidi Kusters-Vandevelde, Willem Vreuls, Peter Bult, Bram Van Ginneken, Jeroen Van Der Laak, and Geert Litjens. From Detection of Individual Metastases to Classification of Lymph Node Status at the Patient Level: The CAMELYON17 Challenge. *IEEE Transactions on Medical Imaging*, 38(2):550–560, 2019.

[28] John D. Banja, Rolf Dieter Hollstein, and Michael A. Bruno. When artificial intelligence models surpass physician performance: Medical malpractice liability in an

era of advanced artificial intelligence. *Journal of the American College of Radiology*, 19(7):816–820, 2022.

[29] Alina Jade Barnett, Fides Regina Schwartz, Chaofan Tao, Chaofan Chen, Yinhao Ren, Joseph Y Lo, and Cynthia Rudin. A case-based interpretable deep learning model for classification of mass lesions in digital mammography. *Nature Machine Intelligence*, 3(12):1061–1070, 2021.

[30] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3319–3327, 2017.

[31] Stephen Bazen and Xavier Joutard. The taylor decomposition: A unified generalization of the oaxaca method to nonlinear models. *HALSHS Sciences Humaines et Soaciales*, 1(1):1–42, 2013.

[32] Babak Ehteshami Bejnordi and et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *Journal of the American Medical Association*, 318(22):2199–2210, 2017.

[33] Yoshua Bengio. Deep Learning of Representations for Unsupervised and Transfer Learning. In *International Conference on Unsupervised and Transfer Learning*, pages 1–20, 2011.

[34] Tarek R Besold and Kai-Uwe Kühnberger. Towards integrated neural–symbolic systems for human-level ai: Two research programs helping to bridge the gaps. *Biologically Inspired Cognitive Architectures*, 14:97–110, 2015.

[35] Adrien Bibal, Michael Lognoul, Alexandre de Streel, and Benoît Frénay. Impact of legal requirements on explainability in machine learning. In *International Conference in Machine Learning Workshop on Law and Machine Learning*, pages 1–3, 2020.

[36] Adrien Bibal, Michael Lognoul, Alexandre de Streel, and Benoît Frénay. Legal requirements on explainability in machine learning. *Artificial Intelligence and Law*, 29(2):1–21, 2020.

[37] Or Biran and Courtenay Cotton. Explanation and justification in machine learning: A survey. In *International Joint Conference on Artificial Intelligence - Workshop on Explainable AI*, pages 8–13, 2017.

[38] Guido Bologna and Yoichi Hayashi. Characterization of symbolic rules embedded in deep DIMLP networks: A challenge to transparency of deep learning. *Journal of Artificial Intelligence and Soft Computing Research*, 7(4):265–286, 2017.

[39] Giovanni Briganti and Olivier Le Moine. Artificial intelligence in medicine: Today and tomorrow. *Frontiers in Medicine*, 7:1–6, 2020.

[40] Titus J. Brinker, Achim Hekler, Alexander H. Enk, Joachim Klode, Axel Hauschild, Carola Berking, Schilling, and et al. A convolutional neural network trained with dermoscopic images performed on par with 145 dermatologists in a clinical melanoma image classification task. *European Journal of Cancer*, 111:148–154, 2019.

[41] C. Bucila, R. Caruana, and A. Niculescu-Mizil. Model compression. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 535–541, 2006.

[42] Zoya Bylinskii, Tilke Judd, Aude Oliva, Antonio Torralba, and Fredo Durand. What Do Different Evaluation Metrics Tell Us about Saliency Models? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(3):740–757, 2019.

[43] Federico Cabitza, Raffaele Rasoini, and Gian Franco Gensini. Unintended Consequences of Machine Learning in Medicine. *Journal of the American Medical Association*, 318(6):517–518, 2017.

[44] Davide Calvaresi, Giovanni Ciatto, Amro Najjar, Reyhan Aydogan, Leon Van der Torre, Andrea Omicini, and Michael Schumacher. Expectation: Personalized explainable artificial intelligence for decentralized agents with heterogeneous knowledge. In *International Workshop on Explainable and Transparent AI and Multi-Agent Systems*, pages 331–343, 2021.

[45] Davide Calvaresi, Mauro Marinoni, Arnon Sturm, Michael Schumacher, and Giorgio Buttazzo. The challenge of real-time multi-agent systems for enabling IoT and CPS. In *International Conference on Web Intelligence*, pages 356–364, 2017.

[46] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *USENIX Security Symposium*, pages 267–284, 2019.

[47] Rich Caruana, H Kangarloo, John Dionisio, U. Sinha, and David Johnson. Case-based explanation of non-case-based learning methods. In *AMIA Symposium*, pages 212–215, 1999.

[48] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1721–1730, 2015.

[49] Manomita Chakraborty, Saroj Kumar Biswas, and Biswajit Purkayastha. Rule extraction from neural network trained using deep belief network and back propagation. *Knowledge and Information Systems*, 62(9):3753–3781, 2020.

[50] Z. Che, S. Purushotham, R. Khemani, and Y. Liu. Interpretable Deep Models for ICU Outcome Prediction. *AMIA Annual Symposium proceedings*, 2016:371–380, 2016.

[51] Chaofan Chen, Oscar Li, Chaofan Tao, Alina Jade Barnett, Jonathan Su, and Cynthia Rudin. This looks like that: Deep learning for interpretable image recognition. In *International Conference on Neural Information Processing Systems*, page 8930–8941, 2019.

[52] Hu Chen, Yi Zhang, Mannudeep K. Kalra, Feng Lin, Yang Chen, Peixi Liao, Jiliu Zhou, and Ge Wang. Low-dose ct with a residual encoder-decoder convolutional neural network. *IEEE Transactions on Medical Imaging*, 36(12):2524–2535, 2017.

[53] Michael Chromik and Martin Schuessler. A taxonomy for human subject evaluation of black-box explanations in xai. In *ACM Intelligent User Interfaces Conference*, pages 1–7, 2020.

[54] Giovanni Ciatto, Roberta Calegari, Andrea Omicini, and Davide Calvaresi. Towards XMAS: explainability through multi-agent systems. In *Workshop on Artificial Intelligence and Internet of Things co-located - International Conference of the Italian Association for Artificial Intelligence*, pages 40–53, 2019.

[55] Giovanni Ciatto, Michael I Schumacher, Andrea Omicini, and Davide Calvaresi. Agent-based explanations in ai: Towards an abstract framework. In *International Workshop on Explainable, Transparent Autonomous Agents and Multi-Agent Systems*, pages 3–20, 2020.

[56] Kenneth Clark and et al. The cancer imaging archive (TCIA): Maintaining and operating a public information repository. *Journal of Digital Imaging*, 26(6):1045–1057, 2013.

[57] Miruna-Adriana Clinciu and Helen Hastie. A survey of explainable ai terminology. In *Workshop on Interactive Natural Language Technology for Explainable Artificial Intelligence*, pages 8–13, 2019.

