Informed Machine Learning – A Survey and Taxonomy of Integrating Knowledge into Learning Systems

Laura von Rueden, Sebastian Mayer, Katharina Beckh, Bogdan Georgiev, Sven Giesselbach, Raoul Heese, Birgit Kirsch, Julius Pfrommer, Annika Pick, Rajkumar Ramamurthy, Michał Walczak, Jochen Garcke, Christian Bauckhage and Jannis Schuecker

Abstract—Despite its great success, machine learning can have its limits when dealing with insufficient training data. A potential solution is the additional integration of prior knowledge into the training process, which leads to the notion of *informed machine learning*. In this paper, we present a structured overview over various approaches in this field. First, we provide a definition and propose a concept for informed machine learning, which illustrates its building blocks and distinguishes it from conventional machine learning. Second, we introduce a taxonomy that serves as a classification framework for informed machine learning approaches. It considers the source of knowledge, its representation, and its integration into the machine learning pipeline. Third, we survey related research and describe how different knowledge representations such as algebraic equations, logic rules, or simulation results can be used in informed machine learning. This evaluation of numerous papers on the basis of our taxonomy uncovers key methods in this field.

Index Terms—Machine Learning, Prior Knowledge, Expert Knowledge, Informed, Hybrid, Survey.

✦

1 INTRODUCTION

MACHINE learning has shown great success in building models for pattern recognition in domains ranging from computer vision [1] over speech recognition [2] and text understanding [3] to Game AI [4]. In addition to these classical domains, machine learning and in particular deep learning are increasingly important and successful in engineering and the sciences [5], [6], [7]. These success stories are grounded in the data-based nature of the approach of learning from a tremendous number of examples.

However, there are many circumstances where purely data-driven approaches can reach their limits or lead to unsatisfactory results. The most obvious scenario is that not enough data is available to train well-performing and sufficiently generalized models. Another important aspect is that a purely data-driven model might not meet constraints such as dictated by natural laws, or given through regulatory or security guidelines. With machine learning models becoming more and more complex, there is also a growing need for models to be interpretable and explainable [8].

These issues have led to increased research on how to improve machine learning models by additionally incorporating prior knowledge into the learning process. For example, logic rules [9], [10] or algebraic equations [11], [12] have been added as constraints to loss functions. Knowledge graphs can enhance neural networks with information about relations between instances [13], which is of interest in image classification [14], [15]. Furthermore, physical simulations have been used to enrich data in machine learning [16], [17], [18]. This heterogeneity in approaches leads to some redundancy in nomenclature; for instance, we find terms such as physics-informed deep learning [19], physics-guided neural networks [11], or semantic-based regularization [20]. The growing number and increasing variety of research papers in this field thus motivates a systematic survey.

Here, we therefore provide a structured overview based on a survey of a large number of research papers on how to integrate additional, prior knowledge into the machine learning pipeline. As an umbrella term for such methods, we henceforth use *informed machine learning*.

Our contributions are threefold: First, we propose a concept for informed machine learning that clarifies its building blocks and relation to conventional machine learning. Second, we introduce a taxonomy that classifies informed machine learning approaches according to the source of knowledge, its representation, and its integration into the machine learning pipeline. We put a special emphasis on categorizing various knowledge representations, since this may enable practitioners to incorporate their domain knowledge into machine learning processes. Third, we present a research survey and describe how different knowledge representations, e.g., algebraic equations, logic rules, or simulation results, can be used in informed machine learning.

- *All authors are with the Fraunhofer Center for Machine Learning.*
- *Laura von Rueden, Katharina Beckh, Bogdan Georgiev, Sven Giesselbach, Birgit Kirsch, Annika Pick, Rajkumar Ramamurthy, Christian Bauckhage and Jannis Schuecker are with the Fraunhofer IAIS, Institute for Intelligent Analysis and Information Systems, 53757 Sankt Augustin, Germany.*
- *Sebastian Mayer and Jochen Garcke are with the Fraunhofer SCAI, Institute for Algorithms and Scientific Computing, 53757 Sankt Augustin, Germany.*
- *Raoul Heese and Michał Walczak are with the Fraunhofer ITWM, Institute for Industrial Mathematics, 67663 Kaiserslautern, Germany.*
- *Julius Pfrommer is with the Fraunhofer IOSB, Institute for Optronics, System Technologies and Image Exploitation, 76131 Karlsruhe, Germany.*
- *Corresponding author: laura.von.rueden@iais.fraunhofer.de*

Our goal is to equip potential new users of informed machine learning with established and successful methods. As we intend to survey a broad spectrum of methods in this field, we cannot describe all methical details and we do not claim to have covered all available research papers. We rather aim to analyze and describe common grounds as well as the diversity of approaches in order to identify the main research directions in informed machine learning.

In Section 2, we begin with a brief historical account. In Section 3, we then formalize our concept of *informed machine learning* and define our notion of knowledge and its integration into machine learning. Section 4 introduces the taxonomy we distilled from surveying a large number of research papers. In Section 5, we present the core of our survey by describing the methods for the integration of knowledge into machine learning classified according to the taxonomy and in particular structured along the knowledge representation. Finally, we conclude in Section 6 with a discussion and a summary of our findings.

2 HISTORICAL OVERVIEW

The idea of integrating knowledge into learning has a long history. Historically, AI research roughly considered the two antipodal paradigms of symbolism and connectionism. The former dominated up until the 1980s and refers to reasoning based on symbolic knowledge; the latter became more popular in the 1990s and considers data-driven decision making using neural networks. Especially Minsky [21] pointed out limitations of symbolic AI and promoted a stronger focus on data-driven methods to allow for causal and fuzzy reasoning. Already in the 1990s were knowledge data bases then used together with training data to obtain knowledge-based artificial neural networks [22]. In the 2000s, when support vector machines (SVMs) were the de-facto paradigm in classification, there was interest in incorporating knowledge into this formalism [23]. Moreover, in the geosciences, and most prominently in weather forecasting, knowledge integration dates back to the 1950s. There is indeed a whole discipline called data assimilation that deals with techniques, which combine statistical and mechanistic models to improve prediction accuracy [24], [25].

The recent growth of research activities shows that the combination of data- and knowledge-driven approaches becomes relevant in more and more areas. A recent survey synthesizes this into a new paradigm of theory-guided data science and points out the importance of enforcing scientific consistency in machine learning [26]. Also the fusion of symbolic and connectionist AI seems approachable only now with modern computational resources and ability to handle non-Euclidean data. In this regard, we refer to a survey on graph neural networks and a research direction framed as relational inductive bias [27].

Our work complements the aforementioned surveys by providing a systematic categorization of knowledge representations that are integrated into machine learning.

3 CONCEPT OF INFORMED MACHINE LEARNING

In this section, we present our concept of *informed machine learning*. We first state our notion of knowledge and then present our descriptive definition of its integration into machine learning.

3.1 Knowledge

The meaning of "knowledge" is difficult to define in general. In epistemology, the branch of philosophy dealing with the theory of knowledge, it is classically defined as true, justified belief [29]. Yet, there are numerous modifications and an ongoing debate as to this definition [30], [31].

Here, we therefore assume a computer-scientific perspective and understand "knowledge" as validated information about relations between entities in certain contexts. During its generation, knowledge first appears as "useful information" [32], which is subsequently validated. People validate information about the world using the brain's inner statistical processing capabilities [33], [34] or by consulting trusted authorities. Explicit forms of validation are given by empirical studies or scientific experiments [31], [35].

Regarding its use in machine learning, an important aspect of knowledge is its formalization. The degree of formalization depends on whether knowledge has been put into writing, how structured the writing is, and how formal and strict the language is that was used (e.g., natural language vs. mathematical formula). The more formally knowledge is represented, the more easily it can be integrated into machine learning.

3.2 Integrating Prior Knowledge into Machine Learning

Apart from the usual information source in a machine learning pipeline, the training data, one can additionally integrate formal knowledge. If this knowledge is pre-existent and independent of learning algorithms, it can be called prior knowledge. Machine learning that explicitly integrates such prior knowledge will henceforth be called *informed machine learning*.

Definition. *Informed machine learning* describes learning from a hybrid information source that consists of data and prior knowledge. The prior knowledge is pre-existent and separated from the data and is explicitly integrated into the machine learning pipeline.

This notion of informed machine learning thus describes the flow of information in Figure 1 and is distinct from conventional machine learning.

3.2.1 Conventional Machine Learning

As illustrated in Figure 1, conventional machine learning starts with a specific problem for which there is training data. These are fed into the machine learning pipeline, which delivers a solution. Problems are typically formalized as regression tasks where inputs X have to be mapped to outputs Y. Training data is generated or collected and then processed by algorithms, which try to approximate the unknown mapping. This pipeline comprises four main components, namely the training data, the hypothesis set, the learning algorithm, and the final hypothesis [28].

In traditional approaches, knowledge is used for feature engineering and training data preprocessing. However, this kind of integration is involved and deeply intertwined with the choice of the hypothesis set as well as the learning

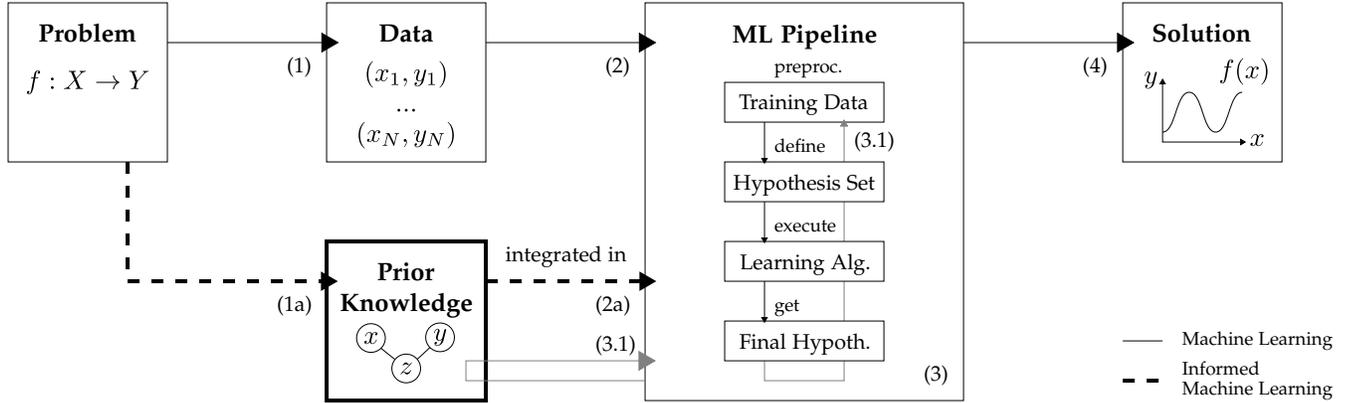**Information Flow of (Informed) Machine Learning**

Figure 1: **Information Flow in Informed Machine Learning.** The informed machine learning pipeline requires two independent sources of information: Data and prior knowledge. Conventional machine learning (continuous arrows) considers training data (1), which are fed into a machine learning pipeline (2). This pipeline produces a final hypothesis (3), which solves the the original problem by means of a model (4). The learning pipeline itself consists of four components, namely the preprocessing of the training data, the definition of a hypothesis set, the execution of a learning algorithm, and the generation of a final hypothesis [28]. Knowledge is generally used in this pipeline, however, often only for preprocessing and, in particular, feature engineering, which can also happen in loops (3.1). In contrast to this common form of machine learning, in *informed machine learning* (dashed arrows), prior knowledge about the problem already exists in advance (1a), for instance in form of knowledge graphs, simulation results, or logic rules. This prior knowledge constitutes a second, independent source of information and is integrated into the machine learning pipeline via explicit interfaces (2a).

algorithm. Hence, it is not independent of the design of the machine learning pipeline itself.

3.2.2 Informed Machine Learning

Figure 1 also shows that the information flow of informed machine learning consists of two lines originating from the problem. These involve the usual training data and additional prior knowledge. The latter exists independently of the learning task and can be provided in form of logic rules, simulation results, knowledge graphs, etc.

The essence of *informed machine learning* is that this prior knowledge is explicitly integrated into the machine learning pipeline, ideally via clear interfaces defined by the knowledge representations. Theoretically, this applies to each of the four components of the machine learning pipeline.

4 TAXONOMY

In this section, we introduce our taxonomy for *informed machine learning* where we focus on the process of integrating prior knowledge into the machine learning pipeline and divide this process into three stages. Each of these stages comes along with a corresponding question:

1) What kind of **knowledge source** is integrated?
2) How is this **knowledge represented**?
3) Where is the **knowledge integrated** in the machine learning pipeline?

To systematically answer these questions, we analyzed a large number of research papers describing informed machine learning approaches. Based on a comparative and iterative approach, we find that specific answers occur frequently. Hence, we propose a taxonomy, which can serve as a classification framework for informed machine learning approaches (see Figure 2). Guided by the above questions, our taxonomy consists of the three dimensions *knowledge source*, *knowledge representation* and *knowledge integration*. While an extensive literature categorization according to this taxonomy will be presented in the next section (Section 5), we first discuss the taxonomy on a conceptual level.

4.1 Knowledge Source

The category *knowledge source* refers to the origin of prior knowledge to be integrated in machine learning. With respect to the information flow in Figure 1, the knowledge source is related to problem specification.

