A Neural-Symbolic learning framework to produce interpretable predictions for image classification

Adrien Bennetot

École doctorale 130 : Informatique, Télécommunications et Électronique de Paris

THÈSE

pour obtenir le grade de docteur délivré par

Sorbonne Université

Spécialité doctorale « Informatique»

présentée et soutenue publiquement par

Adrien Bennetot

le 12 Juillet 2022

A Neural-Symbolic learning framework to produce interpretable predictions for image classification

Directeur de thèse : **Raja CHATILA**
Co-encadrant de thèse : **Natalia DÍAZ-RODRÍGUEZ**

Jury

M. Raja Chatila,	Professeur, Sorbonne Université	Directeur de Thèse
Mme. Natalia Díaz-Rodríguez,	Maîtresse de Conférences, Université de Grenade	Co-encadrant de Thèse
M. Patrick Gallinari,	Professeur, Sorbonne Université	Examinateur
Mme. Siham Tabik,	Maîtresse de Conférences, Université de Grenade	Examinateur
M. Javier Del Ser,	Professeur, Université du Pays Basque	Examinateur
Mme. Monreale Anna,	Maîtresse de Conférences (HDR), Université de Pise	Rapporteur
M. Holzinger Andreas,	Professeur, Université de Vienne	Rapporteur

Abstract

Artificial Intelligence has been developing exponentially over the last decade. Its evolution is mainly linked to the progress of computer graphics card processors, allowing to accelerate the calculation of learning algorithms, and to the access to massive volumes of data. This progress has been principally driven by a search for quality prediction models, making them extremely accurate but opaque. Their large-scale adoption is hampered by their lack of transparency, thus causing the emergence of eXplainable Artificial Intelligence (XAI). This new line of research aims at fostering the use of learning models based on mass data by providing methods and concepts to obtain explanatory elements concerning their functioning. However, the youth of this field causes a lack of consensus and cohesion around the key definitions and objectives governing it. This thesis contributes to the field through two perspectives, one through a theory of what is XAI and how to achieve it and one practical. The first is based on a thorough review of the literature, resulting in two contributions: 1) the proposal of a new definition for Explainable Artificial Intelligence and 2) the creation of a new taxonomy of existing explainability methods. The practical contribution consists of two learning frameworks, both based on a paradigm aiming at linking the connectionist and symbolic paradigms. The first framework, Greybox, sequentially combines an opaque and a transparent model in order to obtain an interpretable image classification based on the different parts of objects that can be recognized on the image. The second framework, Transparent Distillation, aims at distilling the expert knowledge contained in a transparent classifier into a deep learning model. These two frameworks fill a gap in the state of the art by allowing the user to obtain a classification with similar performance to opaque models while having a valid and faithful explanation of why the prediction was made. Ultimately, the achievements detailed in this thesis contribute to the general knowledge on the explainability of learning models at a time when the main challenge of Artificial Intelligence is to meet ethical, trust and reliability criteria.

Acknowledgements

The thesis would have not become what it is without a large amount of people involved and that I am grateful to.

I would especially like to thank my thesis director Pr. Raja Chatila and my co-supervisor Dr. Natalia Díaz-Rodríguez who guided me during these years of research. I owe them original lines of research as well as a scientific and methodological formation of high standard. I am grateful for their support, their precious advice, their critics and the trust they gave me during this thesis. I had the honor of being one of the last doctoral students of Raja and the very first of Natalia.

I would like to express my sincere thanks to the two rapporteurs of this thesis, Pr. Andreas Holzinger and Dr. Anna Monreale for dedicating their time to the reading of this manuscript. I would also like to thank the members of the jury Pr. Patrick Gallinari, Pr. Javier Del Ser and Dr. Siham Tabik for having accepted to examine this thesis.

I would also like to thank Jean-Luc Laurent who was decisive to launch this CIFRE thesis project and who put me on the right track during my first months of thesis, while continuing to give me his support until the end of the writing of my manuscript. I would like to take this opportunity to thank the company Segula Technologies for having financed this thesis, as well as the laboratories U2IS of ENSTA Paris and ISIR of Sorbonne University for having welcomed me.

I would also like to thank Javier Del Ser and Gianni Franchi who have spontaneously dedicated a considerable amount of time to help me in my work and whose technical and theoretical expertise has been decisive in the advancement of this thesis.

My warmest thanks to Siham Tabik, Alberto Barbado, Salvador Garcia, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins and Francisco Herrera for their contributions to this thesis.

To my friend from the Master's degree, Benoît Geslain, who started his thesis between Segula and ISIR at the same time as me and with whom we have lived and shared the same adventures during these 3 years.

Thanks to Caroline, Rémi, Gaël, Thibault and my other colleagues at ENSTA where I spent a large part of my time, and to Alexandre Chapoutot, David Fillat and Goran Frehse

for their support and advices in the different stages of my PhD.

Thank you to my family, especially to my sister Camille and my parents for their support and encouragement, they have been key elements of my motivation during all these years. A little wink to Zizou le lapin who allowed me to have a lot of free illustrations.

And finally for Mathilde. Thank you for having been by my side during these 3 years, for having made so many sacrifices especially during these last weeks of writing the manuscript. Clearly all this would not have been possible without you.

Contents

List of Symbols

AI Artificial Intelligence: the theory and development of computer systems able to perform tasks normally requiring human intelligence such as visual perception speech recognition decision-making and translation between languages.

DL Deep Learning: part of a broader family of machine learning methods based on artificial neural networks with representation learning.

DNN Deep Neural Network: A deep neural network is a neural network with a certain level of complexity, with more than two layers

KB Knowledge Base: the underlying set of facts assumptions and rules which a computer system has available to solve a problem.

KG Knowledge Graph: A knowledge graph, also known as a semantic network, represents a network of real-world entities—i.e. objects, events, situations, or concepts—and illustrates the relationship between them. This information is usually stored in a graph database and visualized as a graph structure, prompting the term knowledge "graph."

ML Machine Learning: part of artificial intelligence. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so.

NB Naive Bayes: family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the feature.

NeSy Neural Symbolic Artificial Intelligence: Hybrid AI approach combining neural networks and classical rule-based symbolic AI.

OWL Web Ontology Language: a family of knowledge representation languages or ontology languages for authoring ontologies or knowledge. bases. The languages

are characterized by formal semantics and RDF/ XML-based serializations for the Semantic Web.

RDF Resource Description Framework: it consists of a number of tools that use concepts from graph theory to add relationships and semantics to unstructured data such as the World Wide Web. The central aim for the RDF framework is to provide a way for machine inter-operation of cross-domain data and merging information from different sources as effortless as possible. An RDF triple or statement is the foundation of the RDF data model. It consists of a subject a predicate and an object resource that together form a statement. Triples consisting of matching subjects and objects can be linked together to form an RDF graph.

XAI Explainable Artificial Intelligence: given an audience, an explainable Artificial Intelligence is one that produces details or reasons to make its functioning clear or easy to understand.

Chapter 1

Introduction

1.1 Context

Artificial Intelligence (AI) is a seventy year old field covering a variety of sciences, theories and technologies aimed at mimicking human cognitive abilities. The hype around AI in the 1960s quickly died down, mainly due to the limitations of computing at the time. Since 2010, the discipline has experienced a new boom, mainly due to significant improvements in computing power and access to large amounts of data. These two points have allowed a paradigm shift from expert systems, designed to be a logical mirror of human reasoning by following strict and precise rules, to connectionist models discovering by themselves the correlations existing in huge datasets.

At first, AI models were easily readable and explainable, but this trend quickly ended. The term black-box started to appear with expert systems containing several hundred rules, making it difficult for a human to understand how the model works. This opacity has increased tenfold with the paradigm shift and the new prosperity of Deep Learning (DL). The vastness of the parameter space of DL models and the absence of concrete rules governing their decision making means that their predictions cannot be explained, either by an external explanatory element or by the model developer. It makes Deep Neural networks (DNNs) considered as complex *black-box* models [1]. The opposite of *black-box-ness* is *transparency*, i.e., the search for a direct understanding of the mechanism by which a model works [2].

The danger is on creating and using decisions that are not justifiable, legitimate, or that simply do not allow obtaining detailed explanations of their behaviour [3]. Explanations supporting the output of a model are crucial, e.g., in precision medicine, where experts require far more information from the model than a simple binary prediction for supporting their diagnosis [4]. Other examples include autonomous vehicles in transportation, security

and finance, among others. As black-box Machine Learning (ML) models are increasingly being employed to make important predictions in critical contexts [5], the demand for transparency is increasing from the various stakeholders in AI [6]. In critical settings [7], robustness to input data perturbation is an important desideratum for security, while an equally important requirement is to produce explanations that can be understood by a human user [8].

In general, humans are reluctant to adopt techniques that are not directly interpretable, tractable and trustworthy [9], given the increasing demand for ethical AI [10]. It is customary to think that by focusing solely on performance, the systems will be increasingly opaque. This is true in the sense that there is a trade-off between the performance of a model and its transparency [11]. However, an improvement in the understanding of a system can lead to the correction of its deficiencies. When developing a ML model, the consideration of interpretability as an additional design driver can improve its implementability for 3 reasons:

- Interpretability helps ensure impartiality in decision-making, i.e. to detect, and consequently, correct from bias in the training dataset.
- Interpretability facilitates the provision of robustness by highlighting potential adversarial perturbations that could change the prediction.
- Interpretability can act as an insurance that only meaningful variables infer the output, i.e., guaranteeing that an underlying truthful causality exists in the model reasoning.

All these means that the interpretation of the system should, in order to be considered practical, provide either an understanding of the model mechanisms and predictions, a visualization of the model's discrimination rules, or hints on what could perturb the model [12].

In order to avoid limiting the effectiveness of the current generation of AI systems, *eXplainable AI* [3] proposes creating a suite of ML techniques that 1) produce more explainable models while maintaining a high level of learning performance (e.g., prediction accuracy), and 2) enable humans to understand, appropriately trust, and effectively manage the emerging generation of artificially intelligent partners. XAI draws as well insights from the Social Sciences [13] and considers the psychology of explanation.

1.2 Motivation and Objectives

The various milestones in AI in recent years have been driven by a search for performance. This quest for precision and accuracy in predictions has sometimes been done at the expense

of all other considerations. The models are more and more opaque and the state of the art in most domains is achieved by deep architectures of neural networks handling high dimensional data. It has become impossible to do without these models in certain fields, such as image recognition or natural language processing, as they outperform traditional algorithms. However, as explained above, it is not possible to use these high-performance models in critical environments. On some tasks such as image recognition, one finds oneself with high performance models that are opaque and therefore impossible to deploy, and on the other hand, models that are transparent but whose performance is far from the state of the art.

Assuming that it is fundamental to be able to answer the question of "Why was this prediction made?", there are 3 possibilities to try to overcome this trade-off: 1) improve transparent methods to make them as good as black boxes, 2) create post-hoc methods to extract explanatory elements from black boxes, and 3) combine opaque and transparent methods to form a new model drawing on the strengths of both paradigms. The common concern of these different points is that there is no unified framework posing a consensus on what constitutes an explanation and on the different concepts governing the notions within XAI and its need is felt as the field is growing extremely fast. Therefore, this thesis focuses on two main objectives, one theorizing what is XAI and how to achieve it and one practical. First, the creation of a guide to categorize and group all the work that has been done in the field in recent years under a single set of definitions and concepts. Second, to push the current limits of the trade-off between performance and explainability by developing new learning frameworks based on a hybrid combination of opaque and transparent methods.

This thesis is thus organized around 4 main themes that have guided these 3 years of research:

- **Surveying the field of XAI**: A set of concepts that summarize the diverse references found in the literature, trying to build a consensus around it. This is accompanied by a new taxonomy to classify explainability methods.

- **Neural-Symbolic Framework**: The formalization of a neural-symbolic learning framework, allowing to produce predictions through a neural network while making them explicit through a symbolic knowledge base. In this framework, it is necessary that the symbolic knowledge base influences and constrains the learning of the neural network. The question of how to constrain the neural network and the various forms that the knowledge base can take is the basis of the work carried out thereafter.

- **Greybox XAI**: a learning framework that is intended to be a use case extracted from the neural-symbolic formalization proposed previously. Here the expert knowledge

base takes the form of a logical membership relation between objects and their sub-parts, while the constraint is found through a serial composition of a neural network and a logistic regression.

- **Transparent Distillation**: a second neural-symbolic use case. This time the learning constraint takes the form of a knowledge distillation between a transparent teacher and an opaque student, whose role is to predict both the different parts of objects on the image and the major object present on the image.

1.3 Reading this thesis

This manuscript has been written so that each chapter reports on the work that has been done to advance each of the above objectives. Thus, the first chapter reports the theory of what is XAI and how to achieve it by proposing a complete survey of the literature. The second chapter lays the foundations of the position taken with respect to XAI by establishing a modular Neural-Symbolic (NeSy) framework. The NeSy philosophy will be taken up again in the two following chapters in order to propose two learning frameworks with the aim of producing accurate and explainable predictions. Therefore, a field expert will be able to skip to Chapter 2 and focus on the technical and practical contribution of the manuscript.

Chapter 2

Concepts and Taxonomies toward Explainable AI

This first chapter covers the theoretical contribution of this thesis. It proposes a new set of concepts and definitions, and reports a new taxonomy of explainability methods based on these definitions.

2.1 Introduction

This literature outbreak shares its rationale with the research agendas of national governments and agencies. Although some recent surveys [4, 14, 11, 15, 16, 17, 18] summarize the upsurge of activity in XAI across sectors and disciplines, this overview aims to cover the creation of a complete unified framework of categories and concepts that allow for scrutiny and understanding of the field of XAI methods. Furthermore, we pose intriguing thoughts around the explainability of AI models in data fusion contexts with regards to data privacy and model confidentiality. This, along with other research opportunities and challenges identified throughout our study, serve as the pull factor toward Responsible Artificial Intelligence, term by which we refer to a series of AI principles to be necessarily met when deploying AI in real applications. As we will later show in detail, model explainability is among the most crucial aspects to be ensured within this methodological framework. All in all, the novel contributions of this overview can be summarized as follows:

1. Grounded on a first elaboration of concepts and terms used in XAI-related research, we propose a novel definition of explainability that places *audience* (Figure 2.1) as a key aspect to be considered when explaining a ML model. We also elaborate on the diverse purposes sought when using XAI techniques, from trustworthiness to privacy

awareness, which round up the claimed importance of purpose and targeted audience in model explainability.

2. We define and examine the different levels of transparency that a ML model can feature by itself, as well as the diverse approaches to post-hoc explainability, namely, the explanation of ML models that are not transparent by design.

3. We thoroughly analyze the literature on XAI and related concepts published to date, covering approximately 400 contributions arranged into two different taxonomies. The first taxonomy addresses the explainability of ML models using the previously made distinction between transparency and post-hoc explainability, including models that are transparent by themselves, Deep and non-Deep (i.e., *shallow*) learning models. The second taxonomy deals with XAI methods suited for the explanation of Deep Learning models, using classification criteria closely linked to this family of ML methods (e.g. layerwise explanations, representation vectors, attention).

4. We enumerate a series of challenges of XAI that still remain insufficiently addressed to date. Specifically, we identify research needs around the concepts and metrics to evaluate the explainability of ML models, and outline research directions toward making Deep Learning models more understandable. We further augment the scope of our prospects toward the implications of XAI techniques in regards to confidentiality, robustness in adversarial settings, data diversity, and other areas intersecting with explainability.

5. After the previous prospective discussion, we arrive at the concept of Responsible Artificial Intelligence, a manifold concept that imposes the systematic adoption of several AI principles for AI models to be of practical use. In addition to explainability, the guidelines behind Responsible AI establish that fairness, accountability and privacy should also be considered when implementing AI models in real environments.

The remainder of this overview is structured as follows: first, Section 2.2 and subsections therein open a discussion on the terminology and concepts revolving around explainability and interpretability in AI, ending up with the aforementioned novel definition of interpretability (Subsections 2.2.1 and 2.2.2), and a general criterion to categorize and analyze ML models from the XAI perspective. Sections 2.3 and 2.4 proceed by reviewing recent findings on XAI for ML models (on transparent models and post-hoc techniques respectively) that comprise the main division in the aforementioned taxonomy. We also include a review on hybrid approaches among the two, to attain XAI. Benefits and caveats of the synergies among the families of methods are discussed in Section 2.5, where we present a prospect of general challenges and some consequences to be cautious about. Finally, Section 2.6 elaborates on the concept of Responsible Artificial Intelligence. Section

2.7 concludes the survey with an outlook aimed at engaging the community around this vibrant research area, which has the potential to impact society, in particular those sectors that have progressively embraced ML as a core technology of their activity.

2.2 Explainability: What, Why, What For and How?

Before proceeding with our literature study, it is convenient to first establish a common point of understanding on what the term *explainability* stands for in the context of AI and, more specifically, ML. This is indeed the purpose of this section, namely, to pause at the numerous definitions that have been done in regards to this concept (what?), to argue why explainability is an important issue in AI and ML (why? what for?) and to introduce the general classification of XAI approaches that will drive the literature study thereafter (how?).

2.2.1 Terminology Clarification

One of the issues that hinders the establishment of common grounds is the interchangeable misuse of interpretability and explainability in the literature. There are notable differences among these concepts. To begin with, interpretability refers to a passive characteristic of a model referring to the level at which a given model makes sense for a human observer. This feature is also expressed as transparency. By contrast, explainability can be viewed as an active characteristic of a model, denoting any action or procedure taken by a model with the intent of clarifying or detailing its internal functions.

To summarize the most commonly used nomenclature, in this section we clarify the distinction and similarities among terms often used in the ethical AI and XAI communities.

- **Understandability** (or equivalently, **intelligibility**) denotes the characteristic of a model to make a human understand its function – how the model works – without any need for explaining its internal structure or the algorithmic means by which the model processes data internally [19].

- **Comprehensibility**: when conceived for ML models, comprehensibility refers to the ability of a learning algorithm to represent its learned knowledge in a human understandable fashion [20, 21, 22]. This notion of model comprehensibility stems from the postulates of Michalski [23], which stated that *"the results of computer induction should be symbolic descriptions of given entities, semantically and structurally similar to those a human expert might produce observing the same entities. Components of these descriptions should be comprehensible as single 'chunks' of information, directly interpretable in*

natural language, and should relate quantitative and qualitative concepts in an integrated fashion". Given its difficult quantification, comprehensibility is normally tied to the evaluation of the model complexity [18].

- **Interpretability**: it is defined as the ability to explain or to provide the meaning in understandable terms to a human.

- **Explainability**: explainability is associated with the notion of explanation as an interface between humans and a decision maker that is, at the same time, both an accurate proxy of the decision maker and comprehensible to humans [18].

- **Transparency**: a model is considered to be transparent if by itself it is understandable. Since a model can feature different degrees of understandability, transparent models in Section 2.3 are divided into three categories: simulatable models, decomposable models and algorithmically transparent models [2].

In all the above definitions, *understandability* emerges as the most essential concept in XAI. Both transparency and interpretability are strongly tied to this concept: while transparency refers to the characteristic of a model to be, on its own, understandable for a human, understandability measures the degree to which a human can understand a decision made by a model. Comprehensibility is also connected to understandability in that it relies on the capability of the audience to understand the knowledge contained in the model. All in all, understandability is a two-sided matter: model understandability and human understandability. This is the reason why the definition of XAI given in Section 2.2.2 refers to the concept of *audience*, as the cognitive skills and pursued goal of the users of the model have to be taken into account jointly with the intelligibility and comprehensibility of the model in use. This prominent role taken by understandability makes the concept of *audience* the cornerstone of XAI, as we next elaborate in further detail.

2.2.2 What?

Although it might be considered to be beyond the scope of this paper, it is worth noting the discussion held around general theories of explanation in the realm of philosophy [24]. Many proposals have been done in this regard, suggesting the need for a general, unified theory that approximates the structure and intent of an explanation. However, nobody has stood the critique when presenting such a general theory. For the time being, the most agreed-upon thought blends together different approaches to explanation drawn from diverse knowledge disciplines. A similar problem is found when addressing interpretability in AI. It appears from the literature that there is not yet a common point of understanding

on what interpretability or explainability are. However, many contributions claim the achievement of interpretable models and techniques that empower explainability.

To shed some light on this lack of consensus, it might be interesting to place the reference starting point at the definition of the term Explainable Artificial Intelligence given by D. Gunning in [3]:

> *"XAI will create a suite of machine learning techniques that enables human users to understand, appropriately trust, and effectively manage the emerging generation of artificially intelligent partners"*

This definition brings together two concepts (understanding and trust) that need to be addressed in advance. However, it misses to consider other purposes motivating the need for interpretable AI models, such as causality, transferability, informativeness, fairness and confidence [2, 25, 26, 27]. We will later delve into these topics, mentioning them here as a supporting example of the incompleteness of the above definition.

As exemplified by the definition above, a thorough, complete definition of explainability in AI still slips from our fingers. A broader reformulation of this definition (e.g. *"An explainable Artificial Intelligence is one that produces explanations about its functioning"*) would fail to fully characterize the term in question, leaving aside important aspects such as its purpose. To build upon the completeness, a definition of explanation is first required.

As extracted from the Cambridge Dictionary of English Language, an explanation is *"the details or reasons that someone gives to make something clear or easy to understand"* [28]. In the context of an ML model, this can be rephrased as: *"the details or reasons a model gives to make its functioning clear or easy to understand"*. It is at this point where opinions start to diverge. Inherently stemming from the previous definitions, two ambiguities can be pointed out. First, the details or the reasons used to explain, are completely dependent of the audience to which they are presented. Second, whether the explanation has left the concept clear or easy to understand also depends completely on the audience. Therefore, the definition must be rephrased to reflect explicitly the dependence of the explainability of the model on the audience. To this end, a reworked definition could read as:

> *Given a certain audience, explainability refers to the details and reasons a model gives to make its functioning clear or easy to understand.*

Since explaining, as argumenting, may involve weighting, comparing or convincing an audience with logic-based formalization of (counter) arguments [29], explainability might convey us into the realm of cognitive psychology and the *psychology of explanations* [3], since measuring whether something has been understood or put clearly is a hard task to be gauged objectively. However, measuring to which extent the internals of a model can be

explained could be tackled objectively. Any means to reduce the complexity of the model or to simplify its outputs should be considered as an XAI approach. How big this leap is in terms of complexity or simplicity will correspond to how explainable the resulting model is. An underlying problem that remains unsolved is that the interpretability gain provided by such XAI approaches may not be straightforward to quantify: for instance, a model simplification can be evaluated based on the reduction of the number of architectural elements or number of parameters of the model itself (as often made, for instance, for DNNs). On the contrary, the use of visualization methods or natural language for the same purpose does not favor a clear quantification of the improvements gained in terms of interpretability. The derivation of general metrics to assess the quality of XAI approaches remain as an open challenge that should be under the spotlight of the field in forthcoming years. We will further discuss on this research direction in Section 2.5.

Explainability is linked to post-hoc explainability since it covers the techniques used to convert a non-interpretable model into a explainable one. In the remaining of this manuscript, explainability will be considered as the main design objective, since it represents a broader concept. A model can be explained, but the interpretability of the model is something that comes from the design of the model itself. Bearing these observations in mind, explainable AI can be defined as follows:

*Given an audience, an **explainable** Artificial Intelligence is one that produces details or reasons to make its functioning clear or easy to understand.*

This definition is posed here as a first contribution of the present overview, implicitly assumes that the ease of understanding and clarity targeted by XAI techniques for the model at hand reverts on different application purposes, such as a better trustworthiness of the model's output by the audience.

2.2.3 Why?

As stated in the introduction, explainability is one of the main barriers AI is facing nowadays in regards to its practical implementation. The inability to explain or to fully understand the reasons by which state-of-the-art ML algorithms perform as well as they do, is a problem that find its roots in two different causes, which are conceptually illustrated in Figure 2.1.

Without a doubt, the first cause is the gap between the research community and business sectors, impeding the full penetration of the newest ML models in sectors that have traditionally lagged behind in the digital transformation of their processes, such as banking, finances, security and health, among many others. In general this issue occurs in strictly regulated sectors with some reluctance to implement techniques that may put at risk their assets.

The second axis is that of knowledge. AI has helped research across the world with the task of inferring relations that were far beyond the human cognitive reach. Every field dealing with huge amounts of reliable data has largely benefited from the adoption of AI and ML techniques. However, we are entering an era in which results and performance metrics are the only interest shown up in research studies. Although for certain disciplines this might be the fair case, science and society are far from being concerned just by performance. The search for understanding is what opens the door for further model improvement and its practical utility.

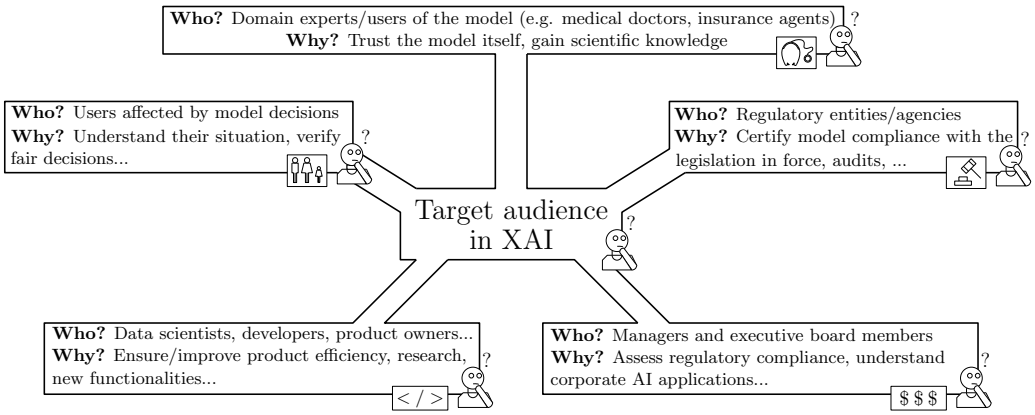


Figure 2.1: Diagram showing the different purposes of explainability in ML models sought by different audience profiles. Two goals occur to prevail across them: need for model understanding, and regulatory compliance. Image partly inspired by the one presented in [30], used with permission from IBM.

The following section develops these ideas further by analyzing the goals motivating the search for explainable AI models.

2.2.4 What for?

The research activity around XAI has so far exposed different goals to draw from the achievement of an explainable model. Almost none of the papers reviewed completely agrees in the goals required to describe what an explainable model should compel. However, all these different goals might help discriminate the purpose for which a given exercise of ML explainability is performed. Unfortunately, scarce contributions have attempted to define such goals from a conceptual perspective [2, 14, 25, 44]. We now synthesize and enumerate definitions for these XAI goals, so as to settle a first classification criteria for the full suit of papers covered in this review:

XAI Goal	**Main target audience** (Fig. 2)	**References**
Trustworthiness	Domain experts, users of the model affected by decisions	[2, 11, 25, 31, 32, 33, 34, 35, 36]
Causality	Domain experts, managers and executive board members, regulatory entities/agencies	[34, 37, 38, 39, 40, 41, 42]
Transferability	Domain experts, data scientists	[2, 7, 22, 27, 43, 44, 31, 36, 37, 38, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84]
Informativeness	All	[2, 7, 22, 26, 27, 43, 44, 31, 33, 34, 36, 37, 40, 45, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 62, 63, 64, 65, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 85, 86, 87, 88, 58, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153]
Confidence	Domain experts, developers, managers, regulatory entities/agencies	[2, 43, 34, 45, 47, 53, 60, 71, 87, 88, 95, 107, 116, 118, 154]
Fairness	Users affected by model decisions, regulatory entities/agencies	[2, 25, 43, 34, 46, 98, 99, 100, 119, 120, 127, 155, 156, 157]
Accessibility	Product owners, managers, users affected by model decisions	[22, 27, 44, 31, 36, 49, 52, 54, 61, 66, 67, 68, 69, 70, 73, 74, 75, 85, 92, 93, 102, 104, 106, 107, 110, 111, 112, 113, 114, 123, 128]
Interactivity	Domain experts, users affected by model decisions	[36, 49, 58, 64, 66, 73, 85, 123]
Privacy awareness	Users affected by model decisions, regulatory entities/agencies	[88]

Table 2.1: Goals pursued in the reviewed literature toward reaching explainability, and their main target audience

.

- *Trustworthiness:* several authors agree upon the search for trustworthiness as the primary aim of an explainable AI model [158, 31]. However, declaring a model as explainable as per its capabilities of inducing trust might not be fully compliant with the requirement of model explainability. Trustworthiness might be considered as the confidence of whether a model will act as intended when facing a given problem. Although it should most certainly be a property of any explainable model, it does not imply that every trustworthy model can be considered explainable on its own, nor is trustworthiness a property easy to quantify. Trust might be far from being the only purpose of an explainable model since the relation among the two, if agreed upon, is not reciprocal. Part of the reviewed papers mention the concept of trust when stating their purpose for achieving explainability. However, as seen in Table 2.1, they do not amount to a large share of the recent contributions related to XAI.

- *Causality:* another common goal for explainability is that of finding causality among data variables. Several authors argue that explainable models might ease the task of finding relationships that, should they occur, could be tested further for a stronger causal link between the involved variables [159, 160]. The inference of causal relationships from observational data is a field that has been broadly studied over time [161]. As widely acknowledged by the community working on this topic, causality requires a wide frame of prior knowledge to prove that observed effects are causal. A ML model only discovers correlations among the data it learns from, and therefore might not suffice for unveiling a cause-effect relationship. However, causation involves correlation, so an explainable ML model could validate the results provided by causality inference techniques, or provide a first intuition of possible causal relationships within the available data. Again, Table 2.1 reveals that causality is not among the most important goals if we attend to the amount of papers that state it explicitly as their goal.

- *Transferability:* models are always bounded by constraints that should allow for their seamless transferability. This is the main reason why a training-testing approach is used when dealing with ML problems [162, 163]. Explainability is also an advocate for transferability, since it may ease the task of elucidating the boundaries that might affect a model, allowing for a better understanding and implementation. Similarly, the mere understanding of the inner relations taking place within a model facilitates the ability of a user to reuse this knowledge in another problem. There are cases in which the lack of a proper understanding of the model might drive the user toward incorrect assumptions and fatal consequences [7, 164]. Transferability should also fall between the resulting properties of an explainable model, but again, not every transferable model should be considered as explainable. As observed in Table 2.1, the amount of papers stating that the ability of rendering a model explainable is to better understand the concepts needed to

reuse it or to improve its performance is the second most used reason for pursuing model explainability.

- *Informativeness:* ML models are used with the ultimate intention of supporting decision making [91]. However, it should not be forgotten that the problem being solved by the model is not equal to that being faced by its human counterpart. Hence, a great deal of information is needed in order to be able to relate the user's decision to the solution given by the model, and to avoid falling in misconception pitfalls. For this purpose, explainable ML models should give information about the problem being tackled. Most of the reasons found among the papers reviewed is that of extracting information about the inner relations of a model. Almost all rule extraction techniques substantiate their approach on the search for a simpler understanding of what the model internally does, stating that the knowledge (information) can be expressed in these simpler proxies that they consider explaining the antecedent. This is the most used argument found among the reviewed papers to back up what they expect from reaching explainable models.

- *Confidence:* as a generalization of robustness and stability, confidence should always be assessed on a model in which reliability is expected. The methods to maintain confidence under control are different depending on the model. As stated in [165, 166, 167], stability is a must-have when drawing interpretations from a certain model. Trustworthy interpretations should not be produced by models that are not stable. Hence, an explainable model should contain information about the confidence of its working regime.

- *Fairness:* from a social standpoint, explainability can be considered as the capacity to reach and guarantee fairness in ML models. In a certain literature strand, an explainable ML model suggests a clear visualization of the relations affecting a result, allowing for a fairness or ethical analysis of the model at hand [10, 99]. Likewise, a related objective of XAI is highlighting bias in the data a model was exposed to [168, 169]. The support of algorithms and models is growing fast in fields that involve human lives, hence explainability should be considered as a bridge to avoid the unfair or unethical use of algorithm's outputs.

- *Accessibility:* a minor subset of the reviewed contributions argues for explainability as the property that allows end users to get more involved in the process of improving and developing a certain ML model [36, 85] . It seems clear that explainable models will ease the burden felt by non-technical or non-expert users when having to deal with algorithms that seem incomprehensible at first sight. This concept is expressed as the third most considered goal among the surveyed literature.

- *Interactivity:* some contributions [49, 58] include the ability of a model to be interactive with the user as one of the goals targeted by an explainable ML model. Once again, this goal is related to fields in which the end users are of great importance, and their ability to tweak and interact with the models is what ensures success.

- *Privacy awareness:* almost forgotten in the reviewed literature, one of the byproducts enabled by explainability in ML models is its ability to assess privacy. ML models may have complex representations of their learned patterns. Not being able to understand what has been captured by the model [1] and stored in its internal representation may entail a privacy breach. Contrarily, the ability to explain the inner relations of a trained model by non-authorized third parties may also compromise the differential privacy of the data origin. Due to its criticality in sectors where XAI is foreseen to play a crucial role, confidentiality and privacy issues will be covered further in Subsection 2.5.4

This subsection has reviewed the goals encountered among the broad scope of the reviewed papers. All these goals are clearly under the surface of the concept of explainability introduced before in this section. To round up this prior analysis on the concept of explainability, the last subsection deals with different strategies followed by the community to address explainability in ML models.

2.2.5 How?

The literature makes a clear distinction among models that are interpretable by design, and those that can be explained by means of external XAI techniques. This duality could also be regarded as the difference between interpretable models and model interpretability techniques; a more widely accepted classification is that of *transparent* models and post-hoc explainability. This same duality also appears in the paper presented in [18] in which the distinction its authors make refers to the methods to solve the transparent box design problem against the problem of explaining the black-box problem. This work, further extends the distinction made among transparent models including the different levels of transparency considered.

Within transparency, three levels are contemplated: algorithmic transparency, decomposability and simulatability[1]. Among post-hoc techniques we may distinguish among *text explanations*, *visualizations*, *local explanations*, *explanations by example*, *explanations by simplification* and *feature relevance*. In this context, there is a broader distinction proposed by [25] discerning between 1) opaque systems, where the mappings from input to output are invisible to the user; 2) interpretable systems, in which users can mathematically analyze

[1]The alternative term *simulability* is also used in the literature to refer to the capacity of a system or process to be simulated. However, we note that this term does not appear in current English dictionaries.

the mappings; and 3) comprehensible systems, in which the models should output symbols or rules along with their specific output to aid in the understanding process of the rationale behind the mappings being made. This last classification criterion could be considered included within the one proposed earlier, hence this paper will attempt at following the more specific one.

Levels of Transparency in Machine Learning Models

Transparent models convey some degree of interpretability by themselves. Models belonging to this category can be also approached in terms of the domain in which they are interpretable, namely, algorithmic transparency, decomposability and simulatability. As we elaborate next in connection to Figure 2.2, each of these classes contains its predecessors, e.g. a *simulatable* model is at the same time a model that is decomposable and algorithmically transparent:

- *Simulatability* denotes the ability of a model of being simulated or thought about strictly by a human, hence complexity takes a dominant place in this class. This being said, simple but extensive (i.e., with *too large* amount of rules) rule based systems fall out of this characteristic, whereas a single perceptron neural network falls within. This aspect aligns with the claim that sparse linear models are more interpretable than dense ones [170], and that an interpretable model is one that can be easily presented to a human by means of text and *visualizations* [31]. Again, endowing a decomposable model with simulatability requires that the model has to be self-contained enough for a human to think and reason about it as a whole.

- *Decomposability* stands for the ability to explain each of the parts of a model (input, parameter and calculation). It can be considered as intelligibility as stated in [171]. This characteristic might empower the ability to understand, interpret or explain the behavior of a model. However, as occurs with algorithmic transparency, not every model can fulfill this property. Decomposability requires every input to be readily interpretable (e.g. cumbersome features will not fit the premise). The added constraint for an algorithmically transparent model to become decomposable is that every part of the model must be understandable by a human without the need for additional tools.

- *Algorithmic Transparency* can be seen in different ways. It deals with the ability of the user to understand the process followed by the model to produce any given output from its input data. Put it differently, a linear model is deemed transparent because its error surface can be understood and reasoned about, allowing the user to understand how the model will act in every situation it may face [163]. Contrarily, it is not possible to understand it in deep architectures as the loss landscape might be opaque [172, 173] since it cannot be

fully observed and the solution has to be approximated through heuristic optimization (e.g. through stochastic gradient descent). The main constraint for algorithmically transparent models is that the model has to be fully searchable by means of mathematical analysis and methods.

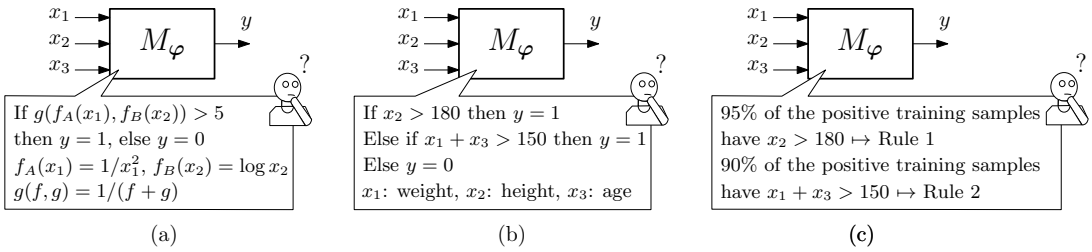


Figure 2.2: Conceptual diagram exemplifying the different levels of transparency characterizing a ML model M_φ, with φ denoting the parameter set of the model at hand: (a) simulatability; (b) decomposability; (c) algorithmic transparency. Without loss of generality, the example focuses on the ML model as the explanation target. However, other targets for explainability may include a given example, the output classes or the dataset itself.

Post-hoc Explainability Techniques for Machine Learning Models

Post-hoc explainability targets models that are not readily interpretable by design by resorting to diverse means to enhance their interpretability, such as *text explanations*, *visual explanations*, *local explanations*, *explanations by example*, *explanations by simplification* and *feature relevance explanations* techniques. Each of these techniques covers one of the most common ways humans explain systems and processes by themselves.

Further along this river, actual techniques, or better put, actual group of techniques are specified to ease the future work of any researcher that intends to look up for an specific technique that suits its knowledge. Not ending there, the classification also includes the type of data in which the techniques has been applied. Note that many techniques might be suitable for many different types of data, although the categorization only considers the type used by the authors that proposed such technique. Overall, post-hoc explainability techniques are divided first by the intention of the author (explanation technique e.g. Explanation by simplification), then, by the method utilized (actual technique e.g. sensitivity analysis) and finally by the type of data in which it was applied (e.g. images).

- *Text explanations* deal with the problem of bringing explainability for a model by means of learning to generate *text explanations* that help explaining the results from the model

[169]. *Text explanations* also include every method generating symbols that represent the functioning of the model. These symbols may portrait the rationale of the algorithm by means of a semantic mapping from model to symbols.

- *Visual explanation* techniques for post-hoc explainability aim at visualizing the model's behavior. Many of the visualization methods existing in the literature come along with dimensionality reduction techniques that allow for a human interpretable simple visualization. Visualizations may be coupled with other techniques to improve their understanding, and are considered as the most suitable way to introduce complex interactions within the variables involved in the model to users not acquainted to ML modeling.

- *Local explanations* tackle explainability by segmenting the solution space and giving explanations to less complex solution subspaces that are relevant for the whole model. These explanations can be formed by means of techniques with the differentiating property that these only explain part of the whole system's functioning.

- *Explanations by example* consider the extraction of data examples that relate to the result generated by a certain model, enabling to get a better understanding of the model itself. Similarly to how humans behave when attempting to explain a given process, *explanations by example* are mainly centered in extracting representative examples that grasp the inner relationships and correlations found by the model being analyzed.

- *Explanations by simplification* collectively denote those techniques in which a whole new system is rebuilt based on the trained model to be explained. This new, simplified model usually attempts at optimizing its resemblance to its antecedent functioning, while reducing its complexity, and keeping a similar performance score. An interesting byproduct of this family of post-hoc techniques is that the simplified model is, in general, easier to be implemented due to its reduced complexity with respect to the model it represents.

- Finally, *feature relevance explanation* methods for post-hoc explainability clarify the inner functioning of a model by computing a relevance score for its managed variables. These scores quantify the affection (sensitivity) a feature has upon the output of the model. A comparison of the scores among different variables unveils the importance granted by the model to each of such variables when producing its output. *Feature relevance* methods can be thought to be an indirect method to explain a model.

The above classification (portrayed graphically in Figure 2.3) will be used when reviewing specific/agnostic XAI techniques for ML models in the following sections (Table 2.2). For each ML model, a distinction of the propositions to each of these categories is presented in order to pose an overall image of the field's trends.

Model	Transparent ML Models			Post-hoc analysis
	Simulatability	**Decomposability**	**Algorithmic Transparency**	
Linear/Logistic Regression	Predictors are human readable and interactions among them are kept to a minimum	Variables are still readable, but the number of interactions and predictors involved in them have grown to force decomposition	Variables and interactions are too complex to be analyzed without mathematical tools	Not needed
Decision Trees	A human can simulate and obtain the prediction of a decision tree on his/her own, without requiring any mathematical background	The model comprises rules that do not alter data whatsoever, and preserves their readability	Human-readable rules that explain the knowledge learned from data and allows for a direct understanding of the prediction process	Not needed
K-Nearest Neighbors	The complexity of the model (number of variables, their understandability and the similarity measure under use) matches human naive capabilities for simulation	The amount of variables is too high and/or the similarity measure is too complex to be able to simulate the model completely, but the similarity measure and the set of variables can be decomposed and analyzed separately	The similarity measure cannot be decomposed and/or the number of variables is so high that the user has to rely on mathematical and statistical tools to analyze the model	Not needed
Rule Based Learners	Variables included in rules are readable, and the size of the rule set is manageable by a human user without external help	The size of the rule set becomes too large to be analyzed without decomposing it into small rule chunks	Rules have become so complicated (and the rule set size has grown so much) that mathematical tools are needed for inspecting the model behaviour	Not needed
General Additive Models	Variables and the interaction among them as per the smooth functions involved in the model must be constrained within human capabilities for understanding	Interactions become too complex to be simulated, so decomposition techniques are required for analyzing the model	Due to their complexity, variables and interactions cannot be analyzed without the application of mathematical and statistical tools	Not needed
Bayesian Models	Statistical relationships modeled among variables and the variables themselves should be directly understandable by the target audience	Statistical relationships involve so many variables that they must be decomposed in marginals so as to ease their analysis	Statistical relationships cannot be interpreted even if already decomposed, and predictors are so complex that model can be only analyzed with mathematical tools	Not needed
Tree Ensembles	✗	✗	✗	Needed: Usually *Model simplification* or *Feature relevance* techniques
Support Vector Machines	✗	✗	✗	Needed: Usually *Model simplification* or *Local explanations* techniques
Multi–layer Neural Network	✗	✗	✗	Needed: Usually *Model simplification*, *Feature relevance* or *Visualization* techniques
Convolutional Neural Network	✗	✗	✗	Needed: Usually *Feature relevance* or *Visualization* techniques
Recurrent Neural Network	✗	✗	✗	Needed: Usually *Feature relevance* techniques

Table 2.2: Overall picture of the classification of ML models attending to their level of explainability.

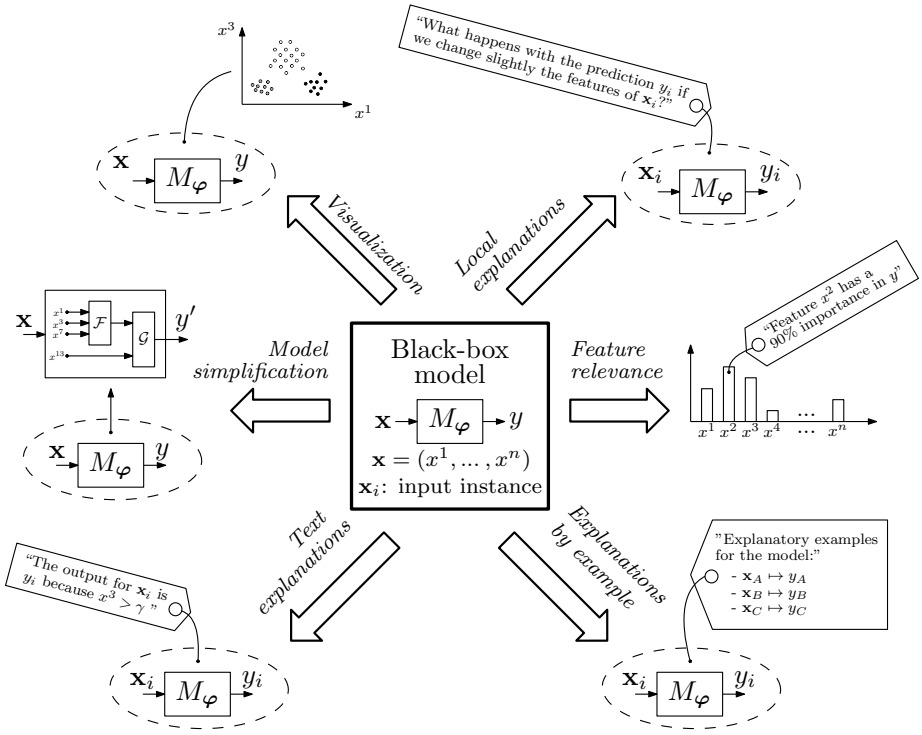


Figure 2.3: Conceptual diagram showing the different post-hoc explainability approaches available for a ML model M_φ.

2.3 Transparent Machine Learning Models

The previous section introduced the concept of *transparent* models. A model is considered to be transparent if by itself it is understandable. The models surveyed in this section are a suit of transparent models that can fall in one or all of the levels of model transparency described previously (namely, simulatability, decomposability and algorithmic transparency). In what follows we provide reasons for this statement, with graphical support given in Figure 2.4.

2.3.1 Linear/Logistic Regression

Logistic Regression (LR) is a classification model to predict a dependent variable (category) that is dichotomous (binary). However, when the dependent variable is continuous, linear regression would be its homonym. This model takes the assumption of linear dependence between the predictors and the predicted variables, impeding a flexible fit to the data.

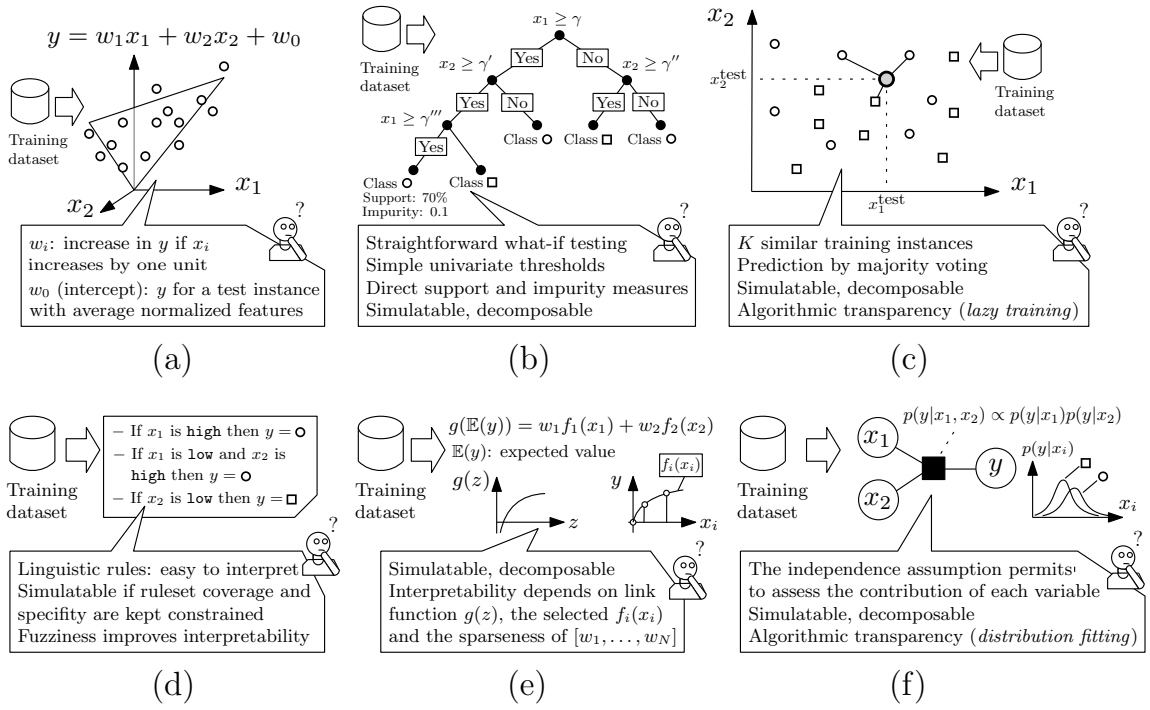


Figure 2.4: Graphical illustration of the levels of transparency of different ML models considered in this overview: (a) Linear regression; (b) Decision trees; (c) K-Nearest Neighbors; (d) Rule-based Learners; (e) Generalized Additive Models; (f) Bayesian Models.

