Informed Machine Learning – A Taxonomy and Survey of Integrating Prior 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, Michal Walczak, Jochen Garcke, Christian Bauckhage, *Member, IEEE*, 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 of various approaches in this field. We provide a definition and propose a concept for informed machine learning which illustrates its building blocks and distinguishes it from conventional machine learning. 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. Based on this taxonomy, we survey related research and describe how different knowledge representations such as algebraic equations, logic rules, or simulation results can be used in learning systems. This evaluation of numerous papers on the basis of our taxonomy uncovers key methods in the field of informed machine learning.

Index Terms—Machine learning, prior knowledge, expert knowledge, informed, hybrid, neuro-symbolic, survey, taxonomy

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, which are important for trustworthy AI [8]. With machine learning models becoming more and more complex, there is also a growing need for models to be interpretable and explainable [9].

These issues have led to increased research on how to improve machine learning models by additionally incorporating prior knowledge into the learning process. Although integrating knowledge into machine learning is common, e.g., through labelling or feature engineering, we observe a growing interest in the integration of more knowledge, and especially of further formal knowledge representations. For example, logic rules [10], [11] or algebraic equations [12], [13] have been added as constraints to loss functions. Knowledge graphs can enhance neural networks with information about relations between instances [14], which is of interest in image classification [15], [16]. Furthermore, physical simulations have been used to enrich training data [17], [18], [19]. This heterogeneity in approaches leads to some redundancy in nomenclature; for instance, we find terms such as physics-informed deep learning [20], physics-guided neural networks [12], or semantic-based regularization [21]. The recent growth of research activities shows that the combination of data- and knowledge-driven approaches becomes relevant in more and more areas. However, the growing number and increasing variety of research papers in this field motivates a systematic survey.

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 [22]. Even for support vector machines there exists a survey about the incorporation of knowledge into this formalism [23]. The fusion of symbolic and connectionist AI seems more and

- *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. E-mail: laura.von.rueden@iais.fraunhofer.de.*
- *Sebastian Mayer and Jochen Garcke are with the Fraunhofer SCAI, Institute for Algorithms and Scientific Computing, 53757 Sankt Augustin, Germany.*
- *Raoul Heese and Michal 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.*

Manuscript received 5 Feb. 2020; revised 15 Mar. 2021; accepted 26 Apr. 2021. Date of publication 12 May 2021; date of current version 7 Dec. 2022.
(Corresponding author: Laura von Rueden.)
Recommended for acceptance by L. Chen.
Digital Object Identifier no. 10.1109/TKDE.2021.3079836

