applied sciences

MDPI

Article

Certifiable AI

Jobst Landgrebe

Department of Philosophy, University at Buffalo, 135 Park Hall, Buffalo, NY 14260, USA; jobstlan@buffalo.edu; Tel.: +49-15123525359

Abstract: Implicit stochastic models, including both 'deep neural networks' (dNNs) and the more recent unsupervised foundational models, cannot be explained. That is, it cannot be determined *how they work*, because the interactions of the millions or billions of terms that are contained in their equations cannot be captured in the form of a causal model. Because users of stochastic AI systems would like to understand how they operate in order to be able to use them safely and reliably, there has emerged a new field called 'explainable AI' (XAI). When we examine the XAI literature, however, it becomes apparent that its protagonists have redefined the term 'explanation' to mean something else, namely: 'interpretation'. Interpretations are indeed sometimes possible, but we show that they give at best only a subjective understanding of how a model works. We propose an alternative to XAI, namely certified AI (CAI), and describe how an AI can be specified, realized, and tested in order to become certified. The resulting approach combines ontologies and formal logic with statistical learning to obtain reliable AI systems which can be safely used in technical applications.

Keywords: Artificial Intelligence; explainable AI; prior knowledge; formal logic; ontology

Citation: Landgrebe, J. Certifiable AI. *Appl. Sci.* **2022**, *12*, 1050. https://doi.org/10.3390/app12031050

Academic Editors: Jose Antonio Iglesias Martinez and Giancarlo Mauri

Received: 2 November 2021
Accepted: 18 January 2022
Published: 20 January 2022

1. Introduction

Since the so-called 'deep neural networks' became broadly usable thanks to a huge supply of training data and computation power, we have experienced a new wave of Artificial Intelligence (AI) enthusiasm in research and applied technology. Despite their name, they function in a way that has nothing to do with the way the central nervous systems of animals works. Like all models resulting from statistical learning [1], dNNs are the outputs of an optimization algorithm that uses supervised or non-supervised data and training hyperparameters to find the local minimum of a loss-function [2]. Supervised models are trained with data tuples of the form $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{N}$, where N indicates the number of observations, $\mathbf{x}$ the input data (usually a vector) and $\mathbf{y}$ the output associated with the input data, where $\mathbf{y}$ may be either a scalar or a vector. Input data are independent variables, and the output (also called 'outcome') is then dependent thereon. For example, the input variables might be quantitatively expressible features of a product sold online and the outcome of its rating by customers. Such tuples can be obtained from data accumulating on the internet, from production processes in a factory, or through manual annotation efforts. Unsupervised models are obtained by using instead an $N \times p$ data matrix X which contains no dependent variables (where N again represents the number of observations and p is the number of variables in the matrix).

The resulting stochastic models are implicit, non-linear, and have millions or more parameters – for example, the sequence-generating unsupervised model GTP-3 has 175 billion parameters [3]. Here 'implicit' means that the models are not generated explicitly by creating a symbolic, structure such as a differential equation or a syllogism. Rather, the model is obtained by using an optimization procedure. Of course, the model must in every case be executable on a computer (Turing-machine). Because the models are created implicitly, and because of their huge size, there is no way in which the processes by which they estimate a stochastic output $\hat{\mathbf{y}}$ from an input vector $\mathbf{x}$ could be made explicit and understandable for example to a human being.

This situation has been still further aggravated in more recent times by the development of so-called foundation models [4], which are, unlike supervised regression models or supervised dNNs, unsupervised. There are two types of foundation models: (i) generative models, which model multivariate or conditional multivariate distributions present in a given body of training data [5]; and (ii) discriminative models which aim at modeling latent higher-level patterns shared by similar outcome sequences with different data at the lowest (input) level in order to enable predictions [6].

These unsupervised models are often specialized to certain tasks by conditioning them on specific data or using them in transfer learning, where the models are retrained in a supervised setting [4] (pp. 85–90). But no matter how they are obtained, they are in every case either a functional or an operator consisting—in the unfolded equational view that can always be obtained from the network view—of an equation with billions of terms and parameters for which it is, again, impossible to tell how they create the output estimate $\hat{\mathbf{y}}$ from a given input $\mathbf{x}$.

For these reasons, certain aspects of using dNNs in production systems in any sector of the economy, including the public sector, have been identified as possible areas of concern. The most important of these are [4] (pp. 151–159):

- the lack of explainability, leading to attempts to create explainable AI [7],
- attitude bias [8] (see Section 3.5 below),
- social injustice and unfairness,
- abuse,
- carbon-dioxide footprint,
- legal aspects,
- economic aspects,
- as well as ethical aspects such as mass surveillance, the concentration of power, and automated decision-making.

We note in passing that many of these aspects will become critical only under the assumption that dNNs and foundation models are in fact used to the extent envisaged by their respective developer communities. We shall return to two of the items on the list in Section 4.3.

This paper focuses on the question of how explainable AI and attitude bias, but also other concerns on this list, can be addressed. It is structured as follows: Section 2 defines the concepts of explanation and interpretation in the context of implicit stochastic models. Section 3 explains why the attempts to explain or even interpret stochastic AI must fail and are irredeemably futile if the aim is to obtain AI that can be used safely in real-world environments (the goal of any technology). Section 4 introduces the idea of *Certifiable AI* and explains how stochastic AI systems can be used to construct reliable and safe systems for real-world usage. It discusses how the design and testing of AI systems can eliminate the need for attempts at XAI and at the same time serve to prevent attitude bias in such systems, in order to obtain *certified AI*.

We will provide details of how to combine deterministic with stochastic algorithms in order to obtain seamless, reliable hybrid AI systems. For this purpose, *prior knowledge* in the form of ontologies associated with mathematical knowledge is critical.

2. Implicit Stochastic Model Explanation and Interpretation

Max Weber [9] proposed what has become a standard distinction in the philosophy of science between four different families of scientific goals, namely: description, explanation, interpretation, and prediction [10,11]. Following Weber, I thus *define* the phrase 'to explain AI' as the activity of obtaining a causal understanding of how a given AI (model or algorithm) generates a certain output given a certain input. Explanation in general answers the question: How does this work? It does this either exactly or almost exactly, in the way Maxwell's equations explain the behaviors of electromagnetic fields using a handful of variables and universal constants.

Interpretability is also well-defined (it is wrong to assert, as is commonly done in the XAI literature, that 'interpretability is a domain-specific notion' [7]), namely as *the ability to formulate a model*, which can take the form of an equation or a portion of text, *that enables users of the interpretation to experience subjectively* the meaning of what is being interpreted [12] (p. 1069).