This specific reason (stiffness of the model) is the one that maintains the model under the umbrella of transparent methods. However, as stated in Section 2, explainability is linked to a certain audience, which makes a model fall under both categories depending who is to interpret it. This way, logistic and linear regression, although clearly meeting the characteristics of transparent models (algorithmic transparency, decomposability and simulatability), may also demand post-hoc explainability techniques (mainly, visualization), particularly when the model is to be explained to non-expert audiences.

The usage of this model has been largely applied within Social Sciences for quite a long time, which has pushed researchers to create ways of explaining the results of the models to non-expert users. Most authors agree on the different techniques used to analyze and express the soundness of LR [174, 175, 176, 177], including the overall model evaluation, statistical tests of individual predictors, goodness-of-fit statistics and validation of the predicted probabilities. The overall model evaluation shows the improvement of the applied model over a baseline, showing if it is in fact improving the model without predictions. The statistical significance of single predictors is shown by calculating the Wald chi-square

statistic. The goodness-of-fit statistics show the quality of fitness of the model to the data and how significant this is. This can be achieved by resorting to different techniques e.g. the so-called Hosmer-Lemeshow (H-L) statistic. The validation of predicted probabilities involves testing whether the output of the model corresponds to what is shown by the data. These techniques show mathematical ways of representing the fitness of the model and its behavior.

Other techniques from other disciplines besides Statistics can be adopted for explaining these regression models. Visualization techniques are very powerful when presenting statistical conclusions to users not well-versed in statistics. For instance, the work in [178] shows that the usage of probabilities to communicate the results, implied that the users where able to estimate the outcomes correctly in 10% of the cases, as opposed to 46% of the cases when using natural frequencies. Although logistic regression is among the simplest classification models in supervised learning, there are concepts that must be taken care of.

In this line of reasoning, the authors of [179] unveil some concerns with the interpretations derived from LR. They first mention how dangerous it might be to interpret log odds ratios and odd ratios as substantive effects, since they also represent unobserved heterogeneity. Linked to this first concern, [179] also states that a comparison between these ratios across models with different variables might be problematic, since the unobserved heterogeneity is likely to vary, thereby invalidating the comparison. Finally they also mention that the comparison of these odds across different samples, groups and time is also risky, since the variation of the heterogeneity is not known across samples, groups and time points. This last paper serves the purpose of visualizing the problems a model's interpretation might entail, even when its construction is as simple as that of LR.

Also interesting is to note that, for a model such as logistic or linear regression to maintain decomposability and simulatability, its size must be limited, and the variables used must be understandable by their users. As stated in Section 2, if inputs to the model are highly engineered features that are complex or difficult to understand, the model at hand will be far from being *decomposable*. Similarly, if the model is so large that a human cannot think of the model as a whole, its simulatability will be put to question.

2.3.2 Decision Trees

Decision trees are another example of a model that can easily fulfill every constraint for transparency. Decision trees are hierarchical structures for decision making used to support regression and classification problems [131, 180]. In the simplest of their flavors, decision trees are *simulatable* models. However, their properties can render them *decomposable* or *algorithmically transparent*.

Decision trees have always lingered in between the different categories of transparent

models. Their utilization has been closely linked to decision making contexts, being the reason why their complexity and understandability have always been considered a paramount matter. A proof of this relevance can be found in the upsurge of contributions to the literature dealing with decision tree simplification and generation [131, 180, 181, 182]. As noted above, although being capable of fitting every category within transparent models, the individual characteristics of decision trees can push them toward the category of algorithmically transparent models. A *simulatable* decision tree is one that is manageable by a human user. This means its size is somewhat small and the amount of features and their meaning are easily understandable. An increment in size transforms the model into a *decomposable* one since its size impedes its full evaluation (simulation) by a human. Finally, further increasing its size and using complex feature relations will make the model *algorithmically transparent* loosing the previous characteristics.

Decision trees have long been used in decision support contexts due to their off-the-shelf transparency. Many applications of these models fall out of the fields of computation and AI (even information technologies), meaning that experts from other fields usually feel comfortable interpreting the outputs of these models [183, 184, 185]. However, their poor generalization properties in comparison with other models make this model family less interesting for their application to scenarios where a balance between predictive performance is a design driver of utmost importance. Tree ensembles aim at overcoming such a poor performance by aggregating the predictions performed by trees learned on different subsets of training data. Unfortunately, the combination of decision trees looses every transparent property, calling for the adoption of post-hoc explainability techniques as the ones reviewed later in the manuscript.

2.3.3 K-Nearest Neighbors

Another method that falls within transparent models is that of K-Nearest Neighbors (KNN), which deals with classification problems in a methodologically simple way: it predicts the class of a test sample by voting the classes of its K nearest neighbors (where the neighborhood relation is induced by a measure of distance between samples). When used in the context of regression problems, the voting is replaced by an aggregation (e.g. average) of the target values associated with the nearest neighbors.

In terms of model explainability, it is important to observe that predictions generated by KNN models rely on the notion of distance and similarity between examples, which can be tailored depending on the specific problem being tackled. Interestingly, this prediction approach resembles that of experience-based human decision making, which decides upon the result of past similar cases. There lies the rationale of why KNN has also been adopted widely in contexts in which model interpretability is a requirement [186, 187, 188, 189].

Furthermore, aside from being simple to explain, the ability to inspect the reasons by which a new sample has been classified inside a group and to examine how these predictions evolve when the number of neighbors K is increased or decreased empowers the interaction between the users and the model.

One must keep in mind that as mentioned before, KNN's class of transparency depends on the features, the number of neighbors and the distance function used to measure the similarity between data instances. A very high K impedes a full simulation of the model performance by a human user. Similarly, the usage of complex features and/or distance functions would hinder the decomposability of the model, restricting its interpretability solely to the transparency of its algorithmic operations.

2.3.4 Rule-based Learning

Rule-based learning refers to every model that generates rules to characterize the data it is intended to learn from. Rules can take the form of simple conditional *if-then* rules or more complex combinations of simple rules to form their knowledge. Also connected to this general family of models, fuzzy rule based systems are designed for a broader scope of action, allowing for the definition of verbally formulated rules over imprecise domains. Fuzzy systems improve two main axis relevant for this paper. First, they empower more understandable models since they operate in linguistic terms. Second, they perform better that classic rule systems in contexts with certain degrees of uncertainty. Rule based learners are clearly transparent models that have been often used to explain complex models by generating rules that explain their predictions [125, 126, 190, 191].

Rule learning approaches have been extensively used for knowledge representation in expert systems [192]. However, a central problem with rule generation approaches is the coverage (amount) and the specificity (length) of the rules generated. This problem relates directly to the intention for their use in the first place. When building a rule database, a typical design goal sought by the user is to be able to analyze and understand the model. The amount of rules in a model will clearly improve the performance of the model at the stake of compromising its intepretability. Similarly, the specificity of the rules plays also against interpretability, since a rule with a high number of antecedents an/or consequences might become difficult to interpret. In this same line of reasoning, these two features of a rule based learner play along with the classes of transparent models presented in Section 2. The greater the coverage or the specificity is, the closer the model will be to being just *algorithmically transparent*. Sometimes, the reason to transition from classical rules to fuzzy rules is to relax the constraints of rule sizes, since a greater range can be covered with less stress on interpretability.

Rule based learners are great models in terms of interpretability across fields. Their

natural and seamless relation to human behaviour makes them very suitable to understand and explain other models. If a certain threshold of coverage is acquired, a rule wrapper can be thought to contain enough information about a model to explain its behavior to a non-expert user, without forfeiting the possibility of using the generated rules as an standalone prediction model.

2.3.5 General Additive Models

In statistics, a Generalized Additive Model (GAM) is a linear model in which the value of the variable to be predicted is given by the aggregation of a number of unknown smooth functions defined for the predictor variables. The purpose of such model is to infer the smooth functions whose aggregate composition approximates the predicted variable. This structure is easily interpretable, since it allows the user to verify the importance of each variable, namely, how it affects (through its corresponding function) the predicted output.

Similarly to every other transparent model, the literature is replete with case studies where GAMs are in use, specially in fields related to risk assessment. When compared to other models, these are understandable enough to make users feel confident on using them for practical applications in finance [193, 194, 195], environmental studies [196], geology [197], healthcare [7], biology [198, 199] and energy [200]. Most of these contributions use visualization methods to further ease the interpretation of the model. GAMs might be also considered as *simulatable* and *decomposable* models if the properties mentioned in its definitions are fulfilled, but to an extent that depends roughly on eventual modifications to the baseline GAM model, such as the introduction of link functions to relate the aggregation with the predicted output, or the consideration of interactions between predictors.

All in all, applications of GAMs like the ones exemplified above share one common factor: understandability. The main driver for conducting these studies with GAMs is to understand the underlying relationships that build up the cases for scrutiny. In those cases the research goal is not accuracy for its own sake, but rather the need for understanding the problem behind and the relationship underneath the variables involved in data. This is why GAMs have been accepted in certain communities as their *de facto* modeling choice, despite their acknowledged misperforming behavior when compared to more complex counterparts.

2.3.6 Bayesian Models

A Bayesian model usually takes the form of a probabilistic directed acyclic graphical model whose links represent the conditional dependencies between a set of variables. For example, a Bayesian network could represent the probabilistic relationships between diseases and

symptoms. Given symptoms, the network can be used to compute the probabilities of the presence of various diseases. Similar to GAMs, these models also convey a clear representation of the relationships between features and the target, which in this case are given explicitly by the connections linking variables to each other.

Once again, Bayesian models fall below the ceiling of Transparent models. Its categorization leaves it under *simulatable*, *decomposable* and *algorithmically transparent*. However, it is worth noting that under certain circumstances (overly complex or cumbersome variables), a model may loose these first two properties. Bayesian models have been shown to lead to great insights in assorted applications such as cognitive modeling [201, 202], fishery [196, 203], gaming [204], climate [205], econometrics [206] or robotics [207]. Furthermore, they have also been utilized to explain other models, such as averaging tree ensembles [208].

2.4 Post-hoc Explainability Techniques for Machine Learning Models

When ML models do not meet any of the criteria imposed to declare them transparent, a separate method must be devised and applied to the model to explain its decisions. This is the purpose of post-hoc explainability techniques (also referred to as post-modeling explainability), which aim at communicating understandable information about how an already developed model produces its predictions for any given input. In this section we categorize and review different algorithmic approaches for post-hoc explainability, discriminating among 1) those that are designed for their application to ML models of any kind; and 2) those that are designed for a specific ML model and thus, can not be directly extrapolated to any other learner. We now elaborate on the trends identified around post-hoc explainability for different ML models, which are illustrated in Figure 2.5 in the form of hierarchical bibliographic categories and summarized next:

- Model-agnostic techniques for post-hoc explainability (Subsection 2.4.1), which can be applied seamlessly to any ML model disregarding its inner processing or internal representations.

- Post-hoc explainability that are tailored or specifically designed to explain certain ML models. We divide our literature analysis into two main branches: contributions dealing with post-hoc explainability of *shallow* ML models, which collectively refers to all ML models that do not hinge on layered structures of neural processing units (Subsection 2.4.2); and techniques devised for *deep* learning models, which correspondingly denote the family of neural networks and related variants, such as convolutional neural networks,

recurrent neural networks (Subsection 2.4.3) and hybrid schemes encompassing deep neural networks and transparent models. For each model we perform a thorough review of the latest post-hoc methods proposed by the research community, along with a identification of trends followed by such contributions.

- We end our literature analysis with Subsection 2.4.4, where we present a second taxonomy that complements the more general one in Figure 2.5 by classifying contributions dealing with the post-hoc explanation of Deep Learning models. To this end we focus on particular aspects related to this family of black-box ML methods, and expose how they link to the classification criteria used in the first taxonomy.

2.4.1 Model-agnostic Techniques for Post-hoc Explainability

Model-agnostic techniques for post-hoc explainability are designed to be plugged to any model with the intent of extracting some information from its prediction procedure. Sometimes, simplification techniques are used to generate proxies that mimic their antecedents with the purpose of having something tractable and of reduced complexity. Other times, the intent focuses on extracting knowledge directly from the models or simply visualizing them to ease the interpretation of their behavior. Following the taxonomy introduced in Section 2, model-agnostic techniques may rely on *model simplification*, *feature relevance* estimation and *visualization* techniques:

- *Explanation by simplification*. They are arguably the broadest technique under the category of model agnostic post-hoc methods. *Local explanations* are also present within this category, since sometimes, simplified models are only representative of certain sections of a model. Almost all techniques taking this path for *model simplification* are based on rule extraction techniques. Among the most known contributions to this approach we encounter the technique of Local Interpretable Model-Agnostic Explanations (LIME) [31] and all its variations [214, 216]. LIME builds locally linear models around the predictions of an opaque model to explain it. These contributions fall under explanations by simplification as well as under *local explanations*. Besides LIME and related flavors, another approach to rule extraction is G-REX [212]. Although it was not originally intended for extracting rules from opaque models, the generic proposition of G-REX has been extended to also account for model explainability purposes [190, 211]. In line with rule extraction methods, the work in [215] presents a novel approach to learn rules in CNF (Conjunctive Normal Form) or DNF (Disjunctive Normal Form) to bridge from a complex model to a human-interpretable model. Another contribution that falls off the same branch is that in [218], where the authors formulate *model simplification* as

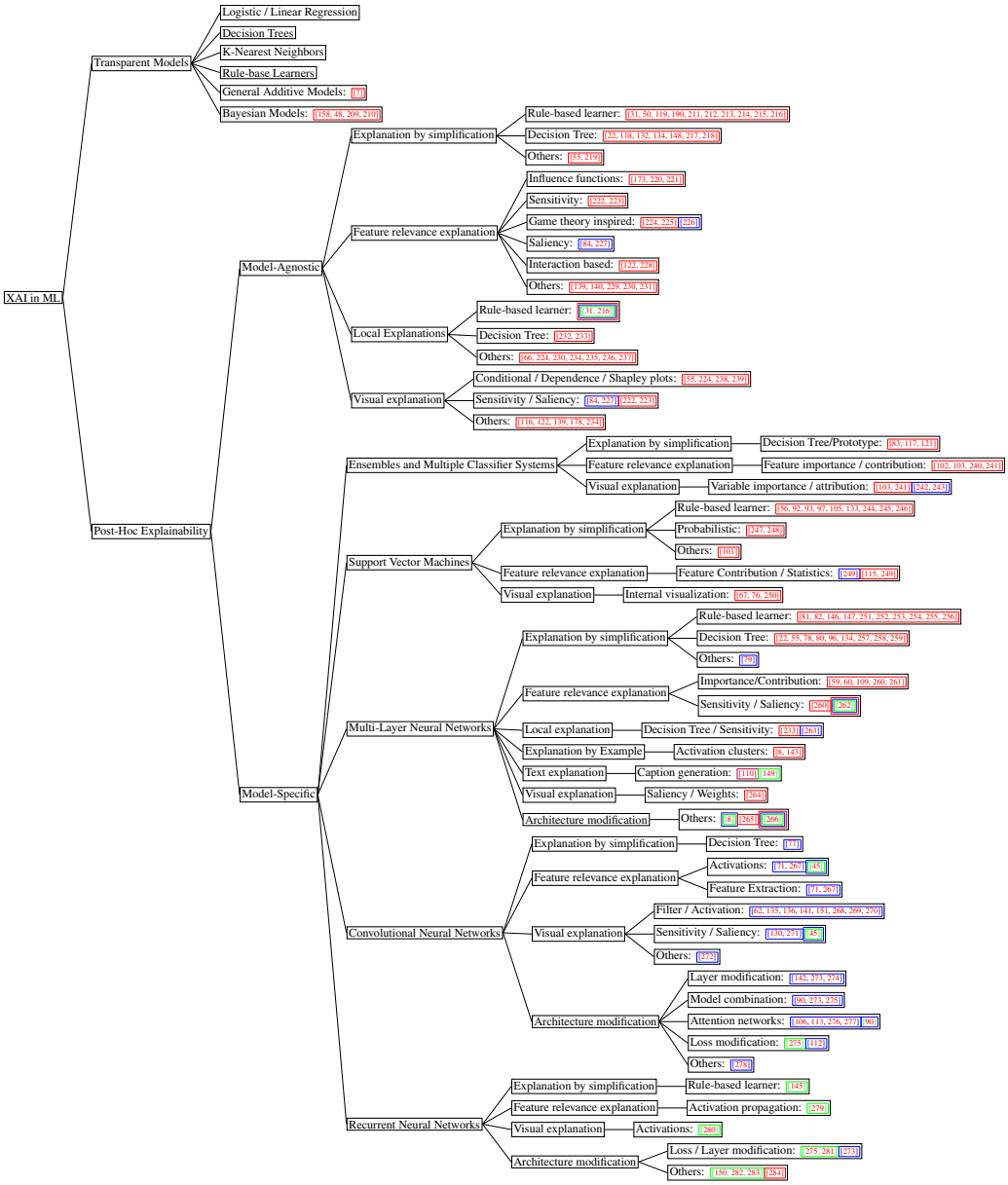


Figure 2.5: Taxonomy of the reviewed literature and trends identified for explainability techniques related to different ML models. References boxed in blue, green and red correspond to XAI techniques using image, text or tabular data, respectively. In order to build this taxonomy, the literature has been analyzed in depth to discriminate whether a post-hoc technique can be seamlessly applied to any ML model, even if, e.g., explicitly mentions *Deep Learning* in its title and/or abstract.

a model extraction process by approximating a transparent model to the complex one. Simplification is approached from a different perspective in [119], where an approach to distill and audit black box models is presented. In it, two main ideas are exposed: a method for model distillation and comparison to audit black-box risk scoring models; and an statistical test to check if the auditing data is missing key features it was trained with. The popularity of *model simplification* is evident, given it temporally coincides with the most recent literature on XAI, including techniques such as LIME or G-REX. This symptomatically reveals that this post-hoc explainability approach is envisaged to continue playing a central role on XAI.

- *Feature relevance explanation* techniques aim to describe the functioning of an opaque model by ranking or measuring the influence, relevance or importance each feature has in the prediction output by the model to be explained. An amalgam of propositions are found within this category, each resorting to different algorithmic approaches with the same targeted goal. One fruitful contribution to this path is that of [224] called SHAP (SHapley Additive exPlanations). Its authors presented a method to calculate an additive feature importance score for each particular prediction with a set of desirable properties (local accuracy, *missingness* and consistency) that its antecedents lacked. Another approach to tackle the contribution of each feature to predictions has been coalitional Game Theory [225] and local gradients [234]. Similarly, by means of local gradients [230] test the changes needed in each feature to produce a change in the output of the model. In [228] the authors analyze the relations and dependencies found in the model by grouping features, that combined, bring insights about the data. The work in [173] presents a broad variety of measures to tackle the quantification of the degree of influence of inputs on outputs of systems. Their QII (Quantitative Input Influence) measures account for correlated inputs while measuring influence. In contrast, in [222] the authors build upon the existing SA (Sensitivity Analysis) to construct a Global SA which extends the applicability of the existing methods. In [227] a real-time image saliency method is proposed, which is applicable to differentiable image classifiers. The study in [122] presents the so-called Automatic STRucture IDentification method (ASTRID) to inspect which attributes are exploited by a classifier to generate a prediction. This method finds the largest subset of features such that the accuracy of a classifier trained with this subset of features cannot be distinguished in terms of accuracy from a classifier built on the original feature set. In [221] the authors use influence functions to trace a model's prediction back to the training data, by only requiring an oracle version of the model with access to gradients and Hessian-vector products. Heuristics for creating counterfactual examples by modifying the input of the model have been also found to contribute to its explainability [236, 237]. Compared to those attempting explanations by simplification,

a similar amount of publications were found tackling explainability by means of *feature relevance* techniques. Many of the contributions date from 2017 and some from 2018, implying that as with *model simplification* techniques, *feature relevance* has also become a vibrant subject study in the current XAI landscape.

- *Visual explanation* techniques are a vehicle to achieve model-agnostic explanations. Representative works in this area can be found in [222], which present a portfolio of visualization techniques to help in the explanation of a black-box ML model built upon the set of extended techniques mentioned earlier (Global SA). Another set of visualization techniques is presented in [223]. The authors present three novel SA methods (data based SA, Monte-Carlo SA, cluster-based SA) and one novel input importance measure (Average Absolute Deviation). Finally, [238] presents ICE (Individual Conditional Expectation) plots as a tool for visualizing the model estimated by any supervised learning algorithm. Visual explanations are less common in the field of model-agnostic techniques for post-hoc explainability. Since the design of these methods must ensure that they can be seamlessly applied to any ML model disregarding its inner structure, creating *visualizations* from just inputs and outputs from an opaque model is a complex task. This is why almost all visualization methods falling in this category work along with *feature relevance* techniques, which provide the information that is eventually displayed to the end user.

Several trends emerge from our literature analysis. To begin with, rule extraction techniques prevail in model-agnostic contributions under the umbrella of post-hoc explainability. This could have been intuitively expected if we bear in mind the wide use of rule based learning as explainability wrappers anticipated in Section 2.3.4, and the complexity imposed by not being able to *get into* the model itself. Similarly, another large group of contributions deals with *feature relevance*. Lately these techniques are gathering much attention by the community when dealing with DL models, with hybrid approaches that utilize particular aspects of this class of models and therefore, compromise the independence of the *feature relevance* method on the model being explained. Finally, visualization techniques propose interesting ways for visualizing the output of *feature relevance* techniques to ease the task of model's interpretation. By contrast, visualization techniques for other aspects of the trained model (e.g. its structure, operations, etc) are tightly linked to the specific model to be explained.

2.4.2 Post-hoc Explainability in Shallow ML Models

Shallow ML covers a diversity of supervised learning models. Within these models, there are strictly interpretable (transparent) approaches (e.g. KNN and Decision Trees, already

discussed in Section 2.3). However, other shallow ML models rely on more sophisticated learning algorithms that require additional layers of explanation. Given their prominence and notable performance in predictive tasks, this section concentrates on two popular shallow ML models (tree ensembles and Support Vector Machines, SVMs) that require the adoption of post-hoc explainability techniques for explaining their decisions.

Tree Ensembles, Random Forests and Multiple Classifier Systems

Tree ensembles are arguably among the most accurate ML models in use nowadays. Their advent came as an efficient means to improve the generalization capability of single decision trees, which are usually prone to overfitting. To circumvent this issue, tree ensembles combine different trees to obtain an aggregated prediction/regression. While it results to be effective against overfitting, the combination of models makes the interpretation of the overall ensemble more complex than each of its compounding tree learners, forcing the user to draw from post-hoc explainability techniques. For tree ensembles, techniques found in the literature are explanation by simplification and *feature relevance* techniques; we next examine recent advances in these techniques.

To begin with, many contributions have been presented to simplify tree ensembles while maintaining part of the accuracy accounted for the added complexity. The author from [118] poses the idea of training a single albeit less complex model from a set of random samples from the data (ideally following the real data distribution) labeled by the ensemble model. Another approach for simplification is that in [117], in which authors create a Simplified Tree Ensemble Learner (STEL). Likewise, [121] presents the usage of two models (simple and complex) being the former the one in charge of interpretation and the latter of prediction by means of Expectation-Maximization and Kullback-Leibler divergence. As opposed to what was seen in model-agnostic techniques, not that many techniques to board explainability in tree ensembles by means of *model simplification*. It derives from this that either the proposed techniques are good enough, or model-agnostic techniques do cover the scope of simplification already.

Following simplification procedures, *feature relevance* techniques are also used in the field of tree ensembles. Breiman [285] was the first to analyze the variable importance within Random Forests. His method is based on measuring MDA (Mean Decrease Accuracy) or MIE (Mean Increase Error) of the forest when a certain variable is randomly permuted in the out-of-bag samples. Following this contribution [241] shows, in an real setting, how the usage of variable importance reflects the underlying relationships of a complex system modeled by a Random Forest. Finally, a crosswise technique among post-hoc explainability, [240] proposes a framework that poses recommendations that, if taken, would convert an example from one class to another. This idea attempts to disentangle the

variables importance in a way that is further descriptive. In the article, the authors show how these methods can be used to elevate recommendations to improve malicious online ads to make them rank higher in paying rates.

Similar to the trend shown in model-agnostic techniques, for tree ensembles again, simplification and *feature relevance* techniques seem to be the most used schemes. However, contrarily to what was observed before, most papers date back from 2017 and place their focus mostly on bagging ensembles. When shifting the focus towards other ensemble strategies, scarce activity has been recently noted around the explainability of boosting and stacking classifiers. Among the latter, it is worth highlighting the connection between the reason why a compounding learner of the ensemble produces an specific prediction on a given data, and its contribution to the output of the ensemble. The so-called Stacking With Auxiliary Features (SWAF) approach proposed in [242] points in this direction by harnessing and integrating explanations in stacking ensembles to improve their generalization. This strategy allows not only relying on the output of the compounding learners, but also on the origin of that output and its consensus across the entire ensemble. Other interesting studies on the explainability of ensemble techniques include model-agnostic schemes such as DeepSHAP [226], put into practice with stacking ensembles and multiple classifier systems in addition to Deep Learning models; the combination of explanation maps of multiple classifiers to produce improved explanations of the ensemble to which they belong [243]; and recent insights dealing with traditional and gradient boosting ensembles [286, 287].

Support Vector Machines

Another shallow ML model with historical presence in the literature is the SVM. SVM models are more complex than tree ensembles, with a much opaquer structure. Many implementations of post-hoc explainability techniques have been proposed to relate what is mathematically described internally in these models, to what different authors considered explanations about the problem at hand. Technically, an SVM constructs a hyper-plane or set of hyper-planes in a high or infinite-dimensional space, which can be used for classification, regression, or other tasks such as outlier detection. Intuitively, a good separation is achieved by the hyperplane that has the largest distance (so-called functional margin) to the nearest training-data point of any class, since in general, the larger the margin, the lower the generalization error of the classifier. SVMs are among the most used ML models due to their excellent prediction and generalization capabilities. From the techniques stated in Section 2, post-hoc explainability applied to SVMs covers explanation by *simplification*, *local explanations*, *visualizations* and *explanations by example*.

Among explanation by simplification, four classes of simplifications are made. Each of them differentiates from the other by how deep they go into the algorithm inner structure.

First, some authors propose techniques to build rule based models only from the support vectors of a trained model. This is the approach of [92], which proposes a method that extracts rules directly from the support vectors of a trained SVM using a modified sequential covering algorithm. In [56] the same authors propose eclectic rule extraction, still considering only the support vectors of a trained model. The work in [93] generates fuzzy rules instead of classical propositional rules. Here, the authors argue that long antecedents reduce comprehensibility, hence, a fuzzy approach allows for a more linguistically understandable result. The second class of simplifications can be exemplified by [97], which proposed the addition of the SVM's hyperplane, along with the support vectors, to the components in charge of creating the rules. His method relies on the creation of hyper-rectangles from the intersections between the support vectors and the hyper-plane. In a third approach to *model simplification*, another group of authors considered adding the actual training data as a component for building the rules. In [125, 244, 246] the authors proposed a clustering method to group prototype vectors for each class. By combining them with the support vectors, it allowed defining ellipsoids and hyper-rectangles in the input space. Similarly in [105], the authors proposed the so-called Hyper-rectangle Rule Extraction, an algorithm based on SVC (Support Vector Clustering) to find prototype vectors for each class and then define small hyper-rectangles around. In [104], the authors formulate the rule extraction problem as a multi-constrained optimization to create a set of non-overlapping rules. Each rule conveys a non-empty hyper-cube with a shared edge with the hyper-plane. In a similar study conducted in [245], extracting rules for gene expression data, the authors presented a novel technique as a component of a multi-kernel SVM. This multi-kernel method consists of feature selection, prediction modeling and rule extraction. Finally, the study in [133] makes use of a growing SVC to give an interpretation to SVM decisions in terms of linear rules that define the space in Voronoi sections from the extracted prototypes.

Leaving aside rule extraction, the literature has also contemplated some other techniques to contribute to the interpretation of SVMs. Three of them (visualization techniques) are clearly used toward explaining SVM models when used for concrete applications. For instance, [76] presents an innovative approach to visualize trained SVM to extract the information content from the kernel matrix. They center the study on Support Vector Regression models. They show the ability of the algorithm to visualize which of the input variables are actually related with the associated output data. In [67] a visual way combines the output of the SVM with heatmaps to guide the modification of compounds in late stages of drug discovery. They assign colors to atoms based on the weights of a trained linear SVM that allows for a much more comprehensive way of debugging the process. In [115] the authors argue that many of the presented studies for interpreting SVMs only account for the weight vectors, leaving the margin aside. In their study they show how this margin is important, and they create an statistic that explicitly accounts for the SVM margin. The

authors show how this statistic is specific enough to explain the multivariate patterns shown in neuroimaging.

Noteworthy is also the intersection between SVMs and Bayesian systems, the latter being adopted as a post-hoc technique to explain decisions made by the SVM model. This is the case of [248] and [247], which are studies where SVMs are interpreted as MAP (Maximum A Posteriori) solutions to inference problems with Gaussian Process priors. This framework makes tuning the hyper-parameters comprehensible and gives the capability of predicting class probabilities instead of the classical binary classification of SVMs. Interpretability of SVM models becomes even more involved when dealing with non-CPD (Conditional Positive Definite) kernels that are usually harder to interpret due to missing geometrical and theoretical understanding. The work in [101] revolves around this issue with a geometrical interpretation of indefinite kernel SVMs, showing that these do not classify by hyper-plane margin optimization. Instead, they minimize the distance between convex hulls in pseudo-Euclidean spaces.

A difference might be appreciated between the post-hoc techniques applied to other models and those noted for SVMs. In previous models, *model simplification* in a broad sense was the prominent method for post-hoc explainability. In SVMs, *local explanations* have started to take some weight among the propositions. However, simplification based methods are, on average, much older than local explanations.

As a final remark, none of the reviewed methods treating SVM explainability are dated beyond 2017, which might be due to the progressive proliferation of DL models in almost all disciplines. Another plausible reason is that these models are already understood, so it is hard to improve upon what has already been done.

2.4.3 Explainability in Deep Learning

Post-hoc *local explanations* and *feature relevance* techniques are increasingly the most adopted methods for explaining DNNs. This section reviews explainability studies proposed for the most used DL models, namely multi-layer neural networks, Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN).

Multi-layer Neural Networks

From their inception, multi-layer neural networks (also known as multi-layer perceptrons) have been warmly welcomed by the academic community due to their huge ability to infer complex relations among variables. However, as stated in the introduction, developers and engineers in charge of deploying these models in real-life production find in their questionable explainability a common reason for reluctance. That is why neural networks

have been always considered as black-box models. The fact that explainability is often a must for the model to be of practical value, forced the community to generate multiple explainability techniques for multi-layer neural networks, including *model simplification* approaches, *feature relevance* estimators, *text explanations*, *local explanations* and model *visualizations*.

Several *model simplification* techniques have been proposed for neural networks with one single hidden layer, however very few works have been presented for neural networks with multiple hidden layers. One of these few works is DeepRED algorithm [257], which extends the decompositional approach to rule extraction (splitting at neuron level) presented in [259] for multi-layer neural network by adding more decision trees and rules.

Some other works use *model simplification* as a post-hoc explainability approach. For instance, [55] presents a simple distillation method called *Interpretable Mimic Learning* to extract an interpretable model by means of gradient boosting trees. In the same direction, the authors in [134] propose a hierarchical partitioning of the feature space that reveals the iterative rejection of unlikely class labels, until association is predicted. In addition, several works addressed the distillation of knowledge from an ensemble of models into a single model [79, 288, 289] .

Given the fact that the simplification of multi-layer neural networks is more complex as the number of layers increases, explaining these models by *feature relevance* methods has become progressively more popular. One of the representative works in this area is [59], which presents a method to decompose the network classification decision into contributions of its input elements. They consider each neuron as an object that can be decomposed and expanded then aggregate and back-propagate these decompositions through the network, resulting in a *deep* Taylor decomposition. In the same direction, the authors in [109] proposed DeepLIFT, an approach for computing importance scores in a multi-layer neural network. Their method compares the activation of a neuron to the reference activation and assigns the score according to the difference.

On the other hand, some works try to verify the theoretical soundness of current explainability methods. For example, the authors in [262], bring up a fundamental problem of most *feature relevance* techniques, designed for multi-layer networks. They showed that two axioms that such techniques ought to fulfill namely, *sensitivity* and *implementation invariance*, are violated in practice by most approaches. Following these axioms, the authors of [262] created *integrated gradients*, a new *feature relevance* method proven to meet the aforementioned axioms. Similarly, the authors in [60] analyzed the correctness of current *feature relevance* explanation approaches designed for Deep Neural Networks, e,g., DeConvNet, Guided BackProp and LRP, on simple linear neural networks. Their analysis showed that these methods do not produce the theoretically correct explanation and presented two new explanation methods *PatternNet* and *PatternAttribution* that are

more theoretically sound for both, simple and deep neural networks.

Convolutional Neural Networks

Currently, CNNs constitute the state-of-art models in all fundamental computer vision tasks, from image classification and object detection to instance segmentation. Typically, these models are built as a sequence of convolutional layers and pooling layers to automatically learn increasingly higher level features. At the end of the sequence, one or multiple fully connected layers are used to map the output features map into scores. This structure entails extremely complex internal relations that are very difficult to explain. Fortunately, the road to explainability for CNNs is easier than for other types of models, as the human cognitive skills favors the understanding of visual data.

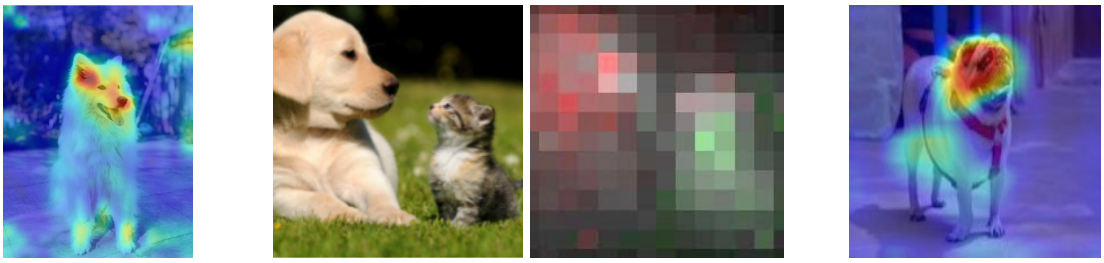
Existing works that aim at understanding what CNNs learn can be divided into two broad categories: 1) those that try to understand the decision process by mapping back the output in the input space to see which parts of the input were discriminative for the output; and 2) those that try to delve inside the network and interpret how the intermediate layers see the external world, not necessarily related to any specific input, but in general.

One of the seminal works in the first category was [290]. When an input image runs feed-forward through a CNN, each layer outputs a number of feature maps with strong and soft activations. The authors in [290] used Deconvnet, a network designed previously by the same authors [141] that, when fed with a feature map from a selected layer, reconstructs the maximum activations. These reconstructions can give an idea about the parts of the image that produced that effect. To visualize these strongest activations in the input image, the same authors used the occlusion sensitivity method to generate a saliency map [135], which consists of iteratively forwarding the same image through the network occluding a different region at a time.

To improve the quality of the mapping on the input space, several subsequent papers proposed simplifying both the CNN architecture and the visualization method. In particular, [95] included a global average pooling layer between the last convolutional layer of the CNN and the fully-connected layer that predicts the object class. With this simple architectural modification of the CNN, the authors built a class activation map that helps identify the image regions that were particularly important for a specific object class by projecting back the weights of the output layer on the convolutional feature maps. Later, in [142], the authors showed that max-pooling layers can be used to replace convolutional layers with a large stride without loss in accuracy on several image recognition benchmarks. They obtained a cleaner visualization than Deconvnet by using a guided backpropagation method.

To increase the interpretability of classical CNNs, the authors in [112] used a loss for each filter in high level convolutional layers to force each filter to learn very specific

object components. The obtained activation patterns are much more interpretable for their exclusiveness with respect to the different labels to be predicted. The authors in [71] proposed visualizing the contribution to the prediction of each single pixel of the input image in the form of a heatmap. They used a Layer-wise Relevance Propagation (LRP) technique, which relies on a Taylor series close to the prediction point rather than partial derivatives at the prediction point itself. To further improve the quality of the visualization, attribution methods such as heatmaps, saliency maps or class activation methods (*GradCAM* [291]) are used (see Figure 2.6). In particular, the authors in [291] proposed a Gradient-weighted Class Activation Mapping (Grad-CAM), which uses the gradients of any target concept, flowing into the final convolutional layer to produce a coarse localization map, highlighting the important regions in the image for predicting the concept.



(a) Heatmap [168] (b) Attribution [292] (c) Grad-CAM [291]

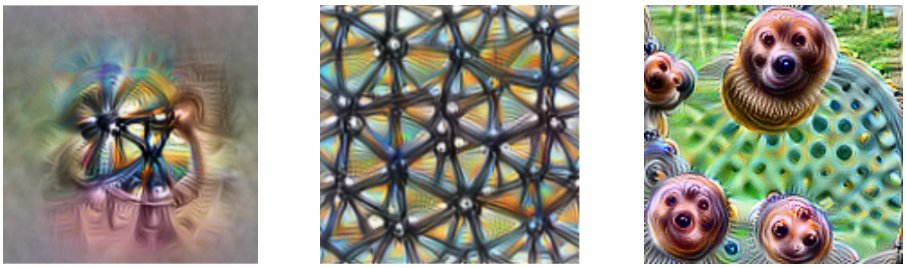
Figure 2.6: Examples of rendering for different XAI visualization techniques on images.

In addition to the aforementioned *feature relevance* and *visual* explanation methods, some works proposed generating *text explanations* of the visual content of the image. For example, the authors in [90] combined a CNN feature extractor with an RNN attention model to automatically learn to describe the content of images. In the same line, [277] presented a three-level attention model to perform a fine-grained classification task. The general model is a pipeline that integrates three types of attention: the object level attention model proposes candidate image regions or patches from the input image, the part-level attention model filters out non-relevant patches to a certain object, and the last attention model localizes discriminative patches. In the task of video captioning, the authors in [110] use a CNN model combined with a bi-directional LSTM model as encoder to extract video features and then feed these features to an LSTM decoder to generate textual descriptions.

One of the seminal works in the second category is [136]. In order to analyse the visual information contained inside the CNN, the authors proposed a general framework that reconstruct an image from the CNN internal representations and showed that several layers retain photographically accurate information about the image, with different degrees of geometric and photometric invariance. To visualize the notion of a class captured by a CNN,

the same authors created an image that maximizes the class score based on computing the gradient of the class score with respect to the input image [271]. In the same direction, the authors in [267] introduced a Deep Generator Network (DGN) that generates the most representative image for a given output neuron in a CNN.

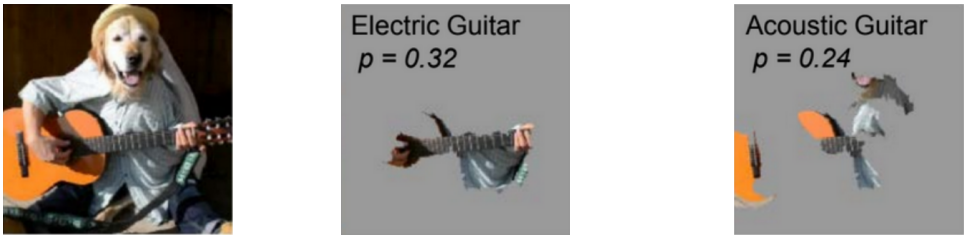
For quantifying the interpretability of the latent representations of CNNs, the authors in [124] used a different approach called network dissection. They run a large number of images through a CNN and then analyze the top activated images by considering each unit as a concept detector to further evaluate each unit for semantic segmentation. This paper also examines the effects of classical training techniques on the interpretability of the learned model. Although many of the techniques examined above utilize *local explanations* to achieve an overall explanation of a CNN model, others explicitly focus on building global explanations based on locally found prototypes. In [263, 293], the authors empirically showed how *local explanations* in deep networks are strongly dominated by their lower level features. They demonstrated that deep architectures provide strong priors that prevent the altering of how these low-level representations are captured. Instead of using one single interpretability technique, the framework proposed in [294] combines several methods to provide much more information about the network. For example, combining feature visualization (*what is a neuron looking for?*) with attribution (*how does it affect the output?*) allows exploring how the network decides between labels. This visual interpretability interface displays different blocks such as feature visualization and attribution depending on the visualization goal. This interface can be thought of as a union of individual elements that belong to layers (input, hidden, output), atoms (a neuron, channel, spatial or neuron group), content (activations – the amount a neuron fires, attribution – which classes a spatial position most contributes to, which tends to be more meaningful in later layers), and presentation (information visualization, feature visualization). Figure 2.7 shows some examples. Attribution methods normally rely on pixel association, displaying what part of an input example is responsible for the network activating in a particular way [292].



(a) Neuron (b) Channel (c) Layer

Figure 2.7: Feature visualization at different levels of a certain network [292].

All in all, *visualization* mixed with *feature relevance* methods are arguably the most adopted approach to explainability in CNNs. A much simpler approach to all the previously cited methods was proposed in LIME framework [70], as was described in Subsection 2.4.1 LIME perturbs the input and sees how the predictions change. In image classification, LIME creates a set of perturbed instances by dividing the input image into interpretable components (contiguous *superpixels*), and runs each perturbed instance through the model to get a probability. A simple linear model learns on this data set, which is locally weighted. At the end of the process, LIME presents the superpixels with highest positive weights as an explanation (see Figure 2.8).



(a) Original image (b) Explaining *electric guitar* (c) Explaining *acoustic guitar*

Figure 2.8: Examples of explanation when using LIME on images [70].

A completely different explainability approach is proposed in adversarial detection. To understand model failures in detecting adversarial examples, the authors in [8] apply the k-nearest neighbors algorithm on the representations of the data learned by each layer of the CNN. A test input image is considered as adversarial if its representations are far from the representations of the training images.

Recurrent Neural Networks

As occurs with CNNs in the visual domain, RNNs have lately been used extensively for predictive problems defined over inherently sequential data, with a notable presence in natural language processing and time series analysis. These types of data exhibit long-term dependencies that are complex to be captured by a ML model. RNNs are able to retrieve such time-dependent relationships by formulating the retention of knowledge in the neuron as another parametric characteristic that can be learned from data.

Few contributions have been made for explaining RNN models. These studies can be divided into two groups: 1) explainability by understanding what a RNN model has learned (mainly via *feature relevance* methods); and 2) explainability by modifying RNN architectures to provide insights about the decisions they make (*local explanations*).

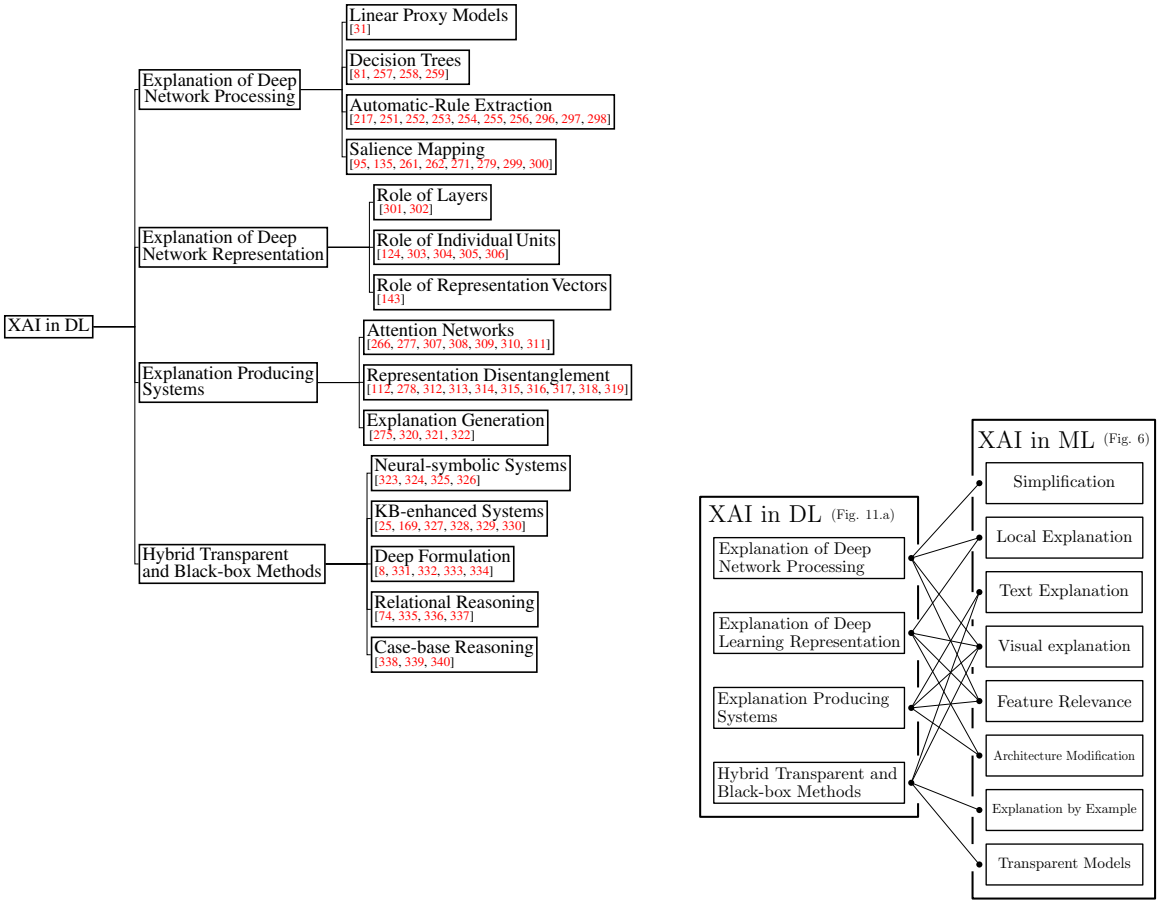
In the first group, the authors in [279] extend the usage of LRP to RNNs. They propose a specific propagation rule that works with multiplicative connections as those in LSTMs (Long Short Term Memory) units and GRUs (Gated Recurrent Units). The authors in [280] propose a visualization technique based on finite horizon n-grams that discriminates interpretable cells within LSTM and GRU networks. Following the premise of not altering the architecture, [295] extends the interpretable mimic learning distillation method used for CNN models to LSTM networks, so that interpretable features are learned by fitting Gradient Boosting Trees to the trained LSTM network under focus.

Aside from the approaches that do not change the inner workings of the RNNs, [284] presents RETAIN (REverse Time AttentIoN) model, which detects influential past patterns by means of a two-level neural attention model. To create an interpretable RNN, the authors in [282] propose an RNN based on SISTA (Sequential Iterative Soft-Thresholding Algorithm) that models a sequence of correlated observations with a sequence of sparse latent vectors, making its weights interpretable as the parameters of a principled statistical model. Finally, [283] constructs a combination of an HMM (Hidden Markov Model) and an RNN, so that the overall model approach harnesses the interpretability of the HMM and the accuracy of the RNN model.

2.4.4 Alternative Taxonomy of Post-hoc Explainability Techniques for Deep Learning

DL is the model family where most research has been concentrated in recent times and they have become central for most of the recent literature on XAI. While the division between model-agnostic and model-specific is the most common distinction made, the community has not only relied on this criteria to classify XAI methods. For instance, some model-agnostic methods such as *SHAP* [224] are widely used to explain DL models. That is why several XAI methods can be easily categorized in different taxonomy branches depending on the angle the method is looked at. An example is LIME which can also be used over CNNs, despite not being exclusive to deal with images. Searching within the alternative DL taxonomy shows us that LIME can explicitly be used for *Explaining a Deep Network Processing*, as a kind of *Linear Proxy Model*. Another type of classification is indeed proposed in [14] with a segmentation based on 3 categories. The first category groups methods explaining the processing of data by the network, thus answering to the question *"why does this particular input leads to this particular output?"*. The second one concerns methods explaining the representation of data inside the network, i.e., answering to the question *"what information does the network contain?"*. The third approach concerns models specifically designed to simplify the interpretation of their own behavior. Such a multiplicity of classification possibilities leads to different ways of constructing XAI

taxonomies.



(a) (b)

Figure 2.9: (a) Alternative Deep Learning specific taxonomy extended from the categorization from [14]; and (b) its connection to the taxonomy in Figure 2.5.

Figure 2.9 shows the alternative Deep Learning taxonomy inferred from [14]. From the latter, it can be deduced the complementarity and overlapping of this taxonomy to Figure 2.5 as:

- Some methods [271, 279] classified in distinct categories (namely *feature relevance for CNN* and *feature relevance for RNN*) in Figure 2.5 are included in a single category (*Explanation of Deep Network Processing with Salience Mapping*) when considering the classification from [14].

- Some methods [81, 143] are classified on a single category (*Explanation by simplification for Multi-Layer Neural Network*) in Figure 2.5 while being in 2 different categories (namely, *Explanation of Deep Network Processing with Decision Trees* and *Explanation of Deep Network Representation with the Role of Representation Vectors*) in [14], as shown in Figure 2.9.

A classification based on explanations of model processing and explanations of model representation is relevant, as it leads to a differentiation between the execution trace of the model and its internal data structure. This means that depending of the failure reasons of a complex model, it would be possible to pick-up the right XAI method according to the information needed: the execution trace or the data structure. This idea is analogous to testing and debugging methods used in regular programming paradigms [341].