[58] Noel C. F. Codella, Chung-Ching Chung-Ching Lin, Allan Halpern, Michael Hind, Rogerio Feris, and John R. Smith. Collaborative Human-AI (CHAI): Evidence-Based Interpretable Melanoma Classification in Dermoscopic Images. In *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*, pages 97–105, 2018.

[59] Mark Coeckelbergh. *AI ethics*. MIT Press, 2020.

[60] Diane Coyle and Adrian Weller. "Explaining" machine learning reveals policy challenges. *Science*, 368(6498):1433–1434, 2020.

[61] Angel Cruz-Roa and et al. Automatic detection of invasive ductal carcinoma in whole slide images with convolutional neural networks. In *Medical Imaging: Digital Pathology*, volume 9041, page 904103, 2014.

[62] Angel Alfonso Cruz-Roa, John Edison Arevalo Ovalle, Anant Madabhushi, and Fabio Augusto González Osorio. A Deep Learning Architecture for Image Representation, Visual Interpretability and Automated Basal-Cell Carcinoma Cancer Detection. In *Medical Image Computing and Computer-Assisted Intervention*, pages 403–410, 2013.

[63] Sunan Cui, Yi Luo, Huan-Hsin Tseng, Randall Ten Haken, and Issam El Naqa. Combining handcrafted features with latent variables in machine learning for prediction of radiation-induced lung damage. *Medical Physics*, 46, 2019.

[64] Jeffrey De Fauw, Joseph R. Ledsam, Bernardino Romera-Paredes, Stanislav Nikolov, Nenad Tomasev, Sam Blackwell, Harry Askham, Xavier Glorot, Brendan O'Donoghue, Daniel Visentin, George van den Driessche, Balaji Lakshminarayanan, Clemens Meyer, Faith Mackinder, Simon Bouton, Kareem Ayoub, Reena Chopra, Dominic King, Alan Karthikesalingam, Cían O. Hughes, Rosalind Raine, Julian Hughes, Dawn A. Sim, Catherine Egan, Adnan Tufail, Hugh Montgomery, Demis Hassabis, Geraint Rees, Trevor Back, Peng T. Khaw, Mustafa Suleyman, Julien Cornebise, Pearse A. Keane, and Olaf Ronneberger. Clinically applicable deep learning for diagnosis and referral in retinal disease. *Nature Medicine*, 24(9):1342–1350, 2018.

[65] Luc De Raedt, Robin Manhaeve, Sebastijan Dumancic, Thomas Demeester, and Angelika Kimmig. Neuro-symbolic= neural+ logical+ probabilistic. In *International Workshop on Neural-Symbolic Learning and Reasoning*, pages 1–4, 2019.

[66] Janez Demsar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[67] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[68] Amit Dhurandhar, Pin-Yu Chen, Ronny Luss, Chun-Chen Tu, Paishun Ting, Karthikeyan Shanmugam, and Payel Das. Explanations based on the missing: Towards contrastive explanations with pertinent negatives. In *Advances in Neural Information Processing Systems*, pages 1–12, 2018.

[69] Stephanie Dick. Artificial intelligence. *Harvard Data Science Review*, 1(1):1–8, 2019.

[70] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint*, pages 1–13, 2017.

[71] Mengnan Du, Ninghao Liu, and Xia Hu. Techniques for interpretable machine learning. *Communications of the ACM*, 63(1):68–77, 2020.

[72] Jeffrey M Ede. Deep learning in electron microscopy. *Machine Learning: Science and Technology*, 2(1):1–72, 2021.

[73] Lilian Edwards and Michael Veale. Slave to the algorithm: Why a right to an explanation is probably not the remedy you are looking for. *Duke Law & Technology Review*, 18:1–67, 2017.

[74] Andre Esteva, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, 2017.

[75] M Everson, Lcgp Herrera, W Li, I Muntion Luengo, O Ahmad, M Banks, C Magee, D Alzoubaidi, H M Hsu, D Graham, T Vercauteren, L Lovat, S Ourselin, S Kashin, Hsiu-Po Wang, Wen-Lun Wang, and R J Haidry. Artificial intelligence for the real-time classification of intrapapillary capillary loop patterns in the endoscopic diagnosis of early oesophageal squamous cell carcinoma: A proof-of-concept study. *United European Gastroenterology Journal*, 7(2):297–306, 2019.

[76] FDA. Artificial Intelligence and Machine Learning (AI/ML)-Enabled Medical Devices. `https://www.fda.gov/medical-devices/software-medical-device-samd/artificial-intelligence-and-machine-learning-aiml-enabled-medical-devices`. Accessed: 2023-01-01.

[77] Samuel G. Finlayson, John D. Bowers, Joichi Ito, Jonathan L. Zittrain, Andrew L. Beam, and Isaac S. Kohane. Adversarial attacks on medical machine learning. *Science*, 363(6433):1287–1289, 2019.

[78] Luciano Floridi, Josh Cowls, Monica Beltrametti, Raja Chatila, Patrice Chazerand, Virginia Dignum, Christoph Luetge, Robert Madelin, Ugo Pagallo, Francesca Rossi, et al. Ai4people—an ethical framework for a good ai society: opportunities, risks, principles, and recommendations. *Minds and Machines*, 28(4):689–707, 2018.

[79] Roger Fonollà and et al. Ensemble of Deep Convolutional Neural Networks for Classification of Early Barrett's Neoplasia Using Volumetric Laser Endomicroscopy. *Applied Sciences*, 9(11):2183, 2019.

[80] E. Frank and M. Hall. A simple approach to ordinal classification. *European Conference on Machine Learning*, pages 145–156, 2001.

[81] Stan Franklin and Art Graesser. Is it an agent, or just a program?: A taxonomy for autonomous agents. In *International Workshop on Agent Theories, Architectures, and Languages*, pages 21–35, 1996.

[82] Nicholas Frosst and Geoffrey Hinton. Distilling a neural network into a soft decision tree. In *International Workshop on Comprehensibility and Explanation in AI and ML - International Conference of the Italian Association for Artificial Intelligence*, pages 1–8, 2017.

[83] Yabo Fu, Yang Lei, Tonghe Wang, Walter J. Curran, Tian Liu, and Xiaofeng Yang. Deep learning in medical image registration: A review. *Physics in Medicine & Biology*, 65(20):1–51, 2019.

[84] J. C. Gámez, D. García, A. González, and R. Pérez. Ordinal classification based on the sequential covering strategy. *International Journal of Approximate Reasoning*, 76:96–110, 2016.

[85] Krzysztof J Geras, Ritse M Mann, and Linda Moy. Artificial intelligence for mammography and digital breast tomosynthesis: Current concepts and future perspectives. *Radiology*, 293(2):246—259, 2019.

[86] Ivan Gonzalez Diaz. DermaKNet: Incorporating the knowledge of dermatologists to Convolutional Neural Networks for skin lesion diagnosis. *IEEE Journal of Biomedical and Health Informatics*, 2194:1–14, 2018.

[87] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[88] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. In *International Conference on Learning Representations*, pages 1–11, 2015.

[89] Bryce Goodman and Seth Flaxman. Eu regulations on algorithmic decision-making and a "right to explanation". In *International Conference on Machine Learning Workshop on Human Interpretability in Machine Learning*, pages 1–9, 2016.