For users with similar cultural backgrounds, the meaning experienced upon reading such an interpretation may be similar. For example, the members of a church congregation listening to a sermon interpreting a biblical text may experience a similar meaning and establish that this is so during conversations after the service.

Interpretation is a hermeneutic activity of a sort first systematically described by Schleiermacher and Dilthey [12]. It is nowadays applied not only to objects of the humanities but also to implicit mathematical models. The requirement for interpretation is often raised when explanations are *a priori* impossible.

I define *interpretation power* as the power of an interpretation model to achieve a similar subjective meaning among individuals with a similar cultural background. It can be assessed by performing interrater reliability tests in which metrics can be designed to measure the degree to which members of a group of individuals obtain the same meaning from a given interpretation [13].

Given these definitions, can stochastic implicit models be either explained or interpreted? Unlike the differential equations used in physics, they do not yield (almost) exact representations of the relationships of the modeled processes. Rather, they yield at best highly approximate models. The main problem is that, unlike differential equations, the number of independent variables used as input for stochastic models is often huge. The length of their input and output vectors determine the dimensionality of their domain and range spaces, respectively:

$$f : \mathbb{R}^k \mapsto \mathbb{R}, \text{or} \tag{1}$$

$$\mathcal{O} : \mathbb{R}^k \mapsto \mathbb{R}^\ell \tag{2}$$

Here k-dimensional vectors are related by functionals f to scalars (one-dimensional outputs), and by operators $\mathcal{O}$ to ℓ-dimensional vectors. k or ℓ can be very large, as is the case, for example, in dNN-based image classification (large k) and neural machine translation (large k and ℓ).

It is certain that the workings of implicit stochastic models such as supervised dNNs and foundation models such as GPT-3 cannot be explained in the sense defined above, since the number of independent variables and model parameters and their interactions cannot be presented as a causal model due the sheer number of relations involved. Note that here, we define a causal model as a model that formally describes the cause-and-effect relationship between two physical entities. We do not mean 'causability' to describe 'the measurable extent to which an explanation [given] to a human expert achieves a specified level of causal understanding' [14] (see Section 4.2).

As is often stressed in the psychological literature, humans can only create and understand models with very few variables [15,16]. What, then, is to be said about interpretation?

3. Attempts at Model Interpretation

This section presents three major families of attempts to produce interpretations of models. Table 1 gives an overview and points to the sections and equations in which the resp. models are discussed.

Table 1. Overview: Types of interpretations of models.

Type	Approach	Section/Equation	Significance
Classical model	Z score	Equation (3)	Positive Z scores indicate that the independent variables contribute to the outcome.
	Gini imporance	Section 3.1	Rough understanding of a variable's contribution to the model's prediction output
dNN local	approximation function	Equation (4)	lower interrater reliability than Z-score
dNN global	basis-changing inverse functional	Equation (5)	Feature visualization
	heatmap clustering	Section 3.2.2	Improved feature visualization
	concept-labelling	Equation (6)	Hermeneutic interpretability
	activation atlasses	Section 3.2.2	High-quality intuitive interpretation

3.1. Classic Types of Model Interpretation

Some traditional stochastic models, such as gradient boosted trees or regression models used in medical risk-factor screening [17], can indeed be interpreted in the hermeneutic sense. For forest models, the *Gini importance measure* of an input variable can be used to obtain a rough understanding of that variable's contribution to the model's prediction output [18].

Multiple regression models have the form:

$$f(\mathbf{x}) = \beta_0 + \sum_{i=1}^{M} \beta_i x_i, \tag{3}$$

where β_0, β_i are the regression parameters, and M is the dimension of the input space (the domain of the functional f). With a sufficiently small number of parameters, interpretations of such models can be obtained using statistics such as Z scores associated with the model with the independent variables. Positive Z scores corresponding to a significant rejection of the null hypothesis (as measured by the Wald test [19]) indicate that the independent variables contribute to the outcome. However, even in this case, the interpretation is only indicative and it depends on the presence of other predictors. The interpretation via Z-statistics is not robust; when one removes from the input data any variable to which a significant Z-statistic is associated, the scores of the remaining variables change. Such interpretations are indeed not causal explanations at all, but rather methods to enable users to experience a subjective meaning (in the sense of hermeneutics) related to the models in question.

3.2. dNN Model Interpretation

The XAI community does of course acknowledge that the behavior of dNNs (how they create an output from a given input) can not be causally explained. It recognizes that the ideal 'comprehensive explanation [which] would extract the whole causal chain from input to output' is 'so far not available'. Rather, they seek 'reduced forms of explanation' such as a 'collection of scores indicating the importance of each input pixel/feature for the prediction' [20], so that explanation is in effect replaced by interpretation. Other XAI authors define an 'explanation model as any interpretable approximation of the original model' [21], which is to say that explanation and interpretation are used as synonyms. Consequently, Rudin asks the community to 'stop explaining black-box machine learning models for high stakes decisions and use interpretable models instead' [7].

There are two approaches to creating an interpretation of this type, involving local and global models, respectively [22]. We review what we believe are the most important examples of these approaches in order to identify the limitations of dNN interpretation in general, without however claiming completeness.

3.2.1. Local Interpretation

Local interpretation approaches try to map a given predictive functional $f \in \mathcal{F}, f : \mathbb{R}^k \to \mathbb{R}$, where $\mathcal{F}$ is a metric functional space, to a lower-dimensional functional space $\mathcal{G}$ using a *local approximation* function $\phi : \mathcal{F} \to \mathcal{G}, g(\mathbf{x}_0) = \phi(f(\mathbf{x}_0))$, where $\mathbf{x}_0 \in \mathbb{R}^k$, the domain of the approximated functional f. One can think of the approximation point $\mathbf{x}_0$ as an element of a narrow interval $I \subset \mathbb{R}^k$, with the quality of the approximation declining drastically as the interval size increases.

Lundberg and Lee [21] have provided a useful unified representation of such approximations. The approximation functions ϕ are often composite, for instance of the form $\phi = \chi \circ \psi$, where $\chi : \mathbb{R}^k \to \{0,1\}^M$ maps the input space to a binary space of dimension M and $\psi : \{0,1\} \to \mathbb{R}$ lifts the binary representation to the regression space $\mathbb{R}$, which is the range of the approximated functional f. The full approximation functions then looks like this (adapted for the purpose of this analysis from [21] (Equation (1))):

$$g(\mathbf{x}_0) = \psi_0 + \sum_{i=1}^{M} \psi_i(\chi_i(\mathbf{x}_0)), \tag{4}$$

with $g(\mathbf{x}_0) \approx f(\mathbf{x}_0)$. This means that the approximation functional $g \in \mathcal{G}$ tries to locally map a high-dimensional model to a logistic regression model in order to achieve an interpretability similar to the interpretation obtained with such classical models (using the regression-Z-scores for parameters of Equation (3), as described in Section 3.1 above).