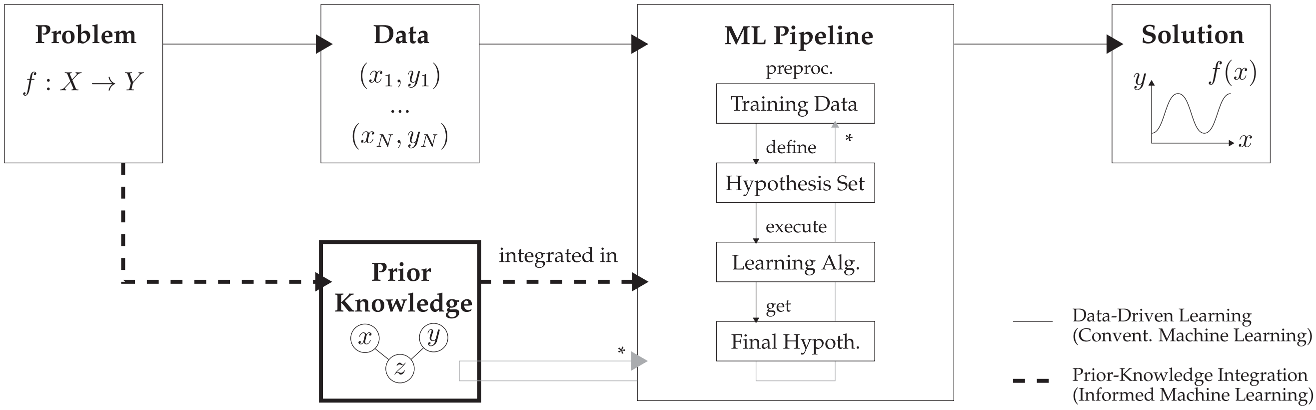


Fig. 1. Information flow in informed machine learning. The informed machine learning pipeline requires a hybrid information source with two components: Data and prior knowledge. In conventional machine learning knowledge is used for data preprocessing and feature engineering, but this process is deeply intertwined with the learning pipeline (*). In contrast, in *informed machine learning* prior knowledge comes from an independent source, is given by formal representations (e.g., by knowledge graphs, simulation results, or logic rules), and is explicitly integrated.

more approachable. In this regard, we refer to recent a survey on graph neural networks and a research direction framed as relational inductive bias [24]. Our work complements the aforementioned surveys by providing a systematic categorization of knowledge representations that are integrated into machine learning. We 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: We propose an abstract concept for informed machine learning that clarifies its building blocks and relation to conventional machine learning. It states that informed learning uses a hybrid information source that consists of data and prior knowledge, which comes from an independent source and is given by formal representations. Our main contribution is the introduction of a taxonomy that classifies informed machine learning approaches, which is novel and the first of its kind. It contains the dimensions of the knowledge source, 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. Moreover, we present a description of available approaches and explain how different knowledge representations, e.g., algebraic equations, logic rules, or simulation results, can be used in informed machine learning.

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 methodical 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 formulation of our concept for *informed machine learning*. In Section 3, we describe how we classified the approaches in terms of our applied surveying methodology and our obtained key insights. Section 4 presents the taxonomy and its elements that we distilled from surveying a large number of research papers. In Section 5, we describe the approaches for the integration of knowledge into machine learning classified according to the taxonomy in more detail. After brief historical account in Section 6, we finally discuss future directions in Section 7 and conclude in Section 8.

2 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.

2.1 Knowledge

The meaning of knowledge is difficult to define in general and is an ongoing debate in philosophy [25], [26], [27]. During the generation of knowledge, it first appears as useful information [28], which is subsequently validated. People validate information about the world using the brain's inner statistical processing capabilities [29], [30] or by consulting trusted authorities. Explicit forms of validation are given by empirical studies or scientific experiments [27], [31].

Here, we assume a computer-scientific perspective and understand knowledge as validated information about relations between entities in certain contexts. 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 versus mathematical formula). The more formally knowledge is represented, the more easily it can be integrated into machine learning.

2.2 Integrating Prior Knowledge into Machine Learning

Apart from the usual information source in a machine learning pipeline, the training data, one can additionally integrate knowledge. If this knowledge is pre-existent and independent of learning algorithms, it can be called prior knowledge. Moreover, such prior knowledge can be given by formal representations, which exist in an external, separated way from the learning problem and the usual training data. Machine learning that explicitly integrates such knowledge representations 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 comes from an independent source, is given by formal representations, and is explicitly integrated into the machine learning pipeline.*

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

2.2.1 *Conventional Machine Learning*

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 can typically be formulated 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 [32].

In traditional approaches, knowledge is generally used in the learning pipeline, however, mainly for training data preprocessing (e.g., labelling) or feature engineering. This kind of integration is involved and deeply intertwined with the whole learning pipeline, such as the choice of the hypothesis set or the learning algorithm, as depicted in Fig. 1. Hence, this knowledge is not really used as an independent source or through separated representations, but is rather used with adaption and as required.

2.2.2 *Informed Machine Learning*

The information flow of informed machine learning comprises an additional prior-knowledge integration and thus consists of two lines originating from the problem, as shown in Fig. 1. 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.