[90] M Graziani, V Andrearczyk, S Marchand-maillet, and H Müller. Concept attribution: Explaining CNN decisions to physicians. *Computers in Biology and Medicine*, 123:103865, 2020.

[91] Mara Graziani. *Interpretability of Deep Learning for Medical Image Classification: Improved Understandability and Generalization*. PhD thesis, University of Geneva, 2021.

[92] Mara Graziani, Vincent Andrearczyk, Stephane Marchand-Maillet, and Henning Müller. Concept attribution: Explaining cnn decisions to physicians. *Computers in Biology and Medicine*, 123:103865, 2020.

[93] Mara Graziani, Vincent Andrearczyk, and Henning Möuller. Regression Concept Vectors for Bidirectional Explanations in Histopathology. In *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*, pages 124–132, 2018.

[94] Mara Graziani, Vincent Andrearczyk, and Henning Müller. Visual interpretability for patch-based classification of breast cancer histopathology images. In *Medical Imaging with Deep Learning*, pages 1–4, 2018.

[95] Mara Graziani, James M. Brown, Vincent Andrearczyk, Veysi Yildiz, J. Peter Campbell, Deniz Erdogmus, Stratis Ioannidis, Michael F. Chiang, Jayashree Kalpathy-Cramer, and Henning Muller. Improved interpretability for computer-aided severity assessment of retinopathy of prematurity. In *Computer-Aided Diagnosis*, pages 1–63, 2019.

[96] Mara Graziani, Lidia Dutkiewicz, Davide Calvaresi, José Pereira Amorim, Katerina Yordanova, Mor Vered, Rahul Nair, Pedro Henriques Abreu, Tobias Blanke, Valeria Pulignano, John O. Prior, Lode Lauwaert, Wessel Reijers, Adrien Depeursinge, Vincent Andrearczyk, and Henning Müller. A global taxonomy of interpretable ai: unifying the terminology for the technical and social sciences. *Artificial Intelligence Review*, 56:1–32, 2022.

[97] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A Survey of Methods for Explaining Black Box Models. *ACM Computing Surveys*, 51(5):1–42, 2018.

[98] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys*, 51(5):1–42, 2018.

[99] H2O. H2O.ai. `https://www.h2o.ai/`, 2019. Accessed: 2019-07-01.

[100] Ronan Hamon, Henrik Junklewitz, Gianclaudio Malgieri, Paul De Hert, Laurent Beslay, and Ignacio Sanchez. Impossible explanations? beyond explainable ai in the gdpr from a covid-19 use case scenario. In *ACM Conference on Fairness, Accountability, and Transparency*, pages 549–559, 2021.

[101] Lichy Han and Maulik R. Kamdar. MRI to MGMT: predicting methylation status in glioblastoma patients using convolutional recurrent neural networks. *Analytical Chemistry*, 25(4):368–379, 2015.

[102] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. In *Advances in Neural Information Processing Systems*, pages 1–9, 2015.

[103] Mohammad Havaei, Axel Davy, David Warde-Farley, Antoine Biard, Aaron Courville, Yoshua Bengio, Chris Pal, Pierre-Marc Jodoin, and Hugo Larochelle. Brain tumor segmentation with deep neural networks. *Medical Image Analysis*, 35:18–31, 2017.

[104] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[105] Mohammad Hesam Hesamian, Wenjing Jia, Xiangjian He, and Paul Kennedy. Deep learning techniques for medical image segmentation: Achievements and challenges. *Journal of Digital Imaging*, 32(4):582–596, 2019.

[106] Denis Hilton. Social attribution and explanation. *The Oxford handbook of causal reasoning*, pages 645–674, 2017.

[107] Denis J Hilton. Conversational processes and causal explanation. *Psychological Bulletin*, 107(1):65, 1990.

[108] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the Knowledge in a Neural Network. In *Deep Learning and Representation Learning Workshop - Conference on Neural Information Processing Systems*, pages 1–9, 2015.

[109] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *International Conference on Neural Information Processing Systems*, pages 1–12, 2019.

[110] J. Hua, Z. Xiong, J. Lowey, E. Suh, and E. R. Dougherty. Optimal number of features as a function of sample size for various classification rules. *Bioinformatics*, 21(8):1509–1515, 2005.

[111] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2261–2269, Los Alamitos, CA, USA, 2017. IEEE Computer Society.

[112] Zhi Huang, Xiaohui Zhan, Shunian Xiang, Travis S. Johnson, Bryan Helm, Christina Y. Yu, Jie Zhang, Paul Salama, Maher Rizkalla, Zhi Han, and Kun Huang. Salmon: Survival analysis learning with multi-omics neural networks on breast cancer. *Frontiers in Genetics*, 10(4):1–13, 2019.

[113] Paolo Inglese and et al. Deep learning and 3D-DESI imaging reveal the hidden metabolic heterogeneity of cancer. *Chemical Science*, 8(5):3500–3511, 2017.

[114] Ayla Al Kabbani, Yuranga Weerakkody, and et. al. Breast imaging-reporting and data system (BI-RADS). *Reston VA: American College of Radiology*, 19(7):816–820, 1998.

[115] Harmanpreet Kaur, Harsha Nori, Samuel Jenkins, Rich Caruana, Hanna Wallach, and Jennifer Wortman Vaughan. Interpreting interpretability: Understanding data scientists' use of interpretability tools for machine learning. In *Conference on Human Factors in Computing Systems*, pages 1–14, 2020.

[116] Christopher J. Kelly, Alan Karthikesalingam, Mustafa Suleyman, Greg Corrado, and Dominic King. Key challenges for delivering clinical impact with artificial intelligence. *BMC Medicine*, 17(1):1–9, 2019.

[117] Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems*, volume 29, pages 1–9, 2016.

[118] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory sayres. Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV). In *International Conference on Machine Learning*, pages 2668–2677, 2018.

[119] Hojin Kim, Jinhong Jung, Jieun Kim, Byungchul Cho, Jungwon Kwak, Jeong Yun Jang, Sang-wook Lee, June-Goo Lee, and Sang Min Yoon. Abdominal multi-organ auto-segmentation using 3d-patch-based deep convolutional neural network. *Scientific Reports*, 10(1):6204, 2020.

[120] Seong Tae Kim, Jae Hyeok Lee, Hakmin Lee, and Yong Man Ro. Visually interpretable deep network for diagnosis of breast masses on mammograms. *Physics in Medicine and Biology*, 63(23):235025, 2018.

[121] Seong Tae Kim, Jae-Hyeok Lee, and Yong Ro. Visual evidence for interpreting diagnostic decision of deep neural network in computer-aided diagnosis. In *Computer-Aided Diagnosis*, pages 1–19, 2019.

[122] Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T. Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. *The (Un)reliability of Saliency Methods*, chapter 4, pages 267–280. Springer, 2019.

[123] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning*, pages 1885–1894, 2017.

[124] Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, and Orion Reblitz-Richardson. Captum: A unified and generic model interpretability library for pytorch, 2020.