However, such approximations are of lower quality, and this is for two reasons. First, they are local, which means that they work only for narrow input interval or single input vectors, where the classical models are global. And second, because they always involve an error factor of approximation, where classical models allow a direct interpretation without approximation (For reasons of space we here leave aside the inherent approximation of linear models, which is a form of bias essential to these models; see Section 3.5, Equation (7)). Their interpretation power is thus lower, which means also that the interrater reliability will be lower.

3.2.2. Global Interpretation

A second approach tries to 'uncover qualities of the data at large that affect model behaviors' [4]. For example, it is analyzed how input features propagate within a neural network [20,23], or how models transform input features using mechanisms similar to basis changes in vector spaces [24,25]. Another more recent approach seeks to characterize dNN nodes in terms of activation patterns to interpret or visualize how the models operate [26,27].

Analyses of this sort are often performed for dNNs that operate on image data or that play games which are presented to the human player as a series of images. This is because here results of the analysis can be visualized and thereby interpreted in some sense by appealing to visual intuition, by analogy with the way we describe the perceptual environment by appealing to its affordance character [28].

Predictor Decomposition

Bach et al. [23] proposed a recursive mechanism to map the predictions of a functional f from its range to its domain space, i.e., they defined a basis-changing inverse functional $g : \mathbb{R} \to \mathbb{R}^k$ (see Equation (1)) with $g(f(\mathbf{x})) = \mathbf{r} \in \mathbb{R}^k$, constrained as follows:

$$\sum_{p=1}^{k} r_p = f(\mathbf{x}), \tag{5}$$

where k is again the dimension of the domain of f and each r_p is obtained by summing over a subset of the nodes of a neural network. The vector $\mathbf{r} = (r_1, \ldots, r_p)$ can be understood as a decomposition (basis change) of the range of the function that is obtained by harvesting and summarising the structure of the neural network. It can be used to visualize prediction-relevant features in the input and in the layers of the neural network to obtain 'relevance maps' (heatmaps) that indicate which features of the input space contribute to the classification output of the network [20] (Suppl. 3.2–3.3). Such maps can be inspected to identify which features of the image are used by the classification algorithm. Lapuschkin et al. went one step further and clustered the heatmaps from classification tasks to obtain sets of images with similar classification-relevant properties [20] (Methods Section).

The inspection and clustering of heatmaps generated from image-classification and Atari-game playing dNNs revealed the features that the dNN had learned. For example, it was shown that dNNs learn to classify horses from notes printed on pictures that designate them as such. If these notes are transposed to pictures of cars, then the cars get classified as horses as well. In another setting, the authors identified that the dNN uses the table nudging mechanism built into a virtual pinball game to move the ball without using the flippers at all in order to obtain an infinite number of points. This strategy would not work if the simulation was adapted to model a real pinball machine more closely, for there the machine tilts upon repeated manipulation of the table in order to thwart the player's attempt to manipulate the game.

Node and Activation Characterisations

Several groups have proposed procedures to assign descriptive concepts to individual nodes of a neural network in order to provide semantics that would make them interpretable for users [26,29]. For example, nodes used in the classification of images of water landscape pictures can be associated with concepts such as blue or river or water. As proposed by Bau et al., dNN nodes can either be associated with atomic concepts C from a vocabulary $\mathcal{C}$ [29] or they can be assigned to formulae $\mathcal{L}(\mathcal{C})$ consisting of combinations of concepts obtained by applying connectives $(\wedge, \vee, \neg)$ of propositional logic over $\mathcal{C}$ to yield node interpretation semantics more adequate to the behavior of input-distal nodes [26]. For example, a node can be associated with a propositional formula such as $((\mathsf{water} \vee \mathsf{river}) \wedge \neg \mathsf{blue})$. The assignment to concepts is performed via an explanation function e that uses a similarity measure δ on a node n:

$$e(n) = \underset{C \in \mathcal{C}}{\operatorname{argmax}}\, \delta(n, C), \tag{6}$$

where the similarity measure δ is computed by selecting activated nodes which are thresholded into a binary segmentation M_n. Using a vocabulary-annotated (verum) image mask. This verum is a map of activated nodes to concepts, the Broden ('Broadly and Densely Labeled') dataset. 'The purpose of Broden is to provide a ground truth set of exemplars for a broad set of visual concepts. The concept labels in Broden are normalized and merged from their original data sets so that every class corresponds to an English word.' [29] (Section 2.1). This verum allows the assignment of activations to concepts. $L_C(\mathbf{x})$, the assignment of activated units to concepts is computed via the Jaccard-index $\delta(n, C) = \frac{\sum_{\mathbf{x}} \mathbb{1}(M(\mathbf{x}) \wedge L_C(\mathbf{x}))}{\sum_{\mathbf{x}} \mathbb{1}(M(\mathbf{x}) \vee L_C(\mathbf{x}))}$, where n indicates the node to which an assignment of concept C is made.

Certainly, the labeling of image-processing dNNs with concepts or propositional-logic formulae in this way provides some hermeneutic interpretability of the model. However, because the verum image mask, L_C is incomplete, and the activated concept- or formula-assigned nodes contribute to the overall result (for image tasks) only to a minor extent and may even be negatively correlated with model performance (in natural language inference), the value of this approach is again at best merely hermeneutic: Looking at the annotated nodes neither enables a causal understanding of how the model works nor can it be used to improve in a useful way predictions of how the dNN will work in a given case.

Another approach to the characterization of image-processing dNNs is the activation atlases proposed by a group at OpenAI [27]. Such atlases visualize features of averaged node cluster activations which are tagged by pseudo-images that describe the visual material that node clusters model. The technique was further refined (for example in [30,31]) so that it is now possible to map out 'multimodal' nodes and node groups in a dNN used to identify faces of prominent persons. Though these approaches provide, again, a human-understandable *interpretation*, they neither explain how the models work nor can they help to alleviate any of the features of stochastic models which have been identified as detrimental to the quality of their outputs, including inexact prediction, unpredictable behavior, and attitude bias.