3 Classification of Approaches

To comprehend how the concept of informed machine learning is implemented, we performed a systematic classification of existing approaches based on an extensive literature survey. Our goals are to uncover different methods, identify their similarities or differences, and to offer guidelines for users and researchers. In this section, we describe our classification methodology and summarize our key insights.

3.1 Methodology

The methodology of our classification is determined by specific analysis questions which we investigated in a systematic literature survey.

3.1.1 *Analysis Questions*

Our guiding question is how prior knowledge can be integrated into the machine learning pipeline. Our answers will particularly focus on three aspects: Since prior knowledge in informed machine learning consists of an independent source and requires some form of explicit representations, we consider knowledge sources and representations. Since it also is essential at which component of the machine learning pipeline what kind of knowledge is integrated, we also consider integration methods. In short, our literature survey addresses the following three questions:

1) *Source*: Which source of knowledge is integrated?
2) *Representation*: How is the knowledge represented?
3) *Integration*: Where in the learning pipeline is it integrated?

3.1.2 *Literature Surveying Procedure*

To systematically answer the above analysis questions, we surveyed a large number of publications describing informed machine learning approaches. We used a comparative and iterative surveying procedure that consisted of different cycles. In the first cycle, we inspected an initial set of papers and took notes as to how each paper answers our questions. Here, we observed that specific answers occur frequently, which then led to the idea of devising a classification framework in the form of a taxonomy. In the second cycle, we inspected an extended set of papers and classified them according to a first draft of the taxonomy. We then further refined the taxonomy to match the observations from the literature. In the third cycle, we re-inspected and re-sorted papers and, furthermore, expanded our set of papers. This resulted in an extensive literature basis in which all papers are classified according to the distilled taxonomy.

3.2 Key Insights

Next, we present an overview over key insights from our systematic classification. As a preview, we refer to Fig. 2, which visually summarizes our findings. A more detailed description of our findings will be given in Sections 4 and 5.

3.2.1 *Taxonomy*

Based on a comparative and iterative literature survey, we identified a taxonomy that we propose as a classification framework for informed machine learning approaches. Guided by the above analysis questions, the taxonomy consists of the three dimensions *knowledge source*, *knowledge representation* and *knowledge integration*. Each dimension contains a set of elements that represent the spectrum of different approaches found in the literature. This is illustrated in the taxonomy in Fig. 2.