We observe that the source of prior knowledge can be an established knowledge domain but also knowledge from an individual group of people with respective experience. We find that prior knowledge often stems from natural or social sciences or is a form of expert or world knowledge, as illustrated on the left in Figure 2. This list is neither complete nor disjoint but intended show a spectrum from explicitly to implicitly validated knowledge. Although particular knowledge can be assigned to more than one of these sources, the goal of this categorization is to identify paths in our taxonomy that describe frequent approaches of knowledge integration into machine learning. In the following we shortly describe each of the knowledge sources.

Natural Sciences. We subsume the subjects of science, technology, engineering, and mathematics under *natural sciences*. Such knowledge is typically validated explicitly

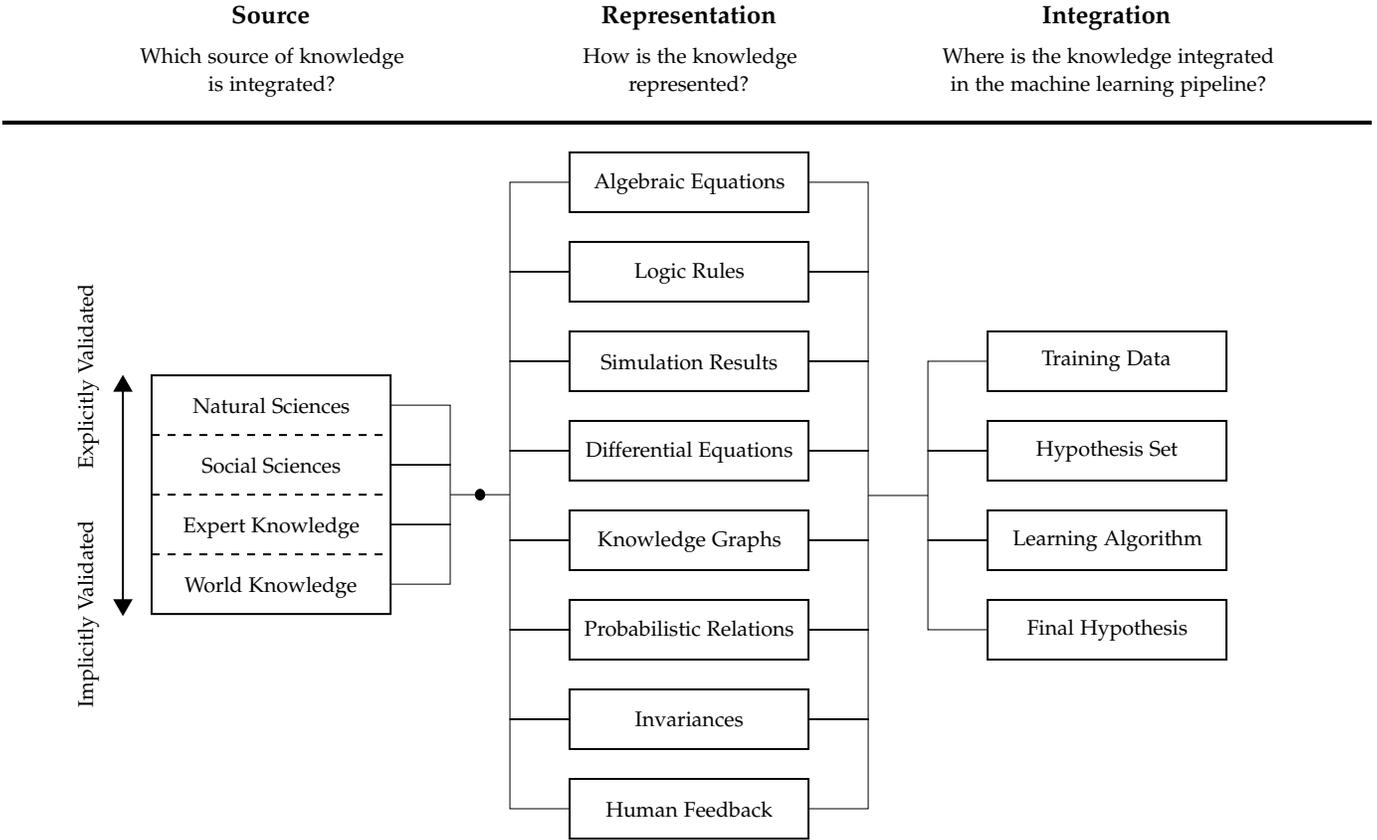| **Source**	**Representation**	**Integration**
Which source of knowledge is integrated?	How is the knowledge represented?	Where is the knowledge integrated in the machine learning pipeline?

Figure 2: **Taxonomy of Informed Machine Learning.** This taxonomy serves as a classification framework for *informed machine learning* and structures known approaches according to the three questions in the top row. For each question, the most frequent answers found in the literature constitute the categories listed here. Lines between building blocks indicate potential combinations of different categories and thus signal paths through the taxonomy. Answers to the question *"Which source of knowledge is integrated?"* are not necessarily disjoint but form a continuous spectrum. The small circle on the outgoing lines indicates that knowledge can be merged for further processing. Answers to the two questions of *"How is the knowledge represented?"* and *"Where is the knowledge integrated into the machine learning pipeline?"* are independent but can occur in combination. The goal of this taxonomy is to organize approaches and research activities so that specific paths along the categories represent research subfields.

through scientific experiments. Examples are the universal laws of physics, bio-molecular descriptions of genetic sequences, or material-forming production processes.

Social Sciences. Under *social sciences*, we subsume subjects like social psychology, economics, and linguistics. Such knowledge is often explicitly validated through empirical studies. Examples are effects in social networks or the syntax and semantics of language.

Expert Knowledge. We consider *expert knowledge* to be knowledge that is held by a particular group of experts. Within the expert's community it can also be called common knowledge. Such knowledge can be validated implicitly through a group of experienced specialists. In the context of cognitive science, this expert knowledge can also become intuitive [33]. For example, an engineer or a physician acquires knowledge over several years of experience working in a specific field.

World Knowledge. By *world knowledge* we refer to facts from everyday life that are known to almost everyone and can thus also be called general knowledge. Such knowledge can be intuitive and validated implicitly by humans reasoning in the world surrounding them. Therefore, world knowledge often describes relations of objects or concepts appearing in the world perceived by humans, for instance, the fact that a bird has feathers and can fly.

4.2 Knowledge Representation

The category *knowledge representation* describes how knowledge is formally represented. With respect to the flow of information in informed machine learning in Figure 1, it directly corresponds to our key element of prior knowledge. This category constitutes the central building block of our taxonomy, because it determines the potential interface to the machine learning pipeline.

In our literature survey, we frequently encountered certain representation types, as listed in the taxonomy in Figure 2 and illustrative more concretely in Table 1. Our goal is to provide a classification framework of informed machine learning approaches including the used knowledge representation types. Although some types can be mathematically transformed into each other, we keep the representation that are closest to those in the reviewed literature. Here we give a first conceptual overview over these types.
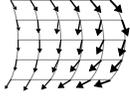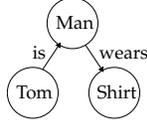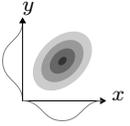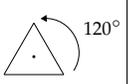
Algebraic Equations	Logic Rules	Simulation Results	Differential Equations	Knowledge Graphs	Probabilistic Relations	Invariances	Human Feedback
$E = m \cdot c^2$ $v \leq c$	$A \wedge B \Rightarrow C$		$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$ $F(x) = m \frac{d^2 x}{dt^2}$				

Table 1: **Illustrative Overview of Knowledge Representations in Informed Machine Learning.** This table shows an overview of the knowledge representations in the informed machine learning taxonomy. Each representation is illustrated by simple or prominent examples[1] in order to give a first intuitive understanding of the different types and their differentiation.

Algebraic Equations. Algebraic equations represent knowledge as equality or inequality relations between mathematical expressions consisting of variables or constants. Equations can be used to describe general functions or to constrain variables to a feasible set and are thus sometimes also called algebraic constraints. Prominent examples in Table 1 are the equation for the mass-energy equivalence and the inequality stating that nothing can travel faster than the speed of light in vacuum.

Logic Rules. Logic provides a way of formalizing knowledge about facts and dependencies and allows for translating ordinary language statements (e.g., IF A THEN B) into formal logic rules ($A \Rightarrow B$). Generally, a logic rule consists of a set of Boolean expressions (A, B) combined with logical connectives ($\wedge$, $\vee$, $\Rightarrow$, ...). Logic rules can be also called logic constraints or logic sentences.

Simulation Results. Simulation results describe the numerical outcome of a computer simulation, which is an approximate imitation of the behavior of a real-world process. A simulation engine typically solves a mathematical model using numerical methods and produces results for situation-specific parameters. Its numerical outcome is the simulation result that we describe here as the final knowledge representation. Examples are the flow field of a simulated fluid or pictures of simulated traffic scenes.

Differential Equations. Differential equations are a subset of algebraic equations, which describe relations between functions and their spatial or temporal derivatives. Two famous examples in Table 1 are the heat equation, which is a partial differential equation (PDE), and Newton's second law, which is an ordinary differential equation (ODE). In both cases, there exists a (possibly empty) set of functions that solve the differential equation for given initial or boundary conditions. Differential equations are often the basis of a numerical computer simulation. We distinguish the taxonomy categories of differential equations and simulation results in the sense that the former represents a compact mathematical model while the latter represents unfolded, data-based computation results.

Knowledge Graphs. A graph is a pair (V, E), where V are its vertices and E denotes edges. In a knowledge graph, vertices (or nodes) usually describe concepts whereas edges represent (abstract) relations between them (as in the example "Man wears shirt" in in Table 1). In an ordinary weighted graph, edges quantify the strength and the sign of a relationship between nodes.

Probabilistic Relations. The core concept of probabilistic relations is a random variable X from which samples x can be drawn according to an underlying probability distribution $P(X)$. Two or more random variables X, Y can be interdependent with joint distribution $(x, y) \sim P(X, Y)$. Prior knowledge could be assumptions on the conditional independence or the correlation structure of random variables or even a full description of the joint probability distributions.

Invariances. Invariances describe properties that do not change under mathematical transformations such as translations and rotations. If a geometric object is invariant under such transformations, it has a symmetry (for example, a rotationally symmetric triangle). A function can be called invariant, if it has the same result for a symmetric transformation of its argument. Connected to invariance is the property of equivariance.

Human Feedback. Human feedback refers to technologies that transform knowledge via direct interfaces between users and machines. The choice of input modalities determines the way information is transmitted. Typical modalities include keyboard, mouse, and touchscreen, followed by speech and computer vision, e.g., tracking devices for motion capturing. In theory, knowledge can also be transferred directly via brain signals using brain-computer interfaces.

4.3 Knowledge Integration

The category *knowledge integration* describes where the knowledge is integrated into the machine learning pipeline.

Our literature survey revealed that integration approaches can be structured according to the four components of training data, hypothesis set, learning algorithm, and final hypothesis. Though we present these approaches more thoroughly in Section 5, the following gives a first conceptual overview.

Training Data. A standard way of incorporating knowledge into machine learning is to embody it in the underlying training data. Whereas a classic approach in traditional machine learning is feature engineering where appropriate features are created from expertise, an informed approach according to our definition is the use of hybrid information in terms of the original data set and an additional, separate source of prior knowledge. This separate source of prior knowledge allows to accumulate information and therefore can create a second data set, which can then be used together with, or in addition to, the original training data. A prominent approach is simulation-assisted machine learning where the training data is augmented through simulation results.
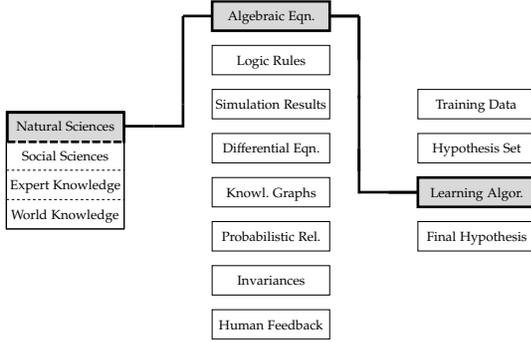
Figure 3: **Example for an Observed Taxonomy Path.** This figure illustrates one of the taxonomy paths we frequently observed in the literature. It highlights the use of knowledge from natural the sciences given in form of algebraic equations, which are then integrated into the learning algorithm. Some papers that follow this path are [11], [12], [36].

Hypothesis Set. Integrating knowledge into the hypothesis set is common, say, through the definition of a neural network's architecture and hyper-parameters. For example, a convolutional neural network applies knowledge as to location and translation invariance of objects in images. More generally, knowledge can be integrated by choosing model structure. A notable example is the design of a network architecture considering a mapping of knowledge elements, such as symbols of a logic rule, to particular neurons.

Learning Algorithm. Learning algorithms typically involve a loss function that can be modified according to additional knowledge, e.g. by designing an appropriate regularizer. A typical approach of informed machine learning is that prior knowledge in form of algebraic equations, for example laws of physics, is integrated by means of additional loss terms.

Final Hypothesis. The output of a learning pipeline, i.e. the final hypothesis, can be benchmarked against existing knowledge. For example, predictions that do not agree with known constraints can be discarded or marked as suspicious so that results are consistent with prior knowledge.