[125] Bruno Korbar, Andrea M. Olofson, Allen P. Miraflor, Catherine M. Nicka, Matthew A. Suriawinata, Lorenzo Torresani, Arief A. Suriawinata, and Saeed Hassanpour. Looking Under the Hood: Deep Neural Network Visualization to Interpret

Whole-Slide Image Analysis Outcomes for Colorectal Polyps. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 821–827, 2017.

[126] Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. Master's thesis, University of Toronto, Canada, 2009.

[127] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[128] Himabindu Lakkaraju, Stephen H Bach, and Jure Leskovec. Interpretable decision sets: A joint framework for description and prediction. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1675–1684, 2016.

[129] J. Lameski, A. Jovanov, E. Zdravevski, P. Lameski, and S. Gievska. Skin lesion segmentation with deep learning. In *International Conference on Smart Technologies*, pages 1–5, 2019.

[130] Jiangwei Lao, Yinsheng Chen, Zhi Cheng Li, Qihua Li, Ji Zhang, Jing Liu, and Guangtao Zhai. A Deep Learning-Based Radiomics Model for Prediction of Survival in Glioblastoma Multiforme. *Scientific Reports*, 7(1):1263, 2017.

[131] Sebastian Lapuschkin, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10, 2015.

[132] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[133] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2323, 1998.

[134] Hyebin Lee, Seong Tae Kim, and Yong Man Ro. Generation of Multimodal Justification Using Visual Word Constraint Model for Explainable Computer-Aided Diagnosis. In *International Workshop on Multimodal Learning for Clinical Decision Support*, pages 21–29, 2019.

[135] Zachary C Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

[136] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A.W.M. van der Laak, Bram van Ginneken, and Clara I. Sánchez. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88, 2017.

[137] Fang Liu, Poonam P. Yadav, Andrew M. Baschnagel, and Alan B. McMillan. Mr-based treatment planning in radiation therapy using a deep learning approach. *Journal of Applied Clinical Medical Physics*, 20:105–114, 2019.

[138] Peng Liu, Lemei Zhang, and Jon Atle Gulla. Dynamic attention-based explainable recommendation with textual and visual fusion. *Information Processing & Management*, 57(6):1–22, 2020.

[139] Qian Liu and Pingzhao Hu. Association analysis of deep genomic features extracted by denoising autoencoders in breast cancer. *Cancers*, 11(4):1–13, 2019.

[140] Shixia Liu, Xiting Wang, Mengchen Liu, and Jun Zhu. Towards better analysis of machine learning models: A visual analytics perspective. *Visual Informatics*, 1(1):48–56, 2017.

[141] Tania Lombrozo. The structure and function of explanations. *Trends in cognitive sciences*, 10(10):464–470, 2006.

[142] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *International Conference on Neural Information Processing Systems*, page 4768–4777, Red Hook, NY, USA, 2017.

[143] Yi Luo, Huan-Hsin Tseng, Sunan Cui, Lise Wei, Randall K Ten Haken, and Issam El Naqa. Balancing accuracy and interpretability of machine learning approaches for radiation treatment outcomes modeling. *Physics in Medicine & Biology*, 63(23):235025, 2019.

[144] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *UMAP: Uniform Manifold Approximation and Projection*, 3(29):1–63, 2018.

[145] Raphael Meudec. tf-explain. `https://github.com/sicara/tf-explain`, 2021.

[146] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38, 2019.

[147] Tim Miller, Piers Howe, and Liz Sonenberg. Explainable ai: Beware of inmates running the asylum or: How i learnt to stop worrying and love the social and behavioural sciences. In *International Conference on Principles of Knowledge Representation and Reasoning*, pages 1–7, 2020.

[148] Mamta Mittal, Lalit Mohan Goyal, Sumit Kaur, Iqbaldeep Kaur, Amit Verma, and D. Jude Hemanth. Deep learning based enhanced tumor segmentation approach for mr brain images. *Applied Soft Computing*, 78:346–354, 2019.

[149] Brent Mittelstadt, Chris Russell, and Sandra Wachter. Explaining explanations in ai. In *Conference on Fairness, Accountability, and Transparency*, pages 279–288, 2019.

[150] Pooya Mobadersany, Safoora Yousefi, Mohamed Amgad, David A. Gutman, Jill S. Barnholtz-Sloan, José E. Velázquez Vega, Daniel J. Brat, and Lee A. D. Cooper. Predicting cancer outcomes from histology and genomics using convolutional networks. *National Academy of Sciences*, 115(13):2970–2979, 2018.

[151] Christoph Molnar. Interpretable Machine Learning, 2019. `https://christophm.github.io/interpretable-ml-book/`.

[152] Christoph Molnar. *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*. Leanpub, 2019. `https://christophm.github.io/interpretable-ml-book`(visited 2021-05-15).

[153] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65:211–222, 2017.

[154] Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1–15, 2018.

[155] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65:211–222, 2017.

[156] Olivier Morin, Martin Vallières, Arthur Jochems, Henry C. Woodruff, Gilmer Valdes, Steve E. Braunstein, Joachim E. Wildberger, Javier E. Villanueva-Meyer, Vasant Kearney, Timothy D. Solberg, and Philippe Lambin. A Deep Look Into the Future of Quantitative Imaging in Oncology: A Statement of Working Principles and Proposal for Change. *International Journal of Radiation Oncology, Biology, Physics*, 102(4):1074–1082, 2018.

[157] W James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. Interpretable machine learning: definitions, methods, and applications. *Proceedings of the National Academy of Sciences*, 116(44):22071–22080, 2019.

[158] Daniel Nemirovsky, Nicolas Thiebaut, Ye Xu, and Abhishek Gupta. Countergan: Generating counterfactuals for real-time recourse and interpretability using residual gans. In *Conference on Uncertainty in Artificial Intelligence*, pages 1488–1497, 2022.

[159] An-phi Nguyen and María Rodríguez Martínez. Mononet: towards interpretable models by learning monotonic features. *Conference on Neural Information Processing Systems - Human-Centric Machine Learning Workshop*, pages 1–5, 2019.

[160] Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. Synthesizing the Preferred Inputs for Neurons in Neural Networks via Deep Generator Networks. In *International Conference on Neural Information Processing Systems*, pages 3395–3403, 2016.

[161] Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *International Conference on Neural Information Processing Systems*, page 3395–3403, 2016.

[162] Toyoaki Nishida, Atsushi Nakazawa, Yoshimasa Ohmoto, and Yasser Mohammad. *Conversational informatics*. Springer, 2014.

[163] Helen Nissenbaum. A contextual approach to privacy online. *Daedalus*, 140(4):32–48, 2011.

[164] M. A. Nogueira. *Creating Evaluation Functions for Oncological Diseases based on PET/CT*. Master thesis in biomedical engineering, University of Coimbra, 2015.

[165] M. A. Nogueira, P. H. Abreu, P. Martins, P. Machado, H. Duarte, and J. Santos. An artificial neural networks approach for assessment treatment response in oncological patients using PET/CT images. *BMC Medical Imaging*, 17(1):17–13, 2017.

[166] OJEU. European Commission - General Data Protection Regulation (2016). `https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679`. Accessed: 2019-07-01.