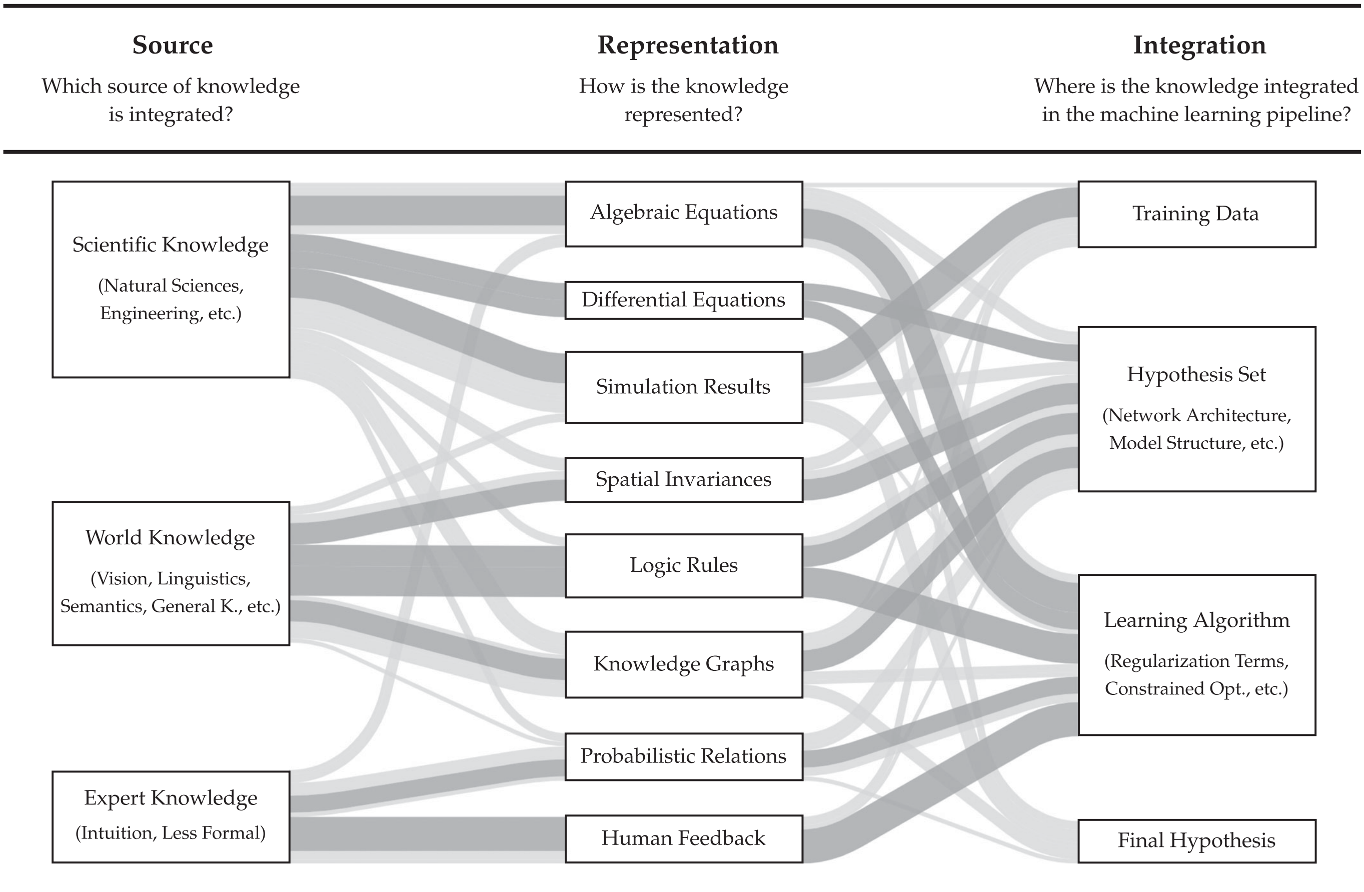


Fig. 2. Taxonomy of informed machine learning. This taxonomy serves as a classification framework for *informed machine learning* and structures approaches according to the three above analysis questions about the *knowledge source*, *knowledge representation* and *knowledge integration*. Based on a comparative and iterative literature survey, we identified for each dimension a set of elements that represent a spectrum of different approaches. The size of the elements reflects the relative count of papers. We combine the taxonomy with a Sankey diagram in which the paths connect the elements across the three dimensions and illustrate the approaches that we found in the analyzed papers. The broader the path, the more papers we found for that approach. Main paths (at least four or more papers with the same approach across all dimensions) are highlighted in darker grey and represent central approaches of informed machine learning.

With respect to knowledge sources, we found three broad categories: Rather specialized and formalized scientific knowledge, everyday life's world knowledge, and more intuitive expert knowledge. For scientific knowledge we found the most informed machine learning papers. With respect to knowledge representations, we found versatile and fine-grained approaches and distilled eight categories (Algebraic equations, differential equations, simulation results, spatial invariances, logic rules, knowledge graphs, probabilistic relations and human feedback). Regarding knowledge integration, we found approaches for all stages of the machine learning pipeline, from the training data and the hypothesis set, over the learning algorithm, to the final hypothesis. However, most informed machine learning papers consider the two central stages.