2.5 XAI: Opportunities, Challenges and Future Research Needs

We now capitalize on the performed literature review to put forward a critique of the achievements, trends and challenges that are still to be addressed in the field of explainability of ML and data fusion models. Actually our discussion on the advances taken so far in this field has already anticipated some of these challenges. In this section we revisit them and explore new research opportunities for XAI, identifying possible research paths that can be followed to address them effectively in years to come:

- When introducing the overview in Section 2.1 we already mentioned the existence of a trade-off between model interpretability and performance, in the sense that making a ML model more understandable could eventually degrade the quality of its produced decisions. In Subsection 2.5.1 we will stress on the potential of XAI developments to effectively achieve an optimal balance between the interpretability and performance of ML models.

- In Subsection 2.2.2 we stressed on the imperative need for reaching a consensus on *what* explainability entails within the AI realm. Reasons for pursuing explainability are also assorted and, under our own assessment of the literature so far, not unambiguously mentioned throughout related works. In Subsection 2.5.2 we will further delve into this important issue.

- Given its notable prevalence in the XAI literature, Subsections 2.4.3 and 2.4.4 revolved on the explainability of Deep Learning models, examining advances reported so far around

a specific bibliographic taxonomy. We go in this same direction with Subsection 2.5.3, which exposes several challenges that hold in regards to the explainability of this family of models.

- Finally, we close up this prospective discussion with Subsections 2.5.4 to 2.5.8, which place on the table several research niches that despite its connection to model explainability, remain insufficiently studied by the community.

Before delving into these identified challenges, it is important to bear in mind that this prospective section is complemented by Section 2.6, which enumerates research needs and open questions related to XAI within a broader context: the need for responsible AI.

2.5.1 On the trade-off between Interpretability and Performance

The matter of interpretability versus performance is one that repeats itself through time, but as any other big statement, has its surroundings filled with myths and misconceptions.

As perfectly stated in [342], it is not necessarily true that models that are more complex are inherently more accurate. This statement is false in cases in which the data is well structured and features at our disposal are of great quality and value. This case is somewhat common in some industry environments, since features being analyzed are constrained within very controlled physical problems, in which all of the features are highly correlated, and not much of the possible landscape of values can be explored in the data [343]. What can be hold as true, is that more complex models enjoy much more flexibility than their simpler counterparts, allowing for more complex functions to be approximated. Now, returning to the statement *"models that are more complex are more accurate"*, given the premise that the function to be approximated entails certain complexity, that the data available for study is greatly widespread among the world of suitable values for each variable and that there is enough data to harness a complex model, the statement presents itself as a true statement. It is in this situation that the trade-off between performance and interpretability can be observed. It should be noted that the attempt at solving problems that do not respect the aforementioned premises will fall on the trap of attempting to solve a problem that does not provide enough data diversity (variance). Hence, the added complexity of the model will only fight against the task of accurately solving the problem.

In this path toward performance, when the performance comes hand in hand with complexity, interpretability encounters itself on a downwards slope that until now appeared unavoidable. However, the apparition of more sophisticated methods for explainability could invert or at least cancel that slope. Figure 2.10 shows a tentative representation inspired by previous works [3], in which XAI shows its power to improve the common trade-off between model interpretability and performance. Another aspect worth mentioning at

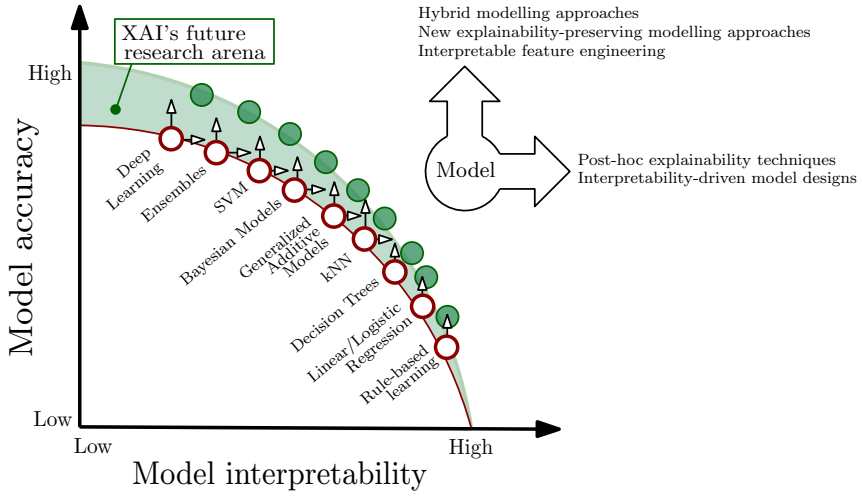


Figure 2.10: Trade-off between model interpretability and performance, and a representation of the area of improvement where the potential of XAI techniques and tools resides.

this point due to its close link to model interpretability and performance is the *approximation dilemma*: explanations made for a ML model must be made drastic and approximate enough to match the requirements of the audience for which they are sought, ensuring that explanations are representative of the studied model and do not oversimplify its essential features.

2.5.2 On the Concept and Metrics

The literature clearly asks for an unified concept of explainability. In order for the field to thrive, it is imperative to place a common ground upon which the community is enabled to contribute new techniques and methods. A common concept must convey the needs expressed in the field. It should propose a common structure for every XAI system. This paper attempted a new proposition of a concept of explainability that is built upon that from Gunning [3]. In that proposition and the following strokes to complete it (Subsection 2.2.2), explainability is defined as the ability a model has to make its functioning clearer to an audience. To address it, post-hoc type methods exist. The concept portrayed in this survey might not be complete but as it stands, allows for a first common ground and reference point to sustain a profitable discussion in this matter. It is paramount that the field of XAI reaches an agreement in this respect combining the shattered efforts of a widespread field behind the same banner.

Another key feature needed to relate a certain model to this concrete concept is the

existence of a metric. A metric, or group of them should allow for a meaningful comparison of how well a model fits the definition of explainable. Without such tool, any claim in this respect dilutes among the literature, not providing a solid ground on which to stand. These metrics, as the classic ones (accuracy, F1, sensitivity...), should express how well the model performs in a certain aspect of explainability. Some attempts have been done recently around the measurement of XAI, as reviewed thoroughly in [344, 345]. In general, XAI measurements should evaluate the goodness, usefulness and satisfaction of explanations, the improvement of the mental model of the audience induced by model explanations, and the impact of explanations on the performance of the model and on the trust and reliance of the audience. Measurement techniques surveyed in [344] and [345] (e.g., goodness checklist, explanation satisfaction scale, elicitation methods for mental models, computational measures for explainer fidelity, explanation trustworthiness and model reliability) seem to be a good push in the direction of evaluating XAI techniques. Unfortunately, conclusions drawn from these overviews are aligned with our prospects on the field: more quantifiable, general XAI metrics are really needed to support the existing measurement procedures and tools proposed by the community.

This survey does not tackle the problem of designing such a suite of metrics, since such a task should be approached by the community as a whole prior acceptance of the broader concept of explainability, which on the other hand, is one of the aims of the current work. Nevertheless, we advocate for further efforts towards new proposals to evaluate the performance of XAI techniques, as well as comparison methodologies among XAI approaches that allow contrasting them quantitatively under different application context, models and purposes.

2.5.3 Challenges to achieve Explainable Deep Learning

While many efforts are currently being made in the area of XAI, there are still many challenges to be faced before being able to obtain explainability in DL models. First, as explained in Subsection 2.2.2, there is a lack of agreement on the vocabulary and the different definitions surrounding XAI. As an example, we often see the terms *feature importance* and *feature relevance* referring to the same concept. This is even more obvious for visualization methods, where there is absolutely no consistency behind what is known as saliency maps, salient masks, heatmaps, neuron activations, attribution, and other approaches alike. As XAI is a relatively young field, the community does not have a standardized terminology yet.

As it has been commented in Subsection 2.5.1, there is a trade-off between interpretability and accuracy [14], i.e., between the simplicity of the information given by the system on its internal functioning, and the exhaustiveness of this description. Whether the observer

is an expert in the field, a policy-maker or a user without machine learning knowledge, intelligibility does not have to be at the same level in order to provide the *audience* an understanding [6]. This is one of the reasons why, as mentioned above, a challenge in XAI is establishing objective metrics on what constitutes a good explanation. A possibility to reduce this subjectivity is taking inspiration from experiments on human psychology, sociology or cognitive sciences to create objectively convincing explanations. Relevant findings to be considered when creating an explainable AI model are highlighted in [13]: First, explanations are better when *constrictive*, meaning that a prerequisite for a good explanation is that it does not only indicate why the model made a decision X, but also why it made decision X rather than decision Y. It is also explained that probabilities are not as important as causal links in order to provide a satisfying explanation. Considering that black box models tend to process data in a quantitative manner, it would be necessary to translate the probabilistic results into qualitative notions containing causal links. In addition, they state that explanations are *selective*, meaning that focusing solely on the main causes of a decision-making process is sufficient. It was also shown that the use of counterfactual explanations can help the user to understand the decision of a model [39, 41, 346].

Combining connectionist and symbolic paradigms seems a favourable way to address this challenge [169, 325, 335, 347, 348]. On one hand, connectionist methods are more precise but opaque. On the other hand, symbolic methods are popularly considered less efficient, while they offer a greater explainability thus respecting the conditions mentioned above:

- The ability to refer to established reasoning rules allows symbolic methods to be constrictive.
- The use of a KB formalized e.g. by an ontology can allow data to be processed directly in a qualitative way.
- Being selective is less straightforward for connectionist models than for symbolic ones.

Recalling that a good explanation needs to influence the mental model of the user, i.e. the representation of the external reality using, among other things, symbols, it seems obvious that the use of the symbolic learning paradigm is appropriate to produce an explanation. Therefore, neural-symbolic interpretability could provide convincing explanations while keeping or improving generic performance [323].

As stated in [25], a truly explainable model should not leave explanation generation to the users as different explanations may be deduced depending on their background knowledge. Having a semantic representation of the knowledge can help a model to have the ability to produce explanations (e.g., in natural language [169]) combining common sense reasoning and human-understandable features.

Furthermore, until an objective metric has been adopted, it appears necessary to make an effort to rigorously formalize evaluation methods. One way may be drawing inspiration from the social sciences, e.g., by being consistent when choosing the evaluation questions and the population sample used [349].

A final challenge XAI methods for DL need to address is providing explanations that are accessible for society, policy makers and the law as a whole. In particular, conveying explanations that require non-technical expertise will be paramount to both handle ambiguities, and to develop the social right to the (not-yet available) right for explanation in the EU General Data Protection Regulation (GDPR) [350].

2.5.4 Explanations for AI Security: XAI and Adversarial Machine Learning

Nothing has been said about confidentiality concerns linked to XAI. One of the last surveys very briefly introduced the idea of algorithm property and trade secrets [15]. However, not much attention has been payed to these concepts. If *confidential* is the property that makes something *secret*, in the AI context many aspects involved in a model may hold this property. For example, imagine a model that some company has developed through many years of research in a specific field. The knowledge synthesized in the model built might be considered to be confidential, and it may be compromised even by providing only input and output access [351]. The latter shows that, under minimal assumptions, *data model functionality stealing* is possible. An approach that has served to make DL models more robust against intellectual property exposure based on a sequence of non accessible queries is in [352]. This recent work exposes the need for further research toward the development of XAI tools capable of explaining ML models while keeping the model's confidentiality in mind.

Ideally, XAI should be able to explain the knowledge within an AI model and it should be able to reason about what the model acts upon. However, the information revealed by XAI techniques can be used both to generate more effective attacks in adversarial contexts aimed at confusing the model, at the same time as to develop techniques to better protect against private content exposure by using such information. Adversarial attacks [353] try to manipulate a ML algorithm after learning what is the specific information that should be fed to the system so as to lead it to a specific output. For instance, regarding a supervised ML classification model, adversarial attacks try to discover the minimum changes that should be applied to the input data in order to cause a different classification. This has happened regarding computer vision systems of autonomous vehicles; a minimal change in a stop signal, imperceptible to the human eye, led vehicles to detect it as a 45 mph signal [354]. For the particular case of DL models, available solutions such as Cleverhans [355] seek to detect

adversarial vulnerabilities, and provide different approaches to harden the model against them. Other examples include AlfaSVMLib [356] for SVM models, and AdversarialLib [357] for evasion attacks. There are even available solutions for unsupervised ML, like clustering algorithms [358].

While XAI techniques can be used to furnish more effective adversarial attacks or to reveal confidential aspects of the model itself, some recent contributions have capitalized on the possibilities of Generative Adversarial Networks (GANs [359]), Variational Autoencoders [360] and other generative models towards explaining data-based decisions. Once trained, generative models can generate instances of what they have learned based on a noise input vector that can be interpreted as a latent representation of the data at hand. By manipulating this latent representation and examining its impact on the output of the generative model, it is possible to draw insights and discover specific patterns related to the class to be predicted. This generative framework has been adopted by several recent studies [361, 362] mainly as an attribution method to relate a particular output of a Deep Learning model to their input variables. Another interesting research direction is the use of generative models for the creation of counterfactuals, i.e., modifications to the input data that could eventually alter the original prediction of the model [363]. Counterfactual prototypes help the user understand the performance boundaries of the model under consideration for his/her improved trust and informed criticism. In light of this recent trend, we definitely believe that there is road ahead for generative ML models to take their part in scenarios demanding understandable machine decisions.

2.5.5 XAI and Output Confidence

Safety issues have also been studied in regards to processes that depend on the output of AI models, such as vehicular perception and self-driving in autonomous vehicles, automated surgery, data-based support for medical diagnosis, insurance risk assessment and cyber-physical systems in manufacturing, among others [364]. In all these scenarios erroneous model outputs can lead to harmful consequences, which has yielded comprehensive regulatory efforts aimed at ensuring that no decision is made solely on the basis of data processing [10].

In parallel, research has been conducted towards minimizing both risk and uncertainty of harms derived from decisions made on the output of a ML model. As a result, many techniques have been reported to reduce such a risk, among which we pause at the evaluation of the model's output confidence to decide upon. In this case, the inspection of the share of epistemic uncertainty (namely, the uncertainty due to lack of knowledge) of the input data and its correspondence with the model's output confidence can inform the user and eventually trigger his/her rejection of the model's output [365, 366]. To this end, explaining

via XAI techniques which region of the input data the model is focused on when producing a given output can discriminate possible sources of epistemic uncertainty within the input domain.

2.5.6 XAI, Rationale Explanation, and Critical Data Studies

When shifting the focus to the research practices seen in Data Science, it has been noted that reproducibility is stringently subject not only to the mere sharing of data, models and results to the community, but also to the availability of information about the full discourse around data collection, understanding, assumptions held and insights drawn from model construction and results' analyses [367]. In other words, in order to transform data into a valuable actionable asset, individuals must engage in collaborative sense-making by sharing the context producing their findings, wherein context refers to sets of narrative stories around how data were processed, cleaned, modeled and analyzed. In this discourse we find also an interesting space for the adoption of XAI techniques due to their powerful ability to describe black-box models in an understandable, hence conveyable fashion towards colleagues from Social Science, Politics, Humanities and Legal fields.

XAI can effectively ease the process of explaining the reasons why a model reached a decision in an accessible way to non-expert users, i.e. the *rationale explanation*. This confluence of multi-disciplinary teams in projects related to Data Science and the search for methodologies to make them appraise the ethical implications of their data-based choices has been lately coined as Critical Data studies [368]. It is in this field where XAI can significantly boost the exchange of information among heterogeneous audiences about the knowledge learned by models.

2.5.7 XAI and Theory-guided Data Science

We envision an exciting synergy between the XAI realm and *Theory-guided Data Science*, a paradigm exposed in [369] that merges both Data Science and the classic theoretical principles underlying the application/context where data are produced. The rationale behind this rising paradigm is the need for data-based models to generate knowledge that is the prior knowledge brought by the field in which it operates. This means that the model type should be chosen according to the type of relations we intend to encounter. The structure should also follow what is previously known. Similarly, the training approach should not allow for the optimization process to enter regions that are not plausible. Accordingly, regularization terms should stand the prior premises of the field, avoiding the elimination of badly represented true relations for spurious and deceptive false relations. Finally, the output of the model should inform about everything the model has come to learn, allowing

to reason and merge the new knowledge with what was already known in the field.

Many examples of the implementation of this approach are currently available with promising results. The studies in [370]-[377] were carried out in diverse fields, showcasing the potential of this new paradigm for data science. Above all, it is relevant to notice the resemblance that all concepts and requirements of Theory-guided Data Science share with XAI. All the additions presented in [369] push toward techniques that would eventually render a model explainable, and furthermore, knowledge consistent. The concept of *knowledge from the beginning*, central to Theory-guided Data Science, must also consider how the knowledge captured by a model should be explained for assessing its compliance with theoretical principles known beforehand. This, again, opens a magnificent window of opportunity for XAI.

2.5.8 Guidelines for ensuring Interpretable AI Models

Recent surveys have emphasized on the multidisciplinary, inclusive nature of the process of making an AI-based model interpretable. Along this process, it is of utmost importance to scrutinize and take into proper account the interests, demands and requirements of all stakeholders interacting with the system to be explained, from the designers of the system to the decision makers consuming its produced outputs and users undergoing the consequences of decisions made therefrom.

Given the confluence of multiple criteria and the need for having the human in the loop, some attempts at establishing the procedural guidelines to implement and explain AI systems have been recently contributed. Among them, we pause at the thorough study in [378], which suggests that the incorporation and consideration of explainability in practical AI design and deployment workflows should comprise four major methodological steps:

1. Contextual factors, potential impacts and domain-specific needs must be taken into account when devising an approach to interpretability: These include a thorough understanding of the purpose for which the AI model is built, the complexity of explanations that are required by the audience, and the performance and interpretability levels of existing technology, models and methods. The latter pose a reference point for the AI system to be deployed in lieu thereof.

2. Interpretable techniques should be preferred when possible: when considering explainability in the development of an AI system, the decision of which XAI approach should be chosen should gauge domain-specific risks and needs, the available data resources and existing domain knowledge, and the suitability of the ML model to meet the requirements of the computational task to be addressed. It is in the confluence of these three design drivers where the guidelines postulated in [378] (and other studies in this

same line of thinking [379]) recommend first the consideration of standard interpretable models rather than sophisticated yet opaque modeling methods. In practice, the aforementioned aspects (contextual factors, impacts and domain-specific needs) can make transparent models preferable over complex modeling alternatives whose interpretability require the application of post-hoc XAI techniques. By contrast, black-box models such as those reviewed in this work (namely, support vector machines, ensemble methods and neural networks) should be selected only when their superior modeling capabilities fit best the characteristics of the problem at hand.

3. If a black-box model has been chosen, the third guideline establishes that ethics-, fairness- and safety-related impacts should be weighed. Specifically, responsibility in the design and implementation of the AI system should be ensured by checking whether such identified impacts can be mitigated and counteracted by supplementing the system with XAI tools that provide the level of explainability required by the domain in which it is deployed. To this end, the third guideline suggests 1) a detailed articulation, examination and evaluation of the applicable explanatory strategies, 2) the analysis of whether the coverage and scope of the available explanatory approaches match the requirements of the domain and application context where the model is to be deployed; and 3) the formulation of an interpretability action plan that sets forth the explanation delivery strategy, including a detailed time frame for the execution of the plan, and a clearance of the roles and responsibilities of the team involved in the workflow.

4. Finally, the fourth guideline encourages to rethink interpretability in terms of the cognitive skills, capacities and limitations of the individual human. This is an important question on which studies on measures of explainability are intensively revolving by considering human mental models, the accessibility of the audience to vocabularies of explanatory outcomes, and other means to involve the expertise of the audience into the decision of what explanations should provide.

We foresee that the set of guidelines proposed in [378] and summarized above will be complemented and enriched further by future methodological studies, ultimately heading to a more *responsible* use of AI. Methodological principles ensure that the purpose for which explainability is pursued is met by bringing the manifold of requirements of all participants into the process, along with other universal aspects of equal relevance such as no discrimination, sustainability, privacy or accountability. A challenge remains in harnessing the potential of XAI to realize a *Responsible AI*, as we discuss in the next section.

2.6 Toward Responsible AI: Principles of Artificial Intelligence, Fairness, Privacy and Data Fusion

Over the years many organizations, both private and public, have published guidelines to indicate how AI should be developed and used. These guidelines are commonly referred to as AI *principles*, and they tackle issues related to potential AI threats to both individuals and to the society as a whole. This section presents some of the most important and widely recognized principles in order to link XAI – which normally appears inside its own principle – to all of them. Should a responsible implementation and use of AI models be sought in practice, it is our firm claim that XAI does not suffice on its own. Other important principles of Artificial Intelligence such as privacy and fairness must be carefully addressed in practice. In the following sections we elaborate on the concept of Responsible AI, along with the implications of XAI and data fusion in the fulfillment of its postulated principles.

2.6.1 Principles of Artificial Intelligence

A recent review of some of the main AI principles published since 2016 appears in [380]. In this work, the authors show a visual framework where different organizations are classified according to the following parameters:

- Nature, which could be private sector, government, inter-governmental organization, civil society or multistakeholder.
- Content of the principles: eight possible principles such as privacy, explainability, or fairness. They also consider the coverage that the document grants for each of the considered principles.
- Target audience: to whom the principles are aimed. They are normally for the organization that developed them, but they could also be destined for another audience (see Figure 2.1).
- Whether or not they are rooted on the International Human Rights, as well as whether they explicitly talk about them.

For instance, [381] is an illustrative example of a document of AI principles for the purpose of this overview, since it accounts for some of the most common principles, and deals explicitly with explainability. Here, the authors propose five principles mainly to guide the development of AI within their company, while also indicating that they could also be used within other organizations and businesses.

The authors of those principles aim to develop AI in a way that it directly reinforces inclusion, gives equal opportunities for everyone, and contributes to the common good. To this end, the following aspects should be considered:

- The outputs after using AI systems should not lead to any kind of discrimination against individuals or collectives in relation to race, religion, gender, sexual orientation, disability, ethnic, origin or any other personal condition. Thus, a fundamental criteria to consider while optimizing the results of an AI system is not only their outputs in terms of error optimization, but also how the system deals with those groups. This defines the principle of *Fair AI*.

- People should always know when they are communicating with a person, and when they are communicating with an AI system. People should also be aware if their personal information is being used by the AI system and for what purpose. It is crucial to ensure a certain level of understanding about the decisions taken by an AI system. This can be achieved through the usage of XAI techniques. It is important that the generated explanations consider the profile of the user that will receive those explanations (the so-called *audience* as per the definition given in Subsection 2.2.2) in order to adjust the transparency level, as indicated in [43]. This defines the principle of *Transparent and Explainable AI*.

- AI products and services should always be aligned with the United Nation's Sustainable Development Goals [382] and contribute to them in a positive and tangible way. Thus, AI should always generate a benefit for humanity and the common good. This defines the principle of *Human-centric AI* (also referred to as *AI for Social Good* [383]).

- AI systems, specially when they are fed by data, should always consider privacy and security standards during all of its life cycle. This principle is not exclusive of AI systems since it is shared with many other software products. Thus, it can be inherited from processes that already exist within a company. This defines the principle of *Privacy and Security by Design*, which was also identified as one of the core ethical and societal challenges faced by Smart Information Systems under the Responsible Research and Innovation paradigm (RRI, [384]). RRI refers to a package of methodological guidelines and recommendations aimed at considering a wider context for scientific research, from the perspective of the lab to global societal challenges such as sustainability, public engagement, ethics, science education, gender equality, open access, and governance. Interestingly, RRI also requires openness and transparency to be ensured in projects embracing its principles, which links directly to the principle of Transparent and Explainable AI mentioned previously.

- The authors emphasize that all these principles should always be extended to any third-party (providers, consultants, partners...).

Going beyond the scope of these five AI principles, the European Commission (EC) has recently published ethical guidelines for Trustworthy AI [385] through an assessment checklist that can be completed by different profiles related to AI systems (namely, product managers, developers and other roles). The assessment is based in a series of principles: 1) human agency and oversight; 2) technical robustness and safety; 3) privacy and data governance; 4) transparency, diversity, non-discrimination and fairness; 5) societal and environmental well-being; 6) accountability. These principles are aligned with the ones detailed in this section, though the scope for the EC principles is more general, including any type of organization involved in the development of AI.

It is worth mentioning that most of these AI principles guides directly approach XAI as a key aspect to consider and include in AI systems. In fact, the overview for these principles introduced before [380], indicates that 28 out of the 32 AI principles guides covered in the analysis, explicitly include XAI as a crucial component. Thus, the work and scope of this chapter deals directly with one of the most important aspects regarding AI at a worldwide level.

2.6.2 Fairness and Accountability

As mentioned in the previous section, there are many critical aspects, beyond XAI, included within the different AI principles guidelines published during the last decade. However, those aspects are not completely detached from XAI; in fact, they are intertwined. This section presents two key components with a huge relevance within the AI principles guides, Fairness and Accountability. It also highlights how they are connected to XAI.

Fairness and Discrimination

It is in the identification of implicit correlations between protected and unprotected features where XAI techniques find their place within discrimination-aware data mining methods. By analyzing how the output of the model behaves with respect to the input feature, the model designer may unveil hidden correlations between the input variables amenable to cause discrimination. XAI techniques such as SHAP [224] could be used to generate counterfactual outcomes explaining the decisions of a ML model when fed with protected and unprotected variables.

Recalling the Fair AI principle introduced in the previous section, [381] reminds that fairness is a discipline that generally includes proposals for bias detection within datasets regarding sensitive data that affect protected groups (through variables like gender, race...).

Indeed, ethical concerns with black-box models arise from their tendency to unintentionally create unfair decisions by considering sensitive factors such as the individual's race, age or gender [386]. Unfortunately, such unfair decisions can give rise to discriminatory issues, either by explicitly considering sensitive attributes or implicitly by using factors that correlate with sensitive data. In fact, an attribute may implicitly encode a protected factor, as occurs with postal code in credit rating [387]. The aforementioned proposals centered on fairness aspects permit to discover correlations between non-sensitive variables and sensitive ones, detect imbalanced outcomes from the algorithms that penalize a specific subgroup of people (*discrimination*), and mitigate the effect of bias on the model's decisions. These approaches can deal with:

- Individual fairness: here, fairness is analyzed by modeling the differences between each subject and the rest of the population.
- Group fairness: it deals with fairness from the perspective of all individuals.
- Counterfactual fairness: it tries to interpret the causes of bias using, for example, causal graphs.

The sources for bias, as indicated in [387], can be traced to:

- Skewed data: bias within the data acquisition process.
- Tainted data: errors in the data modelling definition, wrong feature labelling, and other possible causes.
- Limited features: using too few features could lead to an inference of false feature relationships that can lead to bias.
- Sample size disparities: when using sensitive features, disparities between different subgroups can induce bias.
- Proxy features: there may be correlated features with sensitive ones that can induce bias even when the sensitive features are not present in the dataset.

The next question that can be asked is what criteria could be used to define when AI is not biased. For supervised ML, [388] presents a framework that uses three criteria to evaluate group fairness when there is a sensitive feature present within the dataset:

- Independence: this criterion is fulfilled when the model predictions are independent of the sensitive feature. Thus, the proportion of positive samples (namely, those ones belonging to the class of interest) given by the model is the same for all the subgroups within the sensitive feature.

- Separation: it is met when the model predictions are independent of the sensitive feature given the target variable. For instance, in classification models, the True Positive (TP) rate and the False Positive (FP) rate are the same in all the subgroups within the sensitive feature. This criteria is also known as *Equalized Odds*.

- Sufficiency: it is accomplished when the target variable is independent of the sensitive feature given the model output. Thus, the Positive Predictive Value is the same for all subgroups within the sensitive feature. This criteria is also known as Predictive Rate Parity.

Although not all of the criteria can be fulfilled at the same time, they can be optimized together in order to minimize the bias within the ML model.

There are two possible actions that could be used in order to achieve those criteria. On one hand, evaluation includes measuring the amount of bias present within the model (regarding one of the criteria aforementioned). There are many different metrics that can be used, depending on the criteria considered. Regarding independence criterion, possible metrics are *statistical parity difference* or *disparate impact*. In case of the separation criterion, possible metrics are *equal opportunity difference* and *average odds difference* [388]. Another possible metric is the *Theil index* [389], which measures inequality both in terms of individual and group fairness.

On the other hand, mitigation refers to the process of fixing some aspects in the model in order to remove the effect of the bias in terms of one or several sensitive features. Several techniques exist within the literature, classified in the following categories:

- Pre-processing: these groups of techniques are applied before the ML model is trained, looking to remove the bias at the first step of the learning process. An example is Reweighing [390], which modifies the weights of the features in order to remove discrimination in sensitive attributes. Another example is [391], which hinges on transforming the input data in order to find a good representation that obfuscates information about membership in sensitive features.

- In-processing: these techniques are applied during the training process of the ML model. Normally, they include Fairness optimization constraints along with cost functions of the ML model. An example is Adversarial Debiasing, [392]. This technique optimizes jointly the ability of predicting the target variable while minimizing the ability of predicting sensitive features using a GAN.

- Post-processing: these techniques are applied after the ML model is trained. They are less intrusive because they do not modify the input data or the ML model. An example is Equalized Odds [388]. This techniques allows to adjust the thresholds in the classification

model in order to reduce the differences between the TP rate and the FP rate for each sensitive subgroup.

Even though these references apparently address an AI principle that appears to be independent of XAI, the literature shows that they are intertwined. For instance, the survey in [380] evinces that 26 out of the 28 AI principles that deal with XAI, also talk about fairness explicitly. This fact elucidates that organizations usually consider both aspects together when implementing Responsible AI.

The literature also exploses that XAI proposals can be used for bias detection. For example, [393] proposes a framework to visually analyze the bias present in a model (both for individual and group fairness). Thus, the fairness report is shown just like the visual summaries used within XAI. This explainability approach eases the understanding and measurement of bias. The system must report that there is bias, justify it quantitatively, indicate the degree of fairness, and explain why a user or group would be treated unfairly with the available data. Similarly, XAI techniques such as SHAP [224] could be used to generate counterfactual outcomes explaining the decisions of a ML model when fed with protected and unprotected variables. By identifying implicit correlations between protected and unprotected features through XAI techniques, the model designer may unveil hidden correlations between the input variables amenable to cause discrimination.

Another example is [394], where the authors propose a fair-by-design approach in order to develop ML models that jointly have less bias and include as explanations human comprehensible rules. The proposal is based in self-learning locally generative models that use only a small part of the whole dataset available (weak supervision). It first finds recursively relevant prototypes within the dataset, and extracts the empirical distribution and density of the points around them. Then it generates rules in an IF/THEN format that explain that a data point is classified within a specific category because it is *similar* to some prototypes. The proposal then includes an algorithm that both generates explanations and reduces bias, as it is demonstrated for the use case of recidivism using the Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) dataset [395]. The same goal has been recently pursued in [396], showing that post-hoc XAI techniques can forge fairer explanations from truly unfair black-box models. Finally, CERTIFAI (Counterfactual Explanations for Robustness, Transparency, Interpretability, and Fairness of Artificial Intelligence models) [397] uses a customized genetic algorithm to generate counterfactuals that can help to see the robustness of a ML model, generate explanations, and examine fairness (both at the individual level and at the group level) at the same time.

Strongly linked to the concept of fairness, much attention has been lately devoted to the concept of *data diversity*, which essentially refers to the capability of an algorithmic model to ensure that all different types of objects are represented in its output [398]. Therefore, diversity can be thought to be an indicator of the quality of a collection of items that, when

taking the form of a model's output, can quantify the proneness of the model to produce diverse results rather than highly accurate predictions. Diversity comes into play in human-centered applications with ethical restrictions that permeate to the AI modeling phase [399]. Likewise, certain AI problems (such as content recommendation or information retrieval) also aim at producing diverse recommendations rather than highly-scoring yet similar results [400, 401]. In these scenarios, dissecting the internals of a black-box model via XAI techniques can help identifying the capability of the model to maintain the input data diversity at its output. Learning strategies to endow a model with diversity keeping capabilities could be complemented with XAI techniques in order to shed transparency over the model internals, and assess the effectiveness of such strategies with respect to the diversity of the data from which the model was trained. Conversely, XAI could help to discriminate which parts of the model are compromising its overall ability to preserve diversity.

Accountability

Regarding accountability, the EC [385] defines the following aspects to consider:

- Auditability: it includes the assessment of algorithms, data and design processes, but preserving the intellectual property related to the AI systems. Performing the assessment by both internal and external auditors, and making the reports available, could contribute to the trustworthiness of the technology. When the AI system affects fundamental rights, including safety-critical applications, it should always be audited by an external third party.

- Minimization and reporting of negative impacts: it consists of reporting actions or decisions that yield a certain outcome by the system. It also comprises the assessment of those outcomes and how to respond to them. To address that, the development of AI systems should also consider the identification, assessment, documentation and minimization of their potential negative impacts. In order to minimize the potential negative impact, impact assessments should be carried out both prior to and during the development, deployment and use of AI systems. It is also important to guarantee protection for anyone who raises concerns about an AI system (e.g., *whistle-blowers*). All assessments must be proportionate to the risk that the AI systems pose.

- Trade-offs: in case any tension arises due to the implementation of the above requirements, trade-offs could be considered but only if they are ethically acceptable. Such trade-offs should be reasoned, explicitly acknowledged and documented, and they must be evaluated in terms of their risk to ethical principles. The decision maker must be accountable for the manner in which the appropriate trade-off is being made, and the trade-off decided

should be continually reviewed to ensure the appropriateness of the decision. If there is no ethically acceptable trade-off, the development, deployment and use of the AI system should not proceed in that form.

- Redress: it includes mechanisms that ensure an adequate redress for situations when unforeseen unjust adverse impacts take place. Guaranteeing a redress for those non-predicted scenarios is a key to ensure trust. Special attention should be paid to vulnerable persons or groups.

These aspects addressed by the EC highlight different connections of XAI with accountability. First, XAI contributes to auditability as it can help explaining AI systems for different profiles, including regulatory ones. Also, since there is a connection between fairness and XAI as stated before, XAI can also contribute to the minimization and report of negative impacts.

2.7 Conclusions and Outlook

This overview has revolved around eXplainable Artificial Intelligence, which has been identified in recent times as an utmost need for the adoption of ML methods in real-life applications. Our study has elaborated on this topic by first clarifying different concepts underlying model explainability, as well as by showing the diverse purposes that motivate the search for more interpretable ML methods. These conceptual remarks have served as a solid baseline for a systematic review of recent literature dealing with explainability, which has been approached from two different perspectives: 1) ML models that feature some degree of transparency, thereby interpretable to an extent by themselves; and 2) post-hoc XAI techniques devised to make ML models more interpretable. This literature analysis has yielded a global taxonomy of different proposals reported by the community, classifying them under uniform criteria. Given the prevalence of contributions dealing with the explainability of Deep Learning models, we have inspected in depth the literature dealing with this family of models, giving rise to an alternative taxonomy that connects more closely with the specific domains in which explainability can be realized for Deep Learning models.

We have moved our discussions beyond what has been made so far in the XAI realm toward the concept of Responsible AI, a paradigm that imposes a series of AI principles to be met when implementing AI models in practice, including fairness, transparency, and privacy. We have also discussed the implications of adopting XAI techniques in the context of data fusion, unveiling the potential of XAI to compromise the privacy of protected data involved in the fusion process. Implications of XAI in fairness have also been discussed in

detail. This vision of XAI as a core concept to ensure the aforementioned principles for Responsible AI is summarized graphically in Figure 2.11.

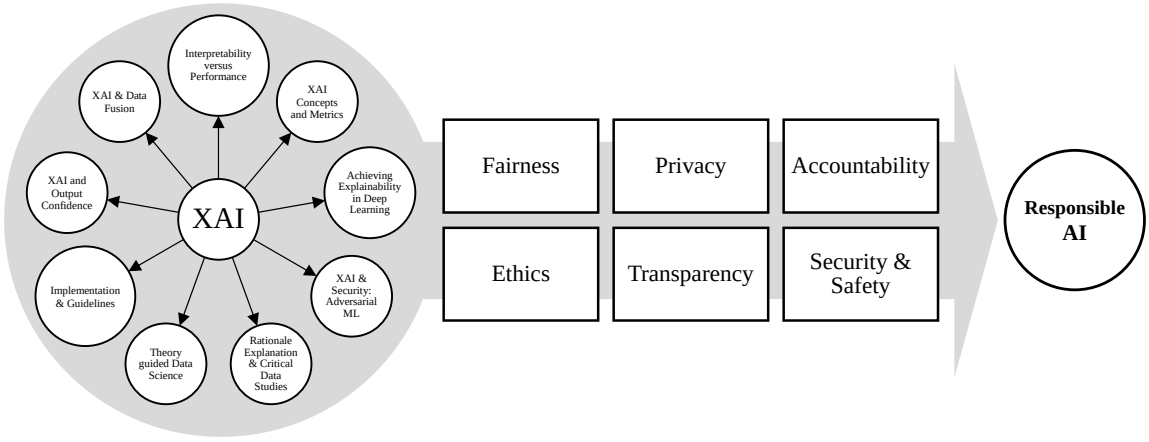


Figure 2.11: Summary of XAI challenges discussed in this overview and its impact on the principles for Responsible AI.

Our reflections about the future of XAI, conveyed in the discussions held throughout this work, agree on the compelling need for a proper understanding of the potentiality and caveats opened up by XAI techniques. It is our vision that model interpretability must be addressed jointly with requirements and constraints related to data privacy, model confidentiality, fairness and accountability. A responsible implementation and use of AI methods in organizations and institutions worldwide will be only guaranteed if all these AI principles are studied jointly.

The following chapter will attempt to fill a gap in the state of the art by proposing to push back the trade-off between explainability and performance with a framework combining an explainable base and a performing model.

Chapter 3

Neural-Symbolic reasoning for XAI

Many high-performance models suffer from a lack of interpretability. There has been an increasing influx of work on explainable artificial intelligence in order to disentangle what is meant and expected by XAI. In his paper, [13] highlighted major findings that should be considered when creating an explainable AI model. First, explanations are better when constrictive, meaning that a prerequisite for a good explanation is that it does not only indicate why the model made a decision X, but also why it made decision X rather than decision Y. The ability to refer to established reasoning rules allows symbolic methods to fulfill this property. It is also explained in Miller's article that probabilities are not as important as causal links in order to provide a satisfying explanation. Considering that black box models tend to process data in a quantitative manner, it would be necessary to translate the probabilistic results into qualitative notions containing causal links. Again, the use of symbols could carry this property as the use of a Knowledge Base such as an ontology can allow data to be processed directly in a qualitative way. In addition, they state that explanations are selective, meaning that focusing solely on the main causes of a decision-making process is sufficient. It is known that there is a trade-off between interpretability and accuracy [14], i.e., between the simplicity of the information given by the system on its internal functioning, and the exhaustiveness of this description. Considering that additional variables and equations must be introduced in order to test whether a correlation between two variables is genuine or spurious [402], being selective is less fast-forward for connectionist models than for symbolic ones. Finally, considering that a good explanation needs to influence the mental model of the user, i.e. the representation of the external reality using, among other things, symbols, it seems obvious that the use of the symbolic learning paradigm is appropriate to produce an explanation.

One of the goals of having interpretability in a model is to explain its reasoning by expressing it in a way that is understandable and readable by human beings, while

highlighting the biases learned by the model, in order to validate or invalidate its decision rationale [18]. It is customary to think that by focusing solely on performance, the systems will be increasingly opaque. This is true in the sense that there is a trade-off between the performance of a model and its transparency [11]. We consider that the advocacy for interpretability may lead to a generic performance improvement for 3 reasons: i) it will help ensure impartiality in decision-making, i.e. to highlight, and consequently, correct from bias in the training data-set, ii) interpretability facilitates the provision of robustness by highlighting potential adversarial perturbations that could change the prediction, and finally, iii) interpretability can act as an insurance that only meaningful variables infer the output, i.e., guaranteeing that an underlying truthful causality exists in the model reasoning. Combining the prediction capabilities of connectionist models with the transparency of symbolic ones could put aside the trade-off by increasing both the interpretability and the performance of AI models. Therefore, neural-symbolic interpretability can provide convincing explanations while keeping or improving generic performance.

How can we combine the performance of Deep Neural Networks (DNNs) and the interpretability of symbolic representations to bring fairness, accountability and transparency to Deep Learning? As it seems intuitive that the presence of a KB is useful to provide an explanation, how to use it to influence a network is not obvious because KBs use a very concrete formalism which is in opposition to the abstract features used by networks. In this chapter, we introduce the *neural-symbolic eXplainable AI framework*, that weaves a DNN and a Knowledge Base such that the prediction of a DNN is explained thanks to the guide of an internal KB. The goal of this framework is to explain the prediction of a DNN thanks to the influence of a KB. As the explanation is directly expressed in terms of the data the network was exposed to, it does not suffer from human-induced nor external sources of unfaithfulness. The DNN output is directly constrained by the extracted semantic knowledge in a way such that the prediction can be explained in natural language by leveraging description logics concepts and relations. In order to set the ground for the required components to produce neural-symbolic explanations, we will first frame the problem and the different building blocks of the framework in Section 3.2. We then propose an example of how to use the framework for an image captioning task on Section 3.3. We accompany this model with an example of its potential use, based on the image captioning method in [168].

3.1 Related Work: Neural-Symbolic interpretability

The use of background knowledge in the form of logical statements or constraints in knowledge bases has shown to not only improve explainability but also performance with

respect to purely data-driven approaches [323, 324, 325]. A positive side effect shown is that this hybrid approach provides robustness to the learning system when errors are present in the training data labels. Other approaches have shown to be able to jointly learn and reason with both symbolic and sub-symbolic representations and inference [347]. The interesting aspect is that this blend allows for expressive probabilistic-logical reasoning in an end-to-end fashion [326]. An example of use case is on dietary recommendations, where explanations are extracted from the reasoning behind (non deep but KB-based) models [327].

Data fusion approaches may thus consider endowing DL models with explainability by externalizing other domain information sources [403]. *Deep* formulations of classical machine learning models have been done, e.g. in Deep Kalman filters (DKFs) [331], Deep Variational Bayes Filters (DVBFs) [332], Structural Variational Autoencoders (SVAE) [333], or CRF as RNNs [334]. These approaches provide deep models with the interpretability inherent to probabilistic graphical models. For instance, SVAE combines probabilistic graphical models in the embedding space with neural networks to enhance the interpretability of DKFs. A particular example of classical ML model enhanced with its Deep Learning counterpart is Deep Nearest Neighbors DkNN [8], where the neighbors constitute a human-interpretable explanation of the predictions through a confidence term defined as *credibility*. The intuition is based on the rationalization of a DNN prediction based on evidence. This evidence consists of a characterization of confidence termed *credibility* that spans the hierarchy of representations within a DNN, that must be supported by the training data [8].

A different perspective on hybrid XAI models consists of enriching black-box models knowledge with that one of transparent ones, as proposed in [404] and further refined in [169]. It allows the network to express what is confident or confused about, in a context that helps to tackle bias [168]. Other examples of hybrid symbolic and sub-symbolic methods where a knowledge-based tool or graph-perspective enhances the neural (e.g., language [328]) model are in [329, 330].

Another hybrid approach consists of mapping an uninterpretable black-box system to a white-box *twin* that is more interpretable. For example, an opaque Artificial Neural Network (ANN) can be combined with a transparent Case Based Reasoning (CBR) system [338, 339]. In [340], the ANN (in this case a DNN) and the CBR (in this case a k-NN) are paired in order to improve interpretability while keeping the same accuracy. The *explanation by example* consists of analyzing the feature weights of the ANN which are then used in the CBR, in order to retrieve nearest-neighbor cases to explain the ANN's prediction. Description Logics [405] have successfully been used for enhancing deep learning models for image interpretation through the use of knowledge bases [406, 348]. Description logics can also help detect inconsistencies in automated knowledge representation and reasoning. An example for automated symbol design and interpretation is in [407]. Some XAI systems

consider counterfactual rule learning and causal signal extractions. Examples of rule learning approaches can include learning from noisy or unstructured data, or learning with constraints [408].

3.2 Neural-Symbolic computation for truly Explainable AI

One of the applications of XAI is correcting models to make them more robust against, for instance, biased data. However, there is also a risk of introducing a human-induced bias when an user is trying to make his model explainable, as it can depend on his background knowledge. Thus, it is necessary that the explanation of a DNN output is given directly by the model, for it to be faithful with respect to what the network actually learned [404].

Truly explainable models should directly integrate reasoning, in order to not leave explanation generation to the human user. In the model proposed by [404], the black box, i.e. the connectionist part, is giving the final output, while the KB is externally provided to the model. This allows the system to generate itself an explanation in natural language, thus linking the high level features identified by the model and the final output. It also highlights the logical path the model should have taken: since the KB is given by the user and (therefore we assume) cannot be incorrect, a reasoning error in the natural language explanation would signify a mistake in the black box between high level features and the final output. In addition, as stated in [404], the inclusion of reasoning in the model eliminates the potential corruption of the explanation that could arise from using external sources to justify the actual model we want to make explainable.

However, we can propose some adjustments in this architecture: the causal links given by the KB do not directly reflect the operations that took place in the black box, and it is therefore impossible to affirm that the model predicted this output for the reasons given in the natural language explanation. Since nothing connects the KB and the black box, therefore it is impossible to link the explanation and the predicted output. The objective of not leaving explanation generation to human is fulfilled, as the model formulates a line of reasoning, but the explanation given is not correct and does not have guarantees of having accurate provenance, as it only explains what the black box should have learned, and not what it actually learned. A possible adaptation would be to not use the output of the black box in the reasoner and solely use the high level features detected by the model so that the natural language explanation would match the reasoning that led to this result. This would mean truncating the potential of the black box. It is possible to link the reasoner and the black box by considering that the output is no longer the final result, but rather high level features. The model would then produce an explanation on what the system should

conclude when seeing those features but not why it detected those features. A last option to achieve an explanation of the model decision would be to directly populate the KB from the data. This would allow to provide an explanation in natural language directly from the black box, emphasizing in the meantime the model's reasoning errors and highlighting possible bias in the dataset or model. This is the option we propose pursuing as we believe it provides the most faithful explanation of how the model actually works. We summarize the different scenarios in Table 3.1.

		Knowledge Base Provenance	
		External	Internal
Model's final output origin	Reasoner	- No explanation about the black box - Does not highlight reasoning mistakes	**+ Explanation about the black box** **+ Highlights reasoning mistakes**
	Black box	- No explanation of the black box + Highlight reasoning mistakes	

Figure 3.1: Scenarios for Neural-Symbolic Reasoning, depending on the origin of the KB and the origin of the final output. The cell with text in bold is our contributed proposed model to achieve faithful neural-symbolic visual reasoning.

We derive two prerequisites that are necessary to create a truly reasoning AI: i) The KB must inherently emerge from the data used by the black-box model in order to conceptually (symbolically) reflect what the model should learn. ii) The symbolic part must influence the connectionist part to be able to explain the predictions of the model.

We propose an adaptation of the architecture in Fig. 3.2. Instead of externally providing a KB to complement the model, we propose to i) directly extract a knowledge base from the data and ii) reflect those rules in a black box by influencing learning according to perceived properties, e.g., by modifying initialization protocols, loss functions or hyperparameters. Therefore, the model's ultimate output would come from the reasoner but would be directly influenced by both the black box and the KB, i.e., it would not truncate the black box potency but reveal, as expected by an explanation, the biases learned by the model and lead to performance improvement while explaining in natural language its prediction.

3.2.1 Required Data

In order to create and populate a knowledge base it is necessary that some terminology axioms $TBox$ and assertion axioms $ABox$ can be extracted from some of the training labels. To this effect, we consider a subset of labels $Y_{\mathcal{KB}} \sqsubseteq \boldsymbol{Y}$, with $\boldsymbol{Y}$ the full set of labels from dataset D, from which a set of Resource Description Framework (RDF) triples of *subject*, *predicate* and *object* (s, p, o) can be extracted. This is the case, for example, when

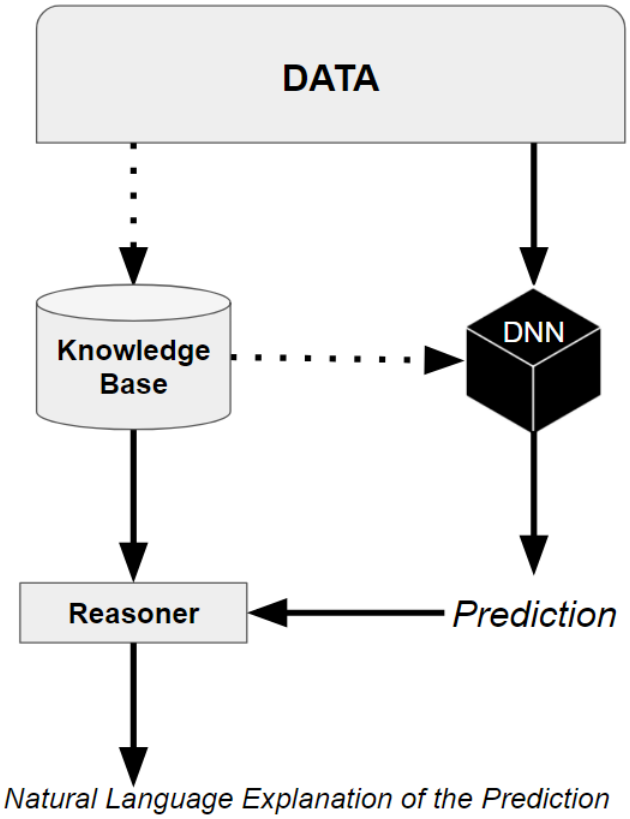


Figure 3.2: Proposed neural-symbolic explainable model extended from [404]: the black box model provides, along with its output, an explanation of its reasoning to highlight bias and improve performance. Our contribution with respect to [404] is the way we populate the KB directly from the data and the way we constraint the DNN thanks to the KB. It can be seen with the dashed lines.

the labels are in a text or caption form[1].