3.3. Reasons for Model Interpretation Failure

The examples we reviewed, as well as other interpretation models for implicit non-linear stochastic models, are very elegant attempts to reduce the stunning dimensionality of the dNN domain and range spaces to dimensions that can be understood by human users. However, these approaches merely provide the user with a narrative that enables a superficial, partial understanding of what is going on inside the dNN, and no causal explanation of *how it works*. This understanding may help some skeptics to develop a kind of emotional trust in the function of dNNs. It can certainly help to understand their limitations. But it can provide no explanation of a sort that would work reliably from one input to the next. Deep learning models cannot be *explained*, because deep learning induces model parameterizations with millions or billions of parameters to master classification or regression tasks. dNNs solve these tasks in a manner that differs completely from the way humans interpret text, language, sounds, images, or smell or somatosensory input.

Some authors think that neural networks are modeled in a way that mimics the structure of human neural systems and that the inspection of the artificial networks can teach us something about the way the nervous system works [32]. Though it is true that convolutional neural networks, for example, were designed by using basic models of the mammalian visual system, the details of the natural and artificial systems are radically different. Consider that one single neuron in the visual cortex (which is supposed to be modeled by one node of a convolutional neural network in its upper layers) contains millions of functionally relevant molecules and interacts directly and indirectly with billions of other neurons, each again endowed with millions of functional molecules, to contribute to the conscious experience of vision. For more details see [33,34] (Ch. 2–3)).

In this context, the results of Moosavi et al. are very relevant. They showed that neural network image classifiers can be nudged into a complete misclassification of images when the input material is mixed with small and universal perturbation vectors [24] generated using an iterative parsimonious perturbation search algorithm. When the image data are perturbed using this universal vector, drastic misclassification occurs. For example, a sock is classified as an elephant or a whale as a bird. Jo and Bengio [25] hypothesized that these perturbations are possible because convolutional neural networks (a CNN is a neural network that can take matrices (image data) as input by using convolutional filter matrices on the input data to map their matrix structure to the nodes of the neural network). use different features of the input space to perform their classifications than humans do—features which they call 'surface statistical regularities'—and that the perturbations alter these features to bring about the mentioned drastic effects.

They used Fourier transformations to construct datasets that share with the original images exactly identical high-level abstractions but differ with regard to surface regularities. For example, a portrait was transformed into an image that preserves the feature contours but uses different colors, with the result that the transformed result maintains the features of the face but presents it in unnatural colors (like the Marylin Monroe series by Andy Warhol).

They found that the dNNs tend to model the 'Fourier image statistics of the training dataset, sometimes exhibiting up to a 28% generalization gap across the various test sets.' This confirms their hypothesis and explains why dNN-image classification can be perturbed so easily. It also provides insight into the way in which these models achieve their results namely—and in contrast to humans—without using any semantics. This provides yet another reason why dNNs are non-explainable.

3.4. Deep Reasons for Deep Model Explanation Failure

Interpretations via the local and global interpretation mechanisms described in Section 2 above yield merely a partial, subjective interpretation. What is the deeper reason for our inability to interpret the behavior of these models in the more objective manner that would be required to come closer to an explanation?

The sorts of implicit stochastic models which researchers have tried to interpret are often those used to emulate human behavior, for example in image or text classification, game playing, in the generation of texts [3] or pictures [35] through the continuation of a given sequence, and also in simulations of inanimate nature, for example, a weather phenomenon such as rain [36]. In all these settings, the models approximate the behavior of complex systems using a logic system, a Turing machine. A Turing machine can only compute algorithms that can be formulated using the basic recursive functions described by Church [37,38]. The problem is that only a certain subset of extended Newtonian mathematics can be expressed as Turing-computable algorithms.

There are indeed *countably many* ($\aleph_0$) Turing-computable functions, but *uncountably many* ($2^{\aleph_0}$) non-Turing-computable functions, and this will not change [39,40]. Implicit stochastic models, no matter how large and complicated they are, remain models approximating complex system behavior using Newtonian mathematics. If this was not the case, then we could not use them inside computers.

The models are generated by using derivatives of loss equations to find local minima in multivariate functionals. The result is a very long differentiable equation. The equation faces some Newtonian requirements which are relaxed: for example, it does not require the interactions between its variables to be always the same. Nor does it require that these interactions have to be homogeneous over the entire neural net. But for neural networks to be computable, it is still required that they satisfy most of the properties of Newtonian models. The manifolds they describe must be differentiable, and the importance of any given interaction must decrease over space or time in a regular fashion; in other words, they must still have a weak Markov property in all spatial and temporal directions.

These properties constrain the quality of the approximation to real-world phenomena that can be achieved with stochastic models. In particular, they are unable to model complex system properties such as changes in phase space, the non-ergodic nature of complex system behavior, and their context-dependence [41].

We do not understand how humans classify texts or images or conduct conversations. Neither do we have mathematical causal (and thus predictive) models of the weather or of the Earth's climate. (Every mathematical causal model is also a predictive model because the explanations it provides can be used for an almost exact prediction.) Causal models for such complex systems are beyond the scope of currently available mathematics [34,42,43]. When we emulate human behavior, or the behavior of the weather system, we create approximations of these complex systems. But, as we have seen from the review of important examples from the XAI literature, we cannot understand these approximations any more than we understand the systems themselves.

This does not, however, mean that we cannot obtain highly functional, reliable, and safe AI systems which avoid undesired bias when we carefully design these systems.

3.5. Attitude Bias in Statistical Learning

In statistical learning, it is assumed that there are true properties of a distribution relation $f(X) = Y$. Against this view, bias is defined as the squared difference between the expectation E of an estimate $\hat{f}(x_0)$ and the real outcome $f(x_0)$:

$$[E\hat{f}(x_0) - f(x_0)]^2. \tag{7}$$

The bias thus indicates the amount by which the average of our estimate differs from the true mean. It is seen as a property, not of the distribution, but rather of the *model* which is interpreted as introducing a systematic deviation from the truth. Models of increasing complexity usually have a lower bias but a higher variance—the expected squared difference between the estimate and its expectation (average) [1] (Section 7.3):

$$E[\hat{f}(x_0) - E\hat{f}(x_0)]^2.$$

In the dNN- and XAI-communities however, bias is understood in the sense of *attitude bias* [4] (Section 5.1), which I define as the presence in humans of attitudes concerning group-related *value judgments* (also called stereotypes or prejudices) such as 'Asians are more intelligent than caucasians', or 'women have a higher emotional intelligence than men.' Attitude bias is thus a matter, not of systematic deviation of a model from the truth of a distribution, but rather a property of the training distribution X itself arising from group-related systematic value judgements. Where such biases are present in the training data, for example in large text corpora taken from the web [8], statistical models obtained from supervised or unsupervised training material will implicitly incorporate them. Of course, models will also incorporate the common-sense knowledge that is present in such data, for example, that the sun rises every day or that all animate organisms need water.