4.4 Classification Paths

Next, we illustrate the use of our taxonomy as a classification framework by looking at examples of common paths.

In our literature survey, we categorized each paper according to one (or more) entries in each of the three taxonomy dimensions. A specific combination of entries across dimensions then represents a methodology for informed learning and constitutes a path through the taxonomy[2]. An example for such a path is shown in Figure 3.

We observed that specific paths, i.e. ways of integrating prior knowledge into machine learning, occur more frequently than others so that we call them main paths. A more detailed description of methods used in the papers will follow in Section 5, where we will focus on knowledge representations and their methodical integration into machine learning.

---

2. We use the word path, since a path results when visually highlighting and connecting the entries in the taxonomy.

5 KNOWLEDGE REPRESENTATIONS IN INFORMED MACHINE LEARNING

In this section, we give a detailed account of the informed machine learning approaches we found in our literature survey. We will focus on methods and therefore structure our presentation according to knowledge representations. This is motivated by the assumption that similar representations are integrated into machine learning in similar ways as they form the mathematical basis for the integration.

For each knowledge representation, we describe the informed machine learning approaches in a separate subsection and present the observed (paths from) knowledge source and the observed (paths to) knowledge integration. We describe each dimension along its entities starting with the main path entity, i.e. the one we found in most papers.

This whole section refers to Table 2, which lists paper references sorted according to our taxonomy.

5.1 Algebraic Equations

Figure 4 summarizes the taxonomy paths we observed in our literature survey, which involve the representation of prior knowledge in form of algebraic equations. The path, i.e. the class of combinations, for which we found the most references comes from natural sciences and goes to the learning algorithm.
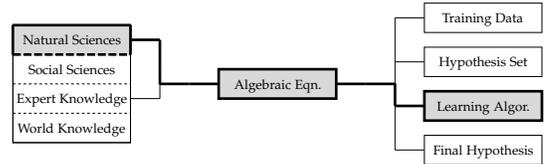

Figure 4: Taxonomy Paths Observed for Algebraic Eqn.

5.1.1 (Paths from) Knowledge Source

Algebraic equations are frequently used to represent knowledge from natural sciences but may also represent other forms of expert knowledge.

Natural Sciences. Equations from the natural sciences that are integrated into machine learning come from diverse areas, particularly from physics [11], [12], [37], [38], [39] but also from biology [36], [40], [41], robotics [42], or production processes [39].

Examples of physical knowledge integrated into machine learning are the following: The trajectory of objects can be described with kinematic laws, e.g., that the position y of a falling object can be described as a function of time t, namely $y(t) = y_0 + v_0 t + a t^2$ [12]. Or, the proportionality of two variables can be expressed via inequality constraints, for example, that the water density ρ at two different depths $d_1 < d_2$ in a lake must obey $\rho(d_1) \leq \rho(d_2)$ [11].

As examples from biology and chemistry, DNA sequence classes can be represented as polyhedral sets, which are specified by linear inequalities [36]. Or, the covalent bonds of a molecule can be constrained such that their number and type agree with the atom's valence electrons [40].

In robotics, the joint positions of a robot arm can be described by linear equations [42].

> **Insert 1: Knowledge-Based Loss Term**
>
> When learning a function f^* from data (x_i, y_i) where the x_i are input features and the y_i are labels, a knowledge-based loss term L_k can be built into the objective function [9], [11]:
>
> $$f^* = \arg\min_f \Big(\overbrace{\lambda_l \sum_i L(f(x_i), y_i)}^{\texttt{Label-based}} + \overbrace{\lambda_r R(f)}^{\texttt{Regul.}} \qquad (1)$$
> $$+ \underbrace{\lambda_k L_k(f(x_i), x_i)}_{\texttt{Knowledge-based}} \Big)$$
>
> Whereas L is the usual label-based loss and R is a regularization function, L_k quantifies the violation of given prior-knowledge equations. Parameters λ_l, λ_r and λ_k determine the weight of the terms.
> Note that L_k only depends on the input features x_i and the learned function f and thus offers the possibility of label-free supervision [12].

In electrochemical micro-machining processes, relations between control parameters (e.g. voltage, pulse duration) and intermediate observables (e.g. current density) are known to influence outcomes and can be expressed as linear equations derived from principles of physical chemistry [39].

Expert Knowledge. Examples for the representation of expert knowledge are also given in [37], [43]: Valid ranges of variables according to experts' intuition can be defined as approximation constraints [37]. Moreover, the experience of, say, an oncological surgeon that the number of metastatized lymph nodes increases with tumor size can be formalized by a linear inequality [43].

5.1.2 (Paths to) Knowledge Integration

We observe that a frequent way of integrating equation-based knowledge into machine learning is via the learning algorithm. Two other approaches we found are the integration into the final hypothesis or into the hypothesis set.

Learning Algorithm. Algebraic equations and inequations can be integrated into learning algorithms via additional loss terms [11], [12], [37] or, more generally, via constrained problem formulation [36], [41], [43].

The integration of algebraic equations as knowledge-based loss terms into the learning objective function is detailed in Insert 1. These knowledge-based terms measure potential inconsistencies w.r.t., say, physical laws [11], [12]. Such an extended loss is usually called physics-based or hybrid loss and fosters the learning from data as well as from prior knowledge. Beyond the measuring inconsistencies with exact formulas, inconsistencies with approximation ranges or general monotonicity constraints, too, can be quantified via rectified linear units [37].

As a further approach, support vector machines can incorporate knowledge by relaxing the optimization problem into a linear minimization problem to which constraints are added in form of linear inequalities [36]. Similarly, it is possible to relax the optimization problem behind certain kernel-based approximation methods to constrain the behavior of a regressor or classifier in a possibly nonlinear region of the input domain [41], [43].

Training Data. Another natural way of integrating algebraic equations into machine learning is to use them for training data generation. While there are many papers in this category, we want to highlight one that integrates prior knowledge as an independent, second source of information by constructing a specific feature vector that directly models physical properties and constraints [44].

Hypothesis Set. An alternative approach is the integration into the hypothesis set. In particular, algebraic equations can be translated into the architecture of neural networks [38], [39], [42], [45]. One idea is to sequence predefined operations leading to a functional decomposition [45]. More specifically, relations between input parameters of a function and intermediate observables can be encoded as linear connections between, say, the input and the first hidden layer in a network model [39]. Similarly, the architecture of a mean of multiple computations (MMC) network can be augmented with transformations reflecting physical constraints on certain input and output variables [42]. Knowledge can also be incorporated by calculating a proper orthogonal decomposition around a particular solution known to satisfy required conditions (e.g. inhomogenous boundary conditions) [38].

Final Hypothesis. Another integration path applies algebraic equations to the final hypothesis, mainly serving as a consistency check with given constraints from a knowledge domain. This can be implemented as an inconsistency measure that quantifies the deviation of the predicted results from given knowledge similar to the above knowledge-based loss terms. It can then be used as an additional performance metric for model comparison [11]. Such a physical consistency check can also comprise an entire diagnostics set of functions describing particular characteristics, e.g. from turbulence modelling [46].

5.1.3 Remarks

With respect to our taxonomy, the representation of knowledge in form of equations is related to the use of rules. While both can work as constraints, the key difference in this survey is that equations describe algebraic constraints, whereas rules describe logic or semantic constraints. Their mathematical representations overlap because predicates in first-order logic are functions and can thus be described as equations. However, logic rules very often consist of more than one predicate so that, in this survey, we describe logic rules and algebraic equations in two separate categories.

For the taxonomy presented here, we distinguish between algebraic equations and differential equations (see Section 5.4).

Learning algorithms, which incorporate knowledge in form of algebraic equations are all variants of regularized empirical risk minimization. Adding algebraic equations to a learning algorithm thus turns empirical risk minimization into a constrained optimization problem for which there exists various solution procedures [47]. Unsupervised learning methods like pattern decomposition using matrix factorization can be enriched by knowledge via constrained optimization [48].

5.2 Logic Rules

Figure 5 summarizes taxonomy paths that we observed in our literature survey that involve logic rules. The particular path for which we found the most references comes from world knowledge and goes to the hypothesis set.
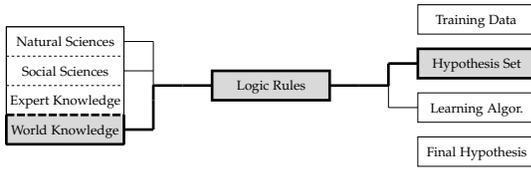

Figure 5: Taxonomy Paths Observed for Logic Rules.

5.2.1 (Path from) Knowledge Source

Logic rules can formalize knowledge from various sources. However, the most frequent ones are world knowledge and social sciences followed by the natural sciences.

World Knowledge. Logic rules often describe knowledge about real-world objects [9], [10], [12], [49], [50] such as seen in images. This can focus on object properties, such as for animals x that $(\text{FLY}(x) \wedge \text{LAYEGGS}(x) \Rightarrow \text{BIRD}(x))$ [9], or that an arrow in an image consists of three lines, two short ones and one longer one [49]. It can also focus on relations between objects such as the co-occurrence of characters in game scenes, e.g. $(\text{PEACH} \Rightarrow \text{MARIO})$ [12].

Natural Sciences. Knowledge from the natural sciences can also be represented by logic rules [22], [48]. For example, DNA sequences can be characterized by a set of rules (e.g., a polymerase can bind to a DNA site only when a specific nucleotide sequence is present [22]).

Social Sciences. Another knowledge domain that can be well represented by logic rules are social sciences and particularly linguistics [51], [52], [53], [54], [55], [56], [57]. Linguistic rules can consider the sentiment of a sentence (e.g., if a sentence consists of two sub-clauses connected with a 'but', then the sentiment of the clause after the 'but' dominates [51]); or the order of tags in a given word sequence (e.g., if a given text element is a citation, then it can only start with an author or editor field [53]). In social context, rules can describe dependencies in networks. For example, people tend to trust other people when they share trusted contacts $(\text{Trusts}(A, B) \wedge \text{Trusts}(B, C) \Rightarrow \text{Trusts}(A, C))$ [58], or, on a scientific research platform, it can be observed that authors citing each other tend to work in the same field $(\text{Cite}(x, y) \wedge \text{hasFieldA}(x) \Rightarrow \text{hasFieldA}(y))$ [20].

5.2.2 (Path to) Knowledge Integration

We observe that logic rules are integrated into learning mainly in the hypothesis set or, alternatively, in the learning algorithm.

Hypothesis Set. Integration into the hypothesis set comprises both deterministic and probabilistic approaches. The former include neural-symbolic systems, which use rules as the basis for the model structure [22], [59], [60], [61]. In Knowledge-Based Artificial Neural Networks (KBANNs), the architecture is constructed from symbolic rules by mapping the components of propositional rules to network components [22] as further explained in Insert 2. Extensions

---

Insert 2: Knowledge-Based Artificial Neural Networks (KBANNs) [22]

Rules can be integrated into neural architectures by mapping the rule's components to the neurons and weights with these steps:

1) Get rules. If needed, rewrite them to have a hierarchical structure.
2) Map rules to a network architecture. Construct (positively/negatively) weighted links for (existing/negated) dependencies.
3) Add nodes. These are not given through the initial rule set and represent hidden units.
4) Perturb the complete set of weights.

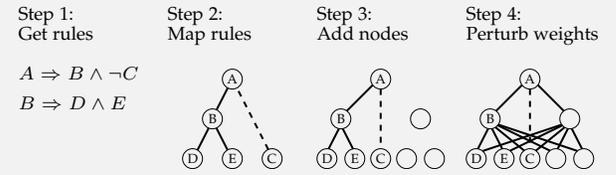After the KBANN's architecture is built, the network is refined with learning algorithms.

Figure 6: Steps of Rules-to-Network Translation [22]. Simple example for integrating rules into a KBANN.

---

are the Connectionist Inductive Learning and Logic programming system (CILP, or precisely, C-IL^{2}P), which also outputs a revised rule set [59] and CILP++, which extends the integration of propositional logic to first-order logic [60]. A recent survey about neural-symbolic computing [61] summarizes further methods.

Parallel network architectures can also integrates logic rules into the hypothesis set. For example, it is possible to train both a teacher network based on rules and a student network that learns to imitate the teacher [51], [52].

Integrating logic rules into the hypothesis set in a probabilistic manner is yet another approach [49], [50], [55], [56], [57], [58]. Here, a major research direction is Statistical Relational Learning or Star AI [62]. Corresponding frameworks provide a logic templating language to define a probability distribution over a set of random variables. Two prominent frameworks are Markov Logic Networks [56] and Probabilistic Soft Logic [55], which translate a set of first-order logic rules to a Markov Random Field. Each rule specifies dependencies between random variables and serves as a template for so called potential functions, which assign probability mass to joint variable configurations. Both frameworks are widely used to incorporate knowledge in form of logic rules [49], [50], [57].

Learning Algorithm. The integration of logic rules into the learning algorithm is often accomplished via additional, semantic loss terms [9], [10], [12], [51], [52]. These augment the objective function similar to the knowledge-based loss terms explained above. However, for logic rules, the additional loss terms evaluate a functional that transforms rules

into continuous and differentiable constraints, for example via the t-norm [9]. Semantic loss functions can also be derived from first principles using a set of axioms [10].