3.2.2 Extracting and populating the Knowledge Base

In Description Logics [411], a terminology box *TBox*, or schema extracted from the dataset, together with the class instanciations (in our case, that compose the datasets seen by the network), i.e., the assertional *ABox*, form the $\mathcal{KB} =< TBox, ABox >$. This particular

[1]This is also possible for plain *named* class labels (e.g., through using external lexicons or semantic parsers such as *DBPedia* [409] or *Wordnet* [410].

KB will serve as database, in form of a knowledge graph (KG) that interlinks individuals of different classes through roles or relationships with possible constraints or semantic restrictions.

The key insight is that knowledge base $\mathcal{KB} =< TBox, ABox >$ serves to guide the *NeSy XAI* framework to explain a DNN output. Concretely, $\mathcal{KB}$ must cover as many concepts and relations as relevantly possible , expressed in terms of each seen dataset D. Therefore, in order for $Y_{\mathcal{KB}}$ to be as semantically rich and accurate as $\boldsymbol{Y}$, we will use a Description Logics [411] formalization of the labels in $\boldsymbol{Y}$ in an ontological form to represent the DNN knowledge in the KB.

Definition: ***Extracting a knowledge base*** $\mathcal{KB} =< TBox, ABox >$ ***from a dataset*** D ***(AKBC Problem)***.

Let $y \in Y_{\mathcal{KB}}$ be a data label component. The *Automatic Knowledge Base Construction (AKBC) from a dataset* problem consists of finding a process $t : Y_{\mathcal{KB}} \rightarrow \{(s, p, o)\}$ able to *triplify* each data point following the RDF language[2]. The triplification process consists of extracting two entities or concepts, subject (s) and object (o), and a predicate (p) that connects them. This predicate expresses a relationship, i.e., a data property or an object property [412] among the subject and the object.

The AKBC problem in the NeSy framework for XAI thus, more generally, consists of finding a process $p : Y_{\mathcal{KB}} \rightarrow TBox \times ABox$ that automatically constructs a KB from labels $Y_{\mathcal{KB}}$ in D whose *TBox* is composed of at least hierarchical (*isA*), compositional (*partOf*), and attribute-based (*hasAttribute*) relations. Analogically, the *ABox* in $\mathcal{KB}$ must be uniquely composed by description logics assertional axioms solely extracted from applying the triplification process on datapoint tuples $(x, y) \in D$.

Example: Applying the KB extraction process to a neural network trained on *MSCOCO* and *KITTI* datasets could lead to obtain, e.g., $Y_{\mathcal{KB}} \sqsubseteq \boldsymbol{Y} = \{Cl \cup Ca \cup SM \cup BB \cup T\}$, where each set corresponds to a label type in the training dataset (Cl are regular class labels, Ca are captions, SM are segmentation masks, BB bounding boxes, and T text labels).

The area of AKBC is a field itself, and existing methods can be adapted to the different datasets and data formats a DNN ingests. Concrete examples include learning a KB through semantic parsing, using Markov Logic [413] or relational probabilistic models [414]. Therefore, we assume that a process p from which obtaining $\mathcal{KB} = p_{AKBC}(\mathcal{X}_i, \mathcal{Y}_i)$ exists (in order to abstract away the input data and dataset-dependent implementation details of the supervised learning problem).

The modelling assumption in this problem is thus that the TBox in $\mathcal{KB}$ must contain the union of concepts and relationships extractable from the samples and labels $(\mathcal{X}, \mathcal{Y})$ in every dataset D_i seen by the DNN. This is the assumption that will allow to generate NLE containing atttribute-based, hierarchical and compositional reasoning. e.g. for the latter,

[2]W3C standard model for data exchange: Resource Description Framework w3.org/RDF/

one NLE could be: "x is of type Y because elements contained by class Y are detected."

Example An example of extracted KB in simplified *Notation3*[3] format of the *subject, predicate, object* triples *(s,p,o)* is in Table 3.1 below.

Table 3.1: Examples of RDF triples contained in a KB terminological (TBox) and assertional (ABox) components [*hasPredLabel* = *hasPredictedLabel*]

KB	RDF *(s,p,o)* triple examples
TBox	`(Cactus Wren, isA, Wren)`
	`(Cactus Wren, hasCrest, False)`
	`(Cactus Wren, hasWing, Brown)`
ABox	`(img01.jpg, hasPredLabel, senior1)`
	`(senior1, isSittingOn, bench1)`
	`(senior1, isA, Senior)`
	`(bench1, isA, Bench)`

3.2.3 Constraining the Deep Neural Network

Since the objective of the framework is to explain the prediction of any DNN, no limitation is made on the type of network used. It is necessary, however, for the KB to constrain the DNN in order to be able to explain the latter's predictions. The main challenge is to do so while minimizing the modification of the DNN accuracy. Diverse mechanisms can be used to constrain the predictions of a neural network. We will show an example using constraints via the use of loss functions.

Given a dataset $D = \{\mathcal{X}, \mathcal{Y}\}$ where $\mathcal{X} \in \boldsymbol{X}$ is a set of features, and $\mathcal{Y} \in \boldsymbol{Y}$ a set of labels, we address supervised learning problems implemented with a black box predictor $f : \mathcal{X} \rightarrow \mathcal{Y}$ that assigns a target value $f(x) = \hat{y}$ for $x \in \mathcal{X}$ and $y \in \mathcal{Y}$.

Concretely, our problem consists of finding a function $f_{KB} : \mathcal{X} \times \mathcal{KB} \rightarrow \mathcal{Y}$ that assigns a target value $f_{KB}(x) = \hat{y'}$ with $\hat{y'}$ as close as possible to $\hat{y}$. As there is a well-known trade-off between the performance of a model and its transparency [11], it is worth noting that a DNN may lose accuracy due to being guided by the KB; however, we hypothesize the explainabilty would improve thanks to the compositional semantic knowledge of the KB. As a concrete example, Section 3.3 will show how constraining the loss functions through the KB can improve highlighting learned bias.

[3]Notation3 (N3) triple format for RDF: https://en.wikipedia.org/wiki/Notation3

3.2.4 Querying the Reasoner to generate a NLE

Once the KB has been extracted and the DNN trained, we need to be able to explain the prediction of any input fed to the network. The following steps will transform the prediction into a natural language explanation (NLE).

Definition: ***Natural language explanation (NLE) generation from a prediction problem***. Generating a NLE $e_{x,y}$ of a DNN output $\hat{y}'$ consists of finding a mapping among the two, i.e., implementing $NLE(m, \mathcal{KB}, x, y)$, where m is the trained DNN model and $\mathcal{KB}$ is the KB extracted from each dataset D used to train m.

In order to select relevant statements to compose the NLE $e_{x,y}$, this problem may involve other subproblems such as datatype learning [415], KG alignment [416] with existing and/or external knowledge sources [323] (e.g., to highlight biased or unfair neural network reasoning contradicting human commonsense reasoning [416]), or other neural models. Explanations may also be produced accounting for graph theory metrics (e.g., influence or novelty in the KG [329]), or via graph-to-text models [417].

Ontologies serve to extract a minimal set of covering models of interpretation from a KB that can explain the observations [418]. Since the set of one or more axiomatic actions (i.e., input x fed into the ABox) are in form of triples, these facilitate not only the generation of a NLE, but also performing further automated reasoning tasks. Among other description logic reasoner capabilities [419], these include tasks such as satisfiability, consistency checking [407], entailment, and many others.

Fig. 3.3 proposes the *NeSy for XAI ontology* underlying the KB. Its objective is twofold: first, to facilitate expressing provenance of a DNN output decision; secondly, conveying a realistic explanation of the output that is amenable of human understanding. For instance, the reification pattern shows how to associate a NLE to a given DNN prediction while preserving its provenance components. This feature facilitates tracking provenance but also tracking or accountability on the decision of the DNN.

The mapping done in function *NLE()* must be application-driven; however it must align with ontology building methodologies such as *TDKGC* (Test Driven KG Construction) [420] (where requirements for the KB ontology to help create e are expressed in form of query-answer pairs $T =< q, a >$. Other method to be used can be by employing competency questions[4], *OMQA* (Ontology Mediated Question Answering) [422], or *CQOA* (Competency Questions Ontology Authoring) [423].

Inspired by the ontology CQs design methodology [423], we formulate the basic kind of possible ontology-based NLEs that the *NLE()* mapping could generate. These require the use of a reasoner containing the *NeSy for XAI ontology* in Fig. 3.3, the KB extracted

[4]CQs: Question expressions an ontology must be able to answer (functional reqs. of the ontology) [421]. They can assess how suitable is a given graph for the purpose of an scenario.

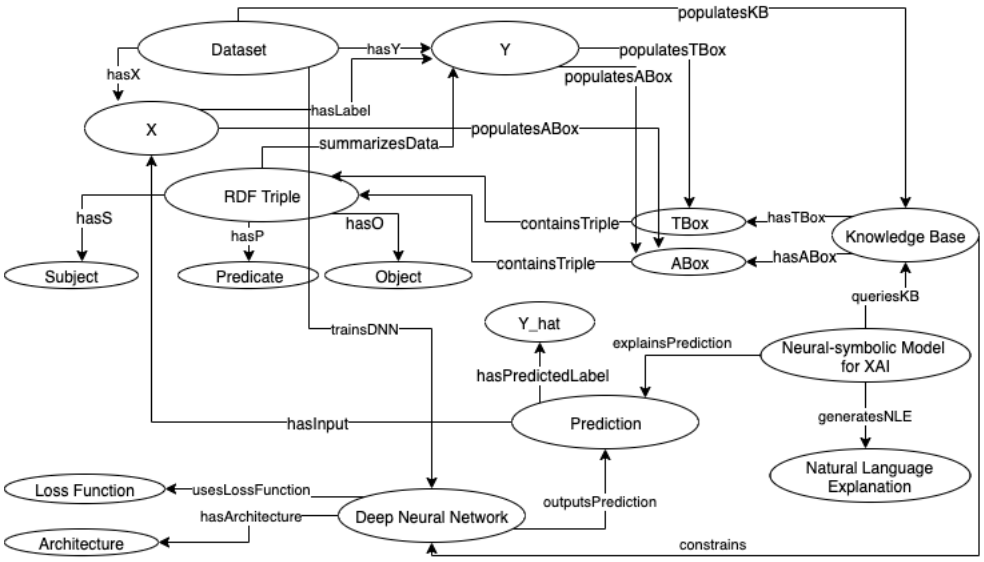


Figure 3.3: Proposed neural-symbolic model ontology expressing the relationships among a deep network components, the dataset it is trained with, and the automated knowledge base extracted to guide the generation of a natural language explanation (NLE).

during training, and queries involving the following OWL[5] entities and relationships:

- C = Concept (ontology class).
- PC = *Part-of* Concept (i.e., *pc* where the triple `(pc, isPartOf, c)` $\in TBox$ is an object property axiom (whose inverse is `(c, hasPart, pc)`).
- UC = Upper Concept (i.e., *uc* such that the triple `(c, isA, uc)` $\in TBox$.
- LC = Lower Concept (i.e., *lc* such that the triple `(lc, isA, c)` $\in TBox$.
- A = Attribute (i.e., *a* such that the triple `(c, hasAttribute, a)` $\in TBox$.

Algorithm 1 summarizes the different steps needed to use the neural-symbolic XAI framework in an end-to-end manner. It shows how to go from a dataset to an explained prediction of a DNN's output.

[5]Web Ontology Language (OWL) [424]

Type of explanation	**Explanation Template**	**NLE Example**
Class inheritance `is-a`-based explanations	It is a [UC] because I can recognize a [LC].	*It is a person because I can recognize a senior.*
`partOf`-based explanations	It is a [C] because I can recognize a [PC].	*It is a car because I can recognize a wheel.*
Contextual/causal explanations	It is [C1] because I can recognize [C2].	*It is a husky because I recognize snow.*
Attribute-based explanations (based on object/data properties)	It is a [C] because I can recognize a [A].	*It is a Downy Woodpecker because it has a red spot on its crown.*

Figure 3.4: Explanation types and examples of natural language explanations (NLEs) to be generated by the NeSy for XAI framework. These are designed to respond to ontology driven competency questions (CQs) [423]. [C = Concept, PC = *Part-of* Concept, UC = Upper Concept (superclass), LC = Lower Concept (subclass), A = Attribute].

Algorithm 1 NeSy Computation for XAI: End-to-end model to explain an output

Input: Dataset $D = (\mathcal{X}, \mathcal{Y})$, a DNN, and a pair (x, y).
Output: a NLE of prediction $\hat{y}'$: $e_{x,y}$
1: Create a sub-set of labels that can be triplified
for all $y \in \boldsymbol{Y}$ **do**
  if $p_{AKBC}(y) \neq \emptyset$ **then**
    $Y_{KB}.append(y)$
  end if
end for
2: Populate the Knowledge Base from labels in D
$KB \leftarrow p_{AKBC}(Y_{KB})$
3: Constrain the training of the DNN via the KB
$\hat{y}' \leftarrow f_{KB}(X, KB)$
4: Query the reasoner with the pertinent competency questions $\mathcal{CQ}$ involving the prediction's entities and relations to obtain an explanation.
for all $QC_i \in \mathcal{CQ}$ **do**
  $e \leftarrow query(KB, CQ, \hat{y}')$
end for
  return e

3.3 Use case: Explaining Image Captioning Outputs

As it seems intuitive that the presence of a KB is useful to provide an explanation, how to use it to influence a network raises some questions. The role of the reasoner linking a KB with its black box is also a post-processing step to be determined and studied further.

One barrier to transparency is a "mismatch between the mathematical optimization using high-dimensionality characteristics of machine learning and the demands of human-scale reasoning and styles of interpretation" [425]. With the objective of reducing this gap, and inspired by the work of [168], we hypothesize that the use of loss functions that have a concrete and more graspable perceptible meaning could make it easier to provide an explanation than a classic non intuitive cross-entropy. In [168], authors introduce two new loss functions: the "Appearance Confusion Loss" and the "Confident Loss" in order to counter-balance gender bias during an image captioning process. The Appearance Confusion Loss is based on the fact that *for an image devoid of gender information, the probability of predicting man or woman should be equal*; and the Confident Loss exists to encourage the model to predict gender words correctly when gender evidence is present.

In order to show an example of how to use the framework, we address an image captioning problem on MSCOCO dataset [426] inspired by [168]. It consists in training a neural network using images *I*, image captions *S*, and image segmentation annotation masks *M*, with a neural image caption network [427] as a base. In this problem, $\boldsymbol{X} = I$ and $\boldsymbol{Y} = \{S, M\}$. In this particular example, the subset of labels $Y_{\mathcal{KB}} \sqsubseteq \boldsymbol{Y}$ is similar to $\boldsymbol{Y}$ as image captions *S* and image segmentation annotation masks *M* can be used together in order to populate the KB. The architecture is shown in Fig 3.5

Populating the Knowledge Base

As captions can be considered as raw text and segmentation masks certify the existence of an entity, the process $t : Y_{\mathcal{KB}} \rightarrow (s, p, o)$ chosen consists of an information extraction problem [428]. This makes it possible to obtain relationships such as shown in Table 3.2.

Constraining the Deep Neural Network through a Knowledge Base

We decide the function $f_{KB} : \mathcal{X} \times \mathcal{KB} \rightarrow \mathcal{Y}$ to be a loss function modification, since these loses were already designed in [168].

As shown in Table 3.2, there is numerous *is-a* predicates bounding different *subjects* and the *Person* object. The *AKBC* process allows us to extract a list $B_{person} =$ [*Man, Teenager, Boy, Senior*] where ontology concepts, i.e., classes *Man*, *Teenager*, *Boy* and *Senior* are subclasses of the class *Person*. This extracted list of basic concepts will

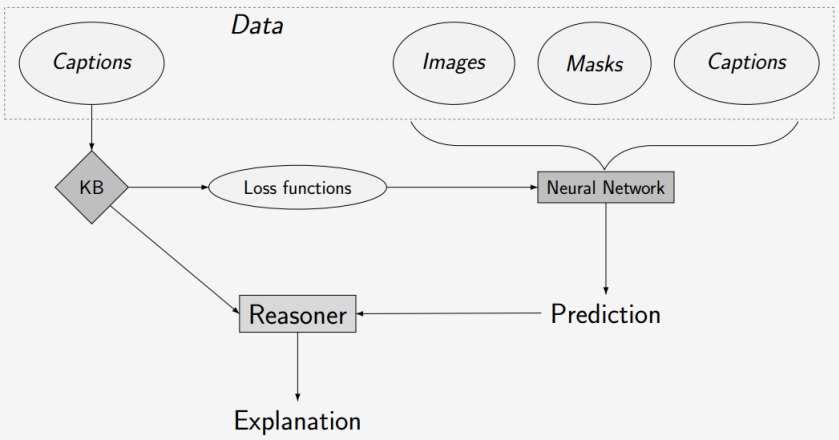


Figure 3.5: Captions are used to create the KB thanks to a Continuous Bag of Word model. The KB influences our Neural Network by modifying its losses, depending on the relationship between words. Images, segmentation masks, captions and the new losses are used to train the neural network. The prediction is used to retrace the KB in order to give an explanation of the outcome.

constitute a set of classes in our KB for which the model will have to hesitate when trying to predict one word rather than another.

We want to force the model to hesitate about which ontological sub-class it should predict when a class is present on an image. In order to not make any mistakes, but to be confident enough to nevertheless predict an ontological sub-class rather than a super-class, in this case the objective is to be as specific as possible. We use masked images I', where the information relevant to making a decision, such as the interior of a segmentation mask for a human in the image (if we are trying to determine whether or not the human classified is a senior or not), is removed. To ensure equiprobability among the different words in B_{word} when there is no appropriate information for the system to predict a specific word, but rather its generic category, we use a confusion function [168]. We denote confusion function C, which operates over the predicted distribution of words $p(\tilde{w}_t)$, to the following function:

$$C(\tilde{w}_t, I') = \sum_{b \in B_{word}} (p(\tilde{w}_t = b | w_{0:t-1}, I') - \frac{1}{J})^2 \tag{3.3.1}$$

where J is the length of B_{word}. As we try to minimize $C(\tilde{w}_t, I')$, we have a sum of squares that tends toward zero. Given that if a sum of squares is zero, each term must be zero, each probability tends to be equal. As in [168], we define the confusion loss

Table 3.2: Examples of triples contained in the Knowledge Base terminological box (TBox) after being populated.

KB	RDF *(s,p,o)* triple examples

TBox	`(Boy, isA, Person)`
	`(Girl, isA, Person)`
	`(Man, isA, Person)`
	`(Woman, isA, Person)`
	`(Person, isA, Thing)`

$\mathcal{L}^{Confusion}$ as:

$$\mathcal{L}^{Confusion} = \frac{1}{N}\sum_{n=0}^{N}\sum_{t=0}^{T}\mathbb{1}(w_t \in B_{word})C(\tilde{w}_t, I'), \tag{3.3.2}$$

with $\mathbb{1}$ an indicator variable that denotes whether or not w_t is a bias-prone word, N the batch size, and T the number of words in the given sentence. As we want the model to be confident about its prediction when there is an appropriate information on the image, this time we use complete (i.e., non masked) images I as input instead of masked ones I'. With j the index of word b in list B_{word}, we have the confidence function F^j.

$$F^j(\tilde{w}_t, I) = \frac{\sum_{b \in B_{word} \setminus b_j} p(\tilde{w}_t = b | w_{0:t-1}, I)}{p(\tilde{w}_t = b_j | w_{0:t-1}, I) + \epsilon} \tag{3.3.3}$$

We add an ϵ for numerical stability. F^j will tend towards zero if $p(\tilde{w}_t = b_j)$ dominates the sum of the predicted distribution of every other *bias-prone* word.

We use F^j to define the confident loss $\mathcal{L}^{Confidence}$:

$$\mathcal{L}^{Confidence} = \frac{1}{N}\sum_{n=0}^{N}\sum_{t=0}^{T}\sum_{j=1}^{J}(\mathbb{1}(w_t = b_j)F^j(\tilde{w}_t, I)) \tag{3.3.4}$$

By adding a standard cross-entropy loss $\mathcal{L}^{CE}$ to non-bias-prone words, we obtain a model able to use context priors when there is no interchangeable word for the predicted one, and to be confused/confident when the question arises, thanks to the loss function $\mathcal{L}$:

$$\mathcal{L} = \alpha\mathcal{L}^{CE} + \beta\mathcal{L}^{Confidence} + \mu\mathcal{L}^{Confusion} \tag{3.3.5}$$

with α, β and μ hyper-parameters.

After being trained with these losses, the DNN is able to make predictions $\hat{y}'$ that can be used to do reverse engineering on the KB by querying it.

Querying the Reasoner

In order to query the reasoner, we use the explanation templates shown in Figure 3.4. As in our example the only type of explanation extractable from the triplification process is *class inheritance-based explanations*, we query the reasoner using triples (uc, isA, lc) and (lc, isA, uc) to produce explanations following the template *It is a [UC] because I can recognize a [LC]* (where upper concept UC is a superclass of lower concept LC). By adding to the generated explanation the definition of our constraints on the DNN, we obtain complete explanations as presented in Fig. 3.6 and Fig 3.7.

This model, when applied to image captioning or object recognition tasks, has several advantages: i) it detects the provenance of bias in a black box model such as a neural network, ii) gives an unbiased prediction for which the context has not been overused, and iii) gives an explanation in natural language on the neural network's functioning; particularly, on its loss-based optimization procedure.

3.4 Results and Discussion

We compare ourselves to Burns et al. [168] on gender a captioning task but adding 2 subclasses to the training, boys and girls. We focus on the captioning of men and women. The dataset is MSCOCO-Bias and Balanced, composed of images from MSCOCO which are labeled as "man" or "woman" with respectively a 1:3 and 1:1 woman to man ratio. The Baseline is the Equalizer model [168], a linear combination of a confusion loss, a confidence loss and a cross-entropy loss. We measure the performance for each class ("men" and women"). We also count instances where men and women are classified as "People", which is not wrong but is not the most accurate answer possible. Results shown in Table 3.3 demonstrate that our model provides a very accurate prediction, making the model less hesitant and less likely to label women and men as "persons".

Models combining connectionism and symbolism are not widely represented in the state of the art of XAI. These paradigms are rarely combined when providing explanations. The use of a symbolic basis with a neural network can provide explanations close to the functioning of human reasoning while maintaining the state-of-the-art performance at the same time. We build upon [404] and extend [168] to further characterize what a neural-symbolic explainable model could output. We propose a model endowed with a non-external KB, i.e., directly built on the learning data of a neural network, that allows to influence its learning and to correct bias thoroughly, while giving a fair explanation

Model (M): A **[person]** on a bench

User (U): Why is it a person ?

M: I recognise a **[person]** and I know from my KB that a **[person]** can be a **[man]**, a **[woman]**, a **[boy]** or a **[girl]**. I was **[not able]** to decide between them so I am **[confused]** over the fact that it's one of them but I'm sure this is a **[person]**

Figure 3.6: Example of prediction and explanation where the model is **not able** to differentiate the subclasses of the class *Person*.

Model (M): A **[man]** on a bench

User (U): Why is it a man ?

M: I recognise a **[person]** and I know from my KB that a **[person]** can be a **[man]**, a **[woman]**, a **[boy]** or a **[girl]**. I was **[able]** to decide between them so I am **[confident]** over the fact that this **[person]** is a **[man]**

Figure 3.7: Example of a prediction and explanation where the model is **able** to differentiate the subclasses of class *Person*.

Model	Women Correct	Women Incorrect	Women Predicted as "Person"	Men Correct	Men Incorrect	Men Predicted as "Person"
Neural-Symbolic Framework	**62.48%**	16.46%	**21.06%**	**69.58%**	**4.96%**	**25.47%**
Equalizer [168]	59.98%	**13.80%**	26.22%	62.47%	5.63%	31.90%

Table 3.3: Mean prediction performance in both MSCOCO-Bias and Balanced for the Neural-Symbolic and the Equalizer model.

from its predictions. As the user or expert external knowledge does not interfere the predictions in the explanation process, it constitutes a truly explainable model that is faithful to communicate the reasoning behind its output decisions.

One of the concerns, however, is the number of predicates that can be used through the loss constraint, and the quality of the generated explanations: we only know that the model is confuse/is confident, and only the *is-a* predicate is being used. In the next chapter, we introduce the Greybox XAI framework which follows the same philosophy as this framework but adds the possibility to use the *is part-of* predicate and thus to produce explanations based on the membership of the subpart of an object to an object.

Chapter 4

Greybox XAI: a Neural-Symbolic learning framework for interpretable image classification

Although Deep Neural Networks have great generalization and prediction capabilities, their functioning does not allow a detailed explanation of their behavior. Opaque deep learning models are increasingly used to make important predictions in critical environments, and the danger is that they make and use predictions that cannot be justified or legitimized. Several eXplainable Artificial Intelligence methods that separate explanations from machine learning models have emerged, but have shortcomings in faithfulness to the model actual functioning and robustness. As a result, there is a widespread agreement on the importance of endowing Deep Learning models with explanatory capabilities so that they can themselves provide an answer to why a particular prediction was made.

Deep Learning models suffer from two kind of bias. The first one is a learning bias, when the data is skewed. This can happen when associations of concepts are over- or under-represented in the training set. For example there were cases of a dataset with women under-represented in offices compared to men, leading a captioning algorithm to assume that a person in an office was necessarily a man while it could also be a woman [429]. One of the applications of XAI is highlighting this bias. The second type of bias is the human induced one, when using common sense knowledge about the world to explain the output of a DNN or when using particular parameters, architectures or loss functions to model a problem [404]. One of the goals of XAI is to correct this bias, for example by forcing the model to be careful when a decision is prone to bias as in the case of gender classification. [169].

A large number of methods for model probing have emerged in recent years. Some have

the advantage of being model-agnostic, i.e. separating the explanation from the machine learning model. This has the advantage of providing flexibility to the user as tools are available to extract explanatory elements from each model [70]. Some of these methods, the best known of which are LIME [430] and SHAP [224], are based on the use of surrogate models. These proxy models will locally mimic the behaviour of the black-box in order to explain individual predictions. While this has the advantage of being easy to use, there are problems of robustness [431, 432]. Moreover, it is not possible to have a global view of the model's behaviour since the explanation is local.

In image recognition, another family of methods widely used is based on visualization. They express an explanation by highlighting characteristics of the image that objectively influence the output of a DNN [433]. The best known of them, Grad-CAM [291], creates class activation map using the gradients of the DNN's output with respect to the last convolutional layer. This provides a visual explanation easy to understand as it allows recognizing the important regions of the image. However it is difficult to know whether an explanation is correct, in the sense that a human non-expert in the field does not necessarily know what the important points of an image are, and a part of the evaluation is subjective. Furthermore, it has been shown that some of the most used methods are insensitive to model and data [293]. In addition, there is also a risk of introducing a human-induced bias when a user is trying to interpret the visual explanation. His or her understanding would depend on his or her own background knowledge. Thus, it is necessary that the explanatory elements of an AI model come directly from the data seen by the network, for it to be faithful with respect to what it actually learned [169].

One of the goals of having interpretability in a model is to explain its reasoning by expressing it in a way that is understandable and readable by human beings, while highlighting the biases learned by the model, in order to validate or invalidate its decision rationale [18]. There is a trade-off between the performance of a model and its transparency [11] but it is also possible to consider that the advocacy for interpretability may lead to a generic performance improvement for 3 reasons: i) it will help ensure impartiality in decision-making, i.e. to highlight, and consequently, correct from bias in the training data-set, ii) interpretability facilitates the provision of robustness by highlighting potential adversarial perturbations that could change the prediction, and finally, iii) interpretability can act as an insurance that only meaningful variables infer the output, i.e., guaranteeing that an underlying truthful causality exists in the model reasoning. Combining the prediction capabilities of connectionist models with the transparency of symbolic ones could put aside the trade-off by increasing either the interpretability or the performance of AI models, the challenge being to increase one without sacrificing too much of the other. It has been proven that using background knowledge within a DNN can bring robustness to the learning system [323, 324, 325]. The use of a Knowledge Base to learn and reason with

symbolic representations has the advantage of promoting the production of explanations while making a prediction [327]. The ability to refer to established reasoning rules allows symbolic methods to fulfill this property.

In order to obtain a model that meets the above criteria, we introduce the *Greybox XAI framework*. This new architecture is transparent by design when used for an image classification task. It combines an encoder-decoder used for the creation of an *Explainable Latent Space* which is then used by a logistic regression. The *Explainable Latent Space* allows knowing for which reasons an image has been classified in a certain way with the help of logistic regression. Moreover, we propose a formalization of the notion of explanation and we pose definitions allowing to judge its quality.

The contribution of this section is threefold:

- A theory of explainability of deep learning models to qualify what is a "good" explanation.
- An explainable by design compositional framework called *Greybox XAI*.
- We show that this new framework provides state-of-the-art results on an image classification task regarding the explainability/accuracy trade-off in various datasets, as its accuracy is close to the existing models while being more explainable.

This section is organized as follows: first we present the literature around XAI and part-based classifiers in Section 4.1. We describe our framework in Section 4.3 and we illustrate its use by experiments on several datasets in Section 4.4.

4.1 Related Work

The literature [434, 435, 436] distinguishes Deep Learning's XAI methods into two categories: transparent models and opaque models that need to be explained thanks to post-hoc methods. As our model is a composition of a transparent model and an opaque model, we will put a particular focus on compositional models.

Compositionality in computer vision refers to the ability to represent complex concepts by combining simpler parts [437, 438]. Compositionality is a desirable property for CNNs as it can improve generalization by encouraging networks to form representations that disentangle the prediction of objects from their surroundings and from each other [439]. For example, handwritten symbols can be learned from only a few examples using a compositional representation of strokes [440]. The compositionality of neural networks is also seen as key to the integration of symbolism and connectionism [441, 442].

Part-based object recognition is an example of semantic compositionality and a classical paradigm where the idea is to collect information at the local level in order to make a global classification. In [443], the authors propose a pipeline that first groups pixels into superpixels, then performs a superpixel-level segmentation, converts this segmentation into a feature vector, and finally classifies the global image thanks to this feature vector. A similar method is proposed by [444], extending it to 3D data. Here, the idea is to classify a part of the image into a predefined class and then use these intermediate predictions to create a classification of the whole image. The authors of [445] also define intermediate-level features that capture local structures such as vertical or horizontal edges, hair filters, and so on. However, they are closer to dictionary learning than to the approach we propose in this paper.

One of the best known models for object part recognition is [446]. It provides object recognition based on mixtures of deformable part models with multiple scales based on data mining of hard negative examples with partially labeled data to train a latent SVM. The evaluation is performed in the PASCAL object detection challenge (PASCAL VOC benchmark [447]).

Semi-supervised methods have been developed more recently, such as [448]. They propose a two-stage neural architecture for fine-grained image classification supported by local detections. The idea is that positive proposal regions highlight different complementary information and that all of this information should be used. For this purpose, an unsupervised recognition model is first built by alternately applying a CRF and a Mask-RCNN (considering an initial approximation with CAM). Then, the recognition model and the positive region proposal are fed to a bidirectional LSTM, which generates a meaningful feature vector that collects information about all regions and is then able to classify the image. This can be considered as unsupervised part-based classification.

Finally,[449] proposes a methodology designed to learn both symbolic and deep representations. It involves a compositional convolutional neural network that makes use of symbolic representations called EXPLANet and SHAP-Backprop, an explainable AI-informed training procedure that corrects and guides the DL process to align with such symbolic representations in form of knowledge graphs. To the best of our knowledge, this model represents the state of the art in terms of compositional learning models.

As introduced in [450], there is a need for causability in certain domains like the medical field for example. Causability is the measurable extent to which an explanation to a human expert achieves a specified level of causal understanding [451]. This notion refers to usability and must not be confused with causality as the relationship between cause and effect [161]. Causability can be measured with the System Causability Scale, a system to measure the quality of explanations based on causability and usability [452].

4.1.1 Existing XAI formalism

Self-Explaining Neural Networks were first defined in [453]. They propose a definition of what a self-explaining model is. They make possible to ensure that any model is explainable if it follows the criteria they have established. [454] established an objective metric for XAI and showcase its robustness in an user-study. [455] proposes a metric [456] proposes a comprehensive taxonomy for XAI, showing and explaining in a clear manner the different terms used in XAI.

4.2 Clarifying the concept of Explainability

As stated in [434], explainability can be considered as an active characteristic of a model, which refers to any action or procedure implemented in order to clarify its internal functions. The explainability of a model thus denotes its capacity to produce an explanation. In order to make a non-transparent model explainable, many post-hoc methods were designed. Post-hoc methods are used on a model after its training and are designed to probe the model in order to improve its explainability.

An accepted definition of explainability is that given a certain audience, explainability refers to the details and reasons a model gives to make its functioning clear or easy to understand [434]. One difficulty arising from this definition is the notion of audience. It implies that an explanation is a social phenomenon [13], which makes its quality to be subjectively assessed and therefore, difficult to measure mathematically and systematically.

An explanation is thus considered to be a transfer of knowledge between an explainer and an explainee. The explainer is the explanation produced by the model and the explainee is the human user receiving this explanation. Since the objective is that the explanation is understood by the user, a second notion appears in addition to the explanation: its interpretation. As a matter of fact, depending on the audience, the explanation will not be perceived in the same way. This is linked to the fact that we do not all have the same beliefs, knowledge or understanding of concepts. Two randomly chosen persons can therefore have a different interpretation of the same and unique explanation given by a model.

Thus, we separate our evaluation of the explainability of a model into two parts: 1/ its explainability, i.e. its capacity to produce an explanation and 2/ the interpretability of this explanation, i.e. how understandable is this explanation for the audience.

While most of the different terms used in explainable AI have been widely debated in the literature, the one about *What* constitutes an explanation has not been widely mathematized. We propose a formalization of the notion of explanation, inspired by [453] in order to establish objective criteria to affirm that an explanation is "good" or not. Whether it comes from the transparency of a model or from a post-hoc method applied on an opaque model,

an explanation must meet certain indispensable characteristics to be considered as a "good" explanation.

4.2.1 Explanation Formalisation for an image classification problem

Let us denote $E = \{(e)|e \in \{0,1\}^*\}$ a set of explanations e. This set is binary in order to be able to encode any type of communication because an explanation can be given in various forms: a text, an image, a graph, etc.

Definition 1. *Let $f : \mathcal{X} \to \mathcal{Y}$ be a classification model with $\mathcal{X}$ the input space and $\mathcal{Y}$ the label space. The explanation function Φ on a $f(x)$ prediction with $x \in \mathcal{X}$ is defined by:*

$$\Phi : \mathcal{Y} \to E \tag{4.2.1}$$

$$f(x) \to \Phi(f(x)) \tag{4.2.2}$$

On the basis of this definition of an explanation function, we define several axioms characterizing an explanation. These axioms intend to formally qualify what a "good" explanation is, following desired properties. Based on the literature [434, 404, 13] and the above definitions, we highlight 3 properties that we consider necessary to obtain a "good" explanation:

1. **Objectivity**. The point of having an explanation that is as objective as possible is to minimize the amount of subjectivity a human might have in interpreting that explanation. This allows for an unbiased explanation that can be understood in the same way by two different users. In order for an explanation to be more objective, we consider that it must be expressed in such a way that it is understood in the same way by the majority of members of a given audience. The real world contains objects and we want compact representations of those objects [457]. We assume this can be obtained in an explanation by the use of logical semantics, using symbols and relations that can be conceptualized by the human user of the explanation. This implies the use of ontologies, specifying what individuals (things, objects) and relationships are assumed to exist and what terminology is used for them.

   Definition 2. *An explanation e of a classification model $f : \mathcal{X} \to \mathcal{Y}$ is said to be more objective if e does contain symbols and/or relationships.*

   Example 4.1 (Objectivity:)**.** *Figure 4.1 shows the example of one explanation making use of symbols and/or relationships and another one not making use of them. A less subjective explanation minimizes the amount of interpretation left to the explainee*

because it uses symbols (words) commonly employed to represent objects. Moreover the subjective explanation based on a visualization of the attention areas of the model leaves a lot to the explainee's interpretation. From one user to another, some will say that the hot area is the head of the rabbit while others will talk about the color of its muzzle, the carrot strand it is holding or its eyes.

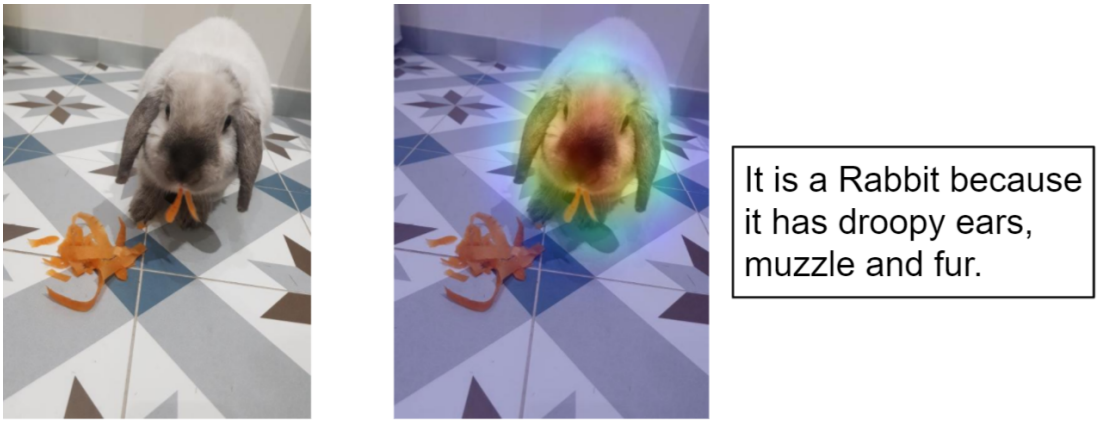


Figure 4.1: Subjective and less subjective/more objective example of explanations. The subjective explanation is a superimposed visualization of Grad-CAM's heatmap and the input image, showing the model mostly used the center-right of the image (where the head of the rabbit is) in order to make its prediction. The more objective explanation is a textual explanation using attributes detected on the rabbit to categorize and describe it.

2. **Intrinsicality**. The complete explanation of a prediction should come directly from the model (or its intrinsic elements) that produced the prediction. In order for the explanation to be totally faithful to what happened in the model, it is necessary that only the inputs, parameters and operations present in the model that we are trying to explain are used. This is essential to ensure that the explanation that is given is what actually happened in the model during its inference rather than the expected behavior. The value of having an intrinsic explanation is to be sure that the explanation exactly describes how the model works, rather than an approximate or desired operation. As a matter of fact, if the explanation depends on something that is not related to the model we wish to explain, it is impossible to ensure that this explanation does not distort the real reasons for which a decision was taken.

   Definition 3. *An explanation e of a classification model $f : \mathcal{X} \to \mathcal{Y}$ is said to be intrinsic if e only depend of elements, parameters and operations present in f, $\mathcal{X}$ or $\mathcal{Y}$.*

Example 4.2 (Intrinsicality)**.** *In order to produce an explanation for a black box model f, the post-hoc method LIME [430] generates a new dataset consisting of perturbed samples and the corresponding predictions of f. On this new dataset, LIME trains an interpretable model h, which is weighted by the proximity of the perturbed samples to the instance of interest. The prediction of the model h should be a good approximation of the predictions of the model f locally, but it does not have to be a good global approximation of the model f. The produced explanation can be expressed as follows:*

$$e = \Phi(f(x)) = \arg\min_{h} \mathcal{L}(f, h, \pi_x) + \Omega(h) \tag{4.2.3}$$

with $\mathcal{L}(f, h, \pi_x)$ the local fidelity, i.e. how close the predictions from h are close to the predictions from f. The proximity measure π_x defines how large is the neighborhood around the explained instance. Therefore, explanation e does not only depend of elements, parameters and operations present in f, $\mathcal{X}$ or $\mathcal{Y}$ since the explanation depends on the surrogate model h. Consequently, this explanation is not intrinsic. It is the prediction of the model h that is explained, not that of the model f.

Example 4.3 (Intrinsicality)**.** *In opposition, the explanation resulting from a linear regression h can be considered as intrinsic because the learned relationships between the inputs and the labels can be written as follows:*

$$e = \Phi(f(x_i)) = \theta_f \times x_i \tag{4.2.4}$$

with θ_f the set of trainable parameters of f and x_i an instance.

3. **Validity and Completeness**. An explanation of a prediction must be valid, meaning that it should assert that the model is Right (or wrong) for the Right Reason (RRR). It must show that the functioning of the model is consistent, that it is not biased by the training data. The explanation should be similar to what an expert in the field would give. To this notion of *validity* we could add a desideratum notion of *completeness*: an explanation could be judged as incomplete if it does not contain enough valid elements in its constitution. However, it has been established in the literature that a "good" explanation is selective [13]. Selectivity means that humans are adept at selecting a few causes from a sometimes infinite number of causes. It is therefore necessary to talk only about the few major causes.

Definition 4. *Given a field expert human being $h : \mathcal{X} \rightarrow \mathcal{Y}$, we define E_{valid} as the set of valid explanations $\Phi : \mathcal{Y} \rightarrow E_{valid}$*

Definition 5. *An explanation e of a classification model $f : \mathcal{X} \rightarrow \mathcal{Y}$ is said to be valid if $e \in E_{valid}$ and complete if e is discriminative enough.*

Example 4.4 (Validity)**.** *To illustrate this, we take the example of the image 4.2 of a ram rabbit that we see in its entirety, from the front, and that the model classifies as a rabbit. We can say that the explanation is valid if it contains elements explaining why it is rabbit, i.e. the ones that an expert looking at the image would use to justify the fact that it is a ram rabbit.*

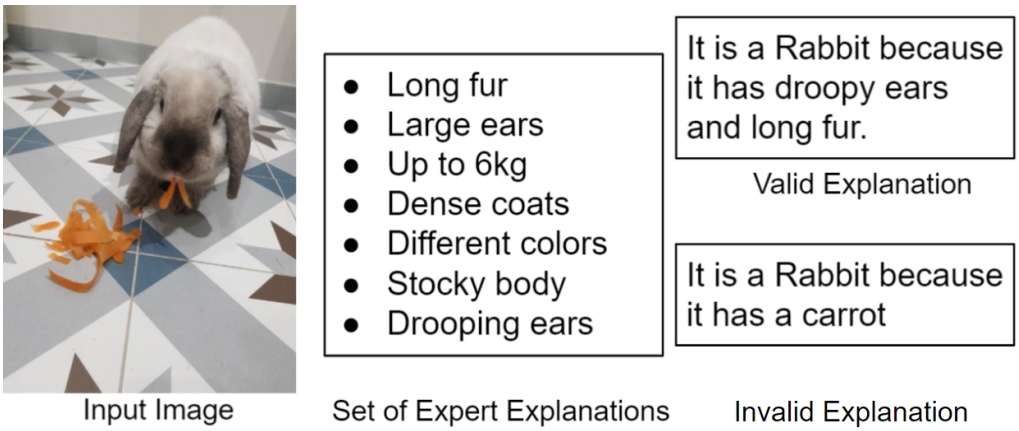


Figure 4.2: Valid and invalid examples of explanations. The valid explanation contains elements that are present in the expert explanation set and that are therefore good reasons to justify why a rabbit is present in the picture. On the contrary, the invalid explanation contains elements out of the expert explanation set.

Example 4.5 (Completeness)**.** *To illustrate the notion of completeness, closely related to validity, we take the example of the image 4.3 of a ram rabbit that we see in its entirety, from the front, and that the model classifies as a rabbit. We can say that the explanation is complete if it contains enough elements explaining why it is rabbit, i.e. some of the most important that an expert looking at the image would notice and use to justify the fact that it is a ram rabbit. Technically, it is not "wrong" to say that a rabbit can be of different colors. However, it is not a discriminating element to recognize a ram rabbit and it would be improbable to see a human giving this explanation. If this justification on the colors had been accompanied by very characteristic elements of the rabbit like its big droopy ears, the explanation would have been complete.*

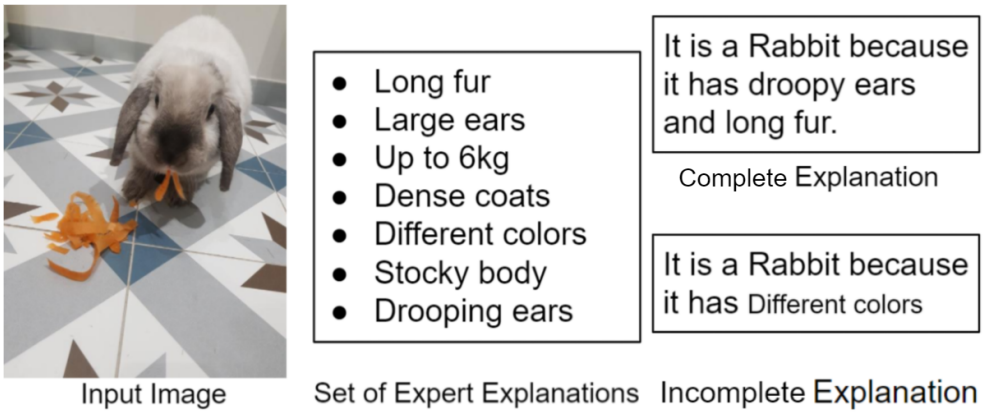


Figure 4.3: Valid and invalid examples of explanations. The valid explanation contains elements that are present in the expert explanation set and that are therefore good reasons to justify why a rabbit is present in the picture. On the contrary, the invalid explanation contains elements out of the expert explanation set.

4.3 Greybox XAI framework

In this section we present the *Greybox XAI framework*. It is designed to be transparent according to the definition of transparency proposed above. This framework is also made to produce "good" explanations in relation to the 3 criteria of objectivity, intrinsicality and validity. The goal of this framework is to perform compositional image classification and to explain its predictions by the different *parts-of* the object that has been classified. It consists of two separately trained models:

- A Deep Neural Network trained to predict a segmentation map from an RGB image input. Its purpose is to detect the different *parts-of* objects that constitute the image.
- A transparent model trained to predict an object, using as input a vector encoding the presence and absence of *parts-of* objects.

These two models are linked in a sequential manner: the output of the DNN is transformed into a vector serving as input to the transparent model. We call the space in which the transformation is carried out the *Explainable Latent Space*. In this space, the predicted segmentation map is transformed into a one-hot vector. This vector indicates all the *parts-of* objects present in the segmentation map. The transparent model classifies this vector. It gives a prediction of the object present on the RGB image according to the different *parts-of*. It is then possible to produce an explanation of this classification based on the *Explainable*

Latent Space and the transparent model computation. In the rest of the chapter, *parts-of* object are called attributes and *objects* are called classes.

First, in Section 4.3.1 we describe the architecture of our framework and the required data to exploit its full potential. Then, in Section 4.3.2 we explain how we sequentially train the different parts of the framework. Finally, we prove in Section 4.3.3 how this *Greybox XAI framework* is explainable.

4.3.1 Greybox Architecture and Data Requirements

Let us denote X and Y two random variables, with $X \sim P_X$ and $Y \sim P_Y$. Without loss of generality we consider the observed samples $\{x_i\}_{i=1}^N \in \mathcal{X}^N$ as vectors and the corresponding labels $\{y_i\}_{i=1}^N \in \mathcal{Y}^N$ as scalars. From the set of observations $\mathcal{X}^N$ and the set of corresponding labels $\mathcal{Y}^N$ we derive a training set denoted $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ with N the number of pair elements (x_i, y_i) of dataset $\mathcal{D}$. The elements of dataset $\mathcal{D}$ are assumed to be independent and identically distributed (i.i.d.) according to an unknown joint distribution $P_{X,Y}$.

Let us denote f a DNN. We assume that a DNN is a function that takes two inputs. The first input is the input data x_i and the second input is the set of trainable weights $\theta = \{\theta_k\}_{k=1}^K$ with K the number of weights of the DNN. Hence we denote $f(x_i, \theta)$ the DNN f applied on x_i with the set of weights θ.

We can consider that a DNN has a probabilistic representation [458]; hence a DNN outputs a likelihood probability function of the random variable Y given X parametrized by $\theta : f(x, \theta) = P(y|x, \theta)$.