There are drastic ugly examples of the effects such *attitude bias* can have on trained models, such as the chatbot *Tay* which Microsoft had to turn off in 2016 because Twitter users had turned it into 'a neo-nazi sexbot'. (https://www.technologyreview.com/2016/03/24/161424/why-microsoft-accidentally-unleashed-a-neo-nazi-sexbot/, accessed on 17 January 2022).

How can such *attitude bias* inherited from training material be avoided? One road is by using human-curated material, which drastically reduces the amount of available training material and thereby prevents the usage of foundation models or supervised dNNs which are enormously data greedy.

But there is another road: through AI certification.

4. Certified AI

To obtain explainable, reliable, and safe AI systems, the systems need to be engineered in the way in which we have built technology since the 17th century, namely on the basis of experiments and the principle of composition. Some technical artifacts used in engineering are exactly understood. Others are understood only approximatively. For example, a jet plane is made of many parts, and many of them can be modeled very well. We also have an understanding of the lift force that enables the plane to fly—but we cannot model exactly how this force works. However, airplanes can be specified and tested, and the conditions under which they can depart, land, and fly can be specified to minimize the risks involved in traveling through the air. This has made air travel with jets the safest means of travel—safer by far than travel by car or train—if the specified conditions are met.

To achieve a comparable level of quality, and thus of safety, AI systems need to be engineered using a similar framework. This means the requirements along all of the following dimensions (the following enumeration is based loosely on ISO 25010 [Software and software quality models]).

- behavior —the suitability and accuracy of its functions,
- security—access control, and similar measures, which include anti-keyloggers, anti-malware, anti-spyware, anti-subversion, anti-tamper, anti-theft, antivirus, and cryptographic software, computer-aided dispatch, firewall, intrusion detection system, intrusion prevention system, log management software, records management, sandbox, security information management, security information and event management, software and operating system updating, and vulnerability management,
- reliability—maturity, fault tolerance, recoverability,
- usability,
- efficiency,
- maintainability, and
- portability

must be clearly stated, and AI systems realized to meet them.

Such systems can then be *certified*, as medical devices or airplanes are certified. We then obtain *certified AI* (CAI), something that is realistic and feasible, instead of XAI, which is impossible. Consider as an example a certain sort of human activity in the domain of precision engineering, which we seek to automate by means of an AI-controlled robot [34] (Chapter 13).

- Let X, Y be finite-dimensional vector spaces in $\mathbb{R}^m$ and $\mathbb{R}^n$, respectively, with $n, m \in \mathbb{N}$.
- Obtain a set of data (input-output tuples) from these spaces: $\langle \mathbf{X}_{k,m}, \mathbf{Y}_{k,n} \rangle$, where $k \in \mathbb{N}$ is the size of the set.
- Let $T_i : \mathbb{R}^p \mapsto \mathbb{R}^q, i \in \mathbb{N}, p, q \in \mathbb{N}_{>1}$ be operators and let $f_j : \mathbb{R}^\ell \mapsto \mathbb{R}, j \in \mathbb{N}_0, \ell \in \mathbb{N}$ be functionals.
- The functionals and operators can be algorithms, differential equations, syllogisms, or stochastic regression models (including neural networks). Note that the domain of T_1 is $\mathbb{R}^m$ and the range of T_n is $\mathbb{R}^n$.
- Let the operator T_0 represent the outcome that the ML or AI model is to emulate in the current context. This is an activity, which is performed to realize a certain step in the production process, for example, the combination of two delicate, small parts into one larger part. The machine which is supposed to replace the human then has to obtain an operator of the form

$$\hat{y} = \hat{T}_0 = T_1 \circ f_1 \circ \cdots \circ f_{m-1}^{\theta} \circ T_{n-1}^{\kappa} \circ f_m \circ T_n^{\lambda} \tag{8}$$

  where $\hat{y}$ is an estimator of T_0.
  The equation describes the composition ($\circ$) of a series of steps, each one consisting of the application of some functional or operator. These might be a stochastic model, an ontology-based mechanical theorem prover, a Bayesian network, a set of rules, an algebraic graph, and there are many other alternatives. The superscripts θ, κ and λ used on some of the functionals and operators indicate the usage of prior distributions or prior knowledge, a crucial point to which we will return below.
- The set of functionals and operators of the model can be trained and tested by using appropriate data-subsets in the usual ratio (see [1] (Chapter 7)).
- Finally, the model $\hat{y}$ is evaluated by using a validation partition from the data.

Each functional f_j and each operator T_i with $i > 0$ of the model represents a part of the relevant human act sequence which is decomposed when the algorithm emulating the animate behavior is designed. (In some cases, a single human act can be represented mathematically through the combination of several functionals or operators. For example, to mimic a human chess player, many operations are computed for one movement.) For example, T_1 might be an act of perception, f_1 corresponds to the first movement pattern of the human act sequence, and so on. Under certain conditions, it is possible to directly train the entire operator $\hat{y} = \hat{T}_0$ end-to-end, though such applications are rare in real-world problems.

4.1. Priors

The most important characteristic of the compositional model (8) is the usage of *priors*. The term *prior distribution* is used in Bayesian inference to describe a prior probability distribution $p(\theta)$, for example the likelihood of an unborn child to be a girl is very close to $p(\theta = 1) = \frac{1}{2}$, where $\theta \in \{0, 1\}$ indicates boy or girl, respectively. (Inherited anomalies of the sex chromosomes are very rare; girl-boy-birth-rate-imbalances can however sometimes be observed.) We know that this prior is correct from sampling the sex of children in human populations. In contrast, *prior knowledge* describes propositions that we know with certainty to be true (given certain conditions), such as Newton's laws. To explain the behavior of a system, we need to use such propositions because they allow a causal understanding of the system's function. To make a compositional system explainable, it suffices that the last functional or operator in the compositional chain uses prior knowledge. For example, if the operator T_n^λ in Equation (8) is a fully deterministic mechanical theorem prover operating as described in Section 4.1, then the results can be evaluated exactly against the requirements using automated testing with pre-defined test cases.