Probabilistic integration of logic rules into learning algorithms is feasible, too [50], [51], [52], [53], [54]. Probabilistic models can be represented as a scoring function, which assigns a value to each input/output combination. Knowledge can be integrated into this scoring function as an additional penalty term that considers logic constraints [53], [54]. Also, expert assignments in a mixture of Gaussian process model can be informed; here, the priors are obtained from a Markov Logic network integrating the logical rules [50].

5.2.3 Remarks

With respect to our taxonomy, logic rules are related to other knowledge representation categories. A rule set can be automatically transformed from and to a knowledge graph.

Logic rules have a long tradition in symbolic AI, in particular in expert systems based on rules [63], [64]. Furthermore, the combination of knowledge-based symbolic systems with data-driven connectionist approaches has long been considered in hybrid systems [65], [66] and corresponds to one main path through our taxonomy, namely the path that represents knowledge in logic rules and integrates it into the architecture, i.e. the hypothesis set.

5.3 Simulation Results

Figure 7 summarizes commonly observed taxonomy paths that involve the use of simulation results. The most common path we found in our literature survey comes from natural sciences and goes to the training data.
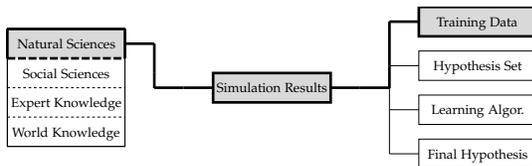

Figure 7: Taxonomy Paths Observed for Sim. Results.

5.3.1 (Paths from) Knowledge Source

Computer simulations have a long tradition in many areas of engineering and natural sciences. While they are also gaining popularity in domains like social sciences, most works on integrating simulation results into machine learning deal with natural sciences and engineering.

Natural Sciences. Examples of simulation results informing machine learning can be found in fluid- and thermodynamics [11], material sciences [18], [67], [68], [69], life sciences [70], [71], mechanics and robotics [16], [72], [73], [74], [75], or autonomous driving [17].

Based on fluid- and thermodynamical models, one can simulate water temperature in a lake [11]. In material sciences, a density functional theory ab-initio simulation can be used to model the energy and stability of potential new material compounds and their crystal structure [67]. Even complex material forming processes can be simulated, for example a composite textile draping process can be simulated based on a finite-element model [18] or metal

---

Insert 3: Simul. Results as Synthetic Training Data

The results from a simulation can be used as synthetic training data and can thus augment the original, real training data. Some papers that follow this approach are [11], [16], [17], [18], [71], [72], [73], [76].
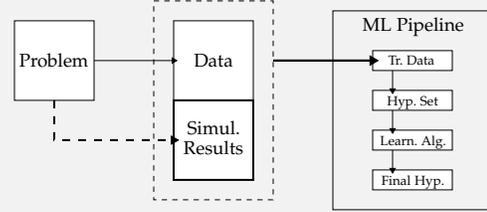

Figure 8: Information flow for synthetic training data from simulations.

---

heating strategies can be explored based on simulations of a warm forming process [69]. In the life sciences, simulations are used to predict the effect of new drugs, for example, on cancer with the radiation cell model (a set of ODEs), or a Boolean cancer model (a discrete dynamical system of cellular states) [71]. For mechanics and robotics there are plenty of examples. For instance, the walking traits of bipedal and hexapod robots can be optimized based on locomotion controller simulation [16], [73], or the stability of stacked objects can be simulated with physics engines [72], [74]. As an example for autonomous driving, urban scenes under different weather and illumination conditions can be simulated [17].

5.3.2 (Paths to) Knowledge Integration

We find that the integration of simulation results into machine learning is most often happens via the augmentation of training data. Another approach that also occurs frequently is the integration into the hypothesis set. Moreover, the integration into the probabilistic learning algorithm or the final hypothesis occurs as well.

Training Data. The integration of simulation results into training data [11], [16], [17], [18], [71], [72], [73], [76] depends on how the simulated, i.e. synthetic, data is combined with the real-world measurements:

Firstly, additional features are simulated and, together with real data, form input features. For example, original features can be transformed by multiple approximate simulations and the similarity of the simulation results can be used to build a kernel [71].

Secondly, target variables are simulated and added to the real data as another feature. This way the model does not necessarily learn to predict targets, e.g. an underlying physical process, but rather the systematic discrepancy between simulated and the true target data [11].

Thirdly, the target variable is simulated and used as a synthetic label. This is often realized with physics engines, for example, neural networks pre-trained on standard data sets such as ImageNet [77] can be tailored towards an application through additional training on simulated data [72]. Synthetic training data generated from simulations can also

be used to pre-train components of Bayesian optimization frameworks [16], [73].

In informed machine learning, training data thus stems from a hybrid information source and contain both simulated and real data points (see Insert 3). The gap between the synthetic and the real domain can be narrowed via adversarial networks such as SimGAN. These improve the realism of, say, synthetic images and can generate large annotated data sets by simulation [76]. The SPIGAN framework goes one step further and uses additional, privileged information from internal data structures of the simulation in order to foster unsupervised domain adaption of deep networks [17].

Hypothesis Set. Another approach we observed integrates simulation results into the hypothesis set [68], [69], [78], [79], which is of particular interest when dealing with low-fidelity simulations. These are simplified simulations that approximate the overall behaviour of a system but ignore intricate details for the sake of computing speed.

When building a machine learning model that reflects the actual, detailed behaviour of a system, low-fidelity simulation results or a response surface (a data-driven model of the simulation results) can be build into the architecture of a knowledge-based neural network (KBANN [22], see Insert 2), e.g. by replacing one or more neurons. This way, parts of the network can be used to learn a mapping from low-fidelity simulation results to a few real-world observations [69], [79]. If target data is not available, results could also come from accurate high-fidelity simulations. Also, parts of the network can be used to learn region-specific parameter choices for the low-fidelity simulation [78].

Simulation results can also be embedded in the hypothesis set of architectures resembling the encoder-decoder framework [68]. Here, The "latent" variables are physical quantities that are difficult or impossible to observe directly so that finite element simulations are used that mimic an encoder or decoder and are combined by training a model.

Learning Algorithm. We also observed the integration of simulation results in generative adversarial network (GAN). The goal is to learn predictions that obey constraints implicit to the simulation [75]. This is achieved by minimizing the distance between the distribution of predictions and the distribution of simulated data points.

Learning Algorithm Furthermore, a simulation can directly be integrated into iterations of a a learning algorithm. For example, a realistic positioning of objects in a 3D scene can be improved by incorporating feedback from a solid-body simulation into learning [74]. By means of reinforcement learning, this is even feasible if there are no gradients available from the simulation. Alternatively, simulations can be used in forward propagation to obtain predictions according to, say, physical laws [70].

Final Hypothesis. A last but important approach that we found in our survey integrates simulation results into the final hypothesis set of a machine learning model. Specifically, simulations can validate results of a trained model [18], [67].

5.3.3 Remarks

With respect to our taxonomy, knowledge representation in the form of simulation results overlaps the representation in the form of algebraic equations and differential equations (see Section 5.1 and Section 5.4). However,w.r.t. simulation results, we do not consider the underlying equations themselves, but knowledge that is derived from the simulation. Simulations are generally very complex software products requiring elaborate knowledge integration (from the underlying simulation model, over numerical solvers to tracing engines, discrete simulation and multi-agent systems). In this section, we focus on knowledge derived from the simulation results where such details need not be known when setting up a learning pipeline.

We also note that we do not consider approaches, which learn purely data-driven models from simulated data, cf. e.g. [80], [81] to be informed machine learning. A requirement for informed machine learning is the use of a hybrid information source of data and separate prior knowledge. Combining learning from both real world and simulated data is explicitly explained in [76].

5.4 Differential Equations

Next,we describe informed machine learning approaches based on knowledge in form of differential equations. Figure 9 summarizes the paths observed in our literature survey. The most common path we found proceeds from natural sciences to the hypothesis set or the learning algorithm.
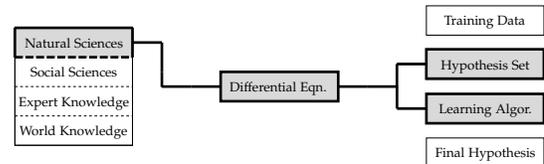

Figure 9: Taxonomy Paths Observed for Differential Eqn.

5.4.1 (Paths from) Knowledge Source

Differential equations model the behavior of dynamical systems by relating state variables to their rate of change. In the literature discussed here, differential equations exclusively represent knowledge from the natural sciences.

Natural Sciences. The work in [19], [82] considers the Burger's equation, which is used in fluid dynamics to model simple one-dimensional currents and in traffic engineering to describe traffic density behavior. Advection-diffusion equations [83] are used in oceanography to model the evolution of sea surface temperatures. The Schrödinger equation studied in [19] describes quantum mechanical phenomena such as wave propagation in optical fibres or the behavior of Bose-Einstein condensates. The Euler-Lagrange equation considered in [84] is the basis of variational approaches in physics and engineering. For instance, it is used to calculate the Brachistochrone curve of a frictionless particle. The work in [85] studies stirred bioreactors, a model for substrate to biomass conversion based on coupled ODEs. In [86], the considered ODE provides a model for the dynamics of gene expression regulation.

5.4.2 (Paths to) Knowledge Integration

Regarding the integration of differential equations, our survey particularly focuses on the integration into neural network models.

Learning Algorithm. A neural network can be trained to approximate the solution of a differential equation. To this end, the governing differential equation is integrated into the loss function similar to Eq. 1 [87]. This requires evaluating derivatives of the network with respect to its inputs, for example, via automatic differentiation, an approach that was recently adapted to deep learning [19], [88]. This ensures the physical plausibility of the neural network output. An extension to generative models is possible, too [82], [89]. Finally, probabilistic models can also be trained by minimizing the distance between the model conditional density and the Boltzmann distribution dictated by a differential equation and boundary conditions [90].

Hypothesis Set. In many applications, differential equations contain unknown time- and space-dependent parameters. Neural networks can model the behavior of such parameters, which then leads to hybrid architectures where the functional form of certain components is analytically derived from (partially) solving differential equations [83], [84], [85]. In other applications, one faces the problem of unknown mappings from input data to quantities whose dynamics are governed by known differential equations, usually called system states. Here, neural networks can learn a mapping from observed data to system states [86], [91]. This also leads to hybrid architectures with knowledge-based modules, e.g. in form of a physics engine.

5.4.3 Remarks

The informed machine learning approaches in this section address long-standing research problems. Approximate solutions to differential equations are studied extensively in numerical mathematics [92] and engineers know the problem of identifying unknown parameters in differential equations as *system identification* [93]. The work in [83] directly adds to the huge body of research on *data assimilation*, which, in geology, refers to methods that allow for estimating system states from indirect observations. From a machine learning perspective, standard data assimilation algorithms such as the Kalman filter, 3DVAR, 4DVAR, and ensemble Kalman filters can all be cast in a Bayesian inference framework [25]. In other words, data assimilation problems are modelled using dynamic Bayesian networks [94] with continuous physically interpretable state spaces where transition kernels and observation operators reflect knowledge about the system dynamics, e.g. the differential equations of atmospheric physics. Some data assimilation algorithms are highly optimized and in daily operational use for weather forecasting. Data assimilation can thus be considered a well-established variant of informed machine learning where knowledge is integrated via differential equations.

5.5 Knowledge Graphs

Figure 10 summarizes the taxonomy paths observed in our literature survey that are related to knowledge representation in form of knowledge graphs. The most common path we found starts with world knowledge and ends in the hypothesis set.

5.5.1 (Paths from) Knowledge Source

Since graphs are very versatile modeling tools, they can represent various kinds of structured knowledge. Typically,
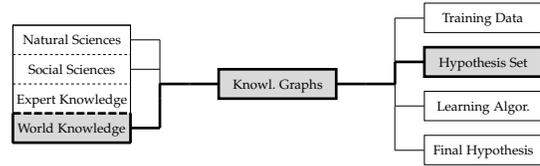

Figure 10: Taxonomy Paths Observed for Knowl. Graphs.

they are constructed from databases, however, the most frequent source we found in informed machine learning papers is world knowledge.

World Knowledge. Since humans perceive the world as composed of entities, graphs are often used to represent relations between entities. For example, the Visual Genome knowledge graph is build from human annotations of object attributes and relations between objects in natural images [14], [15]. Similarly, the MIT ConceptNet [95] encompasses concepts of everyday life and their relations automatically built from text data. Other influential exmaples of knowledge graphs containing general world knowledge are Freebase [96] and DBPedia [97].

Natural Sciences. In physics, graphs can immediately describe physical systems such as spring-coupled masses [13]. In medicine, networks of gene-protein interactions describe biological pathway information [98]. Moreover, medical diagnoses have a hierarchical nature that is captured by classification systems such as the International Classification of Diseases (ICD) [99], [100]. Knowledge incorporated into these systems is a mix of natural science, e.g. the etiology of a disease, and social considerations, e.g. w.r.t. treatment and impact of a disease on daily life. Another knowledge graph is Medical Subject Headings (MeSH) [101], which summarizes knowledge about medical terms.

Social Sciences. In natural language processing, knowledge graphs often represent knowledge about relations among concepts, which can be referred to by words. For example, WordNet [102] represents semantic and lexical relations of words such as synonymy. Such knowledge graphs are often used for information extraction in natural language processing but information extraction can also be used to build new knowledge graphs [103].