Using the fact that the training data is independent and identically distributed according to P_{XY}, the set of training weights θ are optimized by Maximum Likelihood Estimation (MLE) over the training data $\mathcal{D}$.

$$\theta^{MLE} = \arg\max_{\theta} P(\mathcal{D}|\theta) \tag{4.3.1}$$

$$= \arg\max_{\theta} \prod_{i=1}^{N} P(y_i|x_i, \theta) \tag{4.3.2}$$

$$= \arg\max_{\theta} \sum_{i=1}^{N} \log P(y_i|x_i, \theta) \tag{4.3.3}$$

Since the $\arg\max$ of a function does not change if we multiply it by a strictly positive scalar, it is possible to write:

$$\theta^{MLE} = \arg\max_{\theta} \frac{1}{N} \sum_{i=1}^{N} \log P(y_i|x_i, \theta) \tag{4.3.4}$$

in order to get to the definition of Cross Entropy (CE). As a reminder, the CE between $P(y_i)$ and $P(y_i|x_i, \theta)$ in the discrete case is equal to:

$$CE(P(y_i), P(y_i|x_i, \theta)) = -\frac{1}{N} \sum_{i=1}^{N} P(y_i) \log P(y_i|x_i, \theta) \tag{4.3.5}$$

with $P(y_i) = 1$ for the labels $y_i \in \mathcal{D}$ whence:

$$\theta^{MLE} = \arg\max_{\theta} -CE(P(y_i), P(y_i|x_i, \theta)) \tag{4.3.6}$$

$$= \arg\min_{\theta} CE(P(y_i), P(y_i|x_i, \theta)) \tag{4.3.7}$$

The *Greybox XAI framework* is compositional. Hence, let us decompose f into 2 sub-models g and h such that $f = h \circ g$. Let us write g a first model that extracts an *Explainable Latent Space* $\mathcal{Z}$ from the observations $\{x_i\}_{i=1}^N$ and h a second model that maps this Latent Space $\mathcal{Z}$ to labels $\{y_i\}_{i=1}^N$. Hence we have:

$$f : \mathcal{X} \to \mathcal{Y}, f(x_i) = h(g(x_i, \theta_g), \theta_h) = y_i \tag{4.3.8}$$

$$g : \mathcal{X} \to \mathcal{Z}, g(x_i, \theta_g) = z_i \tag{4.3.9}$$

$$h : \mathcal{Z} \to \mathcal{Y}, h(z_i, \theta_h) = y_i \tag{4.3.10}$$

with θ_g and θ_h representing the set of trainable weights of g and h.

Our goal is to map the internal representation of the first model to an *Explainable Latent Space* that will be explainable. In addition, since the prediction of h rely on this *Explainable Latent Space*, we can ensure the prediction to be explainable.

This requires not only to have couples $\mathcal{D}_{couple} = \{(x_i, y_i)\}_{i=1}^N$, directly linking images and classes, but rather triples $\mathcal{D}_{triplet} = \{(x_i, z_i, y_i)\}_{i=1}^N$ with $\{z_i\}_{i=1}^N \in \mathcal{Z}^N$ with $\mathcal{Z}$ being an intermediate *Explainable Latent Space* serving as a bridge between $\mathcal{X}$ and $\mathcal{Y}$. Moreover, it is necessary that each element belonging to $\mathcal{Z}$ is a concept that can be expressed in natural language, and thus that this set $\mathcal{Z}$ represents nameable features. This is necessary in order to obtain objective explanations.

The architecture we propose is therefore a compositional model consisting of two elements. First, an opaque DNN called *Latent Space Predictor* denoted g capable of predicting an *Explainable Latent Space* $\mathcal{Z}$. Second, a transparent model called *Transparent*

Classifier denoted h able of moving from this *Explainable Latent Space* $\mathcal{Z}$ to a final prediction $\mathcal{Y}$. It results in a framework able, for any image x_i, to predict which label y_i it corresponds to. This prediction is justified by an objective and intrinsic explanation based on z_i and on the transparent model's simulatability, due to the composition of both models via $\mathcal{Z}$ being the output of g and the input of h. It is worth noting that while this framework allows explaining the final prediction y_i, based on z_i that acts as the rationale, is not able to explain why z_i was predicted. The *Greybox XAI framework* is therefore a transparent classifier that uses input features from an opaque detector.

Fig. 4.4 shows the Greybox XAI framework used for an image classification task. Here, the dataset $\mathcal{D}_{triplet} = \{(x_i, z_i, y_i)\}_{i=1}^{N}$ consists of triples from a set $\mathcal{X}$ of RGB images, a set $\mathcal{Z}$ of semantic segmentation masks and a set $\mathcal{Y}$ of labels. From $\mathcal{Z}$ we extract a second subset $\mathcal{Z}_{att}$, which we will call the set of attributes. This set contains a list of all attributes, i.e. all different segmentation masks. These are all the *part-of* objects that can be detected by the *Latent Space Predictor*.

Here the *Latent Space Predictor* that constitutes the model g is an Encoder-Decoder model. Its role is to predict from an image x_i a segmentation map z_i. From this segmentation map is extracted the vector $z_{att,i}$ which constitutes a list of all attributes present on the segmentation map z_i. We call this operation a vectorization. The couple $\{z_i, z_{att,i}\}$ is the *Explainable Latent Space*. A logistic regression model here acts as a *Transparent Classifier* h. A Naive Bayes (NB) Classifier can also work, but the experiments conducted in the Section 4.4 gave better results with logistic regression. We then use the inherent transparent nature of h (developed in section 4.3.2) and the *Explainable Latent Space* $\{z_i, z_{att,i}\}$ to explain prediction y_i.

4.3.2 Training Process of the Latent Space Predictor and the Transparent Classifier

The *Latent Space Predictor* g and the *Transparent classifier* h constituting our framework are trained separately. Here are the steps constituting our training:

- Manually extract the subset $\mathcal{Z}_{att}$ from $\mathcal{Z}$ in order to have the attributes in the form of a segmentation mask and in the form of a vector.
- Train the *Transparent Classifier* to predict $\mathcal{Y}$ using $\mathcal{Z}_{att}$. This model is predicting a class based on an attribute vector.
- Train the *Latent Space Predictor* to predict latent space $\mathcal{Z}$ using $\mathcal{X}$. This model is predicting a segmentation map based on an RGB image.

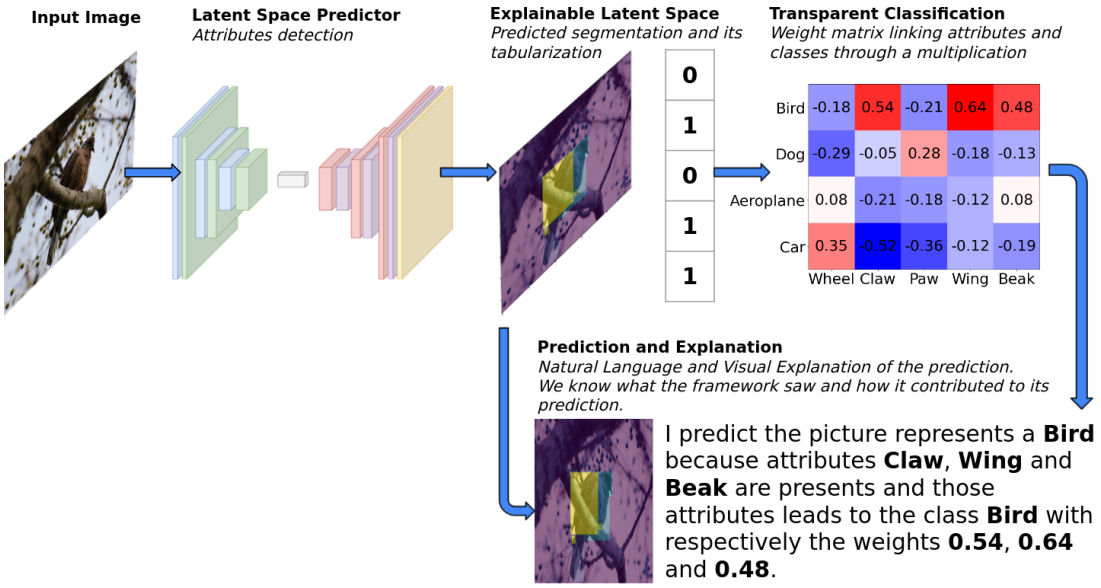


Figure 4.4: Example of use of the Greybox XAI framework for the task of image classification. The framework produces a prediction of what class of object is present on the image while generating a natural language explanation based on the weight matrix of the transparent classification. It also outputs a visual explanation based on a predicted semantic segmentation image representing different object parts of the predicted class.

In Section 4.3.2 we explain how we extract the subset $\mathcal{Z}_{att}$. Section 4.3.2 show how we train the *Transparent Classifier* and in Section 4.3.2 we explain how we train the *Latent Space Predictor*.

Subset Extraction for Knowledge Base Construction

The goal of the dataset subset extraction task is to obtain a Knowledge Base from a Dataset $\mathcal{D}_{triplet} = \{(x_i, z_i, y_i)\}_{i=1}^{N}$ with a set $\mathcal{X}$ of RGB images, a set $\mathcal{Z}$ of semantic segmentation masks and a set of (final, whole object) labels $\mathcal{Y}$. We call a Knowledge Base a common repository of semantic annotations to facilitate a fast and efficient search in the given set of resources [459].

In Description Logics [411], a terminology box *TBox*, or schema extracted from the dataset, together with the class instanciations (in our case, that compose the datasets seen by the network), i.e., the assertional *ABox*, form the $\mathcal{KB} =< TBox, ABox >$. This particular KB will serve as database, in form of a knowledge graph (KG) that interlinks individuals (i.e. instances) of different classes through roles or relationships with possible semantic

restrictions. In order to create and populate a Knowledge Base it is necessary that some terminology axioms $TBox$ and assertion axioms $ABox$ can be extracted from some of the training labels, from which a set of Resource Description Framework (RDF) triples of *subject*, *predicate* and *object* (s, p, o) can be extracted. This is the case, for example, when the labels are in a text or caption form[1]. The *Automatic Knowledge Base Construction (AKBC) from a dataset* problem consists of finding a process $t : Y_{\mathcal{KB}} \rightarrow \{(s, p, o)\}$ able to*triplify* each data point following the RDF language[2]. Concrete examples of AKBC include learning a KB through semantic parsing, using Markov Logic [413] or relational probabilistic models [414]. The triplification process consists of extracting two entities or concepts, subject (s) and object (o), and a predicate (p) that connects them. This predicate expresses a relationship, i.e., a data property or an object property [412] among the subject and the object. The AKBC problem in the *Greybox XAI* framework consists of finding a process that automatically constructs a KB composed of hierarchical (*isA*), *partOf*, and attribute (*hasAttribute*) relations. Analogically, the *ABox* in $\mathcal{KB}$ must be uniquely composed by description logics assertional axioms solely extracted from applying the triplification process on datapoint.

We flatten each semantic segmentation image $z_i \in \mathcal{Z}$ to obtain a single dimension vector and we return $z_{att,i}$, a sorted list of every unique element contained in z_i in order to obtain a set $\mathcal{Z}_{att}$ of each element present on each semantic segmentation image. By using OWL[3] entities and relationships we obtain the following type of KB, linking any element $\{z_{att,i,j}\}_{i,j}^{N,K} \in \mathcal{Z}_{att}^{N,K}$ with the corresponding $x_i \in \mathcal{X}^N$ and $y_i \in \mathcal{Y}^N$, with N the number of triple elements of dataset $\mathcal{D}_{triplet}$ and K the number of different attributes in $\mathcal{Z}$:

This Knowledge Base stands for the *Explainable Latent Space* which will be used to produce the rationale of an explanation.

Our framework is composed of 2 sequential and connected models that do not perform with the same representation of information: while it is possible for the DNN to work with images, it is necessary for the transparent model to work with a much more reduced and compact representation of the data. It is not possible for a model such as a logistic regression to take images as input because it would lose its transparency due to the exploding number of variables and parameters. A logistic regression needs to have human readable predictors and interactions among them kept to a minimum [434]. It can be the case on a compact 2D representation of the semantic segmentation image but not on the image itself because it has too many variables. Therefore, $\mathcal{Z}_{att}$ will be used to train the *Transparent Classifier*.

[1]This is also possible for plain *named* class labels (e.g., through using external lexicons or semantic parsers such as *DBPedia* [409] or *Wordnet* [410].

[2]W3C standard model for data exchange: Resource Description Framework w3.org/RDF/

[3]Web Ontology Language (OWL) [424]

KB	RDF *(s,p,o)* triple examples
TBox	$(z_{att,1,1}$, isPartOf, $y_1)$
	$(z_{att,1,2}$, isPartOf, $y_1)$
	...
	$(z_{att,1,k}$, isPartOf, $y_1)$
	$(z_{att,2,1}$, isPartOf, $y_2)$
	...
	$(z_{att,n,k}$, isPartOf, $y_n)$
ABox	$(x_1$, hasLabel, $y_1)$
	$(x_1$, hasAttributes, $z_{att,1,1...k})$

Table 4.1: Examples of Resource Description Framework (RDF) triples extracted from a Dataset $\mathcal{D} = (\mathcal{X}, \mathcal{Z}, \mathcal{Y})$ with n the number of samples and k the number of different attributes. These triples are contained in a KB terminological (TBox) and assertional (ABox) components.

Training of the Transparent Classifier

We use a logistic regression to statistically fit the attributes $\mathcal{Z}_{att}$ and classes $\mathcal{Y}$ present in the database. The goal is to see if we have fairly discriminating attributes. The choice of using a logistic regression is motivated by the fact that this model predicts probabilities easily interpretable [460].

In binary logistic regression the function $h(z_{att,i}, \theta_h)$ used to model the dependence of a regression target $y_i \in \{0, 1\}$ on features $z_{att,i}$ where $y_i \approx h(z_{att,i}, \theta_h)$ can be written as:

$$h(z_{att,i}, \theta_h) = \frac{1}{1 + \exp(-\theta_h^\intercal z_{att,i})} \tag{4.3.11}$$

with θ_h the set of weights trained to minimize the cost function

$$J(\theta_h) = -\left[\sum_{i=1}^{n} y_i \log h(z_{att,i}, \theta_h) + (1 - y_i) \log(1 - h(z_{att,i}, \theta_h))\right] \tag{4.3.12}$$

Multinomial logistic regression is a generalization of binary logistic regression to multiclass problems, meaning that the label $y_i \in \{1, 2, ..., k\}$ can take K different values depending on the number of classes. A softmax function is used to generalize $h_\theta(z_{att,i})$ from the binary to the multi-class classification problem:

$$h(z_{att,i}, \theta_h) = \begin{bmatrix} P(y=1|z_{att,i};\theta_h) \\ P(y=2|z_{att,i};\theta_h) \\ \vdots \\ P(y=k|z_{att,i};\theta_h) \end{bmatrix} = \frac{1}{\sum_{j=1}^{k} \exp \theta_{h,j}^{\intercal} z_{att,i}} \begin{bmatrix} \exp \theta_{h,1}^{\intercal} z_{att,i} \\ \exp \theta_{h,2}^{\intercal} z_{att,i} \\ \vdots \\ \exp \theta_{h,k}^{\intercal} z_{att,i} \end{bmatrix} \tag{4.3.13}$$

$\theta_{h,1}$, $\theta_{h,2}$, ... $\theta_{h,k} \in \mathcal{R}_n$ are all the parameters of the regression and are represented as a n-by-K matrix:

$$\theta_h = \begin{bmatrix} | & | & | & | \\ \theta_{h,1} & \theta_{h,2} & \dots & \theta_{h,K} \\ | & | & | & | \end{bmatrix} \tag{4.3.14}$$

$$J(\theta_h) = -\left[\sum_{i=1}^{m}\sum_{k=1}^{k} \mathbb{1}\{y_i = k\} \log \frac{\exp \theta_{h,j}^{\intercal} z_{att,i}}{\sum_{j=1}^{K} \exp \theta_{h,j}^{\intercal} z_{att,i}}\right] \tag{4.3.15}$$

Equation 4.3.15 is minimized thanks to an iterative optimization algorithm with the gradient:

$$\nabla_{\theta_{h,k}} J(\theta_h) = -\sum_{i=1}^{m} \left[z_{att,i} \left(\mathbb{1}\{y_i = k\} - P(y_i = k|z_{att,i};\theta_h)\right)\right] \tag{4.3.16}$$

with

$$P(y_i = k|z_{att,i};\theta_h) = \frac{\exp \theta_{h,k}^{\intercal} z_{att_i}}{\sum_{j=1}^{K} \exp \theta_{h,j}^{\intercal} z_{att,i}} \tag{4.3.17}$$

Thus, for any $\{z_{att,i}\}_{i=1}^{N} \in \mathcal{Z}_{att}$ is it possible to find the associated $\{y_i\}_{i=1}^{N} \in \mathcal{Y}$ by using the following equation:

$$P(y_i = k|z_{att,i};\theta_h) = softmax(\theta_{h,j}^{\intercal} z_{att,i}) \tag{4.3.18}$$

Since the softmax function is monotonic, the ranking of probabilities given by $softmax(\theta_{h,j}^{\intercal} z_{att,i})$ is the same as the one given by $\theta_{h,j}^{\intercal} z_{att,i}$. Therefore, we can approximate that:

$$h(z_{att,i}, \theta_h) \approx \theta_{h,j}^{\intercal} z_{att,i} \approx \sum_{j=1}^{K} \theta_{h,j} z_{att,i,j} \tag{4.3.19}$$

Each parameter $\theta_{h,j}$ provides a quantitative contribution of the corresponding attribute $z_{att,i}$ to predicted class.This logistic regression is transparent by design because it meets the criteria of algorithmic transparency, decomposability and simulatability from [434]:

- Algorithmic Transparency: the user can understand the process followed by the logistic regression to produce any given output from its input data. Just multiply the weight matrix θ_h by the attribute vector $z_{att,i}$ to obtain the prediction y_i.
- Decomposability: every part of the logistic regression is understandable by a human without the need for additional tools. The input $z_{att,i}$, the parameters θ_h and the calculation are interpretable.
- Simulatability: the logistic regression has the ability to be totally simulated by a human. It is self-contained enough for a human to think and reason about it as a whole. For this condition to remain true, the complexity of the logistic regression must remain low. Thus, it is necessary that the attribute vector $z_{att,i}$ and the parameters θ_h are not too big.

We can produce explanations following customized templates, for instance:

Example 4.6 (Explanation:)**.** *The model predicts this attribute vector $z_{att,i}$ to belong to class y_i, because attributes $\{z_{att,i,1}, z_{att,i,...}, z_{att,i,k}\}$ are linked to the class y_i with weights $\{\theta_{h,1}, \theta_{h,...}, \theta_{h_k}\}$ in training dataset $\mathcal{D}_{triples}$.*

Regarding the definitions expressed in Section 4.2.1, we can say that the explanation function $\Phi(h(z_{att,i})) = \theta_h^\intercal z_{att,i}$ of our model $h : \mathcal{Z}_{att} \rightarrow \mathcal{Y}$ is:

- Objective: as the explanation e_i uses symbols ($z_{att,i}$) and relationships ($\theta_{h,i}$) that can be conceptualized by the human user.
- Intrinsic: as the explanation e_i only depends of elements ($z_{att,i}$), parameters ($\theta_{h,i}$) and operations present in original model h

The validity of the explanation will have to be measured when applying the *Greybox XAI* framework to a use-case, as this criterion is dataset-dependant. This notion can be easily measured by comparing the weights $\theta_{h,i}$ and an expert knowledge base.

Training of the Latent Space Predictor

The *Latent Space Predictor* must predict a segmentation map z_i from a RGB input image x_i thanks to an Encoder-Decoder architecture. This segmentation map will then be vectorized in an attribute vector $z_{att,i}$ to constitute the *Explainable Latent Space* $\{z_i, z_{att,i}\}$. We choose to use a DeepLabv3+ [461] as a *Latent Space Predictor* with a ResNet101 as backbone model. The specificity of DeepLabv3+ is to use an atrous convolution, allowing the developer to adjust filter's field-of-view in order to capture multi-scale information,

and a depthwise separable convolution. As it is a semantic segmentation task, we use an output stride of 16 for denser feature extraction as it is the best trade-off between speed and accuracy. The objective is to make a pixel-wise prediction over an entire image and the performance is measured in terms of pixel intersection-over-union averaged across the attributes (mIOU).

In order to test the performance of our model, we do not train it with images of semantic segmentation masks but with bounding boxes showing the presence or absence of attributes, in a weekly supervised manner [462]. The main objective is to have an *Explainable Latent Space* that accounts for the presence of each attribute as much as possible. It is therefore necessary that each attribute present on the x_i image is segmented but the segmentation map does not need to be very accurate because it is afterwards vectorized. Pixels that do not belong to any bounding box are considered as background pixels, while the ones belonging to several bounding boxes are accounted as belonging only to the smallest one. We select the smallest rather than the largest in order to not lose the small attributes encompassed by large ones (like eyes in the middle of a face for example).

The input of the Latent Space predictor is an RGB image x_i while its output is a segmentation map z_i of dimension $h * w$ with h and w the dimensions of x_i. As the spatial information is not used by the *Transparent Classifier*, we extract from z_i an attribute vector $z_{att,i}$ containing the list of each unique value contained in z_i. A confidence mask is applied in order to keep only the attributes that were predicted with a confidence above a certain threshold. We finally use a one-hot encoding to obtain a vector of 0s and 1s, describing the prediction of presence or absence of each attribute in the input RGB image x_i. Because this extraction does not allow backpropagation, the *Latent Space Predictor* is trained by maximising its mIOU. It is therefore not directly trained to predict a good attribute vector $z_{att,i}$ but a good segmentation map z_i. Note that this segmentation map prediction is opaque, no explanation is given as to why a certain pixel has been predicted as representing a certain attribute.

4.3.3 Inference prediction and its explanation rendering through a Natural Language Explanation

When the *Latent Space Predictor* g and the *Transparent Classifier* h are trained and provide good results on their own, we freeze their weights and compose them to evaluate the function $(h \circ g) : \mathcal{X}\mathcal{Y}$ in order to predict a class y_i from x_i. As the *Transparent Classifier* h is transparent by design, we are able to generate a natural language explanation while making the prediction $y_i = h(g(x_i))$. As explained earlier we produce an explanation of prediction $\hat{y}$ from 3 elements:

- The intrinsic transparency of h, allowing to know what would imply a change of $z_{att,i}$

on the final prediction $\hat{y}$ thanks to the learned weights in θ_h.

- $z_{att,i}$ which takes the form of a list of attributes that can be named in natural language.
- z_i which is a segmentation map, showing the position of attributes on the RGB image x_i, thus taking over the ease of understanding of the usual visual explanations.

We define the explanation function $\Phi : \mathcal{Y} \rightarrow \mathcal{E}$ with $\mathcal{Y}$ the label space and $\mathcal{E}$ the explanation space. The *Transparent Classifier* h and the *Explainable Latent Space* $\{z_i, z_{att,i}\}$ make it possible to produce explanation e in natural language.

$$e_i = \Phi(h(z_{att,i})) \tag{4.3.20}$$

Therefore, we have the following explanation:

e_i = "Image x_i represents a y_i because attributes $z_{att,i,1}$, $z_{att,i,...}$ and $z_{att,i,m}$ are present, and the classifier h leads those attributes respectively with weights $\theta_{h,1}$, $\theta_{h,...}$ and $\theta_{h,n}$ to class y_i."

In addition, the segmentation map z_i can be displayed as a visual explanation to show the position of attributes $z_{att,i}$.

As a summary, the *Greybox XAI* framework makes a prediction y_i of a random RGB image $x_i \in X$ and produces an explanation e_i by following Algorithm 0:

Algorithm 2 Greybox XAI framework pseudo-algorithm to produce a natural language explanation of a prediction

Require: Input Image x_i, Latent Space Predictor g, Transparent Classifier h, Explanation Function Φ

Step 1: Predict Explainable Latent Space
$z_i \leftarrow g(x_i)$
Step 2: Vectorize Explainable Latent Space to obtain an attribute vector
for $j \in z_i$ **do**
    Append($z_{att,i}$, j) if $j \notin z_{att,i}$
end for
Step 3: Predict Object Class
$y_i \leftarrow h(z_{att,i})$
Step 4: Generate a Natural Language Explanation
$e_i \leftarrow \Phi(h(z_{att,i}))$ **return** Prediction y_i, Natural Language Explanation e_i, Segmented Image z_i

4.4 Experimental Study

We illustrate the use of our framework with 2 datasets: MonuMAI, PASCAL-Part. The extensive use case proving the utility of the model is developed on MonuMAI because this dataset has already been used in the state of the art [449] to prove the utility of compositional models. The hypothesis tested is to verify that the *Greybox XAI* framework is able to produce accurate and explainable predictions. Our goal is to solve a compositional classification problem and to be able to predict for each image which object is present, justifying this prediction by the *parts-of* object (attributes) of this object present on the image.

MonuMAI dataset [463] allows to classify architectural style classification from facade images. The idea here is to be able to classify an image by predicting which type of monument is present in the image based on the distinctive architectural attributes of the different types of monuments. This dataset contains approximately 1500 images labelled with 4 classes (architectural styles) and containing bounding boxes describing the presence of 15 different attributes (visible characteristics of these architectural styles). Each image is labelled with the architectural style of the monument present on the image and bounding boxes inform about the presence and position of the attributes present on the image. We call this dataset $\mathcal{D}_{triple} = \{(x_i, z_i, y_i)\}_{i=1}^{N}$ with a set $\mathcal{X}$ of RGB images representing architectural monuments, a set $\mathcal{Z}$ of bounding boxes representing architectural attributes and a set $\mathcal{Y}$ of architectural styles.

A Knowledge Base 4.2 is built based on the expert knowledge of the MonuMAI dataset [463].

We repeat the various steps described in the Section 4.3.2 in order to train and use the *Greybox XAI* framework:

- A transparent model h is trained to predict an architectural style, using as input a vector encoding the presence and absence of architectural attributes.
- A Deep Neural Network g is trained to predict a segmentation map from an RGB image input. Its purpose is to detect the different architectural attributes that constitute the image.

4.4.1 Logistic Regression as a Transparent Classifier

The purpose of the *Transparent Classifier* is to represent a more accurate and closer version of the dataset than the Knowledge Base 4.2 itself. In fact, the knowledge contained in the knowledge base is very generic (for example, a Hispanic-Muslim monument has a flat arch, an horseshoe arch and a lobed arch). However, while this is true in a general case, not all

KB	RDF *(s,p,o)* triple examples

TBox	`(Ogee Arch, isPartOf, Gothic Monument)`
	`(Pointed Arch, isPartOf, Gothic Monument)`
	`(Trefoil Arch, isPartOf, Gothic Monument)`
	`(Gothic Pinnacle, isPartOf, Gothic Monument)`
	`(Flat Arch, isPartOf, Hispanic-Muslim Monument)`
	`(Lobed Arch, isPartOf, Hispanic-Muslim Monument)`
	`(Horseshoe Arch, isPartOf, Hispanic-Muslim Monument)`
	`(Broken Pediment Arch, isPartOf, Baroque Monument)`
	`(Solomonic Column, isPartOf, Baroque Monument)`
	`(Rounded Arch, isPartOf, Baroque Monument)`
	`(Rounded Arch, isPartOf, Renaissance Monument)`
	`(Porthole Arch, isPartOf, Baroque Monument)`
	`(Porthole Arch, isPartOf, Renaissance Monument)`
	`(Lintelled Doorway Arch, isPartOf, Baroque Monument)`
	`(Lintelled Doorway Arch, isPartOf, Renaissance Monument)`
	`(Serliana, isPartOf, Renaissance Monument)`
	`(Segmental Pediment, isPartOf, Renaissance Monument)`
	`(Triangular Pediment, isPartOf, Renaissance Monument)`

Table 4.2: Examples of RDF triples extracted from the MonuMAI dataset and contained in a KB terminological (TBox) and assertional (ABox) components

such monuments have all these attributes and some representations (i.e. images) of these monuments may have some attributes missing. Also, in this particular case of monuments, it is possible that some images have architectural attributes belonging to several architectural styles. It is the result of the progressive evolution of the construction or reconstruction processes.

Following Section 4.3.2, a logistic regression $h(z_{att}, \theta_h)$ is trained on the set of attributes $\ddagger_{att}$ of the dataset to predict classes from the set of labels $\mathcal{Y}$. We compare the performance of the logistic regression to a Naive Bayes Classifier. The definition of the NB Classifier can be found in Section 5.2.1 of the next chapter.

Model	**Accuracy**
Logistic Regression	**97.65%**
Naive Bayes Classifier	94.32%

Table 4.3: Mean Accuracy of 2 transparent models on MonuMAI dataset. The logistic regression have a better accuracy than the Naive Bayes Classifier. We therefore choose to use it as the *Transparent Classifier* of the *Greybox XAI* framework

Figure 4.5 represent the set of trainable weights θ_h of the logistic regression, linking attributes from z_{att} and classes from $\mathcal{Y}$ thanks to the relationship $y_i \approx \theta_h^{\mathsf{T}} \times z_{att_i}$. These weights provide a statistical link between attributes and classes.

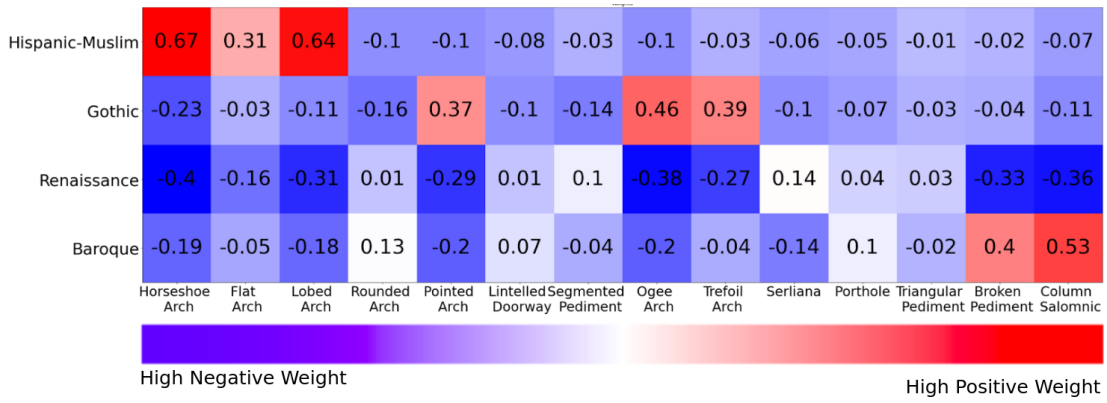


Figure 4.5: Weights of the logistic regression model fitted to link attributes and classes from the MonuMAI dataset.

From the set of trainable weights θ_h of the *Transparent Classifier* we extract a Knowledge Graph (KG) (Figure 4.6) representing the link between attributes and classes. It is

a visual representation of the weights from Figure 4.5. If a weight is superior to 0, an edge is drawn between the 2 concerned nodes. Representing knowledge this way has a simple explanatory interest: when an attribute is detected, it is straightforward to see which classes are linked to this attribute. We see for example that the attributes *Trefoil Arch*, *Pointed Arch* and *Ogee Arch* are only linked to the architectural style *Gothic*. Therefore, if those attributes are present in the vector z_{att_i}, the class *Gothic* will be predicted by the *Transparent Classifier*.

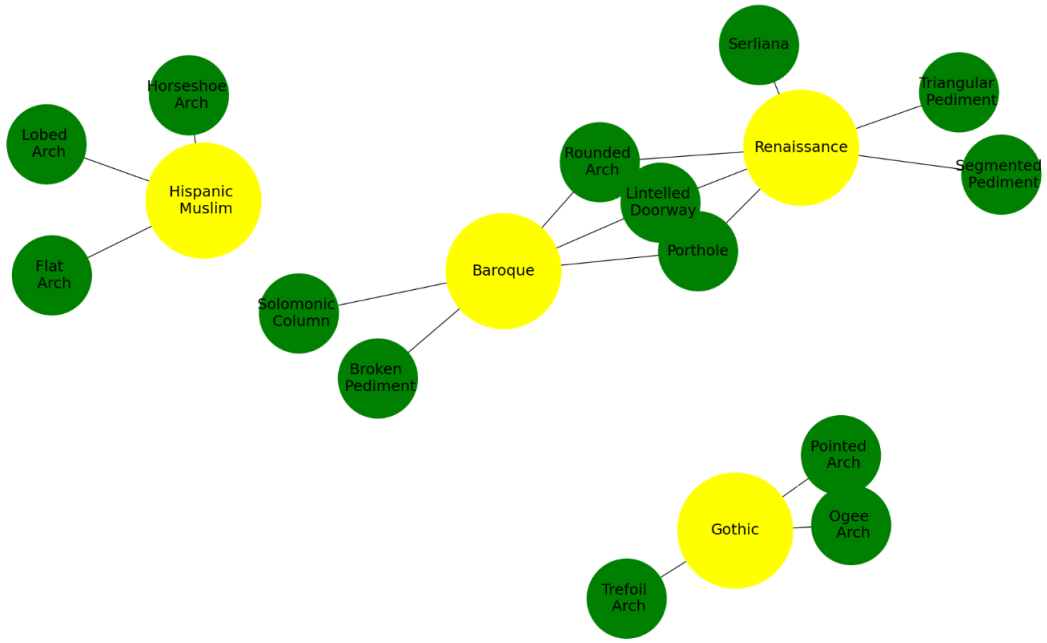


Figure 4.6: Knowledge Graph representation automatically extracted from the Logistic Regression weights on the MonuMAI dataset. Green nodes represent attributes while yellow nodes represent classes. Distances between nodes represent weights linking attributes and classes in the θ matrix of the fitted logistic regression model: the closer 2 nodes are, the larger the weight linking them. An edge is set to black if the associated weight is superior to zero and transparent otherwise.

4.4.2 Deeplabv3+ as a Latent Space Predictor

Logistic regression cannot remain transparent when taking images as input because the number of parameters and variables would be far too large. Moreover, these variables would be pixels rather than symbols. To overcome this and to make the logistic regression take

as input a vector of attributes, we train an Encoder-Decoder on the images on a semantic segmentation task. We chose to use a DeepLabv3+ [461] as it gives the best performance on this dataset.

On this dataset we do not have a semantic segmentation image representing the ground truth but only bounding boxes around each attribute. We use these bounding boxes to predict a segmentation map thanks to a cross-entropy loss. As the model was trained with images annotated with bounding boxes instead of semantic segmentation images, the segmented attributes have a square shape. It is not a problem as unlike most semantic segmentation models, what we are interested in here is not the mIoU or the accuracy of the prediction at the pixel level but rather how well the attribute vector $z_{att,i}$ is predicted. Since the *Transparent Classifier* takes as input a one-hot encoded vector of attributes, the mIoU and the average precision of the Deeplabv3+ have no influence on the classification result: what counts is to detect at least 1 time each attribute, not to detect all pixels of each occurrence of each attribute. Whether we detect 1 pixel of 1 attribute or 3000 pixels of 1 attribute is the same because the segmentation map is put in the form of a vector of attribute presence in order to be used by the *Transparent Classifier*.

To generate this attribute vector, we compare the list of sorted unique elements of the predicted semantic segmentation image and the list of attributes present in the image. Figure 4.7 represents on the left an RGB image used as input to the Deeplabv3+ and on the right the predicted semantic segmentation image. The attribute vector associated with this semantic segmentation image would be, for example, $[1, 0, 1, 1, 0, ..., 0]$ representing the 3 attributes (in dark cyan, pink and blue) present in the image. It is this vector that is subsequently used by the pre-trained *Transparent Classifier* to predict the class of the image.

4.4.3 Performance of the Greybox XAI framework: Accuracy and Explainability

We judge the performance of our framework according to 2 notions: its accuracy during an image classification task and the explainability of its prediction during this same classification task.

Accuracy

We evaluate our model on the image classification task and compare it to several baselines. Our model is the *Greybox XAI Framework*, composed by a Deeplabv3+ and a logistic regression. The first baseline is a ResNet101 classifier, in order to have the example of a total blackbox. We then compare to the EXPLANet [449] model which is the state of the art

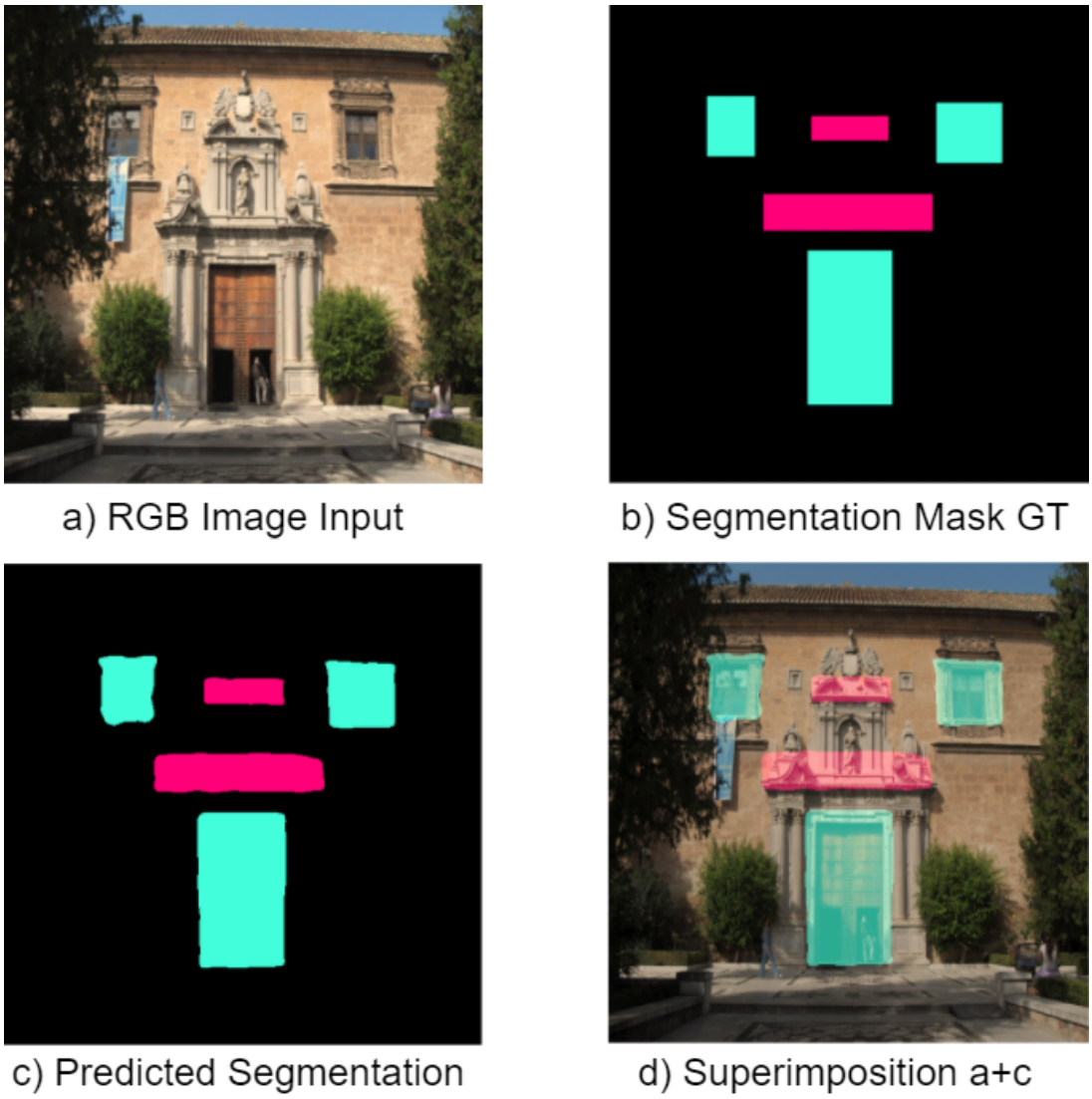


Figure 4.7: Example of results obtained using semantic segmentation. Image a) represents the RGB input data and Image b) represents the Ground Truth of semantic segmentation masks. Image c) represents the result of this segmentation by the *Latent Space Predictor* and image d) is an overlay of the prediction on the input image, in order to see where the detected attributes are. The black pixels represent the background, the cyan and pink pixels represent two architectural attributes.

of compositional XAI models on this database. We also build a new baseline by modifying the DeepLabv3+ to make it a classifier. This Deeplab Classifier is a usual Deeplabv3+ architecture, with a ResNet101 as backbone, with skip connections and atrous convolutions, but instead of predicting the class of each pixel as in a semantic segmentation we use it in a classification role by modifying the last layer. Instead of evaluating it by the MiOU, it is now judged by the accuracy of the global class. By adding an AveragePooling and a Softmax after the decoder, we obtain an end-to-end classification model. Since this is a classification task, we compare ourselves in terms of accuracy. We also add the results on the PASCAL-Part dataset, which contains more attributes and more classes. As our model is transparent by design, we compare ourselves separately to models considered as opaque (CNNs such as ResNet or Encoder-Decoder such as Deeplabv3+) and explainable models (such as EXPLANet).

The results of Greybox XAI classification model based on Deeplabv3+ semantic segmentation backbone and a logistic regression, together with these baseline classification networks are shown in Table 4.4 for MonuMAI and PASCAL-Part dataset. The results show that the Greybox XAI classifier achieves similar accuracy to the opaque models, being outperformed by less than 1% each time. However, its accuracy is far superior to the explainable baseline, which is the EXPLANet model. We can therefore consider that there is a slight loss in accuracy compared to the opaque models but a gain compared to the compositional models.

The loss in accuracy compared to opaque models should be counterbalanced by a benefit in explainability because the *Greybox XAI* is transparent by design when used for a classification task (see Section 4.3.2) and produces "good" explanations (*objective* and *intrinsic*).

Explainability

In order to verify whether the explanations generated by the *Greybox XAI* are *valid* we compare the Knowledge Graph extracted directly from the *Transparent Classifier* weights in Figure 4.6 and the expert Knowledge Graph Figure 4.8. This figure is taken from [449] and represents MonuMAI Knowledge Graph constructed based on art historians expert knowledge [463]. The KG is extracted from the *Transparent Classifier* by creating a node for every *part-of* and *objects* of the weights matrix and by drawing an edge between nodes for every weights superior to 0. The Graph Edit Distance (GED) [464] between the two KG is equal to zero, meaning that the explanation of the *Transparent Classifier* is the same as the ones art historians experts would have produced. Therefore, we can consider that these explanations are globally *valid* if the semantic segmentation is correct.

We also verify if our framework is a *self-explaining prediction model* according to the

Dataset	Model	Accuracy (%)
Comparison with Opaque Models		
MonuMAI	ResNet101	95.69
	Deeplabv3+ Classifier	**96.02**
	Greybox XAI	94.04
	MonuNet Classifier	83.11
PASCAL-Part	ResNet101	90.12
	Deeplabv3+ Classifier	**90.18**
	Greybox XAI	88.30
Comparison with Explainable Models		
MonuMAI	EXPLANet	90.40
	KG Deterministic Classifier	54.79
	Greybox XAI	**94.04**
PASCAL-Part	EXPLANet	82.4
	Greybox XAI	**88.30**

Table 4.4: Explainable compositional vs opaque direct classification: Results of the Greybox XAI model (using semantic segmentation Deeplabv3+ and a logistic regression) on MonuMAI and PASCAL-Part datasets, and comparison with embedded version of the baseline model MonuNet, a vanilla classifier baseline with ResNet101, an expert KG-based deterministic (non-trained) classifier, the compositional model EXPLANet and a classifier derived from Deeplabv3+

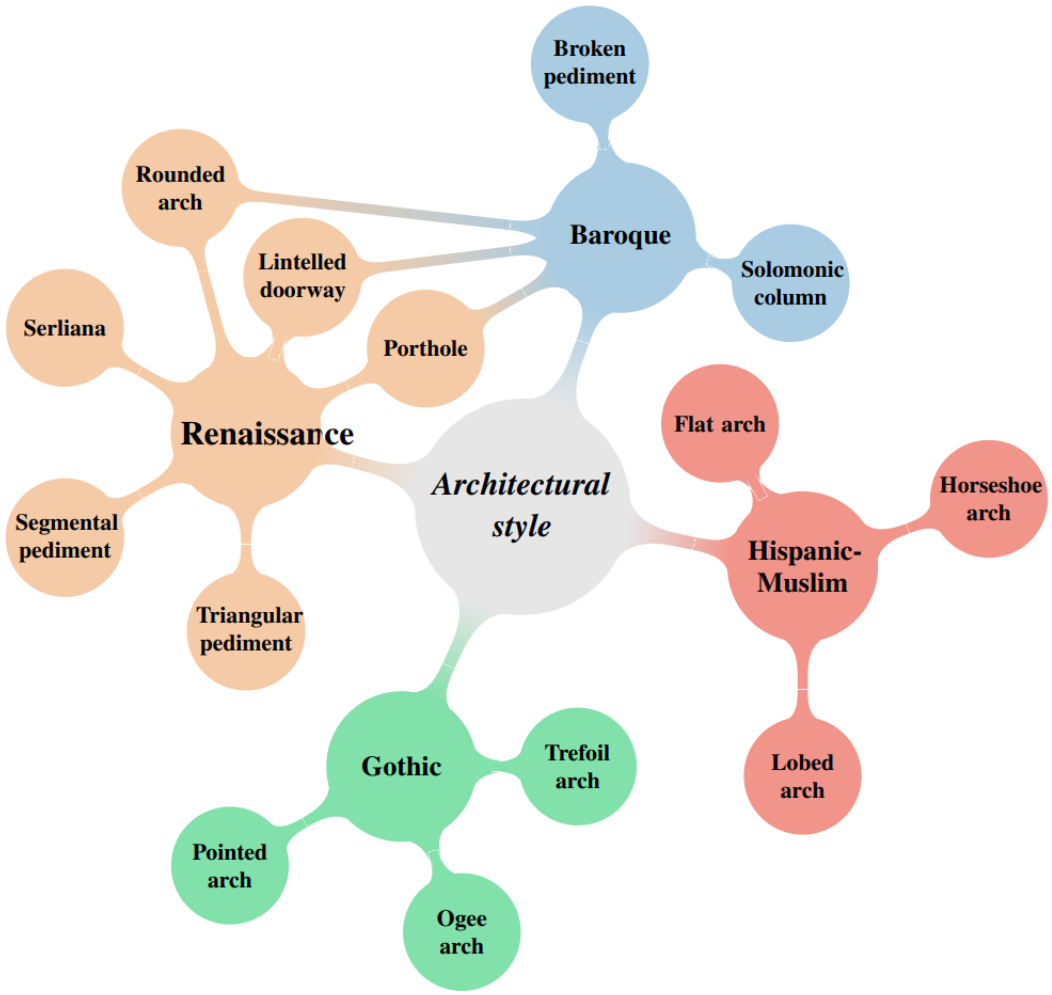


Figure 4.8: Simplified MonuMAI knowledge graph constructed based on art historians expert knowledge [449, 463]. It links attributes and classes in a graphical representation and shows, for example, that a Hispanic-Muslim monument is composed by Flat Arches, Horseshoe Arches and Lobed Arches. This graph representing the expert knowledge of the dataset is similar to the one extracted by the *Transparent Classifier*.

definition of [453], i.e. if it has the form:

$$f(x) = g(\theta(x)_1 h(x)_1, ..., \theta(x)_k h(x)_k) \tag{4.4.1}$$

where:

- g is monotone and completely additively separable
- For every $z_i := \theta_i(x)h_i(x)$, g satisfies $\frac{\partial g}{\partial z_i} \geq$
- θ is locally difference bounded by h
- $h_i(x)$ is an interpretable representation of x
- k is small

Taking Equation 4.3.8, *Greybox XAI* framework can be written:

$$f(x) = (h \circ g)(x) = h(g(x_i, \theta_g), \theta_h) \approx \theta_{h,1} z_{att,1}, ..., \theta_{h,n} z_{att,n} \tag{4.4.2}$$

where:

- The *Transparent Classifier* h is monotone and completely additive separable as it can be approximated with the multiplication between the weight matrix θ_h and the features.
- Partial derivative of h with respect to $\theta_{h,i} z_{att,i}$ is positive.
- θ_h is locally difference-bounded by $z_{att,i}$.
- $z_{att,i}$ is an interpretable representation of x as $z_{att,i}$ are nameable features.
- $n =$ is small as interactions of the logistic regression are kept to a minimum to respect the definition of simulatability.