Depending on the perspective, the taxonomy can be regarded from either one of two sides: An application-oriented user might prefer to read the taxonomy from left to right, starting with some given knowledge source and then selecting representation and integration. Vice versa, a method-oriented developer or researcher might prefer to read the taxonomy from right to left, starting with some given integration method. For both perspectives, knowledge representations are important building blocks and constitute an abstract interface that connects the application- and the method-oriented side.

3.2.2 Frequent Approaches

The taxonomy serves as a classification framework and allows us to identify frequent approaches of informed machine learning. In our literature survey, we categorized each research paper with respect to each of the three taxonomy dimensions.

Paths Through the Taxonomy. When visually highlighting and connecting them, a specific combination of entries across the taxonomy dimensions figuratively results in a path through the taxonomy. Such paths represent specific approaches towards informed learning and we illustrate this by combining the taxonomy with a Sankey diagram, as shown in Fig. 2. We observe that, while various paths through the taxonomy are possible, specific ones occur more frequently and we will call them main paths. For example, we often observed the approach that scientific knowledge is represented in algebraic equations, which are then integrated into the learning algorithm, e.g., the loss function. As another example, we often found that world knowledge such as linguistics is represented by logic rules,

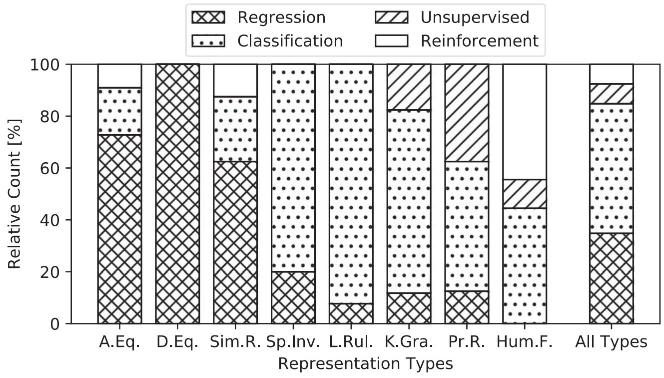


Fig. 3. Knowledge representations and learning tasks.

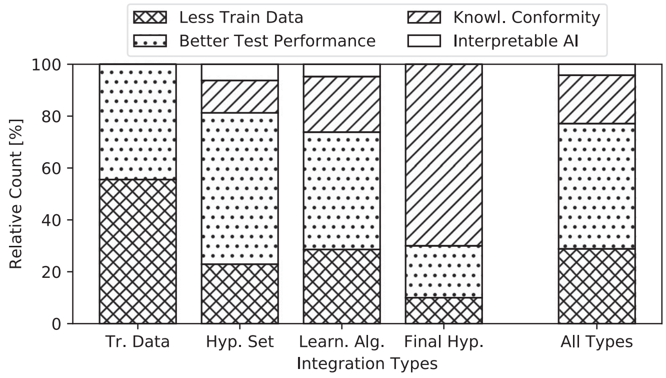


Fig. 4. Knowledge integration and its goals.

which are then integrated into the hypothesis set, e.g., the network architecture. These paths, especially the main paths, can be used as a guideline for users new to the field or provide a set of baseline methods for researchers.

Paths From Source to Representation. We found that the paths from source to representation form groups. That is, for every knowledge source there appear prevalent representation types. Scientific knowledge is mainly represented in terms of algebraic or differential equations or exist in form of simulation results. While other forms of representation are possible, too, there is a clear preference for equations or simulations, likely because most sciences aim at finding natural laws encoded in formulas. For world knowledge, the representation forms of logic rules, knowledge graphs, or spatial invariances are the primary ones. These can be understood as a group of symbolic representations. Expert knowledge is mainly represented by probabilistic relations or human feedback. This is appears reasonable because such representations allow for informality as well as for a degree of uncertainty, both of which might be useful for representing intuition. We also performed an additional analysis on the dependency of the learning task and found a confirmation of the above described representation groups as shown in Fig. 3.