5.5.2 (Paths to) Knowledge Integration

In our survey, we observed the integration of knowledge graphs in all four components of the machine learning pipeline but most prominently in the hypothesis set.

Hypothesis Set. The fact that the world consists of interrelated objects can be integrated by altering the hypothesis set. Graph neural networks operate on graphs and thus feature an object- and relation-centric bias in their architecture [27]. A recent survey [27] gives an overview over this field and explicitly names this knowledge integration "relational inductive bias". This bias is of benefit, e.g. for learning physical dynamics [13], [104] or object detection [15].

In addition, graph neural networks allow for the explicit integration of a given knowledge graph as a second source of information. This allows for multi-label classification in natural images where inference about a particular object is facilitated by using relations to other objects in an image [14] (see Insert 4). More generally, a graph reasoning layer can

Insert 4: Integrating Knowledge Graphs in CNNs for Image Classification

Image classification through convolutional neural networks can be improved by using knowledge graphs that reflect relations between detected objects. Technically, such relations form adjacency matrices in gated graph neural networks [14]. During the detection, the network graph is propagated, starting with detected nodes and then expanding to neighbors [27].
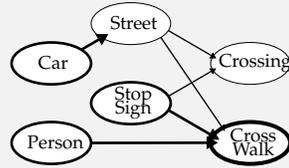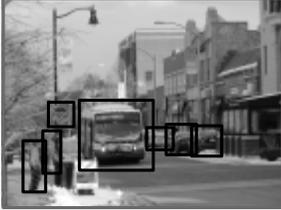

Figure 11: Illustrative application example of using neural networks and knowledge graphs for image classification, similar as in [14]. The image (from the COCO dataset) shows a pedestrian cross walk.

be inserted into any neural network [105]. The main idea is to enhance representations in a given layer by propagating through a given knowledge graph.

Another approach is to use attention mechanisms on a knowledge graph in order to enhance features. In natural language analysis, this facilitates the understanding as well as the generation of conversational text [106]. Similarly, graph-based attention mechanism are used to counteract too few data points by using more general categories [100]. Since feature enhancement requires trainable parameters, we see these approaches as refining the hypothesis set rather than informing the training data.

Training Data. Another direction we observed is the use of knowledge graphs in training data generation. Here, knowledge graphs serve as a second source of training data, which supervises a part of a neural network that learns a graph representation of an image [15]. Another prominent approach is distant supervision where information in a graph is used to automatically annotate texts to train natural language processing systems. This was originally done naïvely by considering each sentence that matches related entities in a graph as a training sample [107]; however, recently attention-based networks have been used to reduce the influence of noisy training samples [108].

Learning Algorithm. Various works discuss the integration of graph knowledge into the learning algorithm. For instance, a regularization term based on the graph Laplacian matrix can enforce strongly connected variables to behave similarly in the model, while unconnected variables are free to contribute differently. This is commonly used in bioinformatics to integrate genetic pathway information [98], [99]. Some natural language models, too, include information from a knowledge graph into the learning algorithm, e.g. when computing word embeddings. Known relations among words can be utilized as augmented contexts [109]

in word2vec training [110]. Also, attention on related knowledge graph embedding can support the training of word embeddings [111], which are fed into language models [112].

Final Hypothesis. Finally, graph can also be used to improve final hypotheses or trained models. For instance, a recent development is to post-process word embeddings based on information from knowledge graphs [113], [114]. Furthermore, in object detection, predicted probabilities of a learning system can be refined using semantic consistency measures [115] derived form knowledge graphs, which indicates whether the co-occurrence of two objects is consistent with available knowledge.

5.5.3   Remarks

Again, there are connections to other knowledge representations. For example, logic rules can be obtained from a knowledge graph and probabilistic graphical models represent statistical relations between variables.

Furthermore, reasoning with knowledge graphs is an established research field with particular use cases in link prediction or node classification. Here, machine learning allows for searching within a knowledge graph. However, here, we focused on the opposite direction, i.e. the use of knowledge graphs in the machine learning pipeline.

5.6   Probabilistic Relations

Figure 12 summarizes taxonomy paths involving the representation type of probabilistic relations. The most frequent pathway found in our literature survey comes from expert knowledge and goes to the learning algorithm.
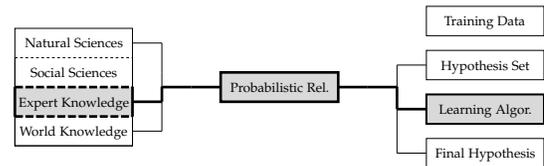

Figure 12: Taxonomy Paths Observed for Probabilistic Rel.

5.6.1   (Paths from) Knowledge Source

Knowledge in form of probabilistic relations originates most prominently from domain experts, but can also come from other sources such as natural sciences.

Expert Knowledge. A human expert has intuitive knowledge over a domain, for example, which entities are related to each other and which are independent. Such relational knowledge, however, is often not quantified and validated and differs from, say, knowledge in natural sciences. Rather, it involves degrees of belief or uncertainty.

Human expertise exists in all domains. Examples include medical expertise [116] where a prominent example is the ALARM network [117], an alarm message system for patient monitoring relating diagnoses to measurements and hidden variables [118], [119]. Further examples are computer expertise for troubleshooting, i.e relating a device status to observations [56]. In the car insurance domain, driver features like age relate to risk aversion [120].

Natural Sciences. Correlation structures can also be obtained from natural sciences knowledge. For example, correlations between genes can be obtained from gene interaction networks [121] or from a gene ontology [122].

5.6.2 (Paths to) Knowledge Integration

We generally observe the integration probabilistic relations into the hypothesis set as well as into the learning algorithm and the final hypothesis.

Hypothesis Set. Expert knowledge is the basis for probabilistic graphical models. For example, Bayesian network structures are typically designed by human experts and thus fall into the category of informing the hypothesis set.

Here, we focus on contributions where knowledge and Bayesian inference are combined in more intricate ways, for instance, by learning network structures from knowledge and from data. A recent overview [123] categorizes the type of prior knowledge about network structures into the presence or absence of edges, edge probabilities, and knowledge about node orders.

Probabilistic knowledge can be used directly in the hypothesis set. For example, the structure of a probabilistic model can be chosen in accordance to given spatio-temporal structures [124], or an extra node can be added to a Bayesian network thus altering the hypothesis set [125]. In other hybrid approaches, the parameters of the conditional distribution of the Bayesian network are either learned from data or obtained from world knowledge [126].

Learning Algorithm. Human knowledge can also be used to define an informative prior [119], which affects the learning algorithm as is has a regularizing effect. This has been extended to obtaining the prior from multiple experts [127]. Furthermore, structural constraints can alter score functions or the selection policies of conditional independence test, informing the search for the network structure [120]. More qualitative knowledge, e.g. observing one variable increases the probability of another, was integrated using isotonic regression, i.e. parameter estimation with order constraints [116]. Causal network inference can make use of ontologies to select the tested interventions [122]. Furthermore, prior causal knowledge can be used to constrain the direction of links in a Bayesian network [128] and, in active learning, the algorithm may directly asks the human for feedback concerning particular links [118]

Final Hypothesis. Finally, predictions obtained from a Bayesian network can be judged by probabilistic relational knowledge in order to refine the model [129].

5.6.3 Remarks

Probabilistic relations can always be formalized in form of a graph where nodes are random variables and edges reflect probability distributions. Hence, we distinguish this probabilistic concept from the ordinary deterministic graphs considered above.

Stochastic Differential Equations [130], too, can describe probabilistic relation between random variables in a time series. If the structure of the equation is known, say, from physical insight into a system, specialized parameter estimation techniques are available to fit a model to given observations [131].

5.7 Invariances

Next, we describe informed machine learning approaches involving the representation type of invariances. Figure 13 summarizes taxonomy paths we observed in our literature survey; the path for which we found most references begins at natural sciences or world knowledge and ends at the hypothesis set.
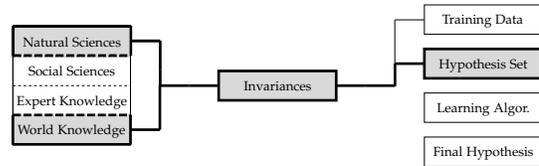

Figure 13: Taxonomy Paths Observed for Invariances.

5.7.1 (Paths from) Knowledge Source

We mainly found references using invariances in the context of world knowledge or natural sciences.

World Knowledge. Knowledge about invariances may fall into the category of world knowledge, for example when modeling facts about local / global pixel correlations in images [132]. Indeed, invariants are often used in image recognition where many characteristics are invariant under metric-preserving transformations. For example, in object recognition, an object should be classified correctly independent of its rotation in an image.

Natural Sciences. In physics, Noether's theorem states that certain symmetries (invariants) lead to conserved quantities (first integrals) and thus integrate Hamiltonian systems or equations of motion [133], [134]. For example, in equations modeling planetary motion, the angular momentum serves as such an invariant.

5.7.2 (Paths to) Knowledge Integration

In most references we found, invariances inform the hypothesis set.

Hypothesis Set. Invariances from physical laws can be integrated into the architecture of a neural network. For example, invariant tensor bases can be used to embed Galilean invariance for the prediction of fluid anisotropy tensors [134], or the physical Minkowski metric that reflects mass invariance can be integrated via a Lorentz layer into a neural network [135].

A recent trend is to integrate knowledge as to invariances into the architecture or layout of convolutional neural networks, which leads to so called geometric deep learning in [136].

A natural generalization of CNNs are group equivariant CNNs (G-CNNs) [137], [138], [139]. G-convolutions provide a higher degree of weight sharing and expressiveness. Simply put, the idea is to define filters based on a more general group-theoretic convolution. Another approach towards rotation invariance in image recognition considers harmonic network architecture where a certain response entanglement (arising from features that rotate at different frequencies) is resolved [140]. The goal is to design CNNs that exhibits equivariance to patch-wise translation and rotation by replacing conventional CNN filters with circular harmonics.

In support vector machines, invariances under group transformations and prior knowledge about locality can be incorporated by the construction of appropriate kernel functions [132]. In this context, local invariance is defined in terms of a regularizer that penalizes the norm of the derivative of the decision function [23].

Training Data. An early example of integrating knowledge as to invariances into machine learning is the creation of virtual examples [141] and it has been shown that data augmentation through virtual examples is mathematically equivalent to incorporating prior knowledge via a regularizer. A similar approach is the creation of meta-features [142]. For instance, in turbulence modelling using the Reynolds stress tensor, a feature can be createad that is rotational, reflectional and Galilean invariant [133]. This is achieved by selecting features fulfilling rotational and Gallilean symmetries and augmenting the training data to ensure reflectional invariance.

5.7.3 Remarks

In general, leveraging of invariants is a venerable scientific technique in wide range of fields. The guiding intuition is that symmetries and invariants diminish (or decrease the dimension of) the space one is supposed to investigate. This allows for an easier navigation through parameter- or model spaces. Geometric properties, invariants and symmetries of the data are the central topic of fields such as geometric learning [136] and topological data analysis [143]. Understanding the shape, curvature and topology of a data set can lead to efficient dimensionality reduction and more suitable representation/embedding constructions.

5.8 Human Feedback

Finally, we look at informed machine learning approaches belonging to the representation type of human feedback. Figure 14 summarizes taxonomy paths identified in our literature survey; the most common path begins with expert knowledge and ends at the learning algorithm.
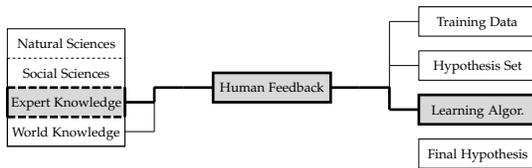

Figure 14: Taxonomy Paths Observed for Hum. Feedback.

5.8.1 (Paths from) Knowledge Source

Compared to other categories in our taxonomy, knowledge representation via human feedback is less formalized; it taps into world knowledge and expert knowledge.

Expert Knowledge. Examples of knowledge that fall into this category include knowledge about topics in text documents [149], agent behaviors [144], [153], [154], [155], [156], and data patterns and hierarchies [148], [149], [151], [152], [157]. Knowledge is often provided in form of relevance or preference feedback and humans in the loop are not required to provide an explanation for their decision.

World Knowledge. World knowledge that is integrated into machine learning via human feedback often is knowledge about objects and object components. This allows for semantic segmentation [147] requiring knowledge about object boundaries, or for object recognition [151] involving knowledge about object categories.

5.8.2 (Paths to) Knowledge Integration

Human feedback for machine learning is usually assumed to be limited to feature engineering and data annotation. However, it can also be integrated into the learning process. This often occurs in three areas of machine learning, namely active learning, reinforcement learning, and visual analytics. In all three cases, human feedback is primarily incorporated into the learning algorithm and, in some cases, also into training data and hypothesis set.

Learning Algorithm. Active learning offers a way to include the "human in the loop" to efficiently learn with minimal human intervention. This is based on iterative strategies where a learning algorithm queries an annotator for labels [158], for example, to indicate if certain data points are relevant or not [150], [159]. We do not consider this standard active learning as an informed learning method because the human knowledge is essentially used for label generation only. However, recent efforts integrate further knowledge into the active learning process. For instance, feedback helps to select instances for labeling according to meaningful patterns in a visualization [151]. This allows the user to actively steer the learning algorithm towards relevant data points.