From these different elements, we can conclude that the Greybox XAI model is transparent and produces "good" explanations when used for the task of image classification. Moreover, it is possible to accompany this textual explanation by a visualization, showing the semantic segmentation image masks used to determine the attribute vector $\ddagger_{att}$ employed by the *Transparent Classifier* to perform the classification. Figure 4.9 shows an example of the visualization that can be produced. This image has been classified as *Gothic* based on the different attributes detected by the Latent Space Predictor. The predicted semantic segmentation map can be overlaid on top of the RGB input to visualize where these features

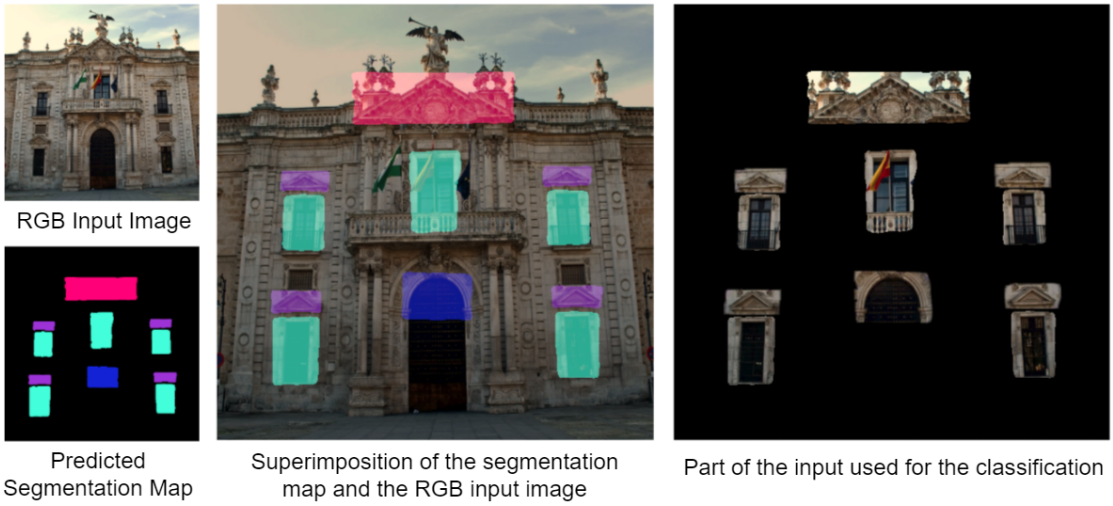


Figure 4.9: Visual explanation of the Greybox XAI model. The image in the upper left corner is the RGB input that the model must classify. In the bottom left corner is the semantic segmentation image predicted by the model, showing in black the pixels classified as part of the background. Cyan and black pixels represents two attributes. In the middle the superposition of the two images on the left, removing black pixels from the background to keep only the elements that will be used in the logistic regression. Finally, the image on the right is the same sample image but replacing the attributes by their RGB value and hiding the background pixels, which are not used during the classification by the Transparent Classifier as it only uses an attribute vector.

are located. The image on the right shows all pixels that were used during classification. These are obtained by removing every pixel segmented as being a background pixel.

Figure 4.9 illustrates the ideal case where the semantic segmentation performed is perfect and the ensuing classification is also perfect. However, there are some cases where the prediction of the segmentation map is either flawed or false, which is sometimes inconsequential but that will sometimes distort the prediction. Below we present the different cases observed and associated example.

Example 4.7 (Incomplete segmentation, correct prediction)**.** *Some occurrences of an attribute are not detected. As vectorization does not take into account the number of occurrences of each attribute, this has no impact on the prediction or the explanation. However, the predicted segmentation map is far from the real segmentation map, which gives an incomplete visual explanation. See Figure 4.10. This is the kind of example that ultimately has no impact on the result: the prediction remains good and the explanation is valid and complete because it addresses the 2 main elements detected. The fact that some occurrences of one of the attributes are forgotten is of little importance for our classification task. However, for a detection task it could have been very problematic. A future improvement could be to use an Optimized Loss Functions for Object detection [465] when training the Latent Space Predictor in order to improve the detection.*

Example 4.8 (Wrong segmentation, correct prediction)**.** *Some attributes are missing but the classification does not change as the detected attributes are already discriminative enough for the Transparent Classifier. See Figure 4.11. Here it is complicated to judge the validity of the explanation without being an expert: is the explanation complete enough by talking about the cyan and red attributes or was the blue attribute indispensable for the classification of this monument? The justification is therefore valid but may lack completeness. Looking at the Transparent Classifier weight matrix, we can see how important the blue attribute should have been in the explanation.*

Example 4.9 (Wrong segmentation, wrong prediction)**.** *Some attributes are detected while they are not existing in the ground truth. It makes the classification totally wrong. See Figure 4.12. This error is more problematic than the previous one because it misleads the user by giving him an invalid explanation. The model is not Right for the Right Reason. This is primarily a semantic segmentation problem: the model should not have seen the yellow attribute and the cyan attribute. The model could thus be improved, maybe with the use of a weighted FocalLoss[466] to penalize more strongly a bad prediction of attributes.*

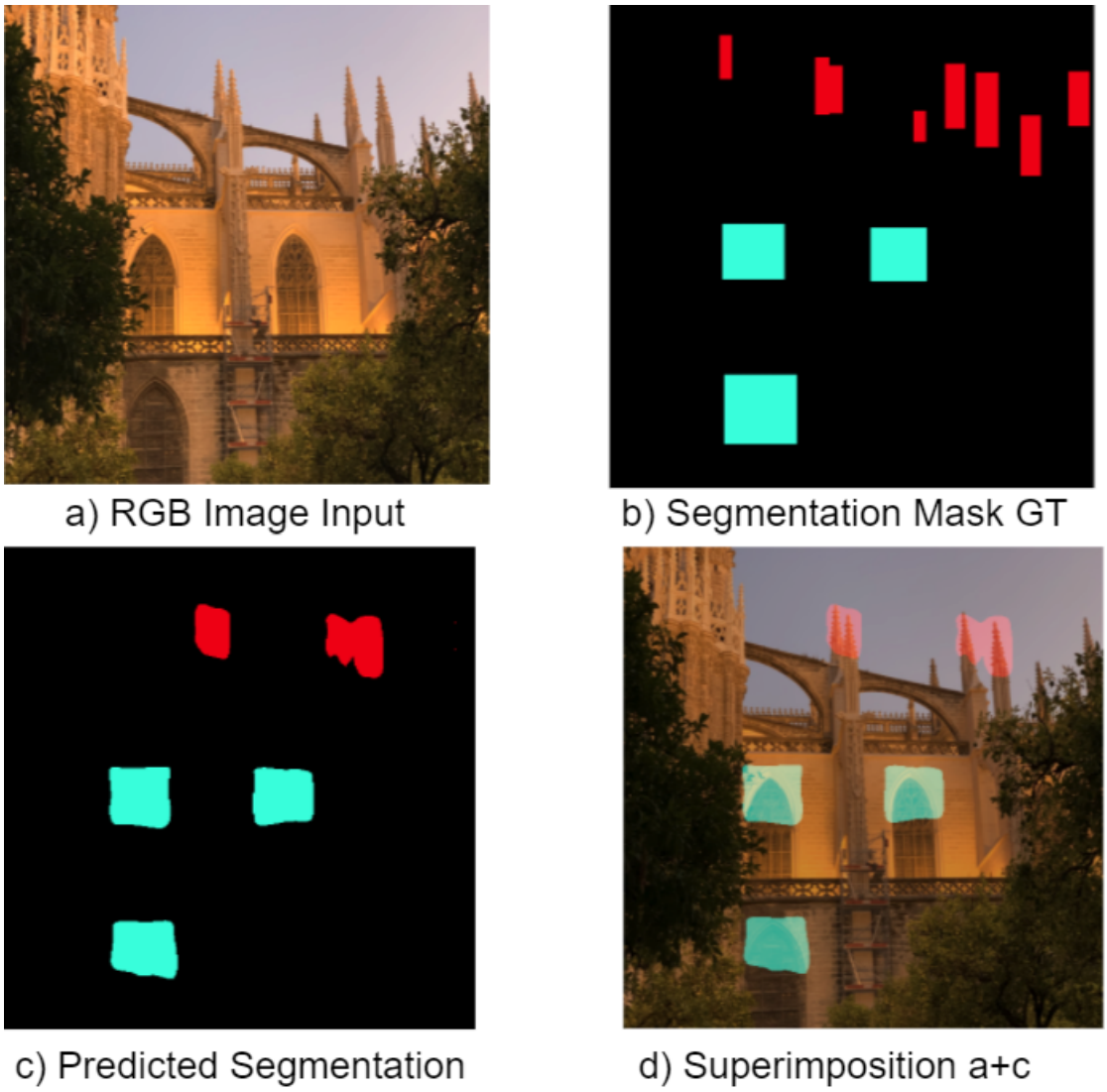


Figure 4.10: **Incomplete segmentation, correct prediction**. Example of results obtained using semantic segmentation. Image a) represents the RGB input data and Image b) represents the semantic segmentation masks of the Ground Truth, which the Latent Space Predictor must predict. Image c) represents the result of this segmentation and image d) is an overlay of the prediction on the input image, in order to see where the detected attributes are. Black pixels represent the background, cyan and black pixels represent two architectural attributes. The red attribute is missing 5 times compared to the GT but it has no impact on the prediction nor the explanation.

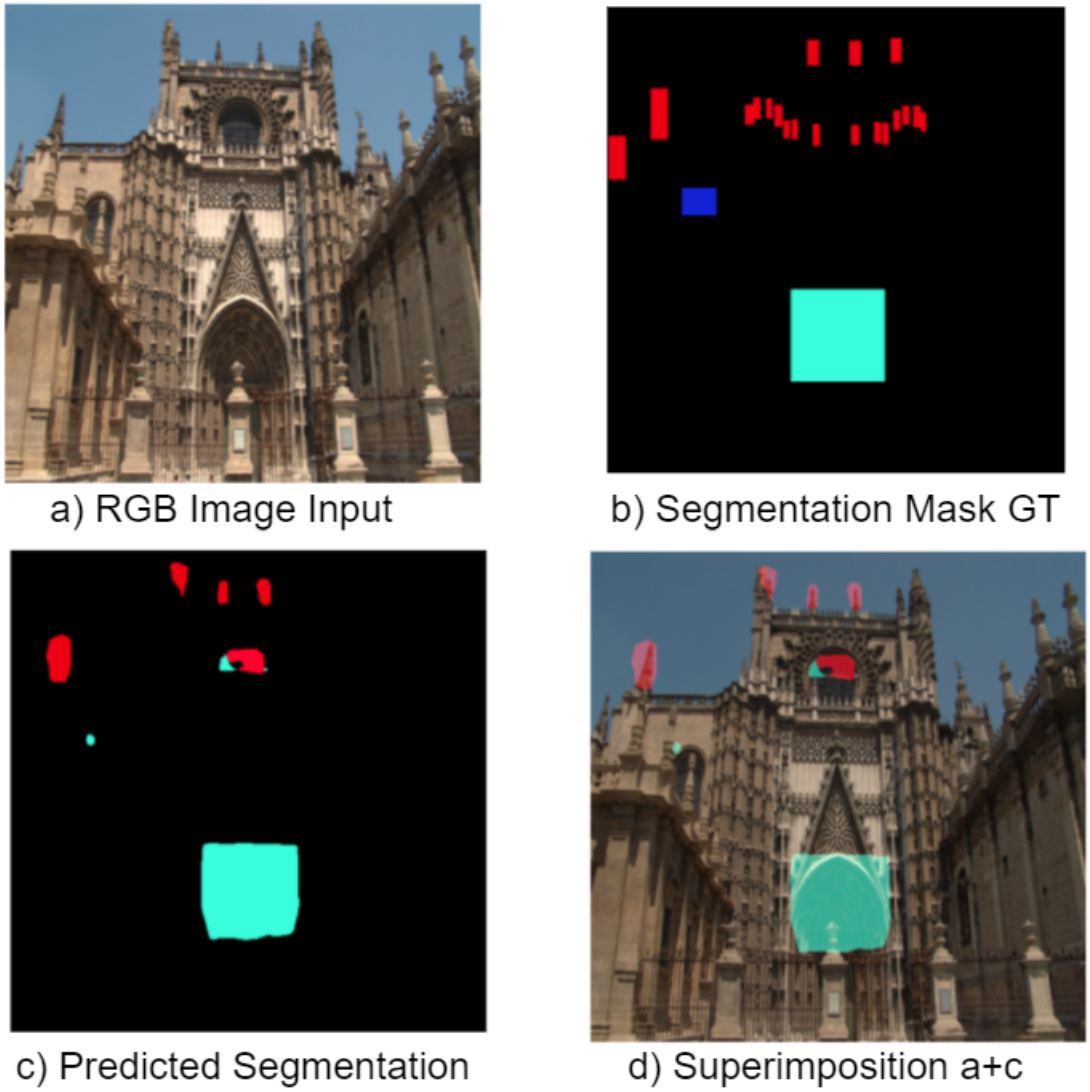


Figure 4.11: **Wrong segmentation, correct prediction**. Example of results obtained using semantic segmentation. Image a) represents the RGB input data and Image b) represents the semantic segmentation masks of the Ground Truth, which the Latent Space Predictor must predict. Image c) represents the result of this segmentation and image d) is an overlay of the prediction on the input image, in order to see where the detected attributes are. The black pixels represent the background, the colored pixels represent architectural attributes. The red attribute is missing several times and the blue one is totally absent compared to the GT but it has no impact on the prediction, as the presence of the red and cyan elements are enough to consider that this monument is Gothic.

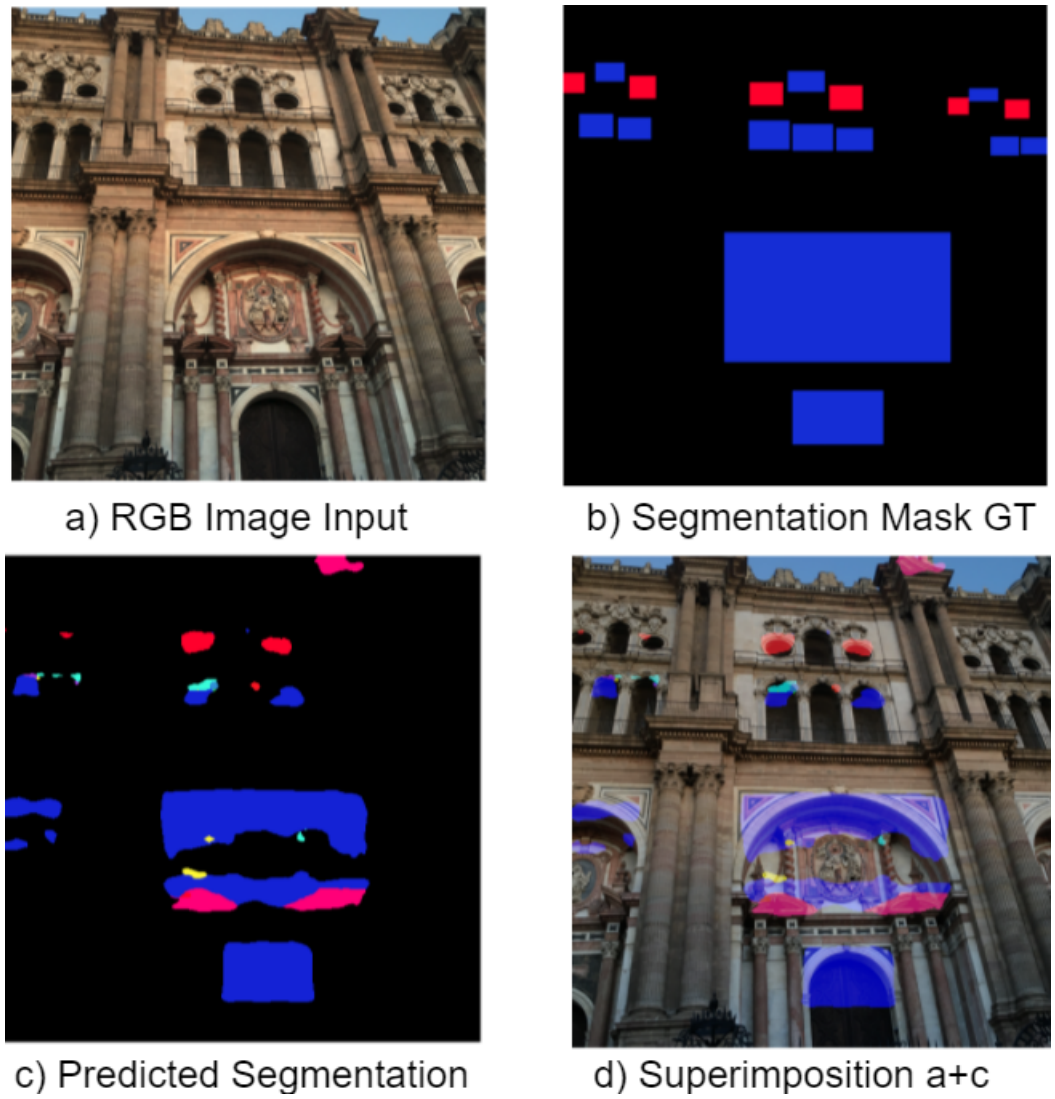


Figure 4.12: **Wrong segmentation, wrong prediction.** Example of results obtained using semantic segmentation. Image a) represents the RGB input data and Image b) represents the semantic segmentation masks of the Ground Truth, which the Latent Space Predictor must predict. Image c) represents the result of this segmentation and image d) is an overlay of the prediction on the input image, in order to see where the detected attributes are. The black pixels represent the background, the colored pixels represent architectural attributes. The red and blue attributes are missing several times and the yellow, pink and cyan where added while they do not exist in the GT. As the *Latent Space Predictor* segments the image badly, the *Transparent Classifier* is not able to correctly predict the class.

4.4.4 Counterfactual Explanations

While an explanation refers to a description of the internal state or logic of an algorithm that leads to a decision, a counterfactual describe a dependency on the external facts that led to that decision [467]. This allows to have an actionable element in the explanation, allowing to know what could have caused a change in the prediction [468]. In short, a counterfactual is the answer to the question "If X had not occurred, Y would not have occurred.". It describes the smallest change to the feature values that changes the prediction to a predefined output [469]. Several methods exist to produce counterfactual explanations [470, 471, 472, 473, 474]. With our *Greybox XAI* framework, we propose to answer the question "Which object would have been predicted if a different object part had been detected?" to find the equivalent of a counterfactual to a specific prediction. We do not propose to calculate what would be the minimal change in the object prediction, or in the initial semantic segmentation, but rather to see what the final prediction would have been if the *Explainable Latent Space* had been different.

For a given attribute vector $z_{att,i}$ detected by the *Latent Space Predictor*, the probability that the predicted class y_i is the class k is expressed as follows:

$$p(y_i = k|z_{att,i}; \theta_h) = \frac{\exp\left(\theta_{h,j}^{\intercal} z_{att,i}\right)}{\sum_{j=1}^{K}(\theta_{h,j}^{\intercal} z_{att,i,j})} \tag{4.4.3}$$

with $(\theta_{h,j}^{\intercal}$ the weight matrix of the *Transparent Classifier*. The attribute vector $z_{att,i}$ being composed of 0 and 1, each probability $y_i = k$ is written as an exponential fraction of a sum of weights. Thus, it is straightforward to calculate what a change between a 1 and a 0 in the attribute vector $z_{att,i}$ would change in the prediction.

4.5 Discussion

Although we have shown that the *Greybox XAI* framework, a compositional transparent model, is capable of achieving an image classification with a satisfying accuracy, this is largely caused by the quality of the attributes present in the dataset describing the set of classes. Without the presence of discriminative attributes in the dataset, it would not be possible to perform this kind of compositional classification.

In addition, the *Transparent Classifier* accuracy depends on the *Latent Space Predictor* performances. Moreover, using a classification based on concrete features forsakes one of the advantages of neural networks, which is to use abstract features. Furthermore, this framework is only compatible with a classification task. Several usual Computer Vision tasks (segmentation, detection) cannot be explained by this framework. Therefore, there is no explainability progress for the *Latent Space Predictor*. Finally, the properties of

the attributes (position, dimension, numbers) are not used by the *Transparent Classifier* resulting in a loss of information.

However, the gain in explainability is important and the possibility to know exactly why a certain prediction happened is very useful. We hope this will motivate the research community to build datasets that inherently provide a greater granularity and hierarchy in the organisation of datasets, in order to move from the widely used (image, class) pair to triples (images, attributes, classes).

An important element to note about this model is its *actionability*. As we have seen in the examples of the Figures 4.10, 4.11 and 4.12, having the possibility to visualize the explanations of the predictions made by the model allows to explore ways to improve its performances, by correcting its predictions, or the validity of the explanations, by correcting the semantic segmentation.

4.6 Conclusions and Future Work

The contribution proposed in this chapter are:

- A formalisation of what is a "good" explanation. We propose 3 criteria - objectivity, intrinsicality, validity - to assert that an explanation is "good" or not.
- The *Greybox XAI*, a compositional framework for explainable classification. It is composed by a black box *Latent Space Predictor* and a *Transparent Classifier*.

We proved that this framework is transparent and produces "good" explanations, based on attributes segmented on the image. We tested it on 2 datasets and showed that it has SOTA results compared to compsotional models. Nevertheless, the accuracy is lower than for opaque end-to-end models like ResNet101.

One of the reasons why accuracy is not at the level of end-to-end models like Resnet101 is that the semantic segmentation performed by the Latent Space Predictor is not perfect. This means that the *Transparent Classifier* takes as input a sometimes distorted or incomplete representation of the RGB image to classify. Since the vectorization of the segmentation map do not include a gradient, it is not possible to perform a direct back-propagation between the *Transparent Classifier* and the *Latent Space Predictor*. Future work could therefore involve finding a way to influence the semantic segmentation using the *Transparent Classifier*. Also, the representation of the image in the *Explainable Latent Space* causes a significant loss of information because the attributes are not enumerated and their relative dimensions and positions are not taken into account. Future work will involve the encoding of this information in the *Explainable Latent Space*, in order to be

able to explain a prediction based on the size and position of the objects. Finally, it would be interesting to test this framework on a new benchmark dataset specifically containing images that differentiate between positional, numbering (counting), objects groupings and relational concepts like Kandinsky Patterns[475, 476]

Finally, let's take look at the Renaissance line in the Figure 4.5. We see that the attributes are not very discriminative in favor of this class: the highest positive value is a weight of 0.14 for the attribute *Serliana*. On the contrary, some attributes like *Ogee Arch* or *Column Salomnic* have a weight of -0.38 and -0.36, which implies that their presence leads to an absence of *Renaissance* monument. Thus, it may be difficult to recognize a renaissance monument because if a Serliana is not detected the class will have difficulty in gaining the upper hand. It is therefore interesting to check whether this particular attribute is detected with good accuracy. An attribute like *Triangular Pediment* which has for biggest absolute weight a 0.03 is almost useless in the prediction. We can therefore give it less importance during the semantic segmentation. Thus, the weight matrix could be used in the future to fine-tune the semantic segmentation so as to give importance to certain attributes in particular (the highest absolute value like *Horseshoe Arch* or *Ogee Arch*) and to neglect others.

Chapter 5

Transparent Distillation to teach expert knowledge

In recent years, neural networks have become increasingly deep. Today's most powerful ones have several million parameters, making them more and more cumbersome and opaque. In order to reduce their size while maintaining their high performance, model compression techniques have been developed. One of them, knowledge distillation, consists in using a large model as a teacher for a smaller, compressed student model. Rather than being trained only on labels and training images, the learning of the student model is supervised by that of a large teacher model. The idea is that by mimicking the teacher's behavior, the student will be able to achieve similar performance while using fewer parameters. Distillation systems are composed of a student-teacher architecture, a distillation algorithm and knowledge. The knowledge distilled from teacher to student is called dark knowledge. We consider that it does not necessarily have to be opaque if the goal is not only to imitate the teacher's performance. Thus, if this method is effective for model compression, our hypothesis is that it can also be effective for interpretability purposes if the knowledge transferred from the teacher to the student is explainable. We therefore propose a student-teacher architecture following a classical distillation protocol but inverting the usual size ratio between student and teacher: instead of distilling a large model into a small model, we distill a Knowledge Base into a Deep Neural Network through a transparent logistic regression. By ensuring that the student is faithful to the transparent model, we obtain explanatory elements on its behavior while having accurate predictions.

5.1 Related Work

Knowledge transfer from one model to another thought distillation has proven to be effective in several domains like data augmentation [477, 478], defense adversarial perturbations [479] and visual recognition [480]. Distillation also can be used to perform state representation learning [481], continual learning [482], sim2real transfer [483].

While the primary purpose of knowledge distillation methods is to do model compression [484, 485], experiments show that distillation affects the student learning [486, 487] and knowledge transfer has been widely used for different purposes [488]. Distillation has empirically proven to be very effective. However, few theoretical works have managed to provide answers as to why it works [489].

The logits are commonly used as a source of teacher information in knowledge distillation. It is called response-based knowledge and the main idea is to train a student to directly mimic the final prediction of the teacher model [490]. The most popular response-based knowledge distillation for classification involves soft targets, where the logits are divided by a temperature factor during the softmax function [79, 491]. Using soft targets is comparable to use label smoothing [492] or regularizers [493, 494].

Knowledge distillation can also be used for transferring representations discovered by huge black-box models into simpler, more interpretable models [495, 496, 497, 498, 499, 500]. Furthermore, we frequently wish to transfer attributes from bigger models, such as well-calibrated uncertainty, so that we may securely deploy more efficient models in their stead. It is critical in both circumstances to achieve high distillation fidelity [501].

5.2 Preliminaries on Knowledge Distillation

Following the formalism introduced in Stanton et al.[501], we focus on a classical supervised classification setting with an input space $\mathcal{X}$ and a label space $\mathcal{Y}$ of size n. We also introduce a sub-label space $\mathcal{Z}$. Let $f : \mathcal{X} \times \Theta \rightarrow \mathcal{R}^n$ be a classifier parametrized by $\theta \in \Theta$ whose outputs define a categorical predicted distribution over $\mathcal{Y}$, $\hat{p}(y = i|x) = \sigma_i(f(x, \theta))$, where:

$$\sigma_i(z) := \frac{\exp(z_i)}{\sum_j(z_j)} \tag{5.2.1}$$

is the softmax function, and z represents the outputs of classifier $f(x, \theta)$; we refer to them as logits.

The knowledge distillation that we will use in this chapter is the one originally described by Hinton et al. [79]. The student learns by minimizing a weighted combination, $\mathcal{L}_{student} :=$

$\alpha\mathcal{L}_{NLL} + (1-\alpha)\mathcal{L}_{KD}$, of 2 losses with $\alpha \in [0,1]$ -with $teacher$ and $student$ shortened to t and s-:

$$\mathcal{L}_{NLL}(z_s, y) := -\sum_{j=1}^{n} y_j \log \sigma_j(z_s) \tag{5.2.2}$$

$$\mathcal{L}_{KD}(z_s, z_t) := -T^2 \sum_{j=1}^{n} (\frac{z_t}{T}) \log \sigma_j(\frac{z_s}{T}) \tag{5.2.3}$$

where $\mathcal{L}_{NLL}$ is the usual cross-entropy between the student logits z_s and the label y and $\mathcal{L}_{KD}$ is the Kullback-Leibler divergence between the teacher logits z_t and the log-logits z_s. The T parameter is called Temperature, it is used to scale the logits and to smooth the probability distribution. If T is equal to 1 then the logits will not be scaled and will take the form of a usual softmax output. As T grows, the max of $\sigma_i(z)$ will decrease and thus let appear a more important granularity between the different possible outputs. This prevents an overconfident teacher from giving one logits close to 1 and all the others close to 0, which would finally have little difference with a one-hot label usually used in classical deep learning.

The hyper-parameter α is used to move the trade-off between using $\mathcal{L}_{NLL}$ and $\mathcal{L}_{KD}$. If it is set to 1, then the learning of the student will only be done through $\mathcal{L}_{NLL}$ and will be a supervised learning by classical labels. If it is set to 0 then the student will never use the labels in his learning, learning only through the teacher's expertise.

5.2.1 Preliminary Study

Our first concern is to verify if it is possible to distill a small model into a bigger one, as it is usually the opposite that is realized in the literature. To do so, we take a Naive Bayes classifier as a transparent teacher and we test different Deep Neural Networks architectures, for a classical image classification task.

Given a certain class C_i and an attribute vector $A_i = a_0, ..., a_n$ of all attributes (parts of objects that can possibly be present in an input), and the naive conditional independence assumption:

$$P(a_i|C, a_1, ..., a_{i-1}, a_{i+1}, ..., a_n) = P(a_i|C), \tag{5.2.4}$$

for all *i* and using the Bayes' theorem, we obtain the following Naive Bayes model:

$$P(C|a_1, ...a_n) = \frac{P(C)\Pi_{i=1}^{n} P(a_i|C)}{P(a_i, ..., a_n)} \tag{5.2.5}$$

As the denominator is constant, the obtained classification rule is [502]:

$$P(C|a_1, ...a_n) \propto P(C)\Pi_{i=1}^{n} P(a_i|C) \tag{5.2.6}$$

and thus,

$$\hat{C} = \underset{C}{\operatorname{argmax}}\, P(C)\Pi_{i=1}^{n} P(a_i|C) \tag{5.2.7}$$

where $\hat{C}$ is the NB model predicted class.

We tested this distillation on a modified version of the PASCAL-Part Dataset, a set of additional annotations for PASCAL VOC 2010. We deleted all attributes with height or width less than 6 pixels and removed the classes that have no attributes or that appear less than 20 times in the whole dataset. We have 12 classes (like Dog or $Train$) represented by 35 attributes (such as Leg or $ArtifactWing$). We tried several networks pre-trained on ImageNet. Table 5.1 shows the mean accuracy obtained, with and without distillation. We chose Xception and MobileNet as baselines in order to test the distillation on a network with a high number of model parameters and a smaller one.

Table 5.1: Mean Accuracy (over 100 runs) of the Student + Naive Bayes (NB) Teacher vs traditional setting (Student alone without distillation) for several networks on PASCAL-Part dataset.

Model	**Accuracy**
DenseNet201	88.59%
DenseNet201 Student + NB Teacher	**88.71**%
Xception	85.85%
Xception Student + NB Teacher	**87.08**%
MobileNet	70.49%
MobileNet Student + NB Teacher	**70.88**%
VGG16	64.92%
VGG16 Student + NB Teacher	**67.22**%

These results show us that it is possible to distill a teacher model into a student model without risking a loss of accuracy even if the teacher has far fewer parameters than the student.

5.3 Proposed Architecture

Inspired by the Neural-Symbolic framework proposed in [169], we propose an adaptation consisting in constraining the learning of a Neural Network through the distillation of a

Knowledge Base as it can be seen in Figure 5.1. It will be demonstrated in the context of a classification task of a set $\mathcal{X}$ of images, which represent classes of a set $\mathcal{Y}$, and which will have been sub-labeled at the pixel level in order to represent attributes of classes from a set $\mathcal{A}$.

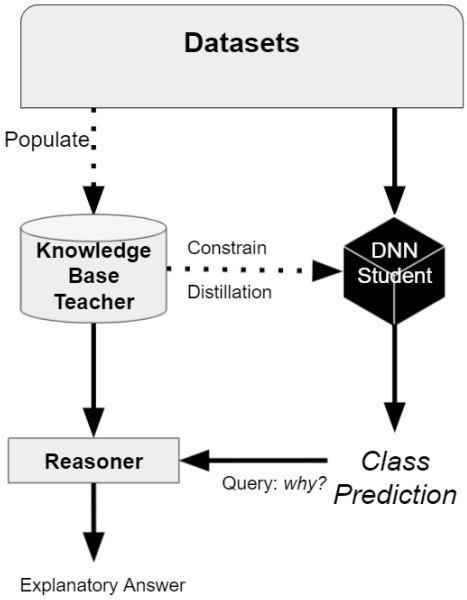


Figure 5.1: Proposed neural-symbolic framework from [169] adapted to constrain the student through distillation.

Since distillation works on the basis of logits as shown in section 5.2, the Knowledge base here is enacted by a transparent model. In the same way as for the Greybox XAI framework proposed in Chapter 4.3, a Transparent Classifier called *Transparent Teacher*, takes the form of a logistic regression model trained on the data in order to act as the statistical link between expert knowledge in form of object parts or attributes of classes, and the final class output.

Since Logistic regression is unable to handle images with ease, the teacher's training requires not only to have couples $\mathcal{D}_{couple} = \{(x_i, y_i)\}_{i=1}^{N}$, directly linking images and classes, but rather triples $\mathcal{D}_{triplet} = \{(x_i, a_i, y_i)\}_{i=1}^{N}$ where $\{a_i\}_{i=1}^{N} \in \mathcal{A}^N$. and $\mathcal{A}$ being attributes serving as a bridge between $\mathcal{X}$ and $\mathcal{Y}$.

The student takes the role of a state of the art Convolutional Neural Network (CNN). It was modified in order to make it multi-task, i.e. in addition to predicting the class y present on the image, it must also predict all attributes a present on the image. This is done with the help of two Fully Connected layers linked to the last convolutional block of the CNN.

Note that the layer for attributes prediction is entirely taken as input by the class prediction layer, so that these cannot be ignored and the final model can aim not only at producing right classifications, but also detecting the right attributes associated to this output class.

The complete architecture is shown in Figure 5.2.

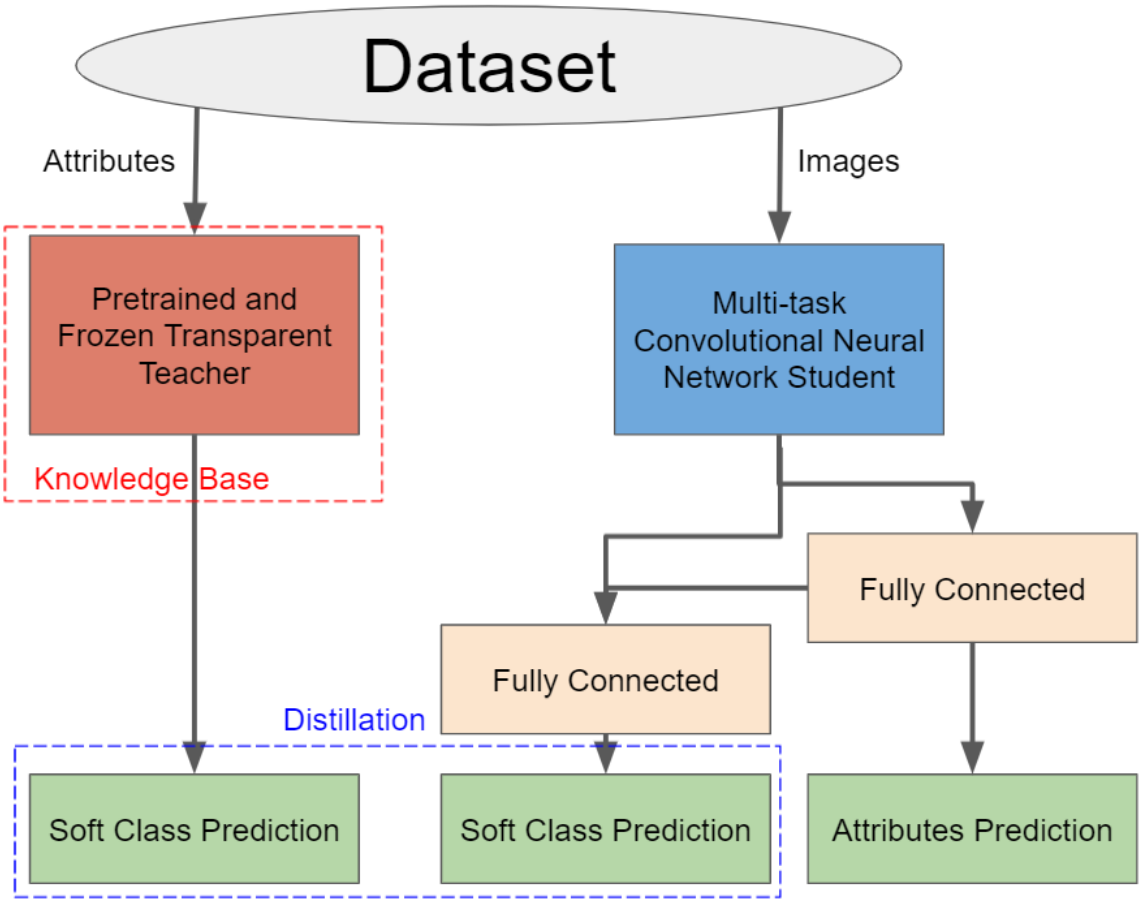


Figure 5.2: Proposed neural-symbolic framework from [169] adapted to constrain the student through distillation.

5.4 Metrics

Metrics considered can be divided in 2 branches , those related to the model performance and those related to the fidelity of the student to the teacher.

5.4.1 Model Performance

As any classification problem, we measure the accuracy of the student in term of class prediction. Since we also want the student to take into account the different attributes present on the image, we measure the accuracy of the prediction of the different attributes. Finally, in order to know if the attributes detected by the student are the most important ones for the classification, we measure the accuracy of the teacher when it takes as input the attributes seen by the student. Proceeding in this way allows to obtain 2 predictions for the same image: one end-to-end by the student and one compositional, with the teacher taking as input the attributes detected by the student

5.4.2 Model Fidelity to the transparent teacher model

We report the following metrics used for distillation defined in [501]:

$$\text{Average Agreement} := \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\{\arg\max_j \sigma_j(z_t, i) = \arg\max_j \sigma_j(z_s, i)\} \tag{5.4.1}$$

$$\text{Average KL Divergence} := \frac{1}{n}\sum_{i=1}^{n} \text{KL}(\hat{p_t}(y|a_i) \| \hat{p_s}(y|x_i)) \tag{5.4.2}$$

The Average Agreement is the average agreement between the teacher's and the student's prediction. The Average KL Divergence is the average divergence from the predictive distribution of the teacher to that of the student.

These two measures are used to account for the fidelity of the student's prediction with respect to the teacher's prediction. The objective is to be as faithful as possible in order to explain one by the other.

5.5 Results

We test our framework on the MonuMAI dataset. MonuMAI allows to classify architectural style classification from facade images [463, 449]; it includes 1092 high quality photographs, where the monument facade is centered and fills most of the image. Most images were taken by smartphone cameras thanks to the MonuMAI citizen science app. The rest of images were selected from the Internet. The dataset was annotated by art historian experts for two tasks, image classification and object detection. All images belong to facades of historical buildings that are labeled as one out of four different styles: Renaissance, Gothic, Baroque and Hispanic-Muslim. Besides this label given to an image, every image is labeled

with key architectural elements belonging to one of fourteen categories with a total of 4583 annotated elements.

We test 2 different configurations: one with distillation and one without distillation. In both cases, the teacher is exactly the same (frozen, with exactly the same weights). It is a simple logistic regression with only 64 parameters (4 classes times 16 attributes). The architecture of the student is similar in both cases with a Resnet152 as a backbone.

In the configuration without distillation, the student is trained by optimising a loss $\mathcal{L} = \alpha\mathcal{L}_{BCE} + (1-\alpha)\mathcal{L}_{NNL}$ with:

$$\mathcal{L}_{NLL}(z_s, y) := -\sum_{j=1}^{n} y_j \log \sigma_j(z_s) \tag{5.5.1}$$

$$\mathcal{L}_{BCE}(z_s, a) := -(a_{j,c} \log \sigma_j(z_{s,c}) + (1 - a_{j,c}) \log(1 - \sigma_j(z_{s,c}))) \tag{5.5.2}$$

where c is the number of attributes. With distillation, it is optimized with $\mathcal{L}_{student}$ from Equation 5.2.3.

The results obtained on this dataset are summarized in Table 5.2.

Configuration	Student Class Accuracy	Teacher Class Accuracy	Student Attributes Accuracy	Agreement	KL Divergence
Resnet152 *with Distillation*	97.68%	**94.30%**	**89.87%**	**95.03%**	**0.0694**
Resnet152 *without Distillation*	97.68%	93.70%	88.80%	94.70%	0.26

Table 5.2: Results on MonuMAI dataset with and without Distillation, with the same backbone network (Resnet152) and a Logistic Regression as a teacher.

From a performance point of view, the difference with and without distillation is not significant although the attributes seem to be slightly better detected with distillation. The major difference is in the interpretability: distillation allows the student to have a prediction that is faithful to the one made by the teacher. Furthermore, the accuracy of the teacher's prediction with the attributes detected by the student is lower than the same being done end-to-end by the student. This is normal in the sense that the teacher has very few parameters and is totally transparent, illustrating again the trade-off between performance and explainability. Also, the teacher-student pair agreed on 95% of the predictions. This means that on 95% of the images, taking the prediction of the teacher or the student is the same.

5.6 Discussion

First of all, it is important to note that the teacher is totally transparent. It is a simple one-layer logistic regression, linking attributes and classes through a weight matrix computed after fitting the regressor.

This framework gives 2 predictions: the student's one, based on the image, and the teacher's one, based on the attributes seen by the student. The student's prediction becomes slightly more accurate than the teacher's one. Both models agree 95% of the time.

We end up with some interesting remarks:

- In 95% of the cases, the teacher and the student agree. In this case, it is therefore possible to substitute the prediction of the Deep Neural Network by that of the Logistic Regression, and thus to have a fully explainable prediction. The explanation would then take the form *This image contains an object is of this class because those attributes were detected*.

- In 5% of the cases, the teacher and the student disagree. Starting from the assumption that the student is slightly more often correct than the teacher, there are 2 possible actions:

  - Trust the teacher. On the basis of the explanation given by the teacher, the human user can check if it is plausible and choose to believe it.
  - Trust the student. In this case, there is no causal explanation of why the prediction was made. We simply know which attributes were detected but not whether they caused the final prediction. We do know, however, from the low KL Divergence that the probability distribution predicted by the student diverges slightly from that given by the teacher, although they do not agree on the Top-1 prediction.

From an XAI perspective, the impact of distillation is not obvious. If it is possible to explain at least 95% of the predictions, this is mainly due to the multi-tasking nature of the framework. The distillation simply served to reconcile the probability distributions of the two models but does not imply the learning of a causal link by the student.

The relationship between interpretability and fidelity can be discussed. Making the student more faithful to the teacher allows us to have insights on his functioning but there are still too many grey areas to explain his decisions.

- Even if the presence/absence of attributes is concatenated in the input of the classifying layer, this represents only few neurons compared to the convolution block.

- There is no evidence that distillation causes a better focus on attributes by the student.
- KL Divergence is a statistical distance and not a metric, as it's not symmetric. Thus, it does not express a distance but rather a loss of information that would occur if we approximated the probability distribution of the student by the one of the teacher. It is not causal or contrastive.

These works on distillation are not yet fully concluding and many avenues of research are open. First of all, it would be interesting to verify if distillation and multi-tasking allow to modify the student's attention, making him focus on more relevant parts of the image. This could be done using a Score-Cam [503]. Then, performing another type of distillation might be wise. We have only used Response-based distillation, which refers to the neural response of the last output layer of the teacher model. However, the interest of using a transparent teacher is more to transfer the expert knowledge it contains, i.e. the causal link between the attributes and the classes. It would thus be more interesting to distil the weight matrix of the logistic regression directly into a layer of the student. Distilling Relation-based knowledge, in order to transmit knowledge on the relations between different instances could thus favour the interest of this distillation for interpretability purposes. Many works exist on Relation-Based Distillation [504, 505, 506, 507, 508, 509, 510, 511, 512] and it would be interesting to integrate it into our framework.

Chapter 6

Conclusion

XAI, as discussed in this thesis, is a rapidly emerging topic of critical relevance to society. The ever-increasing complexity of AI models together with the growing need for their application in a variety of real-world scenarios has given rise to this field. This thesis has contributed to the development of the field on theoretical and practical aspects which are summarized below.

6.1 Summary

- **Surveying the field of XAI**. We proposed a set of concepts that summarize the diverse references found in the literature, trying to build a consensus around it. This is accompanied by a new taxonomy to classify explainability methods. We thoroughly analyze the literature on XAI and related concepts published to date, covering approximately 400 contributions arranged into two different taxonomies. The first taxonomy addresses the explainability of ML models using the previously made distinction between transparency and post-hoc explainability, including models that are transparent by themselves, Deep and non-Deep (i.e., *shallow*) learning models. The second taxonomy deals with XAI methods suited for the explanation of Deep Learning models, using classification criteria closely linked to this family of ML methods (e.g. layerwise explanations, representation vectors, attention). We enumerate a series of challenges of XAI that still remain insufficiently addressed to date. Specifically, we identify research needs around the concepts and metrics to evaluate the explainability of ML models, and outline research directions toward making Deep Learning models more understandable. We further augment the scope of our prospects toward the implications of XAI techniques in regards to confidentiality, robustness in adversarial settings, data diversity, and other areas intersecting with

explainability.

- *Neural-Symbolic learning framework.* Models combining connectionism and symbolism are not widely represented in the state of the art of XAI. These paradigms are rarely combined when providing explanations. The use of a symbolic basis with a neural network can provide explanations close to the functioning of human reasoning while maintaining the state-of-the-art performance at the same time. We proposed a neural-symbolic learning framework, allowing to produce predictions with a neural network while making them explicit through a symbolic knowledge base. This framework is endowed with a non-external KB, i.e., directly built on the learning data of the neural network, that allows to influence its learning and to correct bias thoroughly while giving a fair explanation from its predictions. As the user or expert external knowledge does not interfere the predictions in the explanation process, it constitutes a truly explainable model that is faithful to communicate the reasoning behind its output decisions. This framework is accompanied by a use case showing an example of use.

- *Greybox XAI Framework.* We propose a compositional model designed to be transparent according to the definition of transparency. This framework is also made to produce *good* explanations in relation to the 3 criteria of objectivity, intrinsicality and validity. The goal of this framework is to perform compositional image classification and to explain its predictions by the different *parts-of* the object that has been detected. It consists of two separately trained models: 1) A Deep Neural Network trained to predict a segmentation map from an RGB image input. Its purpose is to detect the different *parts-of* objects that constitute the image. 2) A transparent model trained to predict an object, using as input a vector encoding the presence and absence of *parts-of* objects. These two models are linked in a sequential manner: the output of the DNN is transformed into a vector serving as input to the transparent model. We call the space in which the transformation is carried out the *Explainable Latent Space*. In this space, the predicted segmentation map is transformed into a one-hot vector. This vector indicates all the *parts-of* objects present in the segmentation map. The transparent model classifies this vector. It gives a prediction of the object present on the RGB image according to the different *parts-of* it. It is then possible to produce an explanation of this classification based on the *Explainable Latent Space* and the transparent model computation.

- *Transparent Distillation.* We propose a student-teacher architecture following a classical distillation protocol but inverting the usual size ratio between student and teacher: instead of distilling a large model into a small model, we distill a transparent

model that serves as theoretical and expert-like Knowledge Base of interpretations into a Deep Neural Network through a transparent logistic regression. By ensuring that the student is faithful to the transparent model, we obtain explanatory elements on its behavior while having accurate predictions.

6.2 Future research lines

The different results obtained during this thesis have opened the way to research areas that deserve to be explored. These can be divided into 4 axes, following the structure of the different chapters of this manuscript.

- *Theory around explainability*. While the field has evolved over the past three years, many gaps in the literature still seem to exist. Our reflections have led to the creation of a taxonomy and a set of definitions covering important methods and notions of the field. However, there is still no consensus on what constitutes a *good explanation*. Many metrics have emerged to quantify the quality of explanations and explanatory elements that can be given by models but there is no metric as consensual as those that may exist on other AI tasks, such as accuracy for image classification. Thus, the creation of XAI frameworks is slowed down by the difficulty to find a way to quantify the gain in explainability brought by a method, especially when this gain is done at the expense of other characteristics of the model. Therefore, we believe that it is necessary to work on a standardization of explainability metrics.

- *Neural-Symbolic learning framework*. The neural-symbolic learning paradigm is promising and could explain the predictions of a black-box through an expert knowledge base. The framework proposed in this sense during this thesis faces the difficulty of representing the KB in the neural network, constraining it in some way. Even if attempts have been made to transfer this knowledge through expert optimization functions or knowledge distillation, these two methods present flaws. It would be interesting to try to find new methods allowing to make this link between expert knowledge and neural networks and, in particular, to make sure that it has been well realized, without sacrificing the capacity of generalization and abstraction specific to deep neural networks.

- *Greybox XAI Framework*. This learning framework has shown interesting results. However, it depends heavily on the ability of an opaque model to detect the different sub-parts present in an image. Moreover, this detection requires a very good quality data, with a labeling at the pixel level which can be expensive. Finally, this compositional explanation system raises the question of how far we can explain something.

For example, we can say that we recognize a human because we detect arms, a torso and legs. We can say that we recognize arms because we detect a bicep, a forearm and hands. We recognize a hand by the fingers, the nails, but then, how to justify that we recognize a nail except by its texture? And how to explain why we recognized a texture? This raises the question of the granularity of the explanation, how far can we go back in a recursive way when we justify a classification based on the detection of parts of objects. Whatever level model designers decide to stop explaining, it seems that we will always end up with a not answered "Why?".

- *Transparent Distillation*. Surely the most interesting part as it approaches human development, with a teacher who transmits his knowledge to a student. During this thesis, the approach that was favored was to try to transmit this knowledge by means of a copy of the answers of the teachers by the student. However, this is not exactly how we learn as human beings: teachers transmit a reasoning, a way of seeing things allowing students to understand, adapt and generalize. Thus, it could be interesting to continue on this path of transparent knowledge distillation by trying to transfer knowledge that is more significant than a simple output. Transferring a symbolic representation of knowledge, logical rules or simply a causal link between different notions could improve the use of distillation for interpretability purposes.