From a theoretical point of view, transformations between representations are possible and indeed often apparent within the aforementioned groups. For example, equations can be transformed to simulation results, or logic rules can be represented as knowledge graphs and vice versa. Nevertheless, from a practical point of view, differentiating between forms of representations appears useful as specific representations might already be available in a given set up.

Paths from Representation to Integration. For most of the representation types we found at least one main path to an integration type. The following mappings can be observed. Simulation results are very often integrated into the training data. Knowledge graphs, spatial invariances, and logic rules are frequently incorporated into the hypothesis set. The learning algorithm is mainly enhanced by algebraic or differential equations, logic rules, probabilistic relations, or human feedback. Lastly, the final hypothesis is often checked by knowledge graphs or also by simulation results. However, since we observed various possible types of integration for all representation types, the integration still appears to be problem specific.

Hence, we additionally analyzed the literature for the goal of the prior knowledge integration and found four main goals: Data efficiency, accuracy, interpretability, or knowledge conformity. Although these goals are interrelated or even partially equivalent according to statistical learning theory, it is interesting to examine them as different motivations for the chosen approach. The distribution of goals for the distinct integration types is shown in Fig. 4. We observe that the main goal always is to achieve better performance. The integration of prior knowledge into the training data stands out, because its main goal is to train with less data. The integration into the final hypothesis is also special, because it is mainly used to ensure knowledge conformity for secure and trustworthy AI. All in all, this distribution suggests suitable integration approaches depending on the goal.

4 Taxonomy

In this section, we describe the *informed machine learning* taxonomy that we distilled as a classification framework in our literature survey. For each of the three taxonomy dimensions *knowledge source*, *knowledge representation* and *knowledge integration* we describe the found elements, as shown in Fig. 2. While an extensive approach categorization according to this taxonomy with further concrete examples will be presented in the next section (Section 5), we here describe the taxonomy on a more conceptual level.

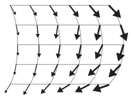
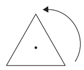
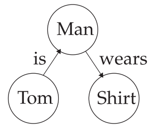
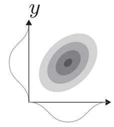
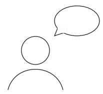
4.1 Knowledge Source

The category *knowledge source* refers to the origin of prior knowledge to be integrated in machine learning. 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 the sciences or is a form of world or expert knowledge, as illustrated on the left in Fig. 2. This list is neither complete nor disjoint but intended show a spectrum from more formal to less formal, or 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.

Scientific Knowledge. We subsume the subjects of science, technology, engineering, and mathematics under *scientific knowledge*. Such knowledge is typically formalized and validated explicitly through scientific experiments. Examples are the universal laws of physics, bio-molecular descriptions of genetic sequences, or material-forming production processes.

TABLE 1
Illustrative Overview of Knowledge Representations in the Informed Machine Learning Taxonomy

Algebraic Equations	Differential Equations	Simulation Results	Spatial Invariances	Logic Rules	Knowledge Graphs	Probabilistic Relations	Human Feedback
$E = m \cdot c^2$ $v \leqslant c$	$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$ $F(x) = m \frac{d^2 x}{dt^2}$			$A \wedge B \Rightarrow C$			

Each representation type is illustrated by a simple or prominent example in order to give a first intuitive understanding.

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. It can be more or less formal. Generally, it 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. Moreover, by world knowledge we also subsume linguistics. Such knowledge can also be explicitly validated through empirical studies. Examples are 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 is rather informal and needs to be formalized, e.g., with human-machine interfaces. It is also validated implicitly through a group of experienced specialists. In the context of cognitive science, this expert knowledge can also become intuitive [29]. For example, an engineer or a physician acquires knowledge over several years of experience working in a specific field.

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 Fig. 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 Fig. 2 and illustrated 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. 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.

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.

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.

Spatial Invariances. Spatial 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.

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.

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 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.

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.

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 or validated 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.