In reinforcement learning, an agent observes an unknown environment and learns to act based on reward signals. The TAMER framework [153] provides the agent with human feedback rather than (predefined) rewards. This way, the agent learns from observations and human knowledge alike. High-dimensional state spaces are addressed in Deep TAMER [154]. While these approaches can quickly learn optimal policies, it is cumbersome to obtain the human feedback for every action. Human preference w.r.t. whole action sequences, i.e. agent behaviors, can circumvent this [155]. This enables the learning of reward functions, which can then be used to learn policies [160]. Expert knowledge can also be incorporated through natural language interfaces [156]. Here, a human provides instructions and agents receive rewards upon completing these instructions.

Visual analytics combines analysis techniques and interactive visual interfaces to enable exploration of- and inference from data [161]. Machine learning is increasingly combined with visual analytics. For example, visual analytics systems allow users to set control points to shape, say, kernel PCA embeddings [152] or to drag similar data points closer in order to learn distance functions [148]. In object recognition, users can provide corrective feedback via brush strokes [147] and for classifying plants as healthy or diseased, correctly identified instances where the interpretation is not in line with human explanations can be altered by a human expert [157].

Lastly, various tools exist for text analysis, in particular for topic modeling [149] where users can create, merge and refine topics or change keyword weights. They thus impart

	Training Data	Hypothesis Set	Learning Algorithm	Final Hypothesis
Algebraic Equations	[44]	[38], [39], [42], [45]	[11], [12], [36], [37], [41], [43]	[11], [46]
Logic Rules		[22], [49], [50], [51], [52], [55], [56], [57], [58], [59], [60], [61]	[9], [10], [12], [50], [51], [52], [53], [54]	
Simulation Results	[11], [16], [17], [18], [71], [72], [73], [76]	[68], [69], [78], [79]	[70], [74], [75]	[18], [67]
Differential Equations		[83], [84], [85], [86], [91]	[19], [82], [87], [88], [89], [90]	
Knowledge Graphs	[15], [107], [108]	[13], [14], [15], [100], [104], [105], [106]	[98], [99], [109], [110], [111], [112]	[113], [114], [115]
Probabilistic Relations		[124], [125], [126]	[116], [118], [119], [120], [122], [127], [128]	[129]
Invariances		[132], [134], [135], [137], [138], [139], [140]	[133], [141], [142]	
Human Feedback	[144], [145], [146]	[144], [146]	[147], [148], [149], [150], [151], [152], [153], [154], [155], [156], [157]	

Table 2: **Classification of Research Papers According to the Informed Machine Learning Taxonomy.** This table lists research papers according to the two taxonomy dimensions of knowledge representation and knowledge integration. Cells with a high number of citations mark prominent research paths.

knowledge by generating new reference matrices (term-by-topic and topic-by-document matrices) that are integrated in a regularization term that penalizes the difference between the new and the old reference matrices. This is similar to the semantic loss term described above.

Training Data and Hypothesis Set. Another approach towards incorporating expert knowledge in reinforcement learning considers human demonstration of problem solving. Expert demonstrations can be used to pre-train a deep Q-network, which accelerates learning [144]. Here, prior knowledge is integrated into the hypothesis set and the training data since the demonstrations inform the training of the Q-network and, at the same time, allow for interactive learning via simulations.

A basic component of visual analytics is to inform training via human feature engineering. For example, a medical expert selects features that are especially relevant for a diagnosis [125] and the BEAMES system enables multi-model steering via weight sliders [146]. Also, systems can recommend model improvements and the user chooses those, which they deem reasonable and refine the model accordingly [145].

5.8.3 Remarks

In practice, reinforcement learning and standard active learning rely on simulations. This means that a labeled data set is available and user feedback is simulated by the machine providing the correct label or action. In active learning this is also called retrospective learning.

Furthermore, human feedback is tightly coupled with the design of the user interface because results have to be visualized in an accessible manner. Hence, techniques such as dimensionality reduction have to be taken into account and usability depends on the performance and response times of these methods. In order to obtain rich feedback, careful design of visual analytics guided machine learning is essential [145], [162], [163] leading to new principles for the interface design of interactive learning systems [164].

Here, too, human knowledge can be integrated in various forms, for instance, using rules.

6 CONCLUSION

In this paper, we presented a structured and systematic overview of methods for the explicit integration of additional prior knowledge into machine learning, which we described using the umbrella term of *informed machine learning*. Our contribution consisted of three parts, namely a conceptual clarification, a taxonomy, and a comprehensive research survey.

We first proposed a concept for informed machine learning that illustrates its building blocks and delineates it from conventional machine learning. In particular, we defined informed machine learning as learning from a hybrid information source that includes data as well as prior knowledge where the latter is exists separate from the data and is additionally, explicitly integrated into the learning pipeline. Although some form of knowledge is always used when designing learning pipelines, especially during feature engineering, we deliberately differentiated informed machine learning as this offers the opportunity to integrate knowledge via potentially automated interfaces.

Second, we introduced a taxonomy that serves as a classification framework for informed machine learning approaches. It organizes approaches found in the literature according to three dimensions, namely the knowledge source, the knowledge representation, and the manner of knowledge integration. For each dimension, we identified sets of most relevant types based on an extensive literature survey. We put a special focus on the typification of knowledge representations, as these form the methodical basis for the integration into the learning pipeline. In summary, we found eight knowledge representation types, ranging from algebraic equations and logic rules, over simulation results, to knowledge graphs that can be integrated in any of the four steps of the machine learning pipeline, from the training

data, over the hypothesis set and the learning algorithm, to the final hypothesis. Categorizing research papers w.r.t. the corresponding entries in the taxonomy illustrates common methodical approaches in informed machine learning.

Third, we presented a comprehensive research survey and described different approaches for integrating prior knowledge into machine learning structured according to our taxonomy. We organized our presentation w.r.t. knowledge representations. For each type, we presented the observed paths from knowledge sources, as well as the observed paths to knowledge integration. We found that specific paths occur frequently. Prominent approaches are, for example, to represent prior knowledge from natural sciences in form of algebraic equations, which are then integrated in the learning algorithm via knowledge-based loss terms, or to represent world knowledge in form of logic rules or knowledge graphs, which are then integrated directly in the hypothesis set.

All in all, we clarified the idea of informed machine learning and presented an overview of methods for the integration of prior knowledge into machine learning. This may help current and future users of informed machine learning to identify the right methods for using their prior knowledge in order to deal with insufficient training data or to make their models even more robust.

ACKNOWLEDGMENTS

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Adv. in Neural Information Processing Systems (NIPS)*, 2012.

[2] G. Hinton, L. Deng, D. Yu, G. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, B. Kingsbury *et al.*, "Deep neural networks for acoustic modeling in speech recognition," *Signal Processing Magazine*, vol. 29, 2012.

[3] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional networks for text classification," *arxiv:1606.01781*, 2016.

[4] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, "Mastering the game of go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, 2016.

[5] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, and A. Walsh, "Machine learning for molecular and materials science," *Nature*, vol. 559, no. 7715, 2018.

[6] T. Ching, D. S. Himmelstein, B. K. Beaulieu-Jones, A. A. Kalinin, B. T. Do, G. P. Way, E. Ferrero, P.-M. Agapow, M. Zietz, M. M. Hoffman *et al.*, "Opportunities and obstacles for deep learning in biology and medicine," *J. Royal Society Interface*, vol. 15, no. 141, 2018.

[7] J. N. Kutz, "Deep learning in fluid dynamics," *J. Fluid Mechanics*, vol. 814, 2017.

[8] R. Roscher, B. Bohn, M. F. Duarte, and J. Garcke, "Explainable machine learning for scientific insights and discoveries," *arxiv:1905.08883*, 2019.

[9] M. Diligenti, S. Roychowdhury, and M. Gori, "Integrating prior knowledge into deep learning," in *Int. Conf. on Machine Learning and Applications (ICMLA)*.   IEEE, 2017.

[10] J. Xu, Z. Zhang, T. Friedman, Y. Liang, and G. V. d. Broeck, "A semantic loss function for deep learning with symbolic knowledge," *arxiv:1711.11157*, 2017.

[11] A. Karpatne, W. Watkins, J. Read, and V. Kumar, "Physics-guided neural networks (pgnn): An application in lake temperature modeling," *arxiv:1710.11431*, 2017.

[12] R. Stewart and S. Ermon, "Label-free supervision of neural networks with physics and domain knowledge," in *Conf. Artificial Intelligence*.   AAAI, 2017.

[13] P. Battaglia, R. Pascanu, M. Lai, D. J. Rezende *et al.*, "Interaction networks for learning about objects, relations and physics," in *Adv. in Neural Information Processing Systems (NIPS)*, 2016.

[14] K. Marino, R. Salakhutdinov, and A. Gupta, "The more you know: Using knowledge graphs for image classification," in *Conf. Computer Vision and Pattern Recognition (CVPR)*.   IEEE, 2017.

[15] C. Jiang, H. Xu, X. Liang, and L. Lin, "Hybrid knowledge routed modules for large-scale object detection," in *Adv. in Neural Information Processing Systems (NIPS)*, 2018.

[16] A. Cully, J. Clune, D. Tarapore, and J.-B. Mouret, "Robots that can adapt like animals," *Nature*, vol. 521, no. 7553, 2015.

[17] K.-H. Lee, J. Li, A. Gaidon, and G. Ros, "Spigan: Privileged adversarial learning from simulation," in *Int. Conf. Learning Representations (ICLR)*, 2019.

[18] J. Pfrommer, C. Zimmerling, J. Liu, L. Kärger, F. Henning, and J. Beyerer, "Optimisation of manufacturing process parameters using deep neural networks as surrogate models," *Procedia CIRP*, vol. 72, no. 1, 2018.

[19] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations," *arxiv:1711.10561*, 2017.

[20] M. Diligenti, M. Gori, and C. Sacca, "Semantic-based regularization for learning and inference," *Artificial Intelligence*, vol. 244, 2017.

[21] M. L. Minsky, "Logical versus analogical or symbolic versus connectionist or neat versus scruffy," *AI magazine*, vol. 12, no. 2, 1991.

[22] G. G. Towell and J. W. Shavlik, "Knowledge-based artificial neural networks," *Artificial Intelligence*, vol. 70, no. 1-2, 1994.

[23] F. Lauer and G. Bloch, "Incorporating prior knowledge in support vector machines for classification: A review," *Neurocomputing*, vol. 71, no. 7-9, 2008.

[24] E. Kalnay, *Atmospheric modeling, data assimilation and predictability*.   Cambridge university press, 2003.

[25] S. Reich and C. Cotter, *Probabilistic forecasting and Bayesian data assimilation*.   Cambridge University Press, 2015.

[26] A. Karpatne, G. Atluri, J. H. Faghmous, M. Steinbach, A. Banerjee, A. Ganguly, S. Shekhar, N. Samatova, and V. Kumar, "Theory-guided data science: A new paradigm for scientific discovery from data," *Trans. Knowledge and Data Engineering*, vol. 29, no. 10, 2017.

[27] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner *et al.*, "Relational inductive biases, deep learning, and graph networks," *arxiv:1806.01261*, 2018.

[28] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin, *Learning from data*.   AMLBook, 2012, vol. 4.

[29] M. Steup, "Epistemologyp," in *The Stanford Encyclopedia of Philosophy (Winter 2018 Edition), Edward N. Zalta (ed.)*, 2018.

[30] L. Zagzebski, *What is Knowledge?*   John Wiley & Sons, 2017.

[31] P. Machamer and M. Silberstein, *The Blackwell guide to the philosophy of science*.   John Wiley & Sons, 2008, vol. 19.

[32] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI magazine*, vol. 17, no. 3, 1996.

[33] D. Kahneman, *Thinking, Fast and Slow*.   Macmillan, 2011.

[34] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman, "Building machines that learn and think like people," *Behavioral and brain sciences*, vol. 40, 2017.

[35] H. G. Gauch, *Scientific method in practice*.   Cambridge University Press, 2003.

[36] G. M. Fung, O. L. Mangasarian, and J. W. Shavlik, "Knowledge-based support vector machine classifiers," in *Adv. in Neural Information Processing Systems (NIPS)*, 2003.

[37] N. Muralidhar, M. R. Islam, M. Marwah, A. Karpatne, and N. Ramakrishnan, "Incorporating prior domain knowledge into deep neural networks," in *Int. Conf. Big Data*. IEEE, 2018.

[38] R. Swischuk, L. Mainini, B. Peherstorfer, and K. Willcox, "Projection-based model reduction: Formulations for physics-based machine learning," *Computers & Fluids*, vol. 179, 2019.

[39] Y. Lu, M. Rajora, P. Zou, and S. Liang, "Physics-embedded machine learning: Case study with electrochemical micro-machining," *Machines*, vol. 5, no. 1, 2017.

[40] T. Ma, J. Chen, and C. Xiao, "Constrained generation of semantically valid graphs via regularizing variational autoencoders," in *Adv. in Neural Information Processing Systems (NIPS)*, 2018.

[41] O. L. Mangasarian and E. W. Wild, "Nonlinear knowledge-based classification," *Trans. Neural Networks*, vol. 19, no. 10, 2008.

[42] R. Ramamurthy, C. Bauckhage, R. Sifa, J. Schücker, and S. Wrobel, "Leveraging domain knowledge for reinforcement learning using mmc architectures," in *Int. Conf. Artificial Neural Networks (ICANN)*. Springer, 2019.