Even if some of the components of the chain described in Equation (8) are stochastic, the result will still be deterministic, testable, and reliable.

An Example of Prior Knowledge: Axiomatised Ontologies

Prior knowledge can be expressed using differential equations or via taxonomic or axiomatic ontologies. Taxonomic ontologies are directed graphs which relate entities (graph-theoretical nodes) to one another using set-membership relations (graph-theoretical edges) such as

$$\mathsf{human} \subset \mathsf{mammal} \subset \mathsf{animal} \subset \mathsf{animate\ being},$$

where each subset is derived from its superset using Aristotelian genus-species definitions that constrain the superset to the subset by adding more properties. Taxonomy-like ontologies using other entity relations are designated according to the relation types (as when meronymy is used for part-whole-taxonomies). Axiomatic ontologies consist of propositions expressed in (propositional, predicate, or modal) logic. Such ontologies are not necessarily graphs, because the propositions are not necessarily connected via relations, though they can be treated as nodes and related to each other explicitly to yield axiomatized taxonomies. This is the type of ontology that is obtained with the web ontology language OWL if all nodes are axiomatically defined using the modal description logic of the $\mathcal{SROIQ}^{(\mathcal{D})}$ standard [44]. Axiomatised ontologies without explicit node connections but common propositional elements can be seen as forming networks in which two propositions that share one or more elements are implicitly connected.

Such axiomatized ontologies or axiomatized taxonomies are extremely useful because they can be used with mechanical theorem provers (MTPs) to compute exact logical inferences. Consider a chatbot built for customer request management on the part of an electricity utility with the aid of an ontology and an MTP. Its penultimate operator T_{n-1}^λ consists of an axiomatized ontology Γ, which captures the most important target statements which customers make in the form of logical formulas and an MTP. $\Gamma = \Gamma_1 \cup \cdots \cup \Gamma_\ell$ is made of ℓ sub-ontologies to increase the expressive power of propositional formulae $\gamma_{ij} \in \Gamma_i, i = 1 \ldots \ell, j = 1 \ldots N$, where N is the number of formulae. These sub-ontologies can contain synonymies and compositional language elements that are regularly used by speakers, such as noun phrases related to time and space or verb phrases describing frequent activities.

For example, a set of target propositions that expresses the intent of a customer to revoke a recently signed contract looks like this (Abbreviations: V—verb, PHO—prepositional object head, S—subject, O—object, POP—preposition.):

$$\begin{aligned}(\gamma_{11}) : \mathsf{electroRequest}(V) \wedge \mathsf{reRevocation}(POH) \wedge (\mathsf{verb}(V,S) \vee ((\mathsf{you}(O)) \wedge \mathsf{verb}(V,S,O)))\\ \wedge \mathsf{mod}(POP, POH, V)\\ (\gamma_{22}) : ((\mathsf{request}(X) \vee \mathsf{ask}(X) \vee \mathsf{solicit}(X) \vee \mathsf{arrogate}(X) \vee \mathsf{demand}(X))\\ \rightarrow \mathsf{electroRequest}(X))\\ (\gamma_{23}) : ((\mathsf{withdrawal}(X) \vee \mathsf{suspension}(X) \vee \mathsf{revocation}(X) \vee \mathsf{annulment}(X) \vee \mathsf{abrogation}(X))\\ \rightarrow \mathsf{reRevocation}(X))\end{aligned}$$

Formula γ_{11} is an abstract representation of sentences like 'I hereby request a contract revocation.' The elements $\mathsf{electroRequest}(V)$ and $\mathsf{reRevocation}(POH)$ are artificial abstract terms used to increase the set of natural language sentences to which the proposition applies; the MTP can use them by replacing them with their synonyms. (The syntactic elements such as verb, you and mod are used to express in logical form the syntactic structure of the natural language sentence.) Formula γ_{22} provides context-specific synonyms for verbs indicating an act of requesting, formula γ_{23} provides such synonyms for nouns indicating contract revocation. An MTP can use the implications ($\rightarrow$) to infer formula γ_{11} from different combinations of the synonyms given in formulae γ_{22} and γ_{23}. The functionals and operators acting before the MTP-operator T^{λ}_{n-1} translate natural language sentences into logical formulae:

$$\text{SENTENCE} \rightsquigarrow \{\phi_1, \ldots, \phi_k\} = \Phi. \tag{9}$$

This allows the MTP to infer a target formula from the set of translations Φ:

$$\Phi \vdash \gamma_{ij} \in \Gamma. \tag{10}$$

The final operator T^{μ}_{n} is a rules engine that automatically directs those sentences to which no formula could be assigned to a human being for inspection. For those with a translation into logic, it uses the inferred formula γ_{ij} to perform operations on system services to fulfill the customer's request automatically.

The last two steps of the chain, the MTP T^{λ}_{n-1} and the rules engine T^{μ}_{n}, guarantee a deterministic, explainable and testable behavior of the system because they are logical steps. Both of them function on the basis of *prior knowledge* stated either in predicate logic as exemplified above in formulae ($\gamma_{11}, \gamma_{22},$ and γ_{23}), or using Horn clauses (these are clauses to express `if ... then` conditions in the form $\neg p_1 \vee \neg p_2 \ldots \neg p_k \vee q$, propositional logic) to run a rules engine.

Even if some of the earlier steps in the compositional chain (Equation (8)) are performed by stochastic AI functionals and operators, so long as the translations in Equation (9) are correct (because they, too, contain the deterministic functionals and operators used in the relevant parts of the chain), this will mean that the last two steps will act like a filter that guarantees that the system can be *certified* using systematic testing based on the specification of the system.

4.2. Other Approaches to Enhance Prior Knowledge in AI Appliactions

Prior distributions and causal stochastic relations (such as the relationship between smoking and increased lung-cancer likelihood) can be built-into stochastic systems also via approaches other than the ontologies described in the previous sections. For example, Bayesian networks [45] can be used to model known relationships between entities in combination with stochastic priors. There are many attempts to include priors in the dNN domain as well. For example, Holzinger et al. [14] propose to use graph neural networks to enable multi-modal information fusion in the medical domain using text, images, and other source types in one model. Another approach uses a 'subset of features

highlighted by domain experts as justifications for predictions, to enforce the alignment between local explanations and rationales.' [46]. Diligenti et al. [47] propose to use first-order logic clauses expressing prior knowledge as constraints integrated into the training process. An important earlier machine learning paper showed that the addition of virtual training examples to a training set is 'equivalent to incorporating the prior knowledge as a regularizer' [48]. When used for quality control of production in manufacturing [49], statistical learning can indeed lead to certifiable stochastic systems because models created from assembly-line data can in some situations be seen as models of logic systems. Such systems can be modeled with very high precision by dNN—because adequate and almost exact mathematical models are possible when deterministic laws of nature dominate the behavior of the system [50]. All these approaches are very important for the further improvement of statistical learning and need to be expanded, but it is unlikely that end-to-end neural network approaches, which remain stochastic models, after all, can compete with stochastic-deterministic hybrids to meet the requirements of certification in complex system modeling tasks and contexts. On the other hand, full stochastic models are much more flexible than such hybrids; which model type to use therefore depends on requirements.