5 Description of Integration Approaches

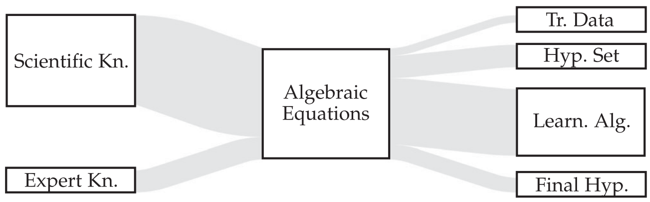
In this section, we give a detailed account of the informed machine learning approaches we found in our literature survey. We will focus on methods and therefore structure our presentation according to knowledge representations. This is motivated by the assumption that similar representations are integrated into machine learning in similar ways as they form the mathematical basis for the integration. Moreover the representations combine both the application- and the method-oriented perspective as described in Section 3.2.1.

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 Tables 2 and 3, which lists the paper references sorted according to our taxonomy.

5.1 Algebraic Equations

The main path for algebraic equations that we found in our literature survey comes from scientific knowledge and goes into the learning algorithm, but also other integration types are possible, as illustrated in the following figure.



5.1.1 (Paths From) Knowledge Source

Algebraic equations are mainly used to represent formalized scientific knowledge, but may also be used to express more intuitive expert knowledge.

Scientific Knowledge. We observed that algebraic equations are used in machine learning in various domains of natural sciences and engineering, particularly in physics [12], [13], [33], [34], [35], but also in biology [36], [37], robotics [38], or manufacturing and production processes [34], [39].

Three representative examples 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 $y(t) = y_0 + v_0 t + at^2$. Such knowledge from Newtonian mechanics can be used to improve object detection and tracking in videos [13]. 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 $\rho(d_1) \leq \rho(d_2)$, which can be used in water temperature prediction [12]. Furthermore, for the prediction of key performance indicators in production 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 [34].

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 [10], [12]:

$$f^* = \arg\min_f \Big(\overbrace{\lambda_l \textstyle\sum_i L(f(x_i), y_i)}^{\texttt{Label-based}} + \overbrace{\lambda_r R(f)}^{\texttt{Regul.}} + \underbrace{\lambda_k L_k(f(x_i), x_i)}_{\texttt{Knowledge-based}} \Big). \tag{1}$$

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 [13].

Expert Knowledge. An example for the representation of expert knowledge is to define valid ranges of variables according to experts' intuition as approximation constraints [33] or monotonicity constraints [39].

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. The integration into the other stages is possible, too, and we describe the approaches here ordered by their occurence.

Learning Algorithm. Algebraic equations and inequations can be integrated into learning algorithms via additional loss terms [12], [13], [33], [35] or, more generally, via constrained problem formulation [36], [37], [39].

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 [12], [13]. 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 [33].

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 [37].

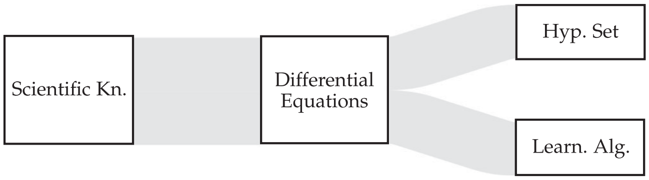
Hypothesis Set. An alternative approach is the integration into the hypothesis set. In particular, algebraic equations can be translated into the architecture of neural networks [34], [38], [40]. One idea is to sequence predefined operations leading to a functional decomposition [40]. More specifically, relations between input parameters, intermediate observables, or output variables reflecting physical constraints can be encoded as linear connections between the layers of a network model [34], [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 [12]. Such a physical consistency check can also comprise an entire diagnostics set of functions describing particular characteristics [41].

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 [42].

5.2 Differential Equations

Next, we describe informed machine learning approaches based on differential equations, which frequently represent scientific knowledge and are integrated into the hypothesis set or the learning algorithm.



5.2.1 (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 represent knowledge from the natural sciences.