[43] O. L. Mangasarian and E. W. Wild, "Nonlinear knowledge in kernel approximation," *Trans. Neural Networks*, vol. 18, no. 1, 2007.

[44] S. Jeong, B. Solenthaler, M. Pollefeys, M. Gross *et al.*, "Data-driven fluid simulations using regression forests," *ACM Trans. Graphics*, vol. 34, no. 6, 2015.

[45] C. Bauckhage, C. Ojeda, J. Schücker, R. Sifa, and S. Wrobel, "Informed machine learning through functional composition."

[46] R. King, O. Hennigh, A. Mohan, and M. Chertkov, "From deep to physics-informed learning of turbulence: Diagnostics," *arxiv:1810.07785*, 2018.

[47] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[48] S. Ermon, R. Le Bras, S. K. Suram, J. M. Gregoire, C. P. Gomes, B. Selman, and R. B. Van Dover, "Pattern decomposition with complex combinatorial constraints: Application to materials discovery," in *Conf. Artificial Intelligence*. AAAI, 2015.

[49] M. Sachan, K. A. Dubey, T. M. Mitchell, D. Roth, and E. P. Xing, "Learning pipelines with limited data and domain knowledge: A study in parsing physics problems," in *Adv. in Neural Information Processing Systems (NIPS)*, 2018.

[50] M. Schiegg, M. Neumann, and K. Kersting, "Markov logic mixtures of gaussian processes: Towards machines reading regression data," in *Artificial Intelligence and Statistics*, 2012.

[51] Z. Hu, X. Ma, Z. Liu, E. Hovy, and E. Xing, "Harnessing deep neural networks with logic rules," *arxiv:1603.06318*, 2016.

[52] Z. Hu, Z. Yang, R. Salakhutdinov, and E. Xing, "Deep neural networks with massive learned knowledge," in *Proc. Conf. Empirical Methods in Natural Language Processing*, 2016.

[53] M.-W. Chang, L. Ratinov, and D. Roth, "Guiding semi-supervision with constraint-driven learning," in *Proc. of the 45th Meeting of the Association of Computational Linguistics*, 2007.

[54] ——, "Structured learning with constrained conditional models," *Machine Learning*, vol. 88, no. 3, 2012.

[55] A. Kimmig, S. Bach, M. Broecheler, B. Huang, and L. Getoor, "A short introduction to probabilistic soft logic," in *Proc. of the NIPS Workshop on Probabilistic Programming: Foundations and Applications*, 2012.

[56] M. Richardson and P. Domingos, "Markov logic networks," *Machine Learning*, vol. 62, no. 1-2, 2006.

[57] D. Sridhar, J. Foulds, B. Huang, L. Getoor, and M. Walker, "Joint models of disagreement and stance in online debate," in *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, 2015.

[58] S. H. Bach, M. Broecheler, B. Huang, and L. Getoor, "Hinge-loss markov random fields and probabilistic soft logic," *arxiv:1505.04406*, 2015.

[59] A. S. A. Garcez and G. Zaverucha, "The connectionist inductive learning and logic programming system," *Applied Intelligence*, vol. 11, no. 1, 1999.

[60] M. V. França, G. Zaverucha, and A. S. d. Garcez, "Fast relational learning using bottom clause propositionalization with artificial neural networks," *Machine Learning*, vol. 94, no. 1, 2014.

[61] A. d. Garcez, M. Gori, L. C. Lamb, L. Serafini, M. Spranger, and S. N. Tran, "Neural-symbolic computing: An effective methodology for principled integration of machine learning and reasoning," *arxiv:1905.06088*, 2019.

[62] L. D. Raedt, K. Kersting, and S. Natarajan, *Statistical Relational Artificial Intelligence: Logic, Probability, and Computation*. Morgan & Claypool Publishers, 2016.

[63] T. M. Mitchell, R. M. Keller, and S. T. Kedar-Cabelli, "Explanation-based generalization: A unifying view," *Machine Learning*, vol. 1, no. 1, 1986.

[64] M. Núñez, "The use of background knowledge in decision tree induction," *Machine Learning*, vol. 6, no. 3, 1991.

[65] K. McGarry, S. Wermter, and J. MacIntyre, "Hybrid neural systems: From simple coupling to fully integrated neural networks," *Neural Computing Surveys*, vol. 2, no. 1, 1999.

[66] R. Sun, "Connectionist implementationalism and hybrid systems," *Encyclopedia of Cognitive Science*, 2006.

[67] G. Hautier, C. C. Fischer, A. Jain, T. Mueller, and G. Ceder, "Finding nature's missing ternary oxide compounds using machine learning and density functional theory," *Chemistry of Materials*, vol. 22, no. 12, 2010.

[68] C. Hoerig, J. Ghaboussi, and M. F. Insana, "An information-based machine learning approach to elasticity imaging," *Biomechanics and modeling in mechanobiology*, vol. 16, no. 3, 2017.

[69] H. S. Kim, M. Koc, and J. Ni, "A hybrid multi-fidelity approach to the optimal design of warm forming processes using a knowledge-based artificial neural network," *Int. J. Machine Tools and Manufacture*, vol. 47, no. 2, 2007.

[70] J. Ingraham, A. Riesselman, C. Sander, and D. Marks, "Learning protein structure with a differentiable simulator," in *Int. Conf. Learning Representations (ICLR)*, 2019.

[71] T. Deist, A. Patti, Z. Wang, D. Krane, T. Sorenson, and D. Craft, "Simulation assisted machine learning." *Bioinformatics (Oxford, England)*, 2019.

[72] A. Lerer, S. Gross, and R. Fergus, "Learning physical intuition of block towers by example," *arxiv:1603.01312*, 2016.

[73] A. Rai, R. Antonova, F. Meier, and C. G. Atkeson, "Using simulation to improve sample-efficiency of bayesian optimization for bipedal robots." *J. Machine Learning Research*, vol. 20, no. 49, 2019.

[74] Y. Du, Z. Liu, H. Basevi, A. Leonardis, B. Freeman, J. Tenenbaum, and J. Wu, "Learning to exploit stability for 3d scene parsing," in *Adv. in Neural Information Processing Systems (NIPS)*, 2018.

[75] H. Ren, R. Stewart, J. Song, V. Kuleshov, and S. Ermon, "Adversarial constraint learning for structured prediction," in *Int. Joint Conf. Artificial Intelligence (IJCAI)*. AAAI, 2018.

[76] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proc. Conf. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[77] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *Int. J. Computer Vision (IJCV)*, vol. 115, no. 3, 2015.

[78] F. Wang and Q.-J. Zhang, "Knowledge-based neural models for microwave design," *Trans. Microwave Theory and Techniques*, vol. 45, no. 12, 1997.

[79] S. J. Leary, A. Bhaskar, and A. J. Keane, "A knowledge-based approach to response surface modelling in multifidelity optimization," *J. Global Optimization*, vol. 26, no. 3, 2003.

[80] J. J. Grefenstette, C. L. Ramsey, and A. C. Schultz, "Learning sequential decision rules using simulation models and competition," *Machine Learning*, vol. 5, no. 4, 1990.

[81] N. Ruiz, S. Schulter, and M. Chandraker, "Learning to simulate," in *Int. Conf. Learning Representations (ICLR)*, 2019.

[82] Y. Yang and P. Perdikaris, "Physics-informed deep generative models," *arxiv:1812.03511*, 2018.

[83] E. de Bezenac, A. Pajot, and P. Gallinari, "Deep learning for physical processes: Incorporating prior scientific knowledge," *arxiv:1711.07970*, 2017.

[84] M. Lutter, C. Ritter, and J. Peters, "Deep lagrangian networks: Using physics as model prior for deep learning," *arxiv:1907.04490*, 2019.

[85] D. C. Psichogios and L. H. Ungar, "A hybrid neural network-first principles approach to process modeling," *AIChE Journal*, vol. 38, no. 10, pp. 1499–1511, 1992.

[86] W. O. Ward, T. Ryder, D. Prangle, and M. A. Álvarez, "Variational bridge constructs for grey box modelling with gaussian processes," *arxiv:1906.09199*, 2019.

[87] I. E. Lagaris, A. Likas, and D. I. Fotiadis, "Artificial neural networks for solving ordinary and partial differential equations," *Trans. Neural Networks*, vol. 9, no. 5, 1998.

[88] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics informed deep learning (part ii): Data-driven discovery of nonlinear partial differential equations," *arxiv:1711.10566*, 2017.

[89] Y. Yang and P. Perdikaris, "Adversarial uncertainty quantification in physics-informed neural networks," *J. Computational Physics*, vol. 394, 2019.

[90] Y. Zhu, N. Zabaras, P.-S. Koutsourelakis, and P. Perdikaris, "Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data," *J. Computational Physics*, vol. 394, 2019.

[91] F. D. A. Belbute-peres, K. R. Allen, K. A. Smith, and J. B. Tenenbaum, "End-to-end differentiable physics for learning and control," in *Adv. in Neural Information Processing Systems (NIPS)*, 2018.

[92] G. Strang, *Computational science and engineering*, 2007, no. Sirsi) i9780961408817.

[93] L. Ljung, "System identification," *Wiley Encyclopedia of Electrical and Electronics Engineering*, 2001.

[94] K. P. Murphy and S. Russell, "Dynamic bayesian networks: Representation, inference and learning," 2002.

[95] R. Speer and C. Havasi, "Conceptnet 5: A large semantic network for relational knowledge," in *The People's Web Meets NLP*. Springer, 2013, pp. 161–176.

[96] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: A collaboratively created graph database for structuring human knowledge," in *Proc. Int. Conf. Management of Data (SIGMOD)*. AcM, 2008.

[97] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "Dbpedia: A nucleus for a web of open data," in *The semantic web*. Springer, 2007.

[98] T. Ma and A. Zhang, "Multi-view factorization autoencoder with network constraints for multi-omic integrative analysis," in *Int. Conf. Bioinformatics and Biomedicine (BIBM)*. IEEE, 2018.

[99] Z. Che, D. Kale, W. Li, M. T. Bahadori, and Y. Liu, "Deep computational phenotyping," in *Proc. Int. Conf. Knowledge Discovery and Data Mining (SIGKDD)*. ACM, 2015.

[100] E. Choi, M. T. Bahadori, L. Song, W. F. Stewart, and J. Sun, "Gram: Graph-based attention model for healthcare representation learning," in *Proc. Int. Conf. Knowledge Discovery and Data Mining (SIGKDD)*. ACM, 2017.

[101] H. J. Lowe and G. O. Barnett, "Understanding and using the medical subject headings (mesh) vocabulary to perform literature searches," *Jama*, vol. 271, no. 14, 1994.

[102] G. A. Miller, "Wordnet: A lexical database for english," *Communications of the ACM*, vol. 38, no. 11, 1995.

[103] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, B. Yang, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel *et al.*, "Never-ending learning," *Communications of the ACM*, vol. 61, no. 5, 2018.

[104] M. B. Chang, T. Ullman, A. Torralba, and J. B. Tenenbaum, "A compositional object-based approach to learning physical dynamics," *arxiv:1612.00341*, 2016.

[105] X. Liang, Z. Hu, H. Zhang, L. Lin, and E. P. Xing, "Symbolic graph reasoning meets convolutions," in *Adv. in Neural Information Processing Systems (NIPS)*, 2018.

[106] H. Zhou, T. Young, M. Huang, H. Zhao, J. Xu, and X. Zhu, "Commonsense knowledge aware conversation generation with graph attention." in *Proc. Int. Joint Conf. Artificial Intelligence (IJCAI)*, 2018.

[107] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," in *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, 2009.

[108] Z.-X. Ye and Z.-H. Ling, "Distant supervision relation extraction with intra-bag and inter-bag attentions," *arxiv:1904.00143*, 2019.

[109] J. Bian, B. Gao, and T.-Y. Liu, "Knowledge-powered deep learning for word embedding," in *Joint European Conf. machine learning and knowledge discovery in databases*. Springer, 2014.

[110] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arxiv:1301.3781*, 2013.

[111] Z. Zhang, X. Han, Z. Liu, X. Jiang, M. Sun, and Q. Liu, "Ernie: Enhanced language representation with informative entities," *arxiv:1905.07129*, 2019.

[112] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arxiv:1810.04805*, 2018.

[113] N. Mrkšić, D. O. Séaghdha, B. Thomson, M. Gašić, L. Rojas-Barahona, P.-H. Su, D. Vandyke, T.-H. Wen, and S. Young, "Counter-fitting word vectors to linguistic constraints," *arxiv:1603.00892*, 2016.

[114] G. Glavaš and I. Vulić, "Explicit retrofitting of distributional word vectors," in *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018.

[115] Y. Fang, K. Kuan, J. Lin, C. Tan, and V. Chandrasekhar, "Object detection meets knowledge graphs," 2017.

[116] A. Feelders and L. C. Van der Gaag, "Learning bayesian network parameters under order constraints," *Int. J. Approximate Reasoning*, vol. 42, no. 1-2, 2006.

[117] R. D. Gardner and D. A. Harle, "Alarm correlation and network fault resolution using the kohonen self-organising map," in *Global Telecommunications Conf.*, vol. 3. IEEE, 1997.

[118] A. R. Masegosa and S. Moral, "An interactive approach for bayesian network learning using domain/expert knowledge," *Int. J. Approximate Reasoning*, vol. 54, no. 8, 2013.