4.3. Legal and Ethical Aspects of Certified AI

Certified AI can be used to meet the legal and ethical requirements for the usage of AI. For example, the Charter of Fundamental Rights (CFR) requires that patients give a 'free and informed consent' to medical procedures performed on them and, as Stöger et al. [51] point out, there should be a 'shared decision-making' by the patient and the physician. The usage of certified AI would certainly greatly serve the goals formulated in the CFR because the deterministic character of the last step of the algorithm allows the physician to explain the way the AI works in the same way a surgical intervention or a model of pharmacodynamics (which is in all cases less accurate than a deterministic computable algorithm due to the interaction of the drug with the complex system of the human body) can be elucidated to the patient.

Certified AI could also support ethical goals for the usage of AI in medicine. Recently, 'Ten commandments' have been proposed for the medical application of AI [52]. Many of these can be well supported if the AI is certified in the way described in Section 4.1. For example, goals 1 (transparency regarding the source of a decision) or goals 4 and 5 (explainability and repeatability) are enabled by CAI with regard to the final result of the computation (but not every intermediate possibly stochastic step) if the last step of a compositional chain is deterministic and adequate tests have been performed.

5. Conclusions

We have seen that implicit stochastic models do not allow explanations, but rather only one or other sort of interpretation in the sense of hermeneutics. Such interpretations are useless in the realm of technical systems. Who among us would be satisfied with a pacemaker or insulin pump whose operations are merely *interpreted*? The very idea that one might be willing to use technical production systems in mission-critical applications using end-to-end stochastic AI is in any case absurd. No one wants a cruise missile flying with a stochastic model as its sole guiding component. All mission-critical technologies need to be certified, and to achieve this, deterministic components are needed at the critical steps of the compositional chain. Many components can and should still be stochastic in order to exploit the enormous potential of statistical learning. But only if the test-relevant components are deterministic can we achieve explainability and obtain the sort of certified AI that we can specify and systematically test using real-world test scenarios.

Prior knowledge expressed, as in the small example given above, using differential equations, fully axiomatized ontologies in predicate logic or in modal logic are crucial to enable certified AI and to avoid the *attitude bias* implicitly built into dNNs from their training material.

Currently, we can take AI systems into mass-production above all in the approximative worlds of, for example, consumer advertisement placing as practiced by Google or Amazon. To take them further and into domains requiring the sort of exact behaviors that we have come to expect from the technical devices which surround us, will require AI systems that can be certified to behave according to our specifications.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: I thank Raija Kramer for the ontology example and Daniel Beck for their critical reviews of the manuscript.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning*, 2nd ed.; Springer: Berlin, Germany, 2008.
2. Goodfellow, I.J.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
3. Brown, T.B.; Mann, B.P.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. *arXiv* **2020**, arXiv:2005.14165.
4. Bommasani, R.; Hudson, D.A.; Adeli, E.; Altman, R.; Arora, S.; von Arx, S.; Bernstein, M.S.; Bohg, J.; Bosselut, A.; Brunskill, E.; et al. On the Opportunities and Risks of Foundation Models. *arXiv* **2021**, arXiv:2108.07258.
5. van den Oord, A.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; Kavukcuoglu, K. WaveNet: A Generative Model for Raw Audio. *arXiv* **2016**, arXiv:1609.03499.
6. van den Oord, A.; Li, Y.; Vinyals, O. Representation Learning with Contrastive Predictive Coding. *arXiv* **2019**, arXiv:1807.03748.
7. Rudin, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.* **2019**, *1*, 206–215. [CrossRef]
8. Caliskan, A.; Bryson, J.J.; Narayanan, A. Semantics derived automatically from language corpora contain human-like biases. *Science* **2017**, *356*, 183–186. [CrossRef]
9. Weber, M. *Gesammelte Aufsätze zur Wissenschaftslehre*; J.C.B. Mohr: Tübingen, Germany, 1988.
10. Hesse, M. *Models and Analogies in Science*; Sheed and Ward: London, UK, 1963.
11. Poser, H. *Wissenschaftstheorie: Eine Philosophische Einführung*; Reclam: Stuttgart, Germany, 2001.
12. Gadammer, H.H. Hermeneutik. In *Historisches Wörterbuch der Philosophie*; Ritter, J.; Gründer, K., Eds.; Schwabe Verlag: Basel, Switzerland , 1974; Volume 3, pp. 1062–74.
13. Tinsley, H.E.; Weiss, D.J. Interrater reliability and agreement. In *Handbook of Applied Multivariate Statistics and Mathematical Modeling*; Elsevier: Amsterdam, The Netherlands, 2000; pp. 95–124.
14. Holzinger, A.; Malle, B.; Saranti, A.; Pfeifer, B. Towards multi-modal causability with Graph Neural Networks enabling information fusion for explainable AI. *Inf. Fusion* **2021**, *71*, 28–37. [CrossRef]
15. Hayek, F.A.V. *The Sensory Order. An Inquiry into the Foundations of Theoretical Psychology*; Chicago University Press: Chicao, IL, USA, 1952.
16. Miller, G.A. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychol. Rev.* **1956**, *63*, 81. [CrossRef]
17. Rousseauw, J.d.; Du Plessis, J.; Benade, A.; Jordaan, P.; Kotze, J.; Jooste, P.; Ferreira, J. Coronary risk factor screening in three rural communities. *S. Afr. Med J.* **1983**, *64*, 216.
18. Nembrini, S.; König, I.R.; Wright, M.N. The revival of the Gini importance? *Bioinformatics* **2018**, *34*, 3711–3718. [CrossRef]
19. Fahrmeir, L.; Kneib, T.; Lang, S.; Marx, B. *Regression Models*; Springer: Berlin, Germany, 2013.
20. Lapuschkin, S.; Wäldchen, S.; Binder, A.; Montavon, G.; Samek, W.; Müller, K.R. Unmasking Clever Hans Predictors and Assessing What Machines Really Learn. *Nat. Commun.* **2019**, *10*, 1–8. [CrossRef] [PubMed]
21. Lundberg, S.; Lee, S.I. A Unified Approach to Interpreting Model Predictions. *arXiv* **2017**, arXiv:1705.07874.
22. Doshi-Velez, F.; Kim, B. Towards a rigorous science of interpretable machine learning. *arXiv* **2017**, arXiv:1702.08608.
23. Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.R.; Samek, W. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE* **2015**, *10*, e0130140. [CrossRef]
24. Moosavi-Dezfooli, S.; Fawzi, A.; Fawzi, O.; Frossard, P. Universal Adversarial Perturbations. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
25. Jo, J.; Bengio, Y. Measuring the Tendency of CNNs to Learn Surface Statistical Regularities. *arXiv* **2017**, arXiv:1711.11561.
26. Mu, J.; Andreas, J. Compositional Explanations of Neurons. *arXiv* **2021**, arXiv:2006.14032.
27. Carter, S.; Armstrong, Z.; Schubert, L.; Johnson, I.; Olah, C. Activation atlas. *Distill* **2019**, *4*, e15. [CrossRef]