[167] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature Visualization. *Distill*, 2017. https://distill.pub/2017/feature-visualization.

[168] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017. https://distill.pub/2017/feature-visualization.

[169] Andrea Omicini. Not just for humans: Explanation for agent-to-agent communication. In *International Conference of the Italian Association for Artificial Intelligence*, pages 1–11, 2020.

[170] Mark L. H. Ong and John B. Schofield. Assessment of lymph node involvement in colorectal cancer. *World Journal of Gastrointestinal Surgery*, 8(3):179–192, 2016.

[171] Olatunji Oni and Sanzheng Qiao. Model-Agnostic Interpretation of Cancer Classification with Multi-Platform Genomic Data. In *ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pages 34–41, 2019.

[172] Sebastian Palacio, Adriano Lucieri, Mohsin Munir, Jörn Hees, Sheraz Ahmed, and Andreas Dengel. Xai handbook: Towards a unified framework for explainable ai. In *International Conference on Computer Vision*, pages 3766–3775, 2021.

[173] Cecilia Panigutti, Alan Perotti, André Panisson, Paolo Bajardi, and Dino Pedreschi. Fairlens: Auditing black-box clinical decision support systems. *Information Processing & Management*, 58(5):1–17, 2021.

[174] Rahul Paul, Ying Liu, Qian Li, Lawrence Hall, and Dmitry Goldgof. Representation of Deep Features using Radiologist defined Semantic Features. In *International Joint Conference on Neural Networks*, pages 1429–1435, 2018.

[175] William Penny and D Frost. Neural networks in clinical medicine. *Medical Decision Making*, 16:386–398, 1996.

[176] Sérgio Pereira, Raphael Meier, Victor Alves, Mauricio Reyes, and Carlos A. Silva. Automatic Brain Tumor Grading from MRI Data Using Convolutional Neural Networks and Quality Assessment. In *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*, pages 106–114, 2018.

[177] Sérgio Pereira, Raphael Meier, Richard McKinley, Roland Wiest, Victor Alves, Carlos A. Silva, and Mauricio Reyes. Enhancing interpretability of automatically extracted machine learning features: application to a RBM-Random Forest system on brain lesion segmentation. *Medical Image Analysis*, 44:228–244, 2018.

[178] Oleg S. Pianykh, Georg Langs, Marc Dewey, Dieter R. Enzmann, Christian J. Herold, Stefan O. Schoenberg, and James A. Brink. Continuous Learning AI in Radiology: Implementation Principles and Early Applications. *Radiology*, 297(1):6–14, 2020.

[179] Rafael Poyiadzi, Kacper Sokol, Raul Santos-Rodriguez, Tijl Bie, and Peter Flach. Face: Feasible and actionable counterfactual explanations. In *AAAI/ACM Conference on AI, Ethics, and Society*, pages 344–350, 2020.

[180] Dian Qin, Jiajun Bu, Zhe Liu, Xin Shen, Sheng Zhou, Jing-Jun Gu, Zhihong Wang, Lei Wu, and Hui-Fen Dai. Efficient medical image segmentation based on knowledge distillation. *IEEE Transactions on Medical Imaging*, 40:3820–3831, 2021.

[181] Adityanarayanan Radhakrishnan, Charles Durham, Ali Soylemezoglu, and Caroline Uhler. Patchnet: Interpretable Neural Networks for Image Classification. In *International Conference on Neural Information Processing Systems - Machine Learning for Health Workshop*, pages 1–15, 2018.

[182] Pranav Rajpurkar and el al. Deep learning for chest radiograph diagnosis: A retrospective comparison of the CheXNeXt algorithm to practicing radiologists. *PLOS Medicine*, 15(11):1–17, 2018.

[183] Daniele Ravì, Charence Wong, Fani Deligianni, Melissa Berthelot, Javier Andreu, Benny Lo, and Guang-Zhong Yang. Deep learning for health informatics. *IEEE Journal of Biomedical and Health Informatics*, 21(1):4–21, 2016.

[184] Mauricio Reyes, Raphael Meier, Sérgio Pereira, Carlos Silva, Michael Dahlweid, Hendrik Tengg-Kobligk, Ronald Summers, and Roland Wiest. On the interpretability of artificial intelligence in radiology: Challenges and opportunities. *Radiology: Artificial Intelligence*, 2:190043, 2020.

[185] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144, 2016.

[186] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In *AAAI Conference on Artificial Intelligence*, pages 1–9, 2018.

[187] Nicolas Riche, Matthieu Duvinage, Matei Mancas, Bernard Gosselin, and Thierry Dutoit. Saliency and human fixations: State-of-the-art and study of comparison metrics. In *IEEE International Conference on Computer Vision*, pages 1153–1160, 2013.

[188] Régis Riveret, Jeremy V Pitt, Dimitrios Korkinof, and Moez Draief. Neuro-symbolic agents: Boltzmann machines and probabilistic abstract argumentation with subarguments. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 1481–1489, 2015.

[189] Scott Robbins. A misdirected principle with a catch: explicability for AI. *Minds and Machines*, 29(4):495–514, 2019.

[190] M. Robnik-Siknja and I. Kononeko. Theoretical and empirical analysis of RelifF and RReliefF. *Machine Learning*, 53:23–69, 2003.

[191] F. Rosenblatt. *The Perceptron, a Perceiving and Recognizing Automaton Project Para*. Report: Cornell Aeronautical Laboratory. Cornell Aeronautical Laboratory, 1957.

[192] Avi Rosenfeld and Ariella Richardson. Explainability in human-agent systems. *Autonomous Agents and Multi-Agent Systems*, 33(6):673–705, 2019.

[193] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

[194] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

[195] Stuart Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Prentice Hall, 2002.

[196] Mahya Sadeghi, Parmit K. Chilana, and M. Stella Atkins. How Users Perceive Content-Based Image Retrieval for Identifying Skin Images. In *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*, pages 141–148, 2018.

[197] Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Bach, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE Transactions on Neural Networks and Learning Systems*, 28(11):2660–2673, 2017.

[198] Md Kamruzzaman Sarker, Lu Zhou, Aaron Eberhart, and Pascal Hitzler. Neuro-symbolic artificial intelligence current trends. *AI Communications*, 34(3):197–208, 2021.

[199] Howard M. Schwartz. *Multi-agent machine learning: A reinforcement approach.* John Wiley & Sons, 2014.

[200] John R Searle, PG Searle, S Willis, John Rogers Searle, et al. *Speech acts: An essay in the philosophy of language*, volume 626. Cambridge university press, 1969.

[201] Andrew Selbst and Julia Powles. "meaningful information" and the right to explanation. In *Conference on Fairness, Accountability and Transparency*, pages 48–48, 2018.

[202] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *IEEE International Conference on Computer Vision*, pages 618–626, 2017.

[203] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359, 2020.

[204] Faezehsadat Shahidi, Salwani Mohd Daud, Hafiza Abas, Noor Azurati Ahmad, and Nurazean Maarop. Breast cancer classification using deep learning approaches and histopathology image: A comparison study. *IEEE Access*, 8:187531–187552, 2020.