Scientific Knowledge. Here we give three prominent examples: The work in [20], [43] 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 [44] are used in oceanography to model the evolution of sea surface temperatures. The Schrödinger equation studied in [20] describes quantum mechanical phenomena such as wave propagation in optical fibres or the behavior of Bose-Einstein condensates.

5.2.2 (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 Equation (1) [45]. 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 [20]. This ensures the physical plausibility of the neural network output. An extension to generative models is possible, too [43]. 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 [46].

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 [44], [47], [48]. 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 [49]. This also leads to hybrid architectures with knowledge-based modules, e.g., in form of a physics engine.

5.3 Simulation Results

Simulation results are also a prominent knowledge representation in informed machine learning. They mainly come from scientific knowledge and are used to extend the training data.

5.3.1 (Paths From) Knowledge Source

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

Scientific Knowledge. Simulation results informing machine learning can be found in fluid- and thermodynamics [12], material sciences [19], [60], [61], life sciences [59], mechanics and robotics [64], [65], [66], or autonomous driving [18]. To make it more concrete, we give three examples: 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 [61]. Even complex material forming processes can be simulated, for example a composite textile draping process can be simulated based on a finite-element model [19]. As an example for autonomous driving, urban traffic scenes under specific weather and illumination conditions, which might be useful for the training of visual perception components, can be simulated with dedicated physics engines [18].

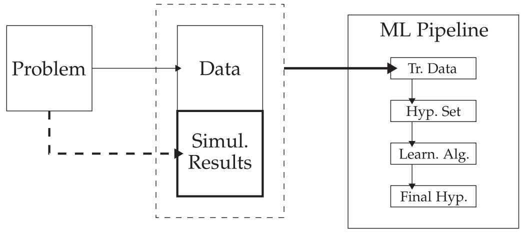


Fig. 5. Information flow for synthetic training data from simulations.

Insert 2: Simulation Results as Synthetic Tr. Data

The results from a simulation can be used as synthetic training data (see Fig. 5) and can thus augment the original, real training data. Some papers that follow this approach are [12], [18], [19], [59], [64], [65], [67].

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. Other approaches that occur frequently are the integration into the hypothesis set or the final hypothesis.

Training Data. The integration of simulation results into training data [12], [18], [19], [59], [64], [65], [67] depends on how the simulated, i.e., synthetic, data is combined with the real-world measurements:

First, additional input 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 [59].

Second, additional 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 [12].

Third, additional target variables are simulated and used as synthetic labels, which is of particular use when the original experiments are very expensive [19]. This approach can also be realized with physics engines, for example, pre-trained neural networks can be tailored towards an application through additional training on simulated data [64]. Synthetic training data generated from simulations can also be used to pre-train components of Bayesian optimization frameworks [65].

In informed machine learning, training data thus stems from a hybrid information source and contains both simulated and real data points (see Insert 2). 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 [67]. 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 [18].

Hypothesis Set. Another approach we observed integrates simulation results into the hypothesis set [60], [68], [69], which is of particular interest when dealing with low-fidelity

TABLE 2
References Classified by Knowledge Representation and (Path From) Knowledge Source

SOURCE	REPRESENTATION							
	Algebraic Equations	Differential Equations	Simulation Results	Spatial Invariances	Logic Rules	Knowledge Graphs	Probabilistic Relations	Human Feedback
Scientific Knowledge	[12], [13], [33] [34], [35], [36] [37], [38], [41] [39], [42]	[20], [43], [44] [45], [46], [47] [48], [49]	[12], [18], [19] [59], [60], [61] [64], [65], [66] [67], [68], [69]	[50], [51], [52]	[53], [54]	[14], [55], [56] [62], [63]	[57], [58]	
World Knowledge			[67], [70]	[71], [72], [73] [75], [76], [77] [83]	[10], [11], [13] [21], [78], [79] [84], [85], [86] [90], [91], [92]	[15], [16], [