[119] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning bayesian networks: The combination of knowledge and statistical data," *Machine Learning*, vol. 20, no. 3, 1995.

[120] L. M. de Campos and J. G. Castellano, "Bayesian network learning algorithms using structural restrictions," *Int. J. Approximate Reasoning*, vol. 45, no. 2, 2007.

[121] M. S. Massa, M. Chiogna, and C. Romualdi, "Gene set analysis exploiting the topology of a pathway," *BMC systems biology*, vol. 4, no. 1, 2010.

[122] M. B. Messaoud, P. Leray, and N. B. Amor, "Integrating ontological knowledge for iterative causal discovery and visualization," in *European Conf. Symbolic and Quantitative Approaches to Reasoning and Uncertainty*. Springer, 2009.

[123] N. Angelopoulos and J. Cussens, "Bayesian learning of bayesian networks with informative priors," *Annals of Mathematics and Artificial Intelligence*, vol. 54, no. 1-3, 2008.

[124] N. Piatkowski, S. Lee, and K. Morik, "Spatio-temporal random fields: Compressible representation and distributed estimation," *Machine Learning*, vol. 93, no. 1, 2013.

[125] A. C. Constantinou, N. Fenton, and M. Neil, "Integrating expert knowledge with data in bayesian networks: Preserving data-driven expectations when the expert variables remain unobserved," *Expert Systems with Applications*, vol. 56, 2016.

[126] B. Yet, Z. B. Perkins, T. E. Rasmussen, N. R. Tai, and D. W. R. Marsh, "Combining data and meta-analysis to build bayesian networks for clinical decision support," *J. Biomedical Informatics*, vol. 52, 2014.

[127] M. Richardson and P. Domingos, "Learning with knowledge from multiple experts," in *Proc. of the 20th Int. Conf. Machine Learning (ICML)*, 2003.

[128] G. Borboudakis and I. Tsamardinos, "Incorporating causal prior knowledge as path-constraints in bayesian networks and maximal ancestral graphs," *arxiv:1206.6390*, 2012.

[129] B. Yet, Z. Perkins, N. Fenton, N. Tai, and W. Marsh, "Not just data: A method for improving prediction with knowledge," *J. Biomedical Informatics*, vol. 48, 2014.

[130] C. Gardiner, *Stochastic methods*. Springer Berlin, 2009, vol. 4.

[131] J. N. Nielsen, H. Madsen, and P. C. Young, "Parameter estimation in stochastic differential equations: An overview," *Annual Reviews in Control*, vol. 24, pp. 83–94, 2000.

[132] B. Schölkopf, P. Simard, A. J. Smola, and V. Vapnik, "Prior knowledge in support vector kernels," in *Adv. in Neural Information Processing Systems (NIPS)*, 1998.

[133] J.-L. Wu, H. Xiao, and E. Paterson, "Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework," *Physical Review Fluids*, vol. 3, no. 7, 2018.

[134] J. Ling, A. Kurzawski, and J. Templeton, "Reynolds averaged turbulence modelling using deep neural networks with embedded invariance," *J. Fluid Mechanics*, vol. 807, 2016.

[135] A. Butter, G. Kasieczka, T. Plehn, and M. Russell, "Deep-learned top tagging with a lorentz layer," *SciPost Phys*, vol. 5, no. 28, 2018.

[136] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: Going beyond euclidean data," *Signal Processing*, vol. 34, no. 4, 2017.

[137] T. S. Cohen and M. Welling, "Group equivariant convolutional networks," in *Proc. Int. Conf. Machine Learning (ICML)*, 2016.

[138] S. Dieleman, J. De Fauw, and K. Kavukcuoglu, "Exploiting cyclic symmetry in convolutional neural networks," *arxiv:1602.02660*, 2016.

[139] J. Li, Z. Yang, H. Liu, and D. Cai, "Deep rotation equivariant network," *Neurocomputing*, vol. 290, 2018.

[140] D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow, "Harmonic networks: Deep translation and rotation equivariance," in *Proc. Conf. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[141] P. Niyogi, F. Girosi, and T. Poggio, "Incorporating prior information in machine learning by creating virtual examples," *Proc. of the IEEE*, vol. 86, no. 11, 1998.

[142] D. L. Bergman, "Symmetry constrained machine learning," in *Proc. of SAI Intelligent Systems Conf.* Springer, 2019.

[143] F. Chazal and B. Michel, "An introduction to topological data analysis: Fundamental and practical aspects for data scientists," *arXiv:1710.04019*, 2017.

[144] T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, I. Osband *et al.*, "Deep q-learning from demonstrations," in *Conf. Artificial Intelligence*. AAAI, 2018.

[145] T. Spinner, U. Schlegel, H. Schäfer, and M. El-Assady, "explainer: A visual analytics framework for interactive and explainable machine learning," *Trans. Visualization and Computer Graphics*, 2019.

[146] S. Das, D. Cashman, R. Chang, and A. Endert, "Beames: Interactive multi-model steering, selection, and inspection for regression tasks." *Computer Graphics and Applications*, 2019.

[147] J. A. Fails and D. R. Olsen Jr, "Interactive machine learning," in *Proc. Int. Conf. Intelligent User Interfaces*. ACM, 2003.

[148] E. T. Brown, J. Liu, C. E. Brodley, and R. Chang, "Dis-function: Learning distance functions interactively," in *Conf. Visual Analytics Science and Technology (VAST)*. IEEE, 2012.

[149] J. Choo, C. Lee, C. K. Reddy, and H. Park, "Utopian: User-driven topic modeling based on interactive nonnegative matrix factorization," *Trans. Visualization and Computer Graphics*, vol. 19, no. 12, 2013.

[150] S. Tong and E. Chang, "Support vector machine active learning for image retrieval," in *Proc. Int. Conf. Multimedia*. ACM, 2001.

[151] J. Bernard, M. Zeppelzauer, M. Sedlmair, and W. Aigner, "Vial: A unified process for visual interactive labeling," *The Visual Computer*, vol. 34, no. 9, 2018.

[152] D. Oglic, D. Paurat, and T. Gärtner, "Interactive knowledge-based kernel pca," in *Joint European Conf. Machine Learning and Knowledge Discovery in Databases*. Springer, 2014.

[153] W. B. Knox and P. Stone, "Interactively shaping agents via human reinforcement: The tamer framework," in *Proc. Int. Conf. Knowledge Cature (K-CAP)*. ACM, 2009.

[154] G. Warnell, N. Waytowich, V. Lawhern, and P. Stone, "Deep tamer: Interactive agent shaping in high-dimensional state spaces," in *Conf. Artificial Intelligence*. AAAI, 2018.

[155] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," in *Adv. in Neural Information Processing Systems (NIPS)*, 2017.

[156] R. Kaplan, C. Sauer, and A. Sosa, "Beating atari with natural language guided reinforcement learning," *arxiv:1704.05539*, 2017.

[157] P. Schramowski, W. Stammer, S. Teso, A. Brugger, H.-G. Luigs, A.-K. Mahlein, and K. Kersting, "Right for the wrong scientific reasons: Revising deep networks by interacting with their explanations," *arXiv preprint arXiv:2001.05371*, 2020.

[158] B. Settles, "Active learning literature survey," University of Wisconsin–Madison, Computer Sciences Technical Report 1648, 2009.

[159] P. Przybyła, A. J. Brockmeier, G. Kontonatsios, M.-A. Le Pogam, J. McNaught, E. von Elm, K. Nolan, and S. Ananiadou, "Prioritising references for systematic reviews with robotanalyst: A user study," *Research synthesis methods*, vol. 9, no. 3, 2018.

[160] C. Wirth, R. Akrour, G. Neumann, and J. Fürnkranz, "A survey of preference-based reinforcement learning methods," *J. Machine Learning Research*, vol. 18, 2017.

[161] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon, "Visual analytics: Definition, process, and challenges," in *Information visualization*. Springer, 2008.

[162] D. Paurat and T. Gärtner, "Invis: A tool for interactive visual data analysis," in *Joint European Conf. Machine Learning and Knowledge Discovery in Databases*. Springer, 2013.

[163] D. Sacha, M. Kraus, D. A. Keim, and M. Chen, "Vis4ml: An ontology for visual analytics assisted machine learning," *Trans. Visualization and Computer Graphics*, vol. 25, no. 1, 2018.

[164] J. J. Dudley and P. O. Kristensson, "A review of user interface design for interactive machine learning," *Trans. Interactive Intelligent Systems (TiiS)*, vol. 8, no. 2, 2018.

Laura von Rueden received her M.Sc. degree in simulation sciences from the RWTH Aachen University in 2015, following a B.Sc. degree in physics. Afterwards, she was a data scientist at Capgemini. Since 2018, she is a research scientist at Fraunhofer IAIS and is a PhD candidate in computer science at the Universität Bonn. Her research interests include machine learning, simulation, and especially the combination of data- and knowledge-based modelling.

Sebastian Mayer received his diploma degree in Mathematics in 2011 from TU Darmstadt and his PhD in Mathematics in 2018 from University Bonn. Since 2017, he is a research scientist at Fraunhofer SCAI. His research interests are machine learning and biologically-inspired algorithms in the context of cyber-physical systems.

Katharina Beckh received her M.Sc. degree in Human-Computer Interaction in 2019 from Julius Maximilian University of Wuerzburg. Since 2019, she is a research scientist at Fraunhofer IAIS. Her research interests include interactive machine learning, human oriented modeling and text mining with a primary focus in the medical domain.

Bogdan Georgiev received his Ph.D. degree in Mathematics in 2018 from Max-Planck-Institute and Bonn University. Since 2018, he is a research scientist at Fraunhofer IAIS. His present research interests include aspects of learning theory such as generalization/compression bounds, geometric learning and quantum computing.

Sven Giesselbach received his M.Sc. degree in computer science in 2012 from university of Bonn. Since 2015, he is a data scientist at Fraunhofer IAIS and is also lead of the team natural language understanding at the department knowledge discovery. His research interests include the use of external knowledge in natural language processing.

Raoul Heese received the Diploma and PhD from the Institute of Quantum Physics, Ulm University, Germany in 2012 and 2016. He is currently working as a research scientist at Fraunhofer ITWM, Kaiserslautern, Germany. His research interests include informed learning, supervised learning and their application to real-world problems.

Rajkumar Ramamurthy received his M.Sc. in Media Informatics in 2016 from RWTH Aachen University. Since 2018, he is a Data Scientist at Fraunhofer IAIS and is also a Doctoral candidate at University of Bonn. His research interests include Reinforcement Learning and Natural Language Processing.

Birgit Kirsch received her M.Sc. degree in business informatics in 2017 from Hochschule Trier. Since 2017, she is a research scientist at Fraunhofer IAIS. Her research interests include Natural Language Processing and Statistical Relational Learning.

Jochen Garcke received his diploma degree in mathematics in 1999, and the Ph.D. degree in mathematics in 2004, both from the Universität Bonn. He was a Postdoctoral Fellow at the Australian National University from 2004 to 2006, a Postdoctoral Researcher from 2006 to 2008 and a Junior Research Group Leader from 2008 to 2011, both at the Technical University Berlin. Since 2011 he is Professor of numerics at the University of Bonn and department head at Fraunhofer SCAI, Sankt Augustin. His research interests include machine learning, scientific computing, reinforcement learning, and high-dimensional approximation. Prof. Garcke is a member of DMV, GAMM, and SIAM. He is a reviewer for IEEE Transactions on Industrial Informatics, IEEE Transactions on Neural Networks, and IEEE Transactions on Pattern Analysis and Machine Intelligence.

Michal Walczak received his PhD degree in physics from the Georg-August University of Goettingen, Germany, in 2014. Since 2016 he is a research scientist at Fraunhofer ITWM in Kaiserslautern, Germany. His research interests include machine learning, decision support, multi-criteria optimization, and their application to radiotherapy planning and process engineering.

Christian Bauckhage (M'02) received received the M.Sc. and Ph.D. degrees in computer science from Bielefeld University in 1998 and 2002, respectively. Since 2008, he is a professor of computer science at the University of Bonn and lead scientist for machine learning at Fraunhofer IAIS. Previously, he was with the Centre for Vision Research in Toronto, Canada, and was a Senior Research Scientist at Deutsche Telekom Laboratories, Berlin. His research focuses on theory and practice of learning systems and next generation computing. He regularly reviews for the IEEE Transactions on Neural Networks and Learning Systems, the IEEE Transactions on Pattern Analysis and Machine Intelligence, and the IEEE Transactions on Games for which he is also an associate editor.

Julius Pfrommer received his PhD degree in computer science in 2019 from Karlsruhe Institute of Technology (KIT). Since 2018 he is head of a research group at Fraunhofer IOSB. His research interests include distributed systems, planning under uncertainty, and optimization theory with its many applications for machine learning and optimal control.

Annika Pick received her M.Sc. degree in Computer Science in 2018 from University of Bonn. Since 2019, she is a data scientist at Fraunhofer IAIS. Her research interests include learning from healthcare data and pattern mining.

Jannis Schuecker received his doctoral degree in physics from the RWTH Aachen University. Until 2019, he was a research scientist at Fraunhofer IAIS. His research interests include machine learning, in particular, time series modeling using neural networks and interpretable machine learning.