[205] Li Shen, Laurie R. Margolies, Joseph H. Rothstein, Eugene Fluder, Russell McBride, and Weiva Sieh. Deep learning to improve breast cancer detection on screening mammography. *Scientific Reports*, 9(1):12495, 2019.

[206] Shiwen Shen, Simon X Han, Denise R Aberle, Alex A T Bui, and Willliam Hsu. An Interpretable Deep Hierarchical Semantic Convolutional Neural Network for Lung Nodule Malignancy Classification. *Expert Systems with Applications*, 128:84–95, 2018.

[207] Wilson Silva, Kelwin Fernandes, and Jaime Silva Cardoso. How to produce complementary explanations using an ensemble model. In *International Joint Conference on Neural Networks*, pages 1–8, 2019.

[208] Wilson Silva, Kelwin Fernandes, Maria J. Cardoso, and Jaime S. Cardoso. Towards Complementary Explanations Using Deep Neural Networks. In *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*, pages 133–140, 2018.

[209] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *International Conference on Learning Representations*, pages 1–18, 2014.

[210] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, pages 1–14, 2015.

[211] John Simpson. Oxford english dictionary, 2009.

[212] Munindar P. Singh. Agent communication languages: Rethinking the principles. *Computer*, 31(12):40–47, 1998.

[213] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. SmoothGrad: removing noise by adding noise. In *Workshop on Visualization for Deep Learning - International Conference on Machine Learning*, pages 1–10, 2017.

[214] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. In *International Conference on Learning Representations*, pages 1–14, 2015.

[215] Wolfgang Stammer, Patrick Schramowski, and Kristian Kersting. Right for the right concept: Revising neuro-symbolic concepts by interacting with their explanations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3619–3629, 2021.

[216] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328, 2017.

[217] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic Attribution for Deep Networks. In *International Conference on Machine Learning*, page 3319–3328, 2017.

[218] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[219] Jie Tan, Matthew Ung, Chao Cheng, and Casey Greene. Unsupervised feature construction and knowledge extraction from genome-wide assays of breast cancer with denoising autoencoders. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, 20:132–43, 2015.

[220] Alfred Tarski, Andrzej Mostowski, and Raphael Mitchel Robinson. *Undecidable Theories:Studies in Logic and the Foundation of Mathematics*. Elsevier, 1953.

[221] Erico Tjoa and Cuntai Guan. A Survey on Explainable Artificial Intelligence (XAI): Toward Medical XAI. *IEEE Transactions on Neural Networks and Learning Systems*, 32(11):4793–4813, 2020.

[222] Katarzyna Tomczak, Patrycja Czerwińska, and Maciej Wiznerowicz. The cancer genome atlas (tcga): an immeasurable source of knowledge. *Contemporary oncology*, 19(1A):A68–A77, 2015.

[223] Richard Tomsett, Dave Braines, Dan Harborne, Alun Preece, and Supriyo Chakraborty. Interpretable to whom? a role-based model for analyzing interpretable machine learning systems. *International Conference on Machine Learning Workshop on Human Interpretability in Machine Learning*, pages 1–7, 2018.

[224] Richard J. Tomsett, Daniel Harborne, Supriyo Chakraborty, Prudhvi K. Gurram, and Alun David Preece. Sanity checks for saliency metrics. In *AAAI Conference on Artificial Intelligence*, page 9525–9536, 2020.

[225] Sana Tonekaboni, Shalmali Joshi, Melissa D McCradden, and Anna Goldenberg. What clinicians want: contextualizing explainable machine learning for clinical end use. In *Machine Learning for Healthcare Conference*, pages 359–380, 2019.

[226] Laurens Van Der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[227] P. J. van Diest, C. H. van Deurzen, and G. Cserni. Pathology issues related to SN procedures and increased detection of micrometastases and isolated tumor cells. *Breast Disease*, 31(2):65–81, 2010.

[228] Pieter Van Molle, Miguel De Strooper, Tim Verbelen, Bert Vankeirsbilck, Pieter Simoens, and Bart Dhoedt. Visualizing Convolutional Neural Networks to Improve Decision Support for Skin Lesion Classification. In *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*, pages 115–123, 2018.

[229] Bastiaan S. Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation equivariant cnns for digital pathology. In *International Conference on Medical Image Computing and Computer Assisted Intervention*, pages 210–218, 2018.

[230] Alfredo Vellido. The importance of interpretability and visualization in machine learning for applications in medicine and health care. *Neural Computing and Applications*, 32:18069–18083, 2019.

[231] Mor Vered, Piers Howe, Tim Miller, Liz Sonenberg, and Eduardo Velloso. Demand-driven transparency for monitoring intelligent agents. *IEEE Transactions on Human-Machine Systems*, 50(3):264–275, 2020.

[232] Giulia Vilone and Luca Longo. Notions of explainability and evaluation approaches for explainable artificial intelligence. *Information Fusion*, 76:89–106, 2021.

[233] Giulia Vilone and Luca Longo. Explainable artificial intelligence: a systematic review. *Applied Sciences*, pages 1–38, 2022.

[234] Sandra Wachter, Brent Mittelstadt, and Luciano Floridi. Why a right to explanation of automated decision-making does not exist in the general data protection regulation. *International Data Privacy Law*, 7(2):76–99, 2017.

[235] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harvard Journal of Law & Technology*, 31:841–887, 2017.

[236] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, and Ronald M. Summers. TieNet: Text-Image Embedding Network for Common Thorax Disease Classification and Reporting in Chest X-Rays. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 9049–9058, 2018.

[237] Jamie Ward. *The student's guide to cognitive neuroscience.* Routledge, 2019.

[238] Adrian Weller. Transparency: motivations and challenges. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pages 23–40. Springer, 2019.

[239] Brian Whitworth. Social-technical systems. In *Encyclopedia of human computer interaction*, pages 533–541. IGI Global, 2006.

[240] Martin J Willemink, Wojciech A Koszek, Cailin Hardell, Jie Wu, Dominik Fleischmann, Hugh Harvey, Les R Folio, Ronald M Summers, Daniel L Rubin, and Matthew P Lungren. Preparing Medical Imaging Data for Machine Learning. *Radiology*, 295(1):4–15, 2020.

[241] Jimmy Wu, Diondra Peck, Scott Hsieh, Vandana Dialani M.D., Constance D. Lehman M.D., Bolei Zhou, Vasilis Syrgkanis, Lester Mackey, and Genevieve Patterson. Expert identification of visual primitives used by CNNs during mammogram classification. In *Medical Imaging Computer-Aided Diagnosis*, pages 633–641, 2018.

[242] Xiaozheng Xie, Jianwei Niu, Xuefeng Liu, Zhengsu Chen, Shaojie Tang, and Shui Yu. A survey on incorporating domain knowledge into deep learning for medical image analysis. *Medical Image Analysis*, 69:101985, 2021.

[243] Yiwen Xu, Ahmed Hosny, Roman Zeleznik, Chintan Parmar, Thibaud Coroller, Idalid Franco, Raymond H. Mak, and Hugo J.W.L. Aerts. Deep learning predicts lung cancer treatment response from serial medical imaging. *Clinical Cancer Research*, 25(11):3266–3275, 2019.