6.3 Publications during this thesis

[1] Adrien Bennetot, Jean-Luc Laurent, Raja Chatila, Natalia Díaz-Rodríguez, "Towards Explainable Neural-Symbolic Visual Reasoning", at IJCAI19 Neural-Symbolic Learning and Reasoning Workshop

[2] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio GilLópez, Benjamins Richard Molina Daniel, et al. “Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI.” In: Information Fusion 58 (2020), pp. 82–115.

[3] Adrien Bennetot, Vicky Charisi, Natalia Díaz-Rodríguez, "Should artificial agents ask for help in human-robot collaborative problem-solving?", at Brain-PIL Workshop - ICRA2020

[4] Adrien Bennetot, Ivan Donadello, Ayoub El Qadi, Mauro Dragoni, Thomas Frossard, Benedikt Wagner, Anna Saranti, Silvia Tulli, Maria Trocan, Raja Chatila, Andreas

Holzinger, Artur d'Avila Garcez, Natalia Díaz-Rodríguez, "A Practical Tutorial on Explainable AI Techniques" [UNDER REVIEW]

[5] Adrien Bennetot, Gianni Franchi, Javier Del Ser, Raja Chatila, Natalia Díaz-Rodríguez, "Greybox XAI: a Neural-Symbolic learning framework to produce interpretable predictions for image classification" [UNDER REVIEW]

[6] Ikram Chraibi Kaadoud, Adrien Bennetot, Barbara Mawhin, Vicky Charisi, Natalia Díaz-Rodríguez, "Explaining *Aha!* moments in artificial agents through IKE-XAI: Implicit Knowledge Extraction for eXplainable AI" [UNDER REVIEW]

Bibliography

[1] D. Castelvecchi, Can we open the black box of AI?, Nature News 538 (7623) (2016) 20.

[2] Z. C. Lipton, The mythos of model interpretability, Queue 16 (3) (2018) 30:31–30:57. doi:10.1145/3236386.3241340.
URL http://doi.acm.org/10.1145/3236386.3241340

[3] D. Gunning, Explainable artificial intelligence (xAI), Tech. rep., Defense Advanced Research Projects Agency (DARPA) (2017).

[4] E. Tjoa, C. Guan, A survey on explainable artificial intelligence (XAI): Towards medical XAI (2019). arXiv:1907.07374.

[5] C. Metta, A. Beretta, R. Guidotti, Y. Yin, P. Gallinari, S. Rinzivillo, F. Giannotti, Explainable deep image classifiers for skin lesion diagnosis (2021). doi:10.48550/ARXIV.2111.11863.
URL https://arxiv.org/abs/2111.11863

[6] A. Preece, D. Harborne, D. Braines, R. Tomsett, S. Chakraborty, Stakeholders in Explainable AI (2018). arXiv:1810.00184.

[7] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, N. Elhadad, Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '15, 2015, pp. 1721–1730.

[8] N. Papernot, P. McDaniel, Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning (2018). arXiv:1803.04765.

[9] J. Zhu, A. Liapis, S. Risi, R. Bidarra, G. M. Youngblood, Explainable AI for designers: A human-centered perspective on mixed-initiative co-creation, 2018 IEEE Conference on Computational Intelligence and Games (CIG) (2018) 1–8.

[10] B. Goodman, S. Flaxman, European union regulations on algorithmic decision-making and a "right to explanation", AI Magazine 38 (3) (2017) 50–57.

[11] F. K. Došilović, M. Brčić, N. Hlupić, Explainable artificial intelligence: A survey, in: 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2018, pp. 210–215.

[12] P. Hall, On the Art and Science of Machine Learning Explanations , arXiv e-prints (2018) arXiv:1810.02909arXiv:1810.02909.

[13] T. Miller, Explanation in artificial intelligence: Insights from the social sciences, Artificial Intelligence 267 (2019) 1–38.

[14] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, L. Kagal, Explaining Explanations: An Overview of Interpretability of Machine Learning, arXiv e-prints (2018) arXiv:1806.00069arXiv:1806.00069.

[15] A. Adadi, M. Berrada, Peeking inside the black-box: A survey on explainable artificial intelligence (XAI), IEEE Access 6 (2018) 52138–52160.

[16] O. Biran, C. Cotton, Explanation and justification in machine learning: A survey, in: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI) 2017 workshop on explainable AI (XAI), Vol. 8, 2017, p. 1.

[17] S. T. Shane T. Mueller, R. R. Hoffman, W. Clancey, G. Klein, Explanation in Human-AI Systems: A Literature Meta-Review Synopsis of Key Ideas and Publications and Bibliography for Explainable AI, Tech. rep., Defense Advanced Research Projects Agency (DARPA) XAI Program (2019).

[18] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, D. Pedreschi, A survey of methods for explaining black box models, ACM Comput. Surv. 51 (5) (2018) 93:1–93:42. doi:10.1145/3236009.
URL http://doi.acm.org/10.1145/3236009

[19] G. Montavon, W. Samek, K.-R. Müller, Methods for interpreting and understanding deep neural networks, Digital Signal Processing 73 (2018) 1–15. doi:10.1016/j.dsp.2017.10.011.

[20] A. Fernandez, F. Herrera, O. Cordon, M. Jose del Jesus, F. Marcelloni, Evolutionary fuzzy systems for explainable artificial intelligence: Why, when, what for, and where to?, IEEE Computational Intelligence Magazine 14 (1) (2019) 69–81. doi:10.1109/MCI.2018.2881645.

[21] M. Gleicher, A framework for considering comprehensibility in modeling, Big data 4 (2) (2016) 75–88.

[22] M. W. Craven, Extracting comprehensible models from trained neural networks, Tech. rep., University of Wisconsin-Madison Department of Computer Sciences (1996).

[23] R. S. Michalski, A theory and methodology of inductive learning, in: Machine learning, Springer, 1983, pp. 83–134.

[24] J. Díez, K. Khalifa, B. Leuridan, General theories of explanation: buyer beware, Synthese 190 (3) (2013) 379–396.

[25] D. Doran, S. Schulz, T. R. Besold, What does explainable AI really mean? a new conceptualization of perspectives (2017). arXiv:1710.00794.

[26] F. Doshi-Velez, B. Kim, Towards a rigorous science of interpretable machine learning (2017). arXiv:1702.08608.

[27] A. Vellido, J. D. Martín-Guerrero, P. J. Lisboa, Making machine learning models interpretable., in: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), Vol. 12, Citeseer, 2012, pp. 163–172.

[28] E. Walter, Cambridge advanced learner's dictionary, Cambridge University Press, 2008.

[29] P. Besnard, A. Hunter, Elements of Argumentation, The MIT Press, 2008.

[30] F. Rossi, AI Ethics for Enterprise AI (2019).
URL https://economics.harvard.edu/files/economics/files/rossi-francesca_4-22-19_ai-ethics-for-enterprise-ai_ec3118-hbs.pdf

[31] M. T. Ribeiro, S. Singh, C. Guestrin, Why should I trust you?: Explaining the predictions of any classifier, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2016, pp. 1135–1144.

[32] M. Fox, D. Long, D. Magazzeni, Explainable planning (2017). arXiv:1709.10256.

[33] H. C. Lane, M. G. Core, M. Van Lent, S. Solomon, D. Gomboc, Explainable artificial intelligence for training and tutoring, Tech. rep., University of Southern California (2005).

[34] W. J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, B. Yu, Interpretable machine learning: definitions, methods, and applications (2019). arXiv:1901.04592.

[35] J. Haspiel, N. Du, J. Meyerson, L. P. Robert Jr, D. Tilbury, X. J. Yang, A. K. Pradhan, Explanations and expectations: Trust building in automated vehicles, in: Companion of the ACM/IEEE International Conference on Human-Robot Interaction, ACM, 2018, pp. 119–120.

[36] A. Chander, R. Srinivasan, S. Chelian, J. Wang, K. Uchino, Working with beliefs: AI transparency in the enterprise., in: Workshops of the ACM Conference on Intelligent User Interfaces, 2018.

[37] A. B. Tickle, R. Andrews, M. Golea, J. Diederich, The truth will come to light: Directions and challenges in extracting the knowledge embedded within trained artificial neural networks, IEEE Transactions on Neural Networks 9 (6) (1998) 1057–1068.

[38] C. Louizos, U. Shalit, J. M. Mooij, D. Sontag, R. Zemel, M. Welling, Causal effect inference with deep latent-variable models, in: Proceedings of the International Conference on Neural Information Processing Systems, 2017, pp. 6446–6456.

[39] O. Goudet, D. Kalainathan, P. Caillou, I. Guyon, D. Lopez-Paz, M. Sebag, Learning functional causal models with generative neural networks, in: Explainable and Interpretable Models in Computer Vision and Machine Learning, Springer, 2018, pp. 39–80.

[40] S. Athey, G. W. Imbens, Machine learning methods for estimating heterogeneous causal effects, stat 1050 (5) (2015).

[41] D. Lopez-Paz, R. Nishihara, S. Chintala, B. Scholkopf, L. Bottou, Discovering causal signals in images, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2017, pp. 6979–6987.

[42] C. Barabas, K. Dinakar, J. Ito, M. Virza, J. Zittrain, Interventions over predictions: Reframing the ethical debate for actuarial risk assessment (2017). arXiv:1712.08238.

[43] A. Theodorou, R. H. Wortham, J. J. Bryson, Designing and implementing transparency for real time inspection of autonomous robots, Connection Science 29 (3) (2017) 230–241.

[44] A. Holzinger, C. Biemann, C. S. Pattichis, D. B. Kell, What do we need to build explainable AI systems for the medical domain? (2017). arXiv:1712.09923.

[45] W. Samek, T. Wiegand, K.-R. Müller, Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models (2017). arXiv:1708.08296.

[46] C. Wadsworth, F. Vera, C. Piech, Achieving fairness through adversarial learning: an application to recidivism prediction (2018). arXiv:1807.00199.

[47] X. Yuan, P. He, Q. Zhu, X. Li, Adversarial examples: Attacks and defenses for deep learning, IEEE Transactions on Neural Networks and Learning Systems 30 (9) (2019) 2805–2824.

[48] B. Letham, C. Rudin, T. H. McCormick, D. Madigan, et al., Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model, The Annals of Applied Statistics 9 (3) (2015) 1350–1371.

[49] M. Harbers, K. van den Bosch, J.-J. Meyer, Design and evaluation of explainable BDI agents, in: IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, Vol. 2, IEEE, 2010, pp. 125–132.

[50] M. H. Aung, P. G. Lisboa, T. A. Etchells, A. C. Testa, B. Van Calster, S. Van Huffel, L. Valentin, D. Timmerman, Comparing analytical decision support models through boolean rule extraction: A case study of ovarian tumour malignancy, in: International Symposium on Neural Networks, Springer, 2007, pp. 1177–1186.

[51] A. Weller, Challenges for transparency (2017). arXiv:1708.01870.

[52] A. A. Freitas, Comprehensible classification models: a position paper, ACM SIGKDD explorations newsletter 15 (1) (2014) 1–10.

[53] V. Schetinin, J. E. Fieldsend, D. Partridge, T. J. Coats, W. J. Krzanowski, R. M. Everson, T. C. Bailey, A. Hernandez, Confident interpretation of bayesian decision tree ensembles for clinical applications, IEEE Transactions on Information Technology in Biomedicine 11 (3) (2007) 312–319.

[54] D. Martens, J. Vanthienen, W. Verbeke, B. Baesens, Performance of classification models from a user perspective, Decision Support Systems 51 (4) (2011) 782–793.

[55] Z. Che, S. Purushotham, R. Khemani, Y. Liu, Interpretable deep models for ICU outcome prediction, in: AMIA Annual Symposium Proceedings, Vol. 2016, American Medical Informatics Association, 2016, p. 371.

[56] N. Barakat, J. Diederich, Eclectic rule-extraction from support vector machines, International Journal of Computer, Electrical, Automation, Control and Information Engineering 2 (5) (2008) 1672–1675.

[57] F. J. C. Garcia, D. A. Robb, X. Liu, A. Laskov, P. Patron, H. Hastie, Explain yourself: A natural language interface for scrutable autonomous robots (2018). arXiv:1803.02088.

[58] P. Langley, B. Meadows, M. Sridharan, D. Choi, Explainable agency for intelligent autonomous systems, in: Proceedings of the 30th AAAI Conference on Artificial Intelligence, 2017, pp. 4762–4763.

[59] G. Montavon, S. Lapuschkin, A. Binder, W. Samek, K.-R. Müller, Explaining nonlinear classification decisions with deep taylor decomposition, Pattern Recognition 65 (2017) 211–222.

[60] P.-J. Kindermans, K. T. Schütt, M. Alber, K.-R. Müller, D. Erhan, B. Kim, S. Dähne, Learning how to explain neural networks: Patternnet and patternattribution (2017). arXiv:1705.05598.

[61] G. Ras, M. van Gerven, P. Haselager, Explanation methods in deep learning: Users, values, concerns and challenges, in: Explainable and Interpretable Models in Computer Vision and Machine Learning, Springer, 2018, pp. 19–36.

[62] S. Bach, A. Binder, K.-R. Müller, W. Samek, Controlling explanatory heatmap resolution and semantics via decomposition depth, in: IEEE International Conference on Image Processing (ICIP), IEEE, 2016, pp. 2271–2275.

[63] G. J. Katuwal, R. Chen, Machine learning model interpretability for precision medicine (2016). arXiv:1610.09045.

[64] M. A. Neerincx, J. van der Waa, F. Kaptein, J. van Diggelen, Using perceptual and cognitive explanations for enhanced human-agent team performance, in: International Conference on Engineering Psychology and Cognitive Ergonomics, Springer, 2018, pp. 204–214.

[65] J. D. Olden, D. A. Jackson, Illuminating the "black box": a randomization approach for understanding variable contributions in artificial neural networks, Ecological modelling 154 (1-2) (2002) 135–150.

[66] J. Krause, A. Perer, K. Ng, Interacting with predictions: Visual inspection of black-box machine learning models, in: CHI Conference on Human Factors in Computing Systems, ACM, 2016, pp. 5686–5697.

[67] L. Rosenbaum, G. Hinselmann, A. Jahn, A. Zell, Interpreting linear support vector machine models with heat map molecule coloring, Journal of Cheminformatics 3 (1) (2011) 11.

[68] J. Tan, M. Ung, C. Cheng, C. S. Greene, Unsupervised feature construction and knowledge extraction from genome-wide assays of breast cancer with denoising autoencoders, in: Pacific Symposium on Biocomputing Co-Chairs, World Scientific, 2014, pp. 132–143.

[69] S. Krening, B. Harrison, K. M. Feigh, C. L. Isabell, M. Riedl, A. Thomaz, Learning from explanations using sentiment and advice in RL, IEEE Transactions on Cognitive and Developmental Systems 9 (1) (2017) 44–55.

[70] M. T. Ribeiro, S. Singh, C. Guestrin, Model-agnostic interpretability of machine learning (2016). arXiv:1606.05386.

[71] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, W. Samek, On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation, PloS one 10 (7) (2015) e0130140.

[72] T. A. Etchells, P. J. Lisboa, Orthogonal search-based rule extraction (OSRE) for trained neural networks: a practical and efficient approach, IEEE Transactions on Neural Networks 17 (2) (2006) 374–384.

[73] Y. Zhang, S. Sreedharan, A. Kulkarni, T. Chakraborti, H. H. Zhuo, S. Kambhampati, Plan explicability and predictability for robot task planning, in: 2017 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2017, pp. 1313–1320.

[74] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, T. Lillicrap, A simple neural network module for relational reasoning, in: Proceedings of the International Conference on Neural Information Processing Systems, 2017, pp. 4967–4976.

[75] C.-Y. J. Peng, T.-S. H. So, F. K. Stage, E. P. S. John, The use and interpretation of logistic regression in higher education journals: 1988–1999, Research in Higher Education 43 (3) (2002) 259–293.

[76] B. Üstün, W. Melssen, L. Buydens, Visualisation and interpretation of support vector regression models, Analytica Chimica Acta 595 (1-2) (2007) 299–309.

[77] Q. Zhang, Y. Yang, H. Ma, Y. N. Wu, Interpreting CNNs via decision trees, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2019, pp. 6261–6270.

[78] M. Wu, M. C. Hughes, S. Parbhoo, M. Zazzi, V. Roth, F. Doshi-Velez, Beyond sparsity: Tree regularization of deep models for interpretability, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2018, pp. 1670–1678.

[79] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network (2015). arXiv:1503.02531.

[80] N. Frosst, G. Hinton, Distilling a neural network into a soft decision tree (2017). arXiv:1711.09784.

[81] M. G. Augasta, T. Kathirvalavakumar, Reverse engineering the neural networks for rule extraction in classification problems, Neural Processing Letters 35 (2) (2012) 131–150.

[82] Z.-H. Zhou, Y. Jiang, S.-F. Chen, Extracting symbolic rules from trained neural network ensembles, AI Communications 16 (1) (2003) 3–15.

[83] H. F. Tan, G. Hooker, M. T. Wells, Tree space prototypes: Another look at making tree ensembles interpretable (2016). arXiv:1611.07115.

[84] R. C. Fong, A. Vedaldi, Interpretable explanations of black boxes by meaningful perturbation, in: IEEE International Conference on Computer Vision, 2017, pp. 3429–3437.

[85] T. Miller, P. Howe, L. Sonenberg, Explainable AI: Beware of inmates running the asylum, in: International Joint Conference on Artificial Intelligence, Workshop on Explainable AI (XAI), Vol. 36, 2017, pp. 36–40.

[86] R. Goebel, A. Chander, K. Holzinger, F. Lecue, Z. Akata, S. Stumpf, P. Kieseberg, A. Holzinger, Explainable AI: the new 42?, in: International Cross-Domain Conference for Machine Learning and Knowledge Extraction, Springer, 2018, pp. 295–303.

[87] V. Belle, Logic meets probability: Towards explainable AI systems for uncertain worlds, in: International Joint Conference on Artificial Intelligence, 2017, pp. 5116–5120.

[88] L. Edwards, M. Veale, Slave to the algorithm: Why a right to an explanation is probably not the remedy you are looking for, Duke L. & Tech. Rev. 16 (2017) 18.

[89] Y. Lou, R. Caruana, J. Gehrke, G. Hooker, Accurate intelligible models with pairwise interactions, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2013, pp. 623–631.

[90] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, Y. Bengio, Show, attend and tell: Neural image caption generation with visual attention, in: Proceedings of the International Conference on Machine Learning, PMLR, 2015, pp. 2048–2057.

[91] J. Huysmans, K. Dejaeger, C. Mues, J. Vanthienen, B. Baesens, An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models, Decision Support Systems 51 (1) (2011) 141–154.

[92] N. H. Barakat, A. P. Bradley, Rule extraction from support vector machines: A sequential covering approach, IEEE Transactions on Knowledge and Data Engineering 19 (6) (2007) 729–741.

[93] F. C. Adriana da Costa, M. M. B. Vellasco, R. Tanscheit, Fuzzy rule extraction from support vector machines, in: International Conference on Hybrid Intelligent Systems, IEEE, 2005, pp. 335–340.

[94] D. Martens, B. Baesens, T. Van Gestel, J. Vanthienen, Comprehensible credit scoring models using rule extraction from support vector machines, European Journal of Operational Research 183 (3) (2007) 1466–1476.

[95] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2016, pp. 2921–2929.

[96] R. Krishnan, G. Sivakumar, P. Bhattacharya, Extracting decision trees from trained neural networks, Pattern Recognition 32 (12) (1999) 1999–2009.

[97] X. Fu, C. Ong, S. Keerthi, G. G. Hung, L. Goh, Extracting the knowledge embedded in support vector machines, in: IEEE International Joint Conference on Neural Networks, Vol. 1, IEEE, 2004, pp. 291–296.

[98] B. Green, "Fair" risk assessments: A precarious approach for criminal justice reform, in: 5th Workshop on Fairness, Accountability, and Transparency in Machine Learning, 2018.

[99] A. Chouldechova, Fair prediction with disparate impact: A study of bias in recidivism prediction instruments, Big Data 5 (2) (2017) 153–163.

[100] M. Kim, O. Reingold, G. Rothblum, Fairness through computationally-bounded awareness, in: Proceedings of the International Conference on Neural Information Processing Systems, 2018, pp. 4842–4852.

[101] B. Haasdonk, Feature space interpretation of SVMs with indefinite kernels, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (4) (2005) 482–492.

[102] A. Palczewska, J. Palczewski, R. M. Robinson, D. Neagu, Interpreting random forest classification models using a feature contribution method, in: Integration of Reusable Systems, Springer, 2014, pp. 193–218.

[103] S. H. Welling, H. H. Refsgaard, P. B. Brockhoff, L. H. Clemmensen, Forest floor visualizations of random forests (2016). arXiv:1605.09196.

[104] G. Fung, S. Sandilya, R. B. Rao, Rule extraction from linear support vector machines, in: ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, ACM, 2005, pp. 32–40.

[105] Y. Zhang, H. Su, T. Jia, J. Chu, Rule extraction from trained support vector machines, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2005, pp. 61–70.

[106] D. Linsley, D. Shiebler, S. Eberhardt, T. Serre, Global-and-local attention networks for visual recognition (2018). arXiv:1805.08819.

[107] S.-M. Zhou, J. Q. Gan, Low-level interpretability and high-level interpretability: a unified view of data-driven interpretable fuzzy system modelling, Fuzzy Sets and Systems 159 (23) (2008) 3091–3131.

[108] J. Burrell, How the machine 'thinks': Understanding opacity in machine learning algorithms, Big Data & Society 3 (1) (2016) 1–12.

[109] A. Shrikumar, P. Greenside, A. Shcherbina, A. Kundaje, Not just a black box: Learning important features through propagating activation differences (2016). arXiv:1605.01713.

[110] Y. Dong, H. Su, J. Zhu, B. Zhang, Improving interpretability of deep neural networks with semantic information, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2017, pp. 4306–4314.

[111] G. Ridgeway, D. Madigan, T. Richardson, J. O'Kane, Interpretable boosted naïve bayes classification., in: ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 1998, pp. 101–104.

[112] Q. Zhang, Y. Nian Wu, S.-C. Zhu, Interpretable convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2018, pp. 8827–8836.

[113] S. Seo, J. Huang, H. Yang, Y. Liu, Interpretable convolutional neural networks with dual local and global attention for review rating prediction, in: Proceedings of the Eleventh ACM Conference on Recommender Systems, ACM, 2017, pp. 297–305.

[114] K. Larsen, J. H. Petersen, E. Budtz-Jørgensen, L. Endahl, Interpreting parameters in the logistic regression model with random effects, Biometrics 56 (3) (2000) 909–914.

[115] B. Gaonkar, R. T. Shinohara, C. Davatzikos, A. D. N. Initiative, et al., Interpreting support vector machine models for multivariate group wise analysis in neuroimaging, Medical image analysis 24 (1) (2015) 190–204.

[116] K. Xu, D. H. Park, C. Yi, C. Sutton, Interpreting deep classifier by visual distillation of dark knowledge (2018). arXiv:1803.04042.

[117] H. Deng, Interpreting tree ensembles with intrees (2014). arXiv:1408.5456.

[118] P. Domingos, Knowledge discovery via multiple models, Intelligent Data Analysis 2 (1-4) (1998) 187–202.

[119] S. Tan, R. Caruana, G. Hooker, Y. Lou, Distill-and-compare: Auditing black-box models using transparent model distillation, in: Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, ACM, 2018, pp. 303–310.

[120] R. A. Berk, J. Bleich, Statistical procedures for forecasting criminal behavior: A comparative assessment, Criminology & Public Policy 12 (3) (2013) 513–544.

[121] S. Hara, K. Hayashi, Making tree ensembles interpretable (2016). arXiv:1606.05390.

[122] A. Henelius, K. Puolamäki, A. Ukkonen, Interpreting classifiers through attribute interactions in datasets (2017). arXiv:1707.07576.

[123] H. Hastie, F. J. C. Garcia, D. A. Robb, P. Patron, A. Laskov, MIRIAM: a multimodal chat-based interface for autonomous systems, in: ACM International Conference on Multimodal Interaction, ACM, 2017, pp. 495–496.

[124] D. Bau, B. Zhou, A. Khosla, A. Oliva, A. Torralba, Network dissection: Quantifying interpretability of deep visual representations, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2017, pp. 6541–6549.

[125] H. Núñez, C. Angulo, A. Català, Rule extraction from support vector machines., in: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), 2002, pp. 107–112.

[126] H. Núñez, C. Angulo, A. Català, Rule-based learning systems for support vector machines, Neural Processing Letters 24 (1) (2006) 1–18.

[127] M. Kearns, S. Neel, A. Roth, Z. S. Wu, Preventing fairness gerrymandering: Auditing and learning for subgroup fairness (2017). arXiv:1711.05144.

[128] E. Akyol, C. Langbort, T. Basar, Price of transparency in strategic machine learning (2016). arXiv:1610.08210.

[129] D. Erhan, A. Courville, Y. Bengio, Understanding representations learned in deep architectures, Department dInformatique et Recherche Operationnelle, University of Montreal, QC, Canada, Tech. Rep 1355 (2010) 1.

[130] Y. Zhang, B. Wallace, A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification (2015). arXiv:1510.03820.

[131] J. R. Quinlan, Simplifying decision trees, International journal of man-machine studies 27 (3) (1987) 221–234.

[132] Y. Zhou, G. Hooker, Interpreting models via single tree approximation (2016). arXiv:1610.09036.

[133] A. Navia-Vázquez, E. Parrado-Hernández, Support vector machine interpretation, Neurocomputing 69 (13-15) (2006) 1754–1759.

[134] J. J. Thiagarajan, B. Kailkhura, P. Sattigeri, K. N. Ramamurthy, Treeview: Peeking into deep neural networks via feature-space partitioning (2016). arXiv:1611.07429.

[135] M. D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: European conference on computer vision, Springer, 2014, pp. 818–833.

[136] A. Mahendran, A. Vedaldi, Understanding deep image representations by inverting them, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2015, pp. 5188–5196.

[137] J. Wagner, J. M. Kohler, T. Gindele, L. Hetzel, J. T. Wiedemer, S. Behnke, Interpretable and fine-grained visual explanations for convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 9097–9107.

[138] A. Kanehira, T. Harada, Learning to explain with complemental examples, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2019, pp. 8603–8611.

[139] D. W. Apley, Visualizing the effects of predictor variables in black box supervised learning models (2016). arXiv:1612.08468.

[140] M. Staniak, P. Biecek, Explanations of model predictions with live and breakdown packages (2018). arXiv:1804.01955.

[141] M. D. Zeiler, D. Krishnan, G. W. Taylor, R. Fergus, Deconvolutional networks., in: CVPR, Vol. 10, 2010, p. 7.

[142] J. T. Springenberg, A. Dosovitskiy, T. Brox, M. Riedmiller, Striving for simplicity: The all convolutional net, arXiv preprint arXiv:1412.6806 (2014).

[143] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, R. Sayres, Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV) (2017). arXiv:1711.11279.

[144] A. Polino, R. Pascanu, D. Alistarh, Model compression via distillation and quantization (2018). arXiv:1802.05668.

[145] W. J. Murdoch, A. Szlam, Automatic rule extraction from long short term memory networks (2017). arXiv:1702.02540.

[146] M. W. Craven, J. W. Shavlik, Using sampling and queries to extract rules from trained neural networks, in: Machine learning proceedings 1994, Elsevier, 1994, pp. 37–45.

[147] A. D. Arbatli, H. L. Akin, Rule extraction from trained neural networks using genetic algorithms, Nonlinear Analysis: Theory, Methods & Applications 30 (3) (1997) 1639–1648.

[148] U. Johansson, L. Niklasson, Evolving decision trees using oracle guides, in: 2009 IEEE Symposium on Computational Intelligence and Data Mining, IEEE, 2009, pp. 238–244.

[149] T. Lei, R. Barzilay, T. Jaakkola, Rationalizing neural predictions (2016). arXiv:1606.04155.

[150] A. Radford, R. Jozefowicz, I. Sutskever, Learning to generate reviews and discovering sentiment (2017). arXiv:1704.01444.

[151] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, D. Batra, Grad-CAM: Why did you say that? (2016).

[152] R. Shwartz-Ziv, N. Tishby, Opening the black box of deep neural networks via information (2017). arXiv:1703.00810.

[153] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, H. Lipson, Understanding neural networks through deep visualization (2015). arXiv:1506.06579.

[154] P. E. Pope, S. Kolouri, M. Rostami, C. E. Martin, H. Hoffmann, Explainability methods for graph convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 10772–10781.

[155] P. Gajane, M. Pechenizkiy, On formalizing fairness in prediction with machine learning (2017). arXiv:1710.03184.

[156] C. Dwork, C. Ilvento, Composition of fairsystems (2018). arXiv:1806.06122.

[157] S. Barocas, M. Hardt, A. Narayanan, Fairness and Machine Learning, fairmlbook.org, 2019, http://www.fairmlbook.org.

[158] B. Kim, E. Glassman, B. Johnson, J. Shah, iBCM: Interactive bayesian case model empowering humans via intuitive interaction, Tech. rep., MIT-CSAIL-TR-2015-010 (2015).

[159] H.-X. Wang, L. Fratiglioni, G. B. Frisoni, M. Viitanen, B. Winblad, Smoking and the occurence of alzheimer's disease: Cross-sectional and longitudinal data in a population-based study, American journal of epidemiology 149 (7) (1999) 640–644.

[160] P. Rani, C. Liu, N. Sarkar, E. Vanman, An empirical study of machine learning techniques for affect recognition in human–robot interaction, Pattern Analysis and Applications 9 (1) (2006) 58–69.

[161] J. Pearl, Causality, Cambridge university press, 2009.

[162] M. Kuhn, K. Johnson, Applied predictive modeling, Vol. 26, Springer, 2013.

[163] G. James, D. Witten, T. Hastie, R. Tibshirani, An introduction to statistical learning, Vol. 112, Springer, 2013.

[164] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus, Intriguing properties of neural networks (2013). arXiv:1312.6199.

[165] D. Ruppert, Robust statistics: The approach based on influence functions, Taylor & Francis, 1987.

[166] S. Basu, K. Kumbier, J. B. Brown, B. Yu, Iterative random forests to discover predictive and stable high-order interactions, Proceedings of the National Academy of Sciences 115 (8) (2018) 1943–1948.

[167] B. Yu, et al., Stability, Bernoulli 19 (4) (2013) 1484–1500.

[168] K. Burns, L. A. Hendricks, K. Saenko, T. Darrell, A. Rohrbach, Women also Snowboard: Overcoming Bias in Captioning Models (2018). arXiv:1803.09797.

[169] A. Bennetot, J.-L. Laurent, R. Chatila, N. Díaz-Rodríguez, Towards explainable neural-symbolic visual reasoning, in: Proceedings of the Neural-symbolic learning and Reasoning Workshop, NeSy-2019 at International Joint Conference on Artificial Intelligence (IJCAI), Macau, China, 2019.

[170] R. Tibshirani, Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society: Series B (Methodological) 58 (1) (1996) 267–288.

[171] Y. Lou, R. Caruana, J. Gehrke, Intelligible models for classification and regression, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2012, pp. 150–158.

[172] K. Kawaguchi, Deep learning without poor local minima, in: Proceedings of the International Conference on Neural Information Processing Systems, 2016, pp. 586–594.

[173] A. Datta, S. Sen, Y. Zick, Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems, in: 2016 IEEE symposium on security and privacy (SP), IEEE, 2016, pp. 598–617.

[174] Z. Bursac, C. H. Gauss, D. K. Williams, D. W. Hosmer, Purposeful selection of variables in logistic regression, Source code for biology and medicine 3 (1) (2008) 17.

[175] J. Jaccard, Interaction effects in logistic regression: Quantitative applications in the social sciences, Sage Thousand Oaks, CA, 2001.

[176] D. W. Hosmer Jr, S. Lemeshow, R. X. Sturdivant, Applied logistic regression, Vol. 398, John Wiley & Sons, 2013.

[177] C.-Y. J. Peng, K. L. Lee, G. M. Ingersoll, An introduction to logistic regression analysis and reporting, The journal of educational research 96 (1) (2002) 3–14.

[178] U. Hoffrage, G. Gigerenzer, Using natural frequencies to improve diagnostic inferences, Academic medicine 73 (5) (1998) 538–540.

[179] C. Mood, Logistic regression: Why we cannot do what we think we can do, and what we can do about it, European sociological review 26 (1) (2010) 67–82.

[180] H. Laurent, R. L. Rivest, Constructing optimal binary decision trees is Np-complete, Information processing letters 5 (1) (1976) 15–17.

[181] P. E. Utgoff, Incremental induction of decision trees, Machine learning 4 (2) (1989) 161–186.

[182] J. R. Quinlan, Induction of decision trees, Machine learning 1 (1) (1986) 81–106.

[183] L. Rokach, O. Z. Maimon, Data mining with decision trees: theory and applications, Vol. 69, World scientific, 2014.

[184] S. Rovnyak, S. Kretsinger, J. Thorp, D. Brown, Decision trees for real-time transient stability prediction, IEEE Transactions on Power Systems 9 (3) (1994) 1417–1426.

[185] H. Nefeslioglu, E. Sezer, C. Gokceoglu, A. Bozkir, T. Duman, Assessment of landslide susceptibility by decision trees in the metropolitan area of istanbul, turkey, Mathematical Problems in Engineering 2010 (2010) Article ID 901095.

[186] S. B. Imandoust, M. Bolandraftar, Application of k-nearest neighbor (knn) approach for predicting economic events: Theoretical background, International Journal of Engineering Research and Applications 3 (5) (2013) 605–610.

[187] L. Li, D. M. Umbach, P. Terry, J. A. Taylor, Application of the GA/KNN method to SELDI proteomics data, Bioinformatics 20 (10) (2004) 1638–1640.

[188] G. Guo, H. Wang, D. Bell, Y. Bi, K. Greer, An KNN model-based approach and its application in text categorization, in: International Conference on Intelligent Text Processing and Computational Linguistics, Springer, 2004, pp. 559–570.

[189] S. Jiang, G. Pang, M. Wu, L. Kuang, An improved k-nearest-neighbor algorithm for text categorization, Expert Systems with Applications 39 (1) (2012) 1503–1509.

[190] U. Johansson, R. König, L. Niklasson, The truth is in there-rule extraction from opaque models using genetic programming., in: FLAIRS Conference, Miami Beach, FL, 2004, pp. 658–663.

[191] J. R. Quinlan, Generating production rules from decision trees., in: Proceedings of the International Joint Conference on Artifical Intelligence, Vol. 87, Citeseer, 1987, pp. 304–307.

[192] P. Langley, H. A. Simon, Applications of machine learning and rule induction, Communications of the ACM 38 (11) (1995) 54–64.

[193] D. Berg, Bankruptcy prediction by generalized additive models, Applied Stochastic Models in Business and Industry 23 (2) (2007) 129–143.

[194] R. Calabrese, et al., Estimating bank loans loss given default by generalized additive models, UCD Geary Institute Discussion Paper Series, WP2012/24 (2012).

[195] P. Taylan, G.-W. Weber, A. Beck, New approaches to regression by generalized additive models and continuous optimization for modern applications in finance, science and technology, Optimization 56 (5-6) (2007) 675–698.

[196] H. Murase, H. Nagashima, S. Yonezaki, R. Matsukura, T. Kitakado, Application of a generalized additive model (GAM) to reveal relationships between environmental factors and distributions of pelagic fish and krill: a case study in sendai bay, Japan, ICES Journal of Marine Science 66 (6) (2009) 1417–1424.

[197] N. Tomić, S. Božić, A modified geosite assessment model (M-GAM) and its application on the lazar canyon area (serbia), International journal of environmental research 8 (4) (2014) 1041–1052.

[198] A. Guisan, T. C. Edwards Jr, T. Hastie, Generalized linear and generalized additive models in studies of species distributions: setting the scene, Ecological Modelling 157 (2-3) (2002) 89–100.

[199] P. Rothery, D. B. Roy, Application of generalized additive models to butterfly transect count data, Journal of Applied Statistics 28 (7) (2001) 897–909.

[200] A. Pierrot, Y. Goude, Short-term electricity load forecasting with generalized additive models, in: 16th Intelligent System Applications to Power Systems Conference, ISAP 2011, IEEE, 2011, pp. 410–415.

[201] T. L. Griffiths, C. Kemp, J. B. Tenenbaum, Bayesian models of cognition. (4 2008). doi:10.1184/R1/6613682.v1.
URL https://kilthub.cmu.edu/articles/Bayesian_models_of_cognition/6613682

[202] B. H. Neelon, A. J. O'Malley, S.-L. T. Normand, A bayesian model for repeated measures zero-inflated count data with application to outpatient psychiatric service use, Statistical modelling 10 (4) (2010) 421–439.

[203] M. McAllister, G. Kirkwood, Bayesian stock assessment: a review and example application using the logistic model, ICES Journal of Marine Science 55 (6) (1998) 1031–1060.

[204] G. Synnaeve, P. Bessiere, A bayesian model for opening prediction in RTS games with application to starcraft, in: Computational Intelligence and Games (CIG), 2011 IEEE Conference on, IEEE, 2011, pp. 281–288.

[205] S.-K. Min, D. Simonis, A. Hense, Probabilistic climate change predictions applying bayesian model averaging, Philosophical transactions of the royal society of london a: mathematical, physical and engineering sciences 365 (1857) (2007) 2103–2116.

[206] G. Koop, D. J. Poirier, J. L. Tobias, Bayesian econometric methods, Cambridge University Press, 2007.

[207] A. R. Cassandra, L. P. Kaelbling, J. A. Kurien, Acting under uncertainty: Discrete bayesian models for mobile-robot navigation, in: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS'96, Vol. 2, IEEE, 1996, pp. 963–972.

[208] H. A. Chipman, E. I. George, R. E. McCulloch, Bayesian cart model search, Journal of the American Statistical Association 93 (443) (1998) 935–948.

[209] B. Kim, C. Rudin, J. A. Shah, The bayesian case model: A generative approach for case-based reasoning and prototype classification, in: Advances in Neural Information Processing Systems, 2014, pp. 1952–1960.

[210] B. Kim, R. Khanna, O. O. Koyejo, Examples are not enough, learn to criticize! criticism for interpretability, in: Advances in Neural Information Processing Systems, 2016, pp. 2280–2288.

[211] U. Johansson, L. Niklasson, R. König, Accuracy vs. comprehensibility in data mining models, in: Proceedings of the seventh international conference on information fusion, Vol. 1, 2004, pp. 295–300.

[212] R. Konig, U. Johansson, L. Niklasson, G-rex: A versatile framework for evolutionary data mining, in: 2008 IEEE International Conference on Data Mining Workshops, IEEE, 2008, pp. 971–974.

[213] H. Lakkaraju, E. Kamar, R. Caruana, J. Leskovec, Interpretable & explorable approximations of black box models (2017). arXiv:1707.01154.

[214] S. Mishra, B. L. Sturm, S. Dixon, Local interpretable model-agnostic explanations for music content analysis., in: ISMIR, 2017, pp. 537–543.

[215] G. Su, D. Wei, K. R. Varshney, D. M. Malioutov, Interpretable two-level boolean rule learning for classification (2015). arXiv:1511.07361.

[216] M. T. Ribeiro, S. Singh, C. Guestrin, Nothing else matters: Model-agnostic explanations by identifying prediction invariance (2016). arXiv:1611.05817.

[217] M. W. Craven, Extracting comprehensible models from trained neural networks, Ph.D. thesis, aAI9700774 (1996).

[218] O. Bastani, C. Kim, H. Bastani, Interpretability via model extraction (2017). arXiv:1706.09773.

[219] G. Hooker, Discovering additive structure in black box functions, in: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2004, pp. 575–580.

[220] P. Adler, C. Falk, S. A. Friedler, T. Nix, G. Rybeck, C. Scheidegger, B. Smith, S. Venkatasubramanian, Auditing black-box models for indirect influence, Knowledge and Information Systems 54 (1) (2018) 95–122.

[221] P. W. Koh, P. Liang, Understanding black-box predictions via influence functions, in: Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017, pp. 1885–1894.

[222] P. Cortez, M. J. Embrechts, Opening black box data mining models using sensitivity analysis, in: 2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM), IEEE, 2011, pp. 341–348.

[223] P. Cortez, M. J. Embrechts, Using sensitivity analysis and visualization techniques to open black box data mining models, Information Sciences 225 (2013) 1–17.

[224] S. M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: Advances in Neural Information Processing Systems, 2017, pp. 4765–4774.

[225] I. Kononenko, et al., An efficient explanation of individual classifications using game theory, Journal of Machine Learning Research 11 (Jan) (2010) 1–18.

[226] H. Chen, S. Lundberg, S.-I. Lee, Explaining models by propagating shapley values of local components (2019). arXiv:arXiv:1911.11888.

[227] P. Dabkowski, Y. Gal, Real time image saliency for black box classifiers, in: Proceedings of the International Conference on Neural Information Processing Systems, 2017, pp. 6967–6976.

[228] A. Henelius, K. Puolamäki, H. Boström, L. Asker, P. Papapetrou, A peek into the black box: exploring classifiers by randomization, Data mining and knowledge discovery 28 (5-6) (2014) 1503–1529.

[229] J. Moeyersoms, B. d'Alessandro, F. Provost, D. Martens, Explaining classification models built on high-dimensional sparse data (2016). arXiv:1607.06280.

[230] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, K.-R. MÃžller, How to explain individual classification decisions, Journal of Machine Learning Research 11 (Jun) (2010) 1803–1831.

[231] J. Adebayo, L. Kagal, Iterative orthogonal feature projection for diagnosing bias in black-box models, arXiv preprint arXiv:1611.04967 (2016).

[232] R. Guidotti, A. Monreale, S. Ruggieri, D. Pedreschi, F. Turini, F. Giannotti, Local rule-based explanations of black box decision systems, arXiv preprint arXiv:1805.10820 (2018).

[233] S. Krishnan, E. Wu, Palm: Machine learning explanations for iterative debugging, in: Proceedings of the 2nd Workshop on Human-In-the-Loop Data Analytics, ACM, 2017, p. 4.

[234] M. Robnik-Šikonja, I. Kononenko, Explaining classifications for individual instances, IEEE Transactions on Knowledge and Data Engineering 20 (5) (2008) 589–600.

[235] M. T. Ribeiro, S. Singh, C. Guestrin, Anchors: High-precision model-agnostic explanations, in: AAAI Conference on Artificial Intelligence, 2018.

[236] D. Martens, F. Provost, Explaining data-driven document classifications, MIS Quarterly 38 (1) (2014) 73–100.

[237] D. Chen, S. P. Fraiberger, R. Moakler, F. Provost, Enhancing transparency and control when drawing data-driven inferences about individuals, Big data 5 (3) (2017) 197–212.

[238] A. Goldstein, A. Kapelner, J. Bleich, E. Pitkin, Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation, Journal of Computational and Graphical Statistics 24 (1) (2015) 44–65.

[239] G. Casalicchio, C. Molnar, B. Bischl, Visualizing the feature importance for black box models, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2018, pp. 655–670.

[240] G. Tolomei, F. Silvestri, A. Haines, M. Lalmas, Interpretable predictions of tree-based ensembles via actionable feature tweaking, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2017, pp. 465–474.

[241] L. Auret, C. Aldrich, Interpretation of nonlinear relationships between process variables by use of random forests, Minerals Engineering 35 (2012) 27–42.

[242] N. F. Rajani, R. Mooney, Stacking with auxiliary features for visual question answering, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), 2018, pp. 2217–2226.

[243] N. F. Rajani, R. J. Mooney, Ensembling visual explanations, in: Explainable and Interpretable Models in Computer Vision and Machine Learning, Springer, 2018, pp. 155–172.

[244] H. Núñez, C. Angulo, A. Català, Rule-based learning systems for support vector machines, Neural Processing Letters 24 (1) (2006) 1–18.

[245] Z. Chen, J. Li, L. Wei, A multiple kernel support vector machine scheme for feature selection and rule extraction from gene expression data of cancer tissue, Artificial Intelligence in Medicine 41 (2) (2007) 161–175.

[246] H. Núñez, C. Angulo, A. Català, Support vector machines with symbolic interpretation, in: VII Brazilian Symposium on Neural Networks, 2002. SBRN 2002. Proceedings., IEEE, 2002, pp. 142–147.

[247] P. Sollich, Bayesian methods for support vector machines: Evidence and predictive class probabilities, Machine learning 46 (1-3) (2002) 21–52.

[248] P. Sollich, Probabilistic methods for support vector machines, in: Proceedings of the International Conference on Neural Information Processing Systems, 2000, pp. 349–355.

[249] W. Landecker, M. D. Thomure, L. M. Bettencourt, M. Mitchell, G. T. Kenyon, S. P. Brumby, Interpreting individual classifications of hierarchical networks, in: 2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM), IEEE, 2013, pp. 32–38.

[250] A. Jakulin, M. Možina, J. Demšar, I. Bratko, B. Zupan, Nomograms for visualizing support vector machines, in: Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, ACM, 2005, pp. 108–117.

[251] L. Fu, Rule generation from neural networks, IEEE Transactions on Systems, Man, and Cybernetics 24 (8) (1994) 1114–1124.

[252] G. G. Towell, J. W. Shavlik, Extracting refined rules from knowledge-based neural networks, Machine Learning 13 (1) (1993) 71–101. doi:10.1007/BF00993103.
URL https://doi.org/10.1007/BF00993103

[253] S. Thrun, Extracting rules from artificial neural networks with distributed representations, in: Proceedings of the 7th International Conference on Neural Information Processing Systems, NIPS'94, MIT Press, Cambridge, MA, USA, 1994, pp. 505–512.
URL http://dl.acm.org/citation.cfm?id=2998687.2998750

[254] R. Setiono, W. K. Leow, Fernn: An algorithm for fast extraction of rules from neural networks, Applied Intelligence 12 (1) (2000) 15–25. doi:10.1023/A:1008307919726.
URL https://doi.org/10.1023/A:1008307919726

[255] I. A. Taha, J. Ghosh, Symbolic interpretation of artificial neural networks, IEEE Transactions on Knowledge and Data Engineering 11 (3) (1999) 448–463. doi:10.1109/69.774103.

[256] H. Tsukimoto, Extracting rules from trained neural networks, IEEE Transactions on Neural Networks 11 (2) (2000) 377–389. doi:10.1109/72.839008.

[257] J. R. Zilke, E. L. Mencía, F. Janssen, Deepred–rule extraction from deep neural networks, in: International Conference on Discovery Science, Springer, 2016, pp. 457–473.

[258] G. P. J. Schmitz, C. Aldrich, F. S. Gouws, Ann-dt: an algorithm for extraction of decision trees from artificial neural networks, IEEE Transactions on Neural Networks 10 (6) (1999) 1392–1401. doi:10.1109/72.809084.

[259] M. Sato, H. Tsukimoto, Rule extraction from neural networks via decision tree induction, in: Proceedings of the International Joint Conference on Neural Networks (IJCNN) , Vol. 3, IEEE, 2001, pp. 1870–1875.

[260] R. Féraud, F. Clérot, A methodology to explain neural network classification, Neural networks 15 (2) (2002) 237–246.

[261] A. Shrikumar, P. Greenside, A. Kundaje, Learning Important Features Through Propagating Activation Differences, arXiv e-prints (2017) arXiv:1704.02685arXiv:1704.02685.

[262] M. Sundararajan, A. Taly, Q. Yan, Axiomatic attribution for deep networks, in: International Conference on Machine Learning, Vol. 70, JMLR. org, 2017, pp. 3319–3328.

[263] J. Adebayo, J. Gilmer, I. Goodfellow, B. Kim, Local explanation methods for deep neural networks lack sensitivity to parameter values (2018). arXiv:1810.03307.

[264] J. Li, X. Chen, E. Hovy, D. Jurafsky, Visualizing and understanding neural models in NLP (2015). arXiv:1506.01066.

[265] S. Tan, K. C. Sim, M. Gales, Improving the interpretability of deep neural networks with stimulated learning, in: 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), IEEE, 2015, pp. 617–623.

[266] L. Rieger, C. Singh, W. J. Murdoch, B. Yu, Interpretations are useful: penalizing explanations to align neural networks with prior knowledge (2019). arXiv:arXiv:1909.13584.

[267] A. Nguyen, A. Dosovitskiy, J. Yosinski, T. Brox, J. Clune, Synthesizing the preferred inputs for neurons in neural networks via deep generator networks, in: Proceedings of the International Conference on Neural Information Processing Systems, 2016, pp. 3387–3395.

[268] Y. Li, J. Yosinski, J. Clune, H. Lipson, J. E. Hopcroft, Convergent learning: Do different neural networks learn the same representations?, in: ICLR, 2016.

[269] M. Liu, J. Shi, Z. Li, C. Li, J. Zhu, S. Liu, Towards better analysis of deep convolutional neural networks, IEEE transactions on visualization and computer graphics 23 (1) (2016) 91–100.

[270] Y. Goyal, A. Mohapatra, D. Parikh, D. Batra, Towards transparent AI systems: Interpreting visual question answering models (2016). arXiv:1608.08974.

[271] K. Simonyan, A. Vedaldi, A. Zisserman, Deep inside convolutional networks: Visualising image classification models and saliency maps (2013). arXiv:1312.6034.