28. Gaver, W.W. Technology affordances. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Orleans Louisiana, LA, USA, 27 April–2 May 1991; pp. 79–84.
29. Bau, D.; Zhou, B.; Khosla, A.; Oliva, A.; Torralba, A. Network dissection: Quantifying interpretability of deep visual representations. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6541–6549.
30. Olah, C.; Cammarata, N.; Schubert, L.; Goh, G.; Petrov, M.; .; Carter, S. An introduction to circuits. *Distill* **2020**, *5*, e00024. [CrossRef]
31. Goh, G.; Cammarata, N.; Voss, C.; Carter, S.; Petrov, M.; Schubert, L.; Radford, A.; Olah, C. Multimodal neurons in artificial neural networks. *Distill* **2021**, *6*, e30. [CrossRef]
32. Lindsay, G.W. Convolutional neural networks as a model of the visual system: Past, present, and future. *J. Cogn. Neurosci.* **2021**, *33*, 2017–2031. [CrossRef]
33. Landgrebe, J.; Smith, B. Making AI Meaningful Again. *Synthese* **2021**, *198*, 2061–2081. [CrossRef]
34. Landgrebe, J.; Smith, B. *Why Machines Will Never Rule the Earth*; Routledge: London, UK, 2022.
35. Reed, S.E.; Akata, Z.; Mohan, S.; Tenka, S.; Schiele, B.; Lee, H. Learning What and Where to Draw. In *Advances in Neural Information Processing Systems*; Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R., Eds.; Curran Associates, Inc.: Barcelona, Spain, 2016; Volume 29.
36. Zhai, L.; Juefei-Xu, F.; Guo, Q.; Xie, X.; Ma, L.; Feng, W.; Qin, S.; Liu, Y. It's Raining Cats or Dogs? Adversarial Rain Attack on DNN Perception. *arXiv* **2020**, arXiv:2009.09205.
37. Church, A. A note on the Entscheidungsproblem. *J. Symb. Log.* **1936**, *1*, 40–41. [CrossRef]
38. Boolos, G.S.; Burgess, J.P.; Jeffrey, R.C. *Computability and Logic*; Cambridge University Press: Cambridge, MA, USA, 2007.
39. Davis, M.; Matijasevic, Y.; Robinson, J. Hilbert's Tenth Problem. Diophantine Equations: Positive Aspects of a Negative Solution. In *Proceedings of the Symposia in Pure Mathematics*; Rutgers University: New Brunswick, NJ, USA, 1976; Volume 28, pp. 323–378.
40. Davis, M. The Myth of Hypercomputation. In *Alan Turing: Life and Legacy of a Great Thinker*; Teuscher, C., Ed.; Springer: Heidelberg, Germany, 2004; pp. 195–211.
41. Landgrebe, J.; Smith, B. An argument for the impossibility of machine intelligence. *arXiv* **2021**, arXiv:2111.07765.
42. Hayek, F.A.V. The pretence of knowledge. In *The Market and Other Orders*; Caldwell, B., Ed.; University of Chicago Press: Chicago, IL, USA, 2014; pp. 362–372.
43. Thurner, S.; Klimek, P.; Hanel, R. *Introduction to the Theory of Complex Systems*; Oxford University Press: Oxford, UK, 2018.
44. Baader, F.; Calvanese, D.; McGuinness, D.; Patel-Schneider, P.; Nardi, D. *The Description Logic Handbook: Theory, Implementation and Applications*; Cambridge University Press: Cambridge, MA, USA, 2003.
45. Cowell, R.G.; Dawid, A.P.; Lauritzen, S.L.; Spiegelhalter, D.J. *Probabilistic Networks and Expert Systems*; Springer: Berlin, Germany, 2007.
46. Du, M.; Liu, N.; Yang, F.; Hu, X. Learning credible DNNs via incorporating prior knowledge and model local explanation. *Knowl. Inf. Syst.* **2021**, *63*, 305–332. [CrossRef]
47. Diligenti, M.; Roychowdhury, S.; Gori, M. Integrating prior knowledge into deep learning. In Proceedings of the 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, 18–21 December 2017; pp. 920–923.
48. Niyogi, P.; Girosi, F.; Poggio, T. Incorporating prior information in machine learning by creating virtual examples. *Proc. IEEE* **1998**, *86*, 2196–2209. [CrossRef]
49. Borg, M.; Englund, C.; Wnuk, K.; Duran, B.; Levandowski, C.; Gao, S.; Tan, Y.; Kaijser, H.; Lönn, H.; Törnqvist, J. Safely entering the deep: A review of verification and validation for machine learning and a challenge elicitation in the automotive industry. *arXiv* **2018**, arXiv:1812.05389.
50. Greydanus, S.; Dzamba, M.; Yosinski, J. Hamiltonian Neural Networks. *arXiv* **2019**, arXiv:1906.01563.
51. Stöger, K.; Schneeberger, D.; Holzinger, A. Medical artificial intelligence: the European legal perspective. *Commun. ACM* **2021**, *64*, 34–36. [CrossRef]
52. Muller, H.; Mayrhofer, M.T.; Van Veen, E.B.; Holzinger, A. The Ten Commandments of ethical medical AI. *Computer* **2021**, *54*, 119–123. [CrossRef]