[244] Qingsong Yang, Pingkun Yan, Yanbo Zhang, Hengyong Yu, Yongyi Shi, Xuanqin Mou, Mannudeep K. Kalra, Yi Zhang, Ling Sun, and Ge Wang. Low-dose ct image denoising using a generative adversarial network with wasserstein distance and perceptual loss. *IEEE Transactions on Medical Imaging*, 37(6):1348–1357, 2018.

[245] W. Yang, K. Wang, and W. Zuo. Neighborhood Component Feature Selection for High-Dimensional Data. *Journal of Computers*, 7(1):162–168, 2012.

[246] Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Sai Suggala, David I Inouye, and Pradeep Ravikumar. On the (in)fidelity and sensitivity for explanations. In *Conference on Neural Information Processing Systems*, pages 1–12, 2019.

[247] Gal Yona and Daniel Greenfeld. Revisiting Sanity Checks for Saliency Maps. In *Advances in Neural Information Processing Systems Workshop on eXplainable AI Approaches for Debugging and Diagnosis*, pages 1–10, 2021.

[248] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding Neural Networks Through Deep Visualization. In *International Conference on Machine Learning - Deep Learning Workshop*, pages 1–12, 2015.

[249] Safoora Yousefi, Fatemeh Amrollahi, Mohamed Amgad, Chengliang Dong, Joshua E Lewis, Congzheng Song, David A Gutman, Sameer H Halani, Jose Enrique Velazquez Vega, Daniel J Brat, and Lee A D Cooper. Predicting clinical outcomes from large scale cancer genomic profiles with deep survival models. *Scientific Reports*, 7:11707, 2017.

[250] John R. Zech, Marcus A. Badgeley, Manway Liu, Anthony B. Costa, Joseph J. Titano, and Eric Karl Oermann. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. *PLOS Medicine*, 15(11):1–17, 2018.

[251] Matthew D. Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks. In *European Conference on Computer Vision*, pages 818–833, 2014.

[252] Hai-Miao Zhang and Bin Dong. A review on deep learning in medical image reconstruction. *Journal of the Operations Research Society of China*, 8:311–340, 2020.

[253] Kai Zhang, Xiyang Liu, Fan Liu, Lin He, Lei Zhang, Yahan Yang, Wangting Li, Shuai Wang, Lin Liu, Zhenzhen Liu, Xiaohang Wu, and Haotian Lin. An interpretable and expandable deep learning diagnostic system for multiple ocular diseases: Qualitative study. *Journal of Medical Internet Research*, 20(11):1–13, 2018.

[254] Shijie Zhang, Yukun Luo, Huarui Du, Zhuang Jin, Yaqiong Zhu, Ying Zhang, Fang Xie, Mingbo Zhang, Xiaoqi Tian, and Jue Zhang. A Novel Interpretable Computer-Aided Diagnosis System of Thyroid Nodules on Ultrasound based on Clinical Experience. *IEEE Access*, pages 1–10, 2020.

[255] Xinyang Zhang, Ningfei Wang, Hua Shen, Shouling Ji, Xiapu Luo, and Ting Wang. Interpretable Deep Learning under Fire. In *Symposium on Edge Computing*, page 1659–1676, 2020.

[256] Yunfeng Zhang, Q Vera Liao, and Rachel KE Bellamy. Effect of confidence and explanation on accuracy and trust calibration in ai-assisted decision making. In *Conference on Fairness, Accountability, and Transparency*, pages 295–305, 2020.

[257] Zijiao Zhang, Chong Wu, Shiyou Qu, and Xiaofang Chen. An explainable artificial intelligence approach for financial distress prediction. *Information Processing & Management*, 59(4):24 pages, 2022.

[258] Zizhao Zhang, Pingjun Chen, Mason Mcgough, Fuyong Xing, Chunbao Wang, Marilyn Bui, Yuanpu Xie, Manish Sapkota, Lei Cui, Jasreman Dhillon, Nazeel Ahmad, Farah K Khalil, Shohreh I Dickinson, Xiaoshuang Shi, Fujun Liu, Hai Su, Jinzheng Cai, and Lin Yang. Pathologist-level interpretable whole-slide cancer diagnosis with deep learning. *Nature Machine Intelligence*, 1(1):236–245, 2019.

[259] Zizhao Zhang, Yuanpu Xie, Fuyong Xing, Mason McGough, and Lin Yang. MDNet: A Semantically and Visually Interpretable Medical Image Diagnosis Network. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3549–3557, 2017.

[260] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning Deep Features for Discriminative Localization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2929, 2016.

[261] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2929, 2016.

[262] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision*, pages 2242–2251, 2017.

[263] Wan Zhu, Longxiang Xie, Jianye Han, and Xiangqian Guo. The application of deep learning in cancer prognosis prediction. *Cancers*, 12(3):603, 2020.

This page is intentionally left blank.

Appendices

Appendix A

Saliency metrics for each perturbation scenario

This appendix provides supporting information to the work developed in Chapter 5.

method	level	↓ mse	↓mae	↑ auc_judd	↑ auc_borji	↑ auc_shuff	↑ nss	↑ infogain	↑ sim	↑ cc	↓ kldiv
	8x8	0.066	0.141	0.823	0.640	0.606	1.188	-5.806	0.771	0.865	1.056
GradCAM	16x16	0.077	0.169	0.793	0.686	0.643	1.031	-5.336	0.774	0.728	1.549
	32x32	0.085	0.191	0.737	0.672	0.633	0.805	-5.017	0.778	0.563	1.568
	8x8	0.029	0.104	0.599	0.541	0.534	0.104	-4.396	0.611	0.055	1.022
GradientsInputs	6x16	0.025	0.096	0.548	0.527	0.522	0.090	-3.961	0.590	0.045	0.780
	32x32	0.015	0.072	0.547	0.534	0.530	0.146	-3.889	0.545	0.063	0.839
	8x8	0.062	0.125	0.831	0.660	0.642	1.342	-6.471	0.742	0.872	1.679
Guided-GradCAM	16x16	0.070	0.147	0.805	0.706	0.657	1.167	-7.118	0.734	0.764	2.434
	32x32	0.077	0.171	0.766	0.704	0.638	0.998	-5.226	0.758	0.633	1.867
	8x8	0.030	0.113	0.623	0.556	0.550	0.217	-4.124	0.658	0.101	0.861
IntegratedGradients	16x16	0.028	0.110	0.571	0.547	0.538	0.185	-3.681	0.652	0.087	0.542
	32x32	0.022	0.099	0.553	0.540	0.533	0.169	-3.593	0.631	0.078	0.514
	8x8	0.003	0.003	0.560	0.515	0.502	1.581	-2.534	0.121	1.000	0.002
OcclusionSensitivity	16x16	0.004	0.004	0.567	0.527	0.504	1.403	-2.289	0.134	0.997	0.047
	32x32	0.006	0.007	0.581	0.536	0.512	0.856	-8.049	0.332	0.950	0.088
	8x8	0.018	0.090	0.661	0.586	0.544	0.335	-3.831	0.688	0.184	0.751
SmoothGrad	16x16	0.015	0.080	0.638	0.606	0.541	0.451	-3.390	0.698	0.184	0.429
	32x32	0.014	0.078	0.647	0.622	0.551	0.565	-3.192	0.711	0.240	0.320
	8x8	0.029	0.109	0.607	0.545	0.534	0.140	-4.177	0.648	0.070	0.889
VanillaGradients	6x16	0.024	0.096	0.555	0.535	0.534	0.123	-3.833	0.626	0.056	0.649
	32x32	0.015	0.073	0.544	0.534	0.535	0.144	-4.055	0.565	0.066	0.917