[272] A. Nguyen, J. Yosinski, J. Clune, Deep neural networks are easily fooled: High confidence predictions for unrecognizable images, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2015, pp. 427–436.

[273] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, T. Darrell, Long-term recurrent convolutional networks for visual recognition and description, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition (CVPR), 2015, pp. 2625–2634.

[274] M. Lin, Q. Chen, S. Yan, Network in network (2013). arXiv:1312.4400.

[275] L. A. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, T. Darrell, Generating Visual Explanations, arXiv e-prints (2016) arXiv:1603.08507arXiv:1603.08507.

[276] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, X. Tang, Residual attention network for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2017, pp. 3156–3164.

[277] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, Z. Zhang, The application of two-level attention models in deep convolutional neural network for fine-grained image classification, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2015, pp. 842–850.

[278] Q. Zhang, R. Cao, Y. Nian Wu, S.-C. Zhu, Growing Interpretable Part Graphs on ConvNets via Multi-Shot Learning, arXiv e-prints (2016) arXiv:1611.04246arXiv:1611.04246.

[279] L. Arras, G. Montavon, K.-R. Müller, W. Samek, Explaining recurrent neural network predictions in sentiment analysis (2017). arXiv:1706.07206.

[280] A. Karpathy, J. Johnson, L. Fei-Fei, Visualizing and understanding recurrent networks (2015). arXiv:1506.02078.

[281] J. Clos, N. Wiratunga, S. Massie, Towards explainable text classification by jointly learning lexicon and modifier terms, in: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-17 Workshop on Explainable AI (XAI)), 2017, p. 19.

[282] S. Wisdom, T. Powers, J. Pitton, L. Atlas, Interpretable recurrent neural networks using sequential sparse recovery (2016). arXiv:1611.07252.

[283] V. Krakovna, F. Doshi-Velez, Increasing the interpretability of recurrent neural networks using hidden markov models (2016). arXiv:1606.05320.

[284] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, W. Stewart, Retain: An interpretable predictive model for healthcare using reverse time attention mechanism, in: Proceedings of the International Conference on Neural Information Processing Systems, 2016, pp. 3504–3512.

[285] L. Breiman, Classification and regression trees, Routledge, 2017.

[286] A. Lucic, H. Haned, M. de Rijke, Explaining predictions from tree-based boosting ensembles (2019). arXiv:arXiv:1907.02582.

[287] S. M. Lundberg, G. G. Erion, S.-I. Lee, Consistent individualized feature attribution for tree ensembles (2018). arXiv:arXiv:1802.03888.

[288] C. Buciluǎ, R. Caruana, A. Niculescu-Mizil, Model compression, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2006, pp. 535–541.

[289] R. Traoré, H. Caselles-Dupré, T. Lesort, T. Sun, G. Cai, N. D. Rodríguez, D. Filliat, Discorl: Continual reinforcement learning via policy distillation, CoRR abs/1907.05855 (2019). arXiv:1907.05855.
URL http://arxiv.org/abs/1907.05855

[290] M. D. Zeiler, G. W. Taylor, R. Fergus, et al., Adaptive deconvolutional networks for mid and high level feature learning., in: ICCV, Vol. 1, 2011, p. 6.

[291] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.

[292] C. Olah, A. Mordvintsev, L. Schubert, Feature visualization, DistillHttps://distill.pub/2017/feature-visualization (2017). doi:10.23915/distill.00007.

[293] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, B. Kim, Sanity checks for saliency maps, in: Proceedings of the International Conference on Neural Information Processing Systems, 2018, pp. 9505–9515.

[294] C. Olah, A. Satyanarayan, I. Johnson, S. Carter, L. Schubert, K. Ye, A. Mordvintsev, The building blocks of interpretability, Distill (2018).

[295] Z. Che, S. Purushotham, R. Khemani, Y. Liu, Distilling knowledge from deep networks with applications to healthcare domain (2015). arXiv:1512.03542.

[296] T. Hailesilassie, Rule extraction algorithm for deep neural networks: A review (2016). arXiv:1610.05267.

[297] J. M. Benitez, J. L. Castro, I. Requena, Are artificial neural networks black boxes?, Trans. Neur. Netw. 8 (5) (1997) 1156–1164. doi:10.1109/72.623216. URL https://doi.org/10.1109/72.623216

[298] U. Johansson, R. König, L. Niklasson, Automatically balancing accuracy and comprehensibility in predictive modeling, Vol. 2, 2005, p. 7 pp. doi:10.1109/ICIF.2005.1592040.

[299] D. Smilkov, N. Thorat, B. Kim, F. Viégas, M. Wattenberg, SmoothGrad: removing noise by adding noise, arXiv e-prints (2017) arXiv:1706.03825arXiv:1706.03825.

[300] M. Ancona, E. Ceolini, C. Öztireli, M. Gross, Towards better understanding of gradient-based attribution methods for Deep Neural Networks, arXiv e-prints (2017) arXiv:1711.06104arXiv:1711.06104.

[301] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks?, arXiv e-prints (2014) arXiv:1411.1792arXiv:1411.1792.

[302] A. Sharif Razavian, H. Azizpour, J. Sullivan, S. Carlsson, CNN Features off-the-shelf: an Astounding Baseline for Recognition, arXiv e-prints (2014) arXiv:1403.6382arXiv:1403.6382.

[303] S. Du, H. Guo, A. Simpson, Self-driving car steering angle prediction based on image recognition, Tech. rep., Technical Report, Stanford University (2017).

[304] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Object Detectors Emerge in Deep Scene CNNs, arXiv e-prints (2014) arXiv:1412.6856arXiv:1412.6856.

[305] Y. Zhang, X. Chen, Explainable Recommendation: A Survey and New Perspectives, arXiv e-prints (2018) arXiv:1804.11192arXiv:1804.11192.

[306] J. Frankle, M. Carbin, The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks, arXiv e-prints (2018) arXiv:1803.03635arXiv:1803.03635.

[307] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention Is All You Need, arXiv e-prints (2017) arXiv:1706.03762arXiv:1706.03762.

[308] J. Lu, J. Yang, D. Batra, D. Parikh, Hierarchical question-image co-attention for visual question answering, in: Proceedings of the 30th International Conference on Neural Information Processing Systems, NIPS'16, Curran Associates Inc., USA, 2016, pp. 289–297.
URL http://dl.acm.org/citation.cfm?id=3157096.3157129

[309] A. Das, H. Agrawal, C. L. Zitnick, D. Parikh, D. Batra, Human Attention in Visual Question Answering: Do Humans and Deep Networks Look at the Same Regions?, arXiv e-prints (2016) arXiv:1606.03556arXiv:1606.03556.

[310] D. Huk Park, L. A. Hendricks, Z. Akata, A. Rohrbach, B. Schiele, T. Darrell, M. Rohrbach, Multimodal Explanations: Justifying Decisions and Pointing to the Evidence, arXiv e-prints (2018) arXiv:1802.08129arXiv:1802.08129.

[311] A. Slavin Ross, M. C. Hughes, F. Doshi-Velez, Right for the Right Reasons: Training Differentiable Models by Constraining their Explanations, arXiv e-prints (2017) arXiv:1703.03717arXiv:1703.03717.

[312] I. T. Jolliffe, Principal Component Analysis and Factor Analysis, Springer New York, New York, NY, 1986, pp. 115–128. doi:10.1007/978-1-4757-1904-8_7.
URL https://doi.org/10.1007/978-1-4757-1904-8_7

[313] A. Hyvärinen, E. Oja, Oja, e.: Independent component analysis: Algorithms and applications. neural networks 13(4-5), 411-430, Neural networks : the official journal of the International Neural Network Society 13 (2000) 411–30. doi:10.1016/S0893-6080(00)00026-5.

[314] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, R. J. Plemmons, Algorithms and applications for approximate nonnegative matrix factorization, Computational Statistics & Data Analysis 52 (2007) 155–173.

[315] D. P. Kingma, M. Welling, Auto-Encoding Variational Bayes, arXiv e-prints (2013) arXiv:1312.6114arXiv:1312.6114.

[316] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, A. Lerchner, beta-vae: Learning basic visual concepts with a constrained variational framework, in: ICLR, 2017.

[317] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, P. Abbeel, InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets, arXiv e-prints (2016) arXiv:1606.03657arXiv:1606.03657.

[318] Q. Zhang, Y. Yang, Y. Liu, Y. Nian Wu, S.-C. Zhu, Unsupervised Learning of Neural Networks to Explain Neural Networks, arXiv e-prints (2018) arXiv:1805.07468arXiv:1805.07468.

[319] S. Sabour, N. Frosst, G. E Hinton, Dynamic Routing Between Capsules, arXiv e-prints (2017) arXiv:1710.09829arXiv:1710.09829.

[320] A. Agrawal, J. Lu, S. Antol, M. Mitchell, C. L. Zitnick, D. Batra, D. Parikh, VQA: Visual Question Answering, arXiv e-prints (2015) arXiv:1505.00468arXiv:1505.00468.

[321] A. Fukui, D. Huk Park, D. Yang, A. Rohrbach, T. Darrell, M. Rohrbach, Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding, arXiv e-prints (2016) arXiv:1606.01847arXiv:1606.01847.

[322] D. Bouchacourt, L. Denoyer, EDUCE: explaining model decisions through unsupervised concepts extraction, CoRR abs/1905.11852 (2019). arXiv:1905.11852. URL http://arxiv.org/abs/1905.11852

[323] I. Donadello, L. Serafini, A. D. Garcez, Logic tensor networks for semantic image interpretation, Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI (2017) 1596–1602.

[324] I. Donadello, Semantic image interpretation-integration of numerical data and logical knowledge for cognitive vision, Ph.D. thesis, University of Trento (2018).

[325] A. S. d'Avila Garcez, M. Gori, L. C. Lamb, L. Serafini, M. Spranger, S. N. Tran, Neural-symbolic computing: An effective methodology for principled integration of machine learning and reasoning, Journal of Applied Logics - IfCoLog Journal of Logics and their Applications (FLAP) 6 (4) (2019) 611–632.
URL https://collegepublications.co.uk/ifcolog/?00033

[326] R. Manhaeve, S. Dumancic, A. Kimmig, T. Demeester, L. De Raedt, DeepProbLog: Neural probabilistic logic programming, in: Proceedings of the International Conference on Neural Information Processing Systems, Vol. 31, 2018, pp. 3749–3759.
URL https://proceedings.neurips.cc/paper/2018/file/dc5d637ed5e62c36ecb73b654b05ba2a-Paper.pdf

[327] I. Donadello, M. Dragoni, C. Eccher, Persuasive explanation of reasoning inferences on dietary data, in: First Workshop on Semantic Explainability @ ISWC 2019, 2019.

[328] F. Petroni, T. Rocktäschel, P. Lewis, A. Bakhtin, Y. Wu, A. H. Miller, S. Riedel, Language models as knowledge bases? (2019). arXiv:1909.01066.

[329] K. Bollacker, N. Díaz-Rodríguez, X. Li, Extending knowledge graphs with subjective influence networks for personalized fashion, in: E. Portmann, M. E. Tabacchi, R. Seising, A. Habenstein (Eds.), Designing Cognitive Cities, Springer International Publishing, 2019, pp. 203–233.

[330] W. Shang, A. Trott, S. Zheng, C. Xiong, R. Socher, Learning world graphs to accelerate hierarchical reinforcement learning (2019). arXiv:1907.00664.

[331] R. G. Krishnan, U. Shalit, D. Sontag, Deep Kalman Filters (2015). arXiv:1511.05121.

[332] M. Karl, M. Soelch, J. Bayer, P. van der Smagt, Deep Variational Bayes Filters: Unsupervised Learning of State Space Models from Raw Data (2016). arXiv:1605.06432.

[333] M. J. Johnson, D. K. Duvenaud, A. Wiltschko, R. P. Adams, S. R. Datta, Composing graphical models with neural networks for structured representations and fast inference, in: Proceedings of the 30th International Conference on Neural Information Processing Systems 29, 2016, pp. 2946–2954.

[334] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, P. H. Torr, Conditional random fields as recurrent neural networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1529–1537.

[335] M. Garnelo, K. Arulkumaran, M. Shanahan, Towards deep symbolic reinforcement learning (2016). arXiv:1609.05518.

[336] V. Bellini, A. Schiavone, T. Di Noia, A. Ragone, E. Di Sciascio, Knowledge-aware autoencoders for explainable recommender systems, in: Proceedings of the 3rd Workshop on Deep Learning for Recommender Systems, DLRS 2018, 2018, pp. 24–31.

[337] C.-Z. A. Huang, A. Vaswani, J. Uszkoreit, N. Shazeer, C. Hawthorne, A. M. Dai, M. D. Hoffman, D. Eck, Music transformer: Generating music with long-term structure (2018). arXiv:1809.04281.

[338] A. Aamodt, E. Plaza, Case-based reasoning: Foundational issues, Methodological Variations, and System Approaches 7 (1) (1994) 39–59.

[339] R. Caruana, Case-based explanation for artificial neural nets, in: Artificial Neural Networks in Medicine and Biology, Proceedings of the ANNIMAB-1 Conference, 2000, pp. 303–308.

[340] M. T. Keane, E. M. Kenny, The Twin-System Approach as One Generic Solution for XAI: An Overview of ANN-CBR Twins for Explaining Deep Learning (2019). arXiv:1905.08069.

[341] C. Hofer, M. Denker, S. Ducasse, Design and Implementation of a Backward-In-Time Debugger, in: NODe 2006, Vol. P-88 of Lecture Notes in Informatics, GI, Erfurt, Germany, 2006, pp. 17–32.
URL https://hal.inria.fr/inria-00555768

[342] C. Rudin, Please stop explaining black box models for high stakes decisions (2018). arXiv:1811.10154.

[343] A. Diez-Olivan, J. Del Ser, D. Galar, B. Sierra, Data fusion and machine learning for industrial prognosis: Trends and perspectives towards Industry 4.0, Information Fusion 50 (2019) 92–111.

[344] R. R. Hoffman, S. T. Mueller, G. Klein, J. Litman, Metrics for explainable ai: Challenges and prospects, arXiv preprint arXiv:1812.04608 (2018).

[345] S. Mohseni, N. Zarei, E. D. Ragan, A multidisciplinary survey and framework for design and evaluation of explainable ai systems (2018). arXiv:arXiv:1811.11839.

[346] R. M. J. Byrne, Counterfactuals in explainable artificial intelligence (xai): Evidence from human reasoning, in: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, International Joint Conferences on Artificial Intelligence Organization, 2019, pp. 6276–6282. doi:10.24963/ijcai.2019/876.
URL https://doi.org/10.24963/ijcai.2019/876

[347] M. Garnelo, M. Shanahan, Reconciling deep learning with symbolic artificial intelligence: representing objects and relations, Current Opinion in Behavioral Sciences 29 (2019) 17–23.

[348] G. Marra, F. Giannini, M. Diligenti, M. Gori, Integrating learning and reasoning with deep logic models (2019). arXiv:1901.04195.

[349] K. Kelley, B. Clark, V. Brown, J. Sitzia, Good practice in the conduct and reporting of survey research, International Journal for Quality in Health Care 15 (3) (2003) 261–266. arXiv:http://oup.prod.sis.lan/intqhc/article-pdf/15/3/261/5251095/mzg031.pdf, doi:10.1093/intqhc/mzg031.
URL https://doi.org/10.1093/intqhc/mzg031

[350] S. Wachter, B. Mittelstadt, L. 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) (2017) 76–99.

[351] T. Orekondy, B. Schiele, M. Fritz, Knockoff nets: Stealing functionality of black-box models, CoRR abs/1812.02766 (2018). arXiv:1812.02766.
URL http://arxiv.org/abs/1812.02766

[352] S. J. Oh, B. Schiele, M. Fritz, Towards reverse-engineering black-box neural networks, in: Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, Springer, 2019, pp. 121–144.

[353] I. J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, arXiv preprint arXiv:1412.6572 (2014).

[354] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, D. Song, Robust physical-world attacks on deep learning models (2017). arXiv:1707.08945.

[355] I. J. Goodfellow, N. Papernot, P. D. McDaniel, cleverhans v0.1: an adversarial machine learning library, CoRR abs/1610.00768 (2016). arXiv:1610.00768.
URL http://arxiv.org/abs/1610.00768

[356] H. Xiao, B. Biggio, B. Nelson, H. Xiao, C. Eckert, F. Roli, Support vector machines under adversarial label contamination, Neurocomput. 160 (C) (2015) 53–62. doi:10.1016/j.neucom.2014.08.081.
URL http://dx.doi.org/10.1016/j.neucom.2014.08.081

[357] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, F. Roli, Evasion attacks against machine learning at test time, in: Proceedings of the 2013th European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part III, ECMLPKDD'13, Springer-Verlag, Berlin, Heidelberg, 2013, pp. 387–402. doi:10.1007/978-3-642-40994-3_25.
URL https://doi.org/10.1007/978-3-642-40994-3_25

[358] B. Biggio, I. Pillai, S. R. Bulò, D. Ariu, M. Pelillo, F. Roli, Is data clustering in adversarial settings secure?, CoRR abs/1811.09982 (2018). arXiv:1811.09982.
URL http://arxiv.org/abs/1811.09982

[359] Z. Pan, W. Yu, X. Yi, A. Khan, F. Yuan, Y. Zheng, Recent progress on generative adversarial networks (gans): A survey, IEEE Access 7 (2019) 36322–36333.

[360] D. Charte, F. Charte, S. García, M. J. del Jesus, F. Herrera, A practical tutorial on autoencoders for nonlinear feature fusion: Taxonomy, models, software and guidelines, Information Fusion 44 (2018) 78–96.

[361] C. F. Baumgartner, L. M. Koch, K. Can Tezcan, J. Xi Ang, E. Konukoglu, Visual feature attribution using wasserstein gans, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8309–8319.

[362] C. Biffi, O. Oktay, G. Tarroni, W. Bai, A. De Marvao, G. Doumou, M. Rajchl, R. Bedair, S. Prasad, S. Cook, et al., Learning interpretable anatomical features through deep generative models: Application to cardiac remodeling, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2018, pp. 464–471.

[363] S. Liu, B. Kailkhura, D. Loveland, Y. Han, Generative counterfactual introspection for explainable deep learning (2019). arXiv:arXiv:1907.03077.

[364] K. R. Varshney, H. Alemzadeh, On the safety of machine learning: Cyber-physical systems, decision sciences, and data products, Big data 5 (3) (2017) 246–255.

[365] G. M. Weiss, Mining with rarity: a unifying framework, ACM Sigkdd Explorations Newsletter 6 (1) (2004) 7–19.

[366] J. Attenberg, P. Ipeirotis, F. Provost, Beat the machine: Challenging humans to find a predictive model's "unknown unknowns", Journal of Data and Information Quality (JDIQ) 6 (1) (2015) 1.

[367] G. Neff, A. Tanweer, B. Fiore-Gartland, L. Osburn, Critique and contribute: A practice-based framework for improving critical data studies and data science, Big data 5 (2) (2017) 85–97.

[368] A. Iliadis, F. Russo, Critical data studies: An introduction, Big Data & Society 3 (2) (2016) 2053951716674238.

[369] A. Karpatne, G. Atluri, J. H. Faghmous, M. Steinbach, A. Banerjee, A. Ganguly, S. Shekhar, N. Samatova, V. Kumar, Theory-guided data science: A new paradigm for scientific discovery from data, IEEE Transactions on Knowledge and Data Engineering 29 (10) (2017) 2318–2331.

[370] G. Hautier, C. C. Fischer, A. Jain, T. Mueller, G. Ceder, Finding nature's missing ternary oxide compounds using machine learning and density functional theory, Chemistry of Materials 22 (12) (2010) 3762–3767.

[371] C. C. Fischer, K. J. Tibbetts, D. Morgan, G. Ceder, Predicting crystal structure by merging data mining with quantum mechanics, Nature materials 5 (8) (2006) 641.

[372] S. Curtarolo, G. L. Hart, M. B. Nardelli, N. Mingo, S. Sanvito, O. Levy, The high-throughput highway to computational materials design, Nature materials 12 (3) (2013) 191.

[373] K. C. Wong, L. Wang, P. Shi, Active model with orthotropic hyperelastic material for cardiac image analysis, in: International Conference on Functional Imaging and Modeling of the Heart, Springer, 2009, pp. 229–238.

[374] J. Xu, J. L. Sapp, A. R. Dehaghani, F. Gao, M. Horacek, L. Wang, Robust transmural electrophysiological imaging: Integrating sparse and dynamic physiological models into ecg-based inference, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015, pp. 519–527.

[375] T. Lesort, M. Seurin, X. Li, N. Díaz-Rodríguez, D. Filliat, Unsupervised state representation learning with robotic priors: a robustness benchmark (2017). arXiv:arXiv:1709.05185.

[376] J. Z. Leibo, Q. Liao, F. Anselmi, W. A. Freiwald, T. Poggio, View-tolerant face recognition and hebbian learning imply mirror-symmetric neural tuning to head orientation, Current Biology 27 (1) (2017) 62–67.

[377] F. Schrodt, J. Kattge, H. Shan, F. Fazayeli, J. Joswig, A. Banerjee, M. Reichstein, G. Bönisch, S. Díaz, J. Dickie, et al., Bhpmf–a hierarchical bayesian approach to gap-filling and trait prediction for macroecology and functional biogeography, Global Ecology and Biogeography 24 (12) (2015) 1510–1521.

[378] D. Leslie, Understanding artificial intelligence ethics and safety (2019). arXiv:arXiv:1906.05684, doi:10.5281/zenodo.3240529.

[379] C. Rudin, Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead (2018). arXiv:arXiv:1811.10154.

[380] J. Fjeld, H. Hilligoss, N. Achten, M. L. Daniel, J. Feldman, S. Kagay, Principled artificial intelligence: A map of ethical and rights-based approaches (2019).
URL https://ai-hr.cyber.harvard.edu/images/primp-viz.pdf

[381] R. Benjamins, A. Barbado, D. Sierra, Responsible ai by design (2019). arXiv:1909.12838.

[382] United-Nations, Transforming our world: the 2030 agenda for sustainable development, Tech. rep., eSocialSciences (2015).
URL https://EconPapers.repec.org/RePEc:ess:wpaper:id:7559

[383] G. D. Hager, A. Drobnis, F. Fang, R. Ghani, A. Greenwald, T. Lyons, D. C. Parkes, J. Schultz, S. Saria, S. F. Smith, M. Tambe, Artificial intelligence for social good (2019). arXiv:arXiv:1901.05406.

[384] B. C. Stahl, D. Wright, Ethics and privacy in AI and big data: Implementing responsible research and innovation, IEEE Security & Privacy 16 (3) (2018) 26–33.

[385] High Level Expert Group on Artificial Intelligence, Ethics guidelines for trustworthy ai, Tech. rep., European Commission (2019).

[386] B. d'Alessandro, C. O'Neil, T. LaGatta, Conscientious classification: A data scientist's guide to discrimination-aware classification, Big data 5 (2) (2017) 120–134.

[387] S. Barocas, A. D. Selbst, Big data's disparate impact, Calif. L. Rev. 104 (2016) 671.

[388] M. Hardt, E. Price, N. Srebro, et al., Equality of opportunity in supervised learning, in: Proceedings of the International Conference on Neural Information Processing Systems, 2016, pp. 3315–3323.

[389] T. Speicher, H. Heidari, N. Grgic-Hlaca, K. P. Gummadi, A. Singla, A. Weller, M. B. Zafar, A unified approach to quantifying algorithmic unfairness: Measuring individual group unfairness via inequality indices, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining, ACM, 2018, pp. 2239–2248.

[390] F. Kamiran, T. Calders, Data preprocessing techniques for classification without discrimination, Knowledge and Information Systems 33 (1) (2012) 1–33.

[391] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, C. Dwork, Learning fair representations, in: International Conference on Machine Learning, 2013, pp. 325–333.

[392] B. H. Zhang, B. Lemoine, M. Mitchell, Mitigating unwanted biases with adversarial learning, in: Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, ACM, 2018, pp. 335–340.

[393] Y. Ahn, Y.-R. Lin, Fairsight: Visual analytics for fairness in decision making, IEEE transactions on visualization and computer graphics (2019).

[394] E. Soares, P. Angelov, Fair-by-design explainable models for prediction of recidivism, arXiv preprint arXiv:1910.02043 (2019).

[395] J. Dressel, H. Farid, The accuracy, fairness, and limits of predicting recidivism, Science advances 4 (1) (2018) eaao5580.

[396] U. Aivodji, H. Arai, O. Fortineau, S. Gambs, S. Hara, A. Tapp, Fairwashing: the risk of rationalization, in: International Conference on Machine Learning, 2019, pp. 161–170.

[397] S. Sharma, J. Henderson, J. Ghosh, Certifai: Counterfactual explanations for robustness, transparency, interpretability, and fairness of artificial intelligence models, arXiv preprint arXiv:1905.07857 (2019).

[398] M. Drosou, H. Jagadish, E. Pitoura, J. Stoyanovich, Diversity in big data: A review, Big data 5 (2) (2017) 73–84.

[399] J. Lerman, Big data and its exclusions, Stan. L. Rev. Online 66 (2013) 55.

[400] R. Agrawal, S. Gollapudi, A. Halverson, S. Ieong, Diversifying search results, in: Proceedings of the second ACM international conference on web search and data mining, ACM, 2009, pp. 5–14.

[401] B. Smyth, P. McClave, Similarity vs. diversity, in: International conference on case-based reasoning, Springer, 2001, pp. 347–361.

[402] H. A. Simon, Spurious correlation: A causal interpretation*, Journal of the American Statistical Association 49 (267) (1954) 467–479. arXiv:https://doi.org/10.1080/01621459.1954.10483515, doi:10.1080/01621459.1954.10483515.
URL https://doi.org/10.1080/01621459.1954.10483515

[403] N. Díaz-Rodríguez, A. Härmä, R. Helaoui, I. Huitzil, F. Bobillo, U. Straccia, Couch potato or gym addict? semantic lifestyle profiling with wearables and fuzzy knowledge graphs, in: 6th Workshop on Automated Knowledge Base Construction, AKBC@NIPS 2017, Long Beach, California, 2017.

[404] D. Doran, S. Schulz, T. R. Besold, What Does Explainable AI Really Mean? A New Conceptualization of Perspectives, arXiv e-prints (2017) arXiv:1710.00794arXiv:1710.00794.

[405] The Description Logic Handbook: Theory, Implementation and Applications, 2nd Edition, Cambridge University Press, 2007. doi:10.1017/CBO9780511711787.

[406] I. Donadello, L. Serafini, Integration of numeric and symbolic information for semantic image interpretation, Intelligenza Artificiale 10 (1) (2016) 33–47.

[407] J.-B. Lamy, L. F. Soualmia, Formalization of the semantics of iconic languages: An ontology-based method and four semantic-powered applications, Knowledge-Based Systems 135 (2017) 159–179.

[408] G. Marra, F. Giannini, M. Diligenti, M. Gori, Lyrics: a general interface layer to integrate ai and deep learning, arXiv preprint arXiv:1903.07534 (2019).

[409] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, Z. Ives, Dbpedia: A nucleus for a web of open data, in: The semantic web, Springer, 2007, pp. 722–735.

[410] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, K. J. Miller, Introduction to wordnet: An on-line lexical database, International journal of lexicography 3 (4) (1990) 235–244.

[411] F. Baader, W. Nutt, The description logic handbook, Cambridge University Press, New York, NY, USA, 2003, Ch. Basic Description Logics, pp. 43–95.
URL http://dl.acm.org/citation.cfm?id=885746.885749

[412] P. Hitzler, M. Krtzsch, S. Rudolph, Foundations of Semantic Web Technologies, 1st Edition, Chapman & Hall/CRC, 2009.

[413] C. Kiddon, P. Domingos, Knowledge extraction and joint inference using tractable Markov logic, in: Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX), Association for Computational Linguistics, Montréal, Canada, 2012, pp. 79–83.

[414] N. Balasubramanian, S. Soderland, O. Etzioni, et al., Rel-grams: a probabilistic model of relations in text, in: Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction, Association for Computational Linguistics, 2012, pp. 101–105.

[415] I. Huitzil, U. Straccia, N. Díaz-Rodríguez, F. Bobillo, Datil: learning fuzzy ontology datatypes, in: International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Springer, 2018, pp. 100–112.

[416] M. Qu, J. Tang, Y. Bengio, Weakly-supervised knowledge graph alignment with adversarial learning, CoRR abs/1907.03179 (2019). arXiv:1907.03179.
URL http://arxiv.org/abs/1907.03179

[417] K. Xu, L. Wu, Z. Wang, M. Yu, L. Chen, V. Sheinin, Sql-to-text generation with graph-to-sequence model, arXiv preprint arXiv:1809.05255 (2018).

[418] L. Chen, C. D. Nugent, Ontology-based activity recognition in intelligent pervasive environments, International Journal of Web Information Systems (IJWIS) 5 (4) (2009) 410–430.

[419] S. Rudolph, Foundations of description logics, in: Reasoning Web International Summer School, Springer, 2011, pp. 76–136.

[420] J. Z. Pan, D. Calvanese, T. Eiter, I. Horrocks, M. Kifer, F. Lin, Y. Zhao, Reasoning Web: Logical Foundation of Knowledge Graph Construction and Query Answering:

12th International Summer School 2016, Aberdeen, UK, September 5-9, 2016, Tutorial Lectures, Vol. 9885, Springer, 2017.

[421] J. Z. Pan, G. Vetere, J. M. Gomez-Perez, H. Wu, Exploiting linked data and knowledge graphs in large organisations, Springer, 2017.

[422] M. Bienvenu, Ontology-mediated query answering: harnessing knowledge to get more from data, in: IJCAI: International Joint Conference on Artificial Intelligence, 2016.

[423] Y. Ren, A. Parvizi, C. Mellish, J. Z. Pan, K. van Deemter, R. Stevens, Towards competency question-driven ontology authoring, in: V. Presutti, C. d'Amato, F. Gandon, M. d'Aquin, S. Staab, A. Tordai (Eds.), The Semantic Web: Trends and Challenges, Springer International Publishing, Cham, 2014, pp. 752–767.

[424] G. Antoniou, F. Van Harmelen, Web ontology language: Owl, in: Handbook on ontologies, Springer, 2004, pp. 67–92.

[425] J. Burrell, How the machine 'thinks': Understanding opacity in machine learning algorithms, Big Data & Society 3 (1) (2016) 2053951715622512. arXiv:https://doi.org/10.1177/2053951715622512, doi:10.1177/2053951715622512.
URL https://doi.org/10.1177/2053951715622512

[426] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft COCO: Common objects in context, in: European Conference on Computer Vision, Springer, 2014, pp. 740–755.

[427] O. Vinyals, A. Toshev, S. Bengio, D. Erhan, Show and Tell: A Neural Image Caption Generator, arXiv e-prints (2014) arXiv:1411.4555arXiv:1411.4555.

[428] R. J. Mooney, R. Bunescu, Mining knowledge from text using information extraction, SIGKDD Explor. Newsl. 7 (1) (2005) 3–10. doi:10.1145/1089815.1089817.
URL http://doi.acm.org/10.1145/1089815.1089817

[429] L. A. Hendricks, K. Burns, K. Saenko, T. Darrell, A. Rohrbach, Women also snowboard: Overcoming bias in captioning models, in: European Conference on Computer Vision, Springer, 2018, pp. 793–811.

[430] M. T. Ribeiro, S. Singh, C. Guestrin, Why should i trust you?: Explaining the predictions of any classifier, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2016, pp. 1135–1144.

[431] D. Alvarez-Melis, T. S. Jaakkola, On the Robustness of Interpretability Methods, arXiv e-prints (2018) arXiv:1806.08049arXiv:1806.08049.

[432] D. Slack, S. Hilgard, E. Jia, S. Singh, H. Lakkaraju, Fooling LIME and SHAP: Adversarial Attacks on Post hoc Explanation Methods, arXiv e-prints (2019) arXiv:1911.02508arXiv:1911.02508.

[433] G. Ras, N. Xie, M. van Gerven, D. Doran, Explainable deep learning: A field guide for the uninitiated (2021). arXiv:2004.14545.

[434] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al., Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI, Information Fusion 58 (2020) 82–115.

[435] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, D. Pedreschi, A survey of methods for explaining black box models, ACM computing surveys (CSUR) 51 (5) (2018) 1–42.

[436] V. Buhrmester, D. Münch, M. Arens, Analysis of explainers of black box deep neural networks for computer vision: A survey, arXiv preprint arXiv:1911.12116 (2019).

[437] J. Andreas, Measuring compositionality in representation learning, arXiv preprint arXiv:1902.07181 (2019).

[438] J. A. Fodor, E. Lepore, Compositionality Papers, Oxford University Press UK, 2002.

[439] A. Stone, H. Wang, M. Stark, Y. Liu, D. Scott Phoenix, D. George, Teaching compositionality to CNNs, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2017, pp. 5058–5067.

[440] B. M. Lake, R. Salakhutdinov, J. B. Tenenbaum, Human-level concept learning through probabilistic program induction, Science 350 (6266) (2015) 1332–1338.

[441] D. Hupkes, V. Dankers, M. Mul, E. Bruni, The compositionality of neural networks: integrating symbolism and connectionism, arXiv preprint arXiv:1908.08351 (2019).

[442] J. Mao, C. Gan, P. Kohli, J. B. Tenenbaum, J. Wu, The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision, arXiv preprint arXiv:1904.12584 (2019).

[443] R. De Kok, T. Schneider, U. Ammer, Object-based classification and applications in the alpine forest environment, International Archives of Photogrammetry and Remote Sensing 32 (Part 7) (1999) 4–3.

[444] D. Huber, A. Kapuria, R. Donamukkala, M. Hebert, Parts-based 3d object classification, in: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2004., Vol. 2, IEEE, 2004, pp. II–II.

[445] E. J. Bernstein, Y. Amit, Part-based statistical models for object classification and detection, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 2, IEEE, 2005, pp. 734–740.

[446] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (9) (2009) 1627–1645.

[447] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, A. Zisserman, The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results.
URL http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html

[448] W. Ge, X. Lin, Y. Yu, Weakly supervised complementary parts models for fine-grained image classification from the bottom up, in: Proceedings of the IEEE Conference on Computer Vision and Rattern Recognition, 2019, pp. 3034–3043.

[449] N. Díaz-Rodríguez, A. Lamas, J. Sanchez, , G. Franchi, I. Donadello, S. Tabik, D. Filliat, P. Cruz, R. Montes, F. Herrera, EXplainable Neural-Symbolic Learning (*X-NeSyL*) methodology to fuse deep learning representations with expert knowledge graphs: the MonuMAI cultural heritage use case (2021).

[450] A. Holzinger, G. Langs, H. Denk, K. Zatloukal, H. Müller, Causability and explainability of artificial intelligence in medicine, Wiley Interdiscip. Rev. Data Min. Knowl. Discov. 9 (4) (2019) e1312.

[451] A. Holzinger, B. Malle, A. Saranti, B. Pfeifer, Towards multi-modal causability with graph neural networks enabling information fusion for explainable ai, Information Fusion 71 (2021) 28–37. doi:https:

//doi.org/10.1016/j.inffus.2021.01.008.
URL https://www.sciencedirect.com/science/article/pii/S1566253521000142

[452] A. Holzinger, A. Carrington, H. Müller, Measuring the quality of explanations: The system causability scale (SCS), KI - Künstliche Intelligenz 34 (2) (2020) 193–198. doi:10.1007/s13218-020-00636-z.
URL https://doi.org/10.1007%2Fs13218-020-00636-z

[453] D. Alvarez-Melis, T. S. Jaakkola, Towards robust interpretability with self-explaining neural networks, in: Proceedings of the 32nd International Conference on Neural Information Processing Systems, NIPS'18, Curran Associates Inc., Red Hook, NY, USA, 2018, p. 7786–7795.

[454] F. Sovrano, F. Vitali, An objective metric for explainable ai: How and why to estimate the degree of explainability (2021). arXiv:2109.05327.

[455] A. Rosenfeld, Better metrics for evaluating explainable artificial intelligence, in: Proceedings of the 20th International Conference on Autonomous Agents and Multi-Agent Systems, AAMAS '21, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 2021, p. 45–50.

[456] G. Schwalbe, B. Finzel, A comprehensive taxonomy for explainable artificial intelligence: A systematic survey of surveys on methods and concepts (2021). arXiv:2105.07190.

[457] E. B. Baum, What Is Thought?, Cambridge MA: Bradford Book/MIT Press, 2004.

[458] C. Blundell, J. Cornebise, K. Kavukcuoglu, D. Wierstra, Weight Uncertainty in Neural Networks, arXiv e-prints (2015) arXiv:1505.05424arXiv:1505.05424.

[459] P. Kremen, M. Blaško, Z. Kouba, Semantic Annotation of Objects, Handbook of Research on Social Dimensions of Semantic Technologies and Web Services, IGI Global, Hershey, PA, USA, 2009, pp. 223–238. doi:10.4018/978-1-60566-650-1.ch011.
URL https://doi.org/10.4018/978-1-60566-650-1.ch011

[460] E. C. Norton, B. E. Dowd, Log odds and the interpretation of logit models, Health services research 53 (2) (2018) 859–878, pMC5867187[pmcid]. doi:10.1111/1475-6773.12712.
URL https://doi.org/10.1111/1475-6773.12712

[461] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation (2018). arXiv:1802.02611.

[462] H. Kervadec, J. Dolz, S. Wang, E. Granger, I. ben Ayed, Bounding boxes for weakly supervised segmentation: Global constraints get close to full supervision, in: Medical Imaging with Deep Learning, 2020.
URL https://openreview.net/forum?id=VOQMC3rZtL

[463] A. Lamas, S. Tabik, P. Cruz, R. Montes, Á. Martínez-Sevilla, T. Cruz, F. Herrera, MonuMAI: Dataset, deep learning pipeline and citizen science based app for monumental heritage taxonomy and classification, Neurocomputing 420 (2020) 266–280.

[464] A. Sanfeliu, K.-S. Fu, A distance measure between attributed relational graphs for pattern recognition, IEEE Transactions on Systems, Man, and Cybernetics (1983) 353–362.

[465] S. Jiang, H. Qin, B. Zhang, J. Zheng, Optimized loss functions for object detection and application on nighttime vehicle detection, Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering 236 (7) (2021) 1568–1578. doi:10.1177/09544070211036366.
URL https://doi.org/10.1177%2F09544070211036366

[466] R. Qin, K. Qiao, L. Wang, L. Zeng, J. Chen, B. Yan, Weighted focal loss: An effective loss function to overcome unbalance problem of chest x-ray14, IOP Conference Series: Materials Science and Engineering 428 (2018) 012022. doi:10.1088/1757-899x/428/1/012022.
URL https://doi.org/10.1088/1757-899x/428/1/012022

[467] S. Wachter, B. Mittelstadt, C. Russell, Counterfactual explanations without opening the black box: Automated decisions and the gdpr (2017). doi:10.48550/ARXIV.1711.00399.
URL https://arxiv.org/abs/1711.00399

[468] R. K. Mothilal, A. Sharma, C. Tan, Explaining machine learning classifiers through diverse counterfactual explanations, in: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, 2020, pp. 607–617.

[469] S. Verma, J. Dickerson, K. Hines, Counterfactual explanations for machine learning: A review (2020). doi:10.48550/ARXIV.2010.10596.
URL https://arxiv.org/abs/2010.10596

[470] S. Dandl, C. Molnar, M. Binder, B. Bischl, Multi-objective counterfactual explanations, in: T. Bäck, M. Preuss, A. Deutz, H. Wang, C. Doerr, M. Emmerich, H. Trautmann (Eds.), Parallel Problem Solving from Nature – PPSN XVI, Springer International Publishing, Cham, 2020, pp. 448–469.

[471] A. Van Looveren, J. Klaise, Interpretable counterfactual explanations guided by prototypes (2019). doi:10.48550/ARXIV.1907.02584.
URL https://arxiv.org/abs/1907.02584

[472] A.-H. Karimi, G. Barthe, B. Balle, I. Valera, Model-agnostic counterfactual explanations for consequential decisions (2019). doi:10.48550/ARXIV.1905.11190.
URL https://arxiv.org/abs/1905.11190

[473] T. Laugel, M.-J. Lesot, C. Marsala, X. Renard, M. Detyniecki, Inverse classification for comparison-based interpretability in machine learning (2017). doi:10.48550/ARXIV.1712.08443.
URL https://arxiv.org/abs/1712.08443

[474] M. T. Ribeiro, S. Singh, C. Guestrin, Anchors: High-precision model-agnostic explanations, Proceedings of the AAAI Conference on Artificial Intelligence 32 (1) (Apr. 2018).
URL https://ojs.aaai.org/index.php/AAAI/article/view/11491

[475] H. Müller, A. Holzinger, Kandinsky patterns, Artificial Intelligence 300 (2021) 103546. doi:https://doi.org/10.1016/j.artint.2021.103546.
URL https://www.sciencedirect.com/science/article/pii/S0004370221000977

[476] A. Holzinger, M. Kickmeier-Rust, H. Müller, Kandinsky patterns as iq-test for machine learning, in: A. Holzinger, P. Kieseberg, A. M. Tjoa, E. Weippl (Eds.), Machine Learning and Knowledge Extraction, Springer International Publishing, Cham, 2019, pp. 1–14.

[477] M. A. Gordon, K. Duh, Explaining sequence-level knowledge distillation as data-augmentation for neural machine translation (2019). doi:10.48550/ARXIV.1912.03334.
URL https://arxiv.org/abs/1912.03334

[478] H. Lee, S. J. Hwang, J. Shin, Self-supervised label augmentation via input transformations (2019). doi:10.48550/ARXIV.1910.05872.
URL https://arxiv.org/abs/1910.05872

[479] N. Papernot, P. McDaniel, X. Wu, S. Jha, A. Swami, Distillation as a defense to adversarial perturbations against deep neural networks (2015). doi:10.48550/ARXIV.1511.04508.
URL https://arxiv.org/abs/1511.04508

[480] Z. Li, D. Hoiem, Learning without forgetting, IEEE Transactions on Pattern Analysis and Machine Intelligence 40 (12) (2018) 2935–2947. doi:10.1109/TPAMI.2017.2773081.

[481] T. Lesort, N. Díaz-Rodríguez, J.-F. Goudou, D. Filliat, State representation learning for control: An overview, Neural Networks 108 (2018) 379 – 392. doi:https://doi.org/10.1016/j.neunet.2018.07.006.
URL http://www.sciencedirect.com/science/article/pii/S0893608018302053

[482] T. Lesort, V. Lomonaco, A. Stoian, D. Maltoni, D. Filliat, N. Díaz-Rodríguez, Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges, Information fusion 58 (2020) 52–68.

[483] R. Traoré, H. Caselles-Dupré, T. Lesort, T. Sun, N. Díaz-Rodríguez, D. Filliat, Continual Reinforcement Learning deployed in Real-life using Policy Distillation and Sim2Real Transfer, in: ICML Workshop on "Multi-Task and Lifelong Reinforcement Learning", Long Beach, United States, 2019, accepted to the Workshop on Multi-Task and Lifelong Reinforcement Learning, ICML 2019.
URL https://hal.archives-ouvertes.fr/hal-02285839

[484] C. Bucila, R. Caruana, A. Niculescu-Mizil, Model compression, in: KDD '06, 2006.

[485] G. Urban, K. J. Geras, S. E. Kahou, O. Aslan, S. Wang, R. Caruana, A. Mohamed, M. Philipose, M. Richardson, Do deep convolutional nets really need to be deep and convolutional? (2017). arXiv:1603.05691.

[486] J. H. Cho, B. Hariharan, On the efficacy of knowledge distillation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[487] M. Phuong, C. Lampert, Towards understanding knowledge distillation, in: K. Chaudhuri, R. Salakhutdinov (Eds.), Proceedings of the 36th International Conference on

Machine Learning, Vol. 97 of Proceedings of Machine Learning Research, PMLR, Long Beach, California, USA, 2019, pp. 5142–5151.
URL http://proceedings.mlr.press/v97/phuong19a.html

[488] X. Wang, R. Zhang, Y. Sun, J. Qi, Kdgan: Knowledge distillation with generative adversarial networks, in: S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (Eds.), Advances in Neural Information Processing Systems 31, Curran Associates, Inc., 2018, pp. 775–786.

[489] R. Urner, S. Shalev-Shwartz, S. Ben-David, Access to unlabeled data can speed up prediction time, in: Proceedings of the 28th International Conference on Machine Learning, 2011, pp. 641–648.

[490] J. Gou, B. Yu, S. Maybank, D. Tao, Knowledge distillation: A survey, 2020.

[491] J. Ba, R. Caruana, Do deep nets really need to be deep?, in: Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems, Vol. 27, Curran Associates, Inc., 2014, pp. 2654–2662.
URL https://proceedings.neurips.cc/paper/2014/file/ea8fcd92d59581717e06eb187f10666d-Paper.pdf

[492] S. Kim, H.-E. Kim, Transferring knowledge to smaller network with class-distance loss, in: ICLR, 2017.

[493] W. Ding, X. Jing, Z. Yan, L. T. Yang, A survey on data fusion in internet of things: Towards secure and privacy-preserving fusion, Information Fusion 51 (2019) 129–144.

[494] R. Müller, S. Kornblith, G. Hinton, When does label smoothing help? doi:10.48550/ARXIV.1906.02629.
URL https://arxiv.org/abs/1906.02629

[495] S. Tan, R. Caruana, G. Hooker, Y. Lou, Distill-and-compare, in: Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, ACM, 2018. doi:10.1145/3278721.3278725.
URL https://doi.org/10.1145%2F3278721.3278725

[496] Z. Hu, X. Ma, Z. Liu, E. Hovy, E. Xing, Harnessing deep neural networks with logic rules (2016). doi:10.48550/ARXIV.1603.06318.
URL https://arxiv.org/abs/1603.06318

[497] Z. Hu, Z. Yang, R. Salakhutdinov, E. Xing, Deep neural networks with massive learned knowledge, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2016, pp. 1670–1679. doi:10.18653/v1/D16-1173.
URL https://aclanthology.org/D16-1173

[498] Z. Che, S. Purushotham, R. Khemani, Y. Liu, Distilling knowledge from deep networks with applications to healthcare domain. doi:10.48550/ARXIV.1512.03542.
URL https://arxiv.org/abs/1512.03542

[499] X. Liu, X. Wang, S. Matwin, Improving the interpretability of deep neural networks with knowledge distillation (2018). doi:10.48550/ARXIV.1812.10924.
URL https://arxiv.org/abs/1812.10924

[500] D. Chen, J.-P. Mei, C. Wang, Y. Feng, C. Chen, Online knowledge distillation with diverse peers, Proceedings of the AAAI Conference on Artificial Intelligence 34 (04) (2020) 3430–3437. doi:10.1609/aaai.v34i04.5746.
URL https://ojs.aaai.org/index.php/AAAI/article/view/5746

[501] S. Stanton, P. Izmailov, P. Kirichenko, A. A. Alemi, A. G. Wilson, Does knowledge distillation really work? (2021). doi:10.48550/ARXIV.2106.05945.
URL https://arxiv.org/abs/2106.05945

[502] H. Zhang, The optimality of naive bayes, Vol. 2, 2004.

[503] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, X. Hu, Score-cam: Score-weighted visual explanations for convolutional neural networks (2019). doi:10.48550/ARXIV.1910.01279.
URL https://arxiv.org/abs/1910.01279

[504] S. H. Lee, D. H. Kim, B. C. Song, Self-supervised knowledge distillation using singular value decomposition (2018). doi:10.48550/ARXIV.1807.06819.
URL https://arxiv.org/abs/1807.06819

[505] C. Zhang, Y. Peng, Better and faster: Knowledge transfer from multiple self-supervised learning tasks via graph distillation for video classification (2018). doi:10.48550/ARXIV.1804.10069.
URL https://arxiv.org/abs/1804.10069

[506] L. Zhang, Y. Shi, Z. Shi, K. Ma, C. Bao, Task-oriented feature distillation, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, Vol. 33, Curran Associates, Inc., 2020, pp. 14759–14771.
URL https://proceedings.neurips.cc/paper/2020/file/a96b65a721e561e1e3de768ac819ffbb-Paper.pdf

[507] S. Lee, B. C. Song, Graph-based knowledge distillation by multi-head attention network, in: BMVC, 2019.

[508] N. Passalis, M. Tzelepi, A. Tefas, Heterogeneous knowledge distillation using information flow modeling (2020). doi:10.48550/ARXIV.2005.00727.
URL https://arxiv.org/abs/2005.00727

[509] B. Peng, X. Jin, J. Liu, D. Li, Y. Wu, Y. Liu, S. Zhou, Z. Zhang, Correlation congruence for knowledge distillation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[510] F. Tung, G. Mori, Similarity-preserving knowledge distillation (2019). doi:10.48550/ARXIV.1907.09682.
URL https://arxiv.org/abs/1907.09682

[511] N. Passalis, A. Tefas, Learning deep representations with probabilistic knowledge transfer (2018). doi:10.48550/ARXIV.1803.10837.
URL https://arxiv.org/abs/1803.10837

[512] Y. Liu, K. Chen, C. Liu, Z. Qin, Z. Luo, J. Wang, Structured knowledge distillation for semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.