Table A.1: Average saliency metrics for each saliency map method when perturbing with different levels fron the NN scenario. ↓ indicates the dissimilarity metrics which should be lower and ↑ indicates similarity metrics which should be higher for better methods.

method	level	↑ mse	↑ mae	↓ auc_judd	↓ auc_borji	↓ auc_shuff	↓ nss	↓ infogain	↓ sim	↓ cc	↑ kldiv
	8x8	0.066	0.143	0.820	0.641	0.606	1.172	-5.758	0.767	0.858	1.036
GradCAM	16x16	0.084	0.182	0.768	0.673	0.635	0.936	-5.391	0.765	0.666	1.579
	32x32	0.101	0.218	0.652	0.611	0.584	0.496	-5.516	0.746	0.356	1.772
	8x8	0.031	0.108	0.596	0.539	0.530	0.093	-4.311	0.615	0.048	0.980
GradientsInputs	16x16	0.036	0.124	0.541	0.522	0.518	0.068	-4.058	0.595	0.034	0.790
	32x32	0.028	0.110	0.516	0.507	0.513	0.021	-4.116	0.544	0.012	0.908
	8x8	0.065	0.130	0.828	0.660	0.642	1.326	-6.599	0.735	0.862	1.751
Guided-GradCAM	16x16	0.079	0.162	0.794	0.699	0.650	1.114	-7.363	0.724	0.729	2.663
	32x32	0.108	0.217	0.725	0.673	0.614	0.824	-5.626	0.739	0.537	2.060
	8x8	0.035	0.123	0.619	0.553	0.547	0.200	-4.137	0.656	0.094	0.866
IntegratedGradients	16x16	0.038	0.134	0.561	0.539	0.530	0.145	-3.723	0.646	0.070	0.555
	32x32	0.031	0.122	0.530	0.521	0.517	0.080	-3.681	0.616	0.038	0.550
	8x8	0.004	0.004	0.560	0.515	0.502	1.593	-2.522	0.120	1.000	0.003
OcclusionSensitivity	16x16	0.004	0.005	0.567	0.528	0.504	1.438	-2.242	0.134	0.998	0.035
	32x32	0.008	0.009	0.580	0.536	0.512	0.874	-7.812	0.322	0.935	0.075
	8x8	0.018	0.090	0.662	0.587	0.545	0.342	-3.831	0.689	0.187	0.753
SmoothGrad	6x16	0.014	0.079	0.639	0.606	0.541	0.453	-3.381	0.699	0.185	0.428
	32x32	0.014	0.078	0.644	0.619	0.549	0.553	-3.202	0.710	0.233	0.322
	8x8	0.036	0.123	0.606	0.544	0.531	0.135	-4.255	0.645	0.068	0.936
VanillaGradients	16x16	0.042	0.137	0.550	0.530	0.527	0.091	-3.953	0.622	0.046	0.722
	32x32	0.031	0.116	0.527	0.515	0.520	0.050	-4.295	0.559	0.025	1.007

Table A.2: Average saliency metrics for each saliency map method when perturbing with different levels from NT scenario. ↑ indicates the dissimilarity metrics which should be higher for better methods and ↓ indicates similarity metrics which should be lower for better methods.

method	level	↑ mse	↑ mae	↓ auc_judd	↓ auc_borji	↓ auc_shuff	↓ nss	↓ infogain	↓ sim	↓ cc	↑ kldiv
GradCAM	8x8	0.061	0.132	0.836	0.664	0.658	1.334	-6.300	0.753	0.878	1.575
GradCAM	16x16	0.073	0.160	0.804	0.697	0.681	1.146	-6.876	0.747	0.749	2.369
GradCAM	32x32	0.085	0.191	0.753	0.690	0.681	0.948	-5.226	0.767	0.583	1.810
GradientsInputs	8x8	0.018	0.084	0.606	0.545	0.533	0.130	-4.186	0.622	0.065	0.949
GradientsInputs	16x16	0.013	0.070	0.570	0.547	0.542	0.173	-3.744	0.609	0.085	0.650
GradientsInputs	32x32	0.009	0.055	0.619	0.589	0.589	0.377	-3.470	0.578	0.170	0.687
Guided-GradCAM	8x8	0.057	0.115	0.845	0.626	0.616	1.283	-6.426	0.768	0.908	1.587
Guided-GradCAM	16x16	0.060	0.130	0.827	0.689	0.657	1.151	-7.035	0.768	0.818	2.298
Guided-GradCAM	32x32	0.068	0.156	0.792	0.709	0.669	1.019	-5.024	0.799	0.706	1.643
IntegratedGradients	8x8	0.021	0.096	0.613	0.548	0.540	0.159	-4.156	0.653	0.080	0.872
IntegratedGradients	16x16	0.019	0.091	0.574	0.550	0.544	0.187	-3.662	0.651	0.090	0.537
IntegratedGradients	32x32	0.017	0.086	0.576	0.558	0.553	0.243	-3.506	0.634	0.111	0.500
OcclusionSensitivity	8x8	0.006	0.006	0.558	0.514	0.502	1.550	-2.559	0.117	0.999	0.006
OcclusionSensitivity	16x16	0.009	0.009	0.568	0.530	0.505	1.378	-2.286	0.150	0.992	0.055
OcclusionSensitivity	32x32	0.013	0.014	0.584	0.539	0.512	0.894	-7.889	0.328	0.935	0.109
SmoothGrad	8x8	0.019	0.092	0.658	0.583	0.543	0.323	-3.861	0.689	0.180	0.754
SmoothGrad	16x16	0.015	0.081	0.635	0.603	0.538	0.442	-3.398	0.698	0.178	0.431
SmoothGrad	32x32	0.014	0.079	0.645	0.620	0.548	0.558	-3.199	0.710	0.234	0.320
VanillaGradients	8x8	0.022	0.097	0.611	0.547	0.536	0.151	-4.148	0.652	0.084	0.868
VanillaGradients	16x16	0.015	0.075	0.580	0.555	0.550	0.206	-3.656	0.639	0.100	0.568
VanillaGradients	32x32	0.010	0.060	0.608	0.582	0.574	0.349	-3.515	0.591	0.161	0.652

Table A.3: Average saliency metrics for each saliency map method when perturbing with different levels from a TN scenario. ↑ indicates the dissimilarity metrics which should be higher for better methods and ↓ indicates similarity metrics which should be lower for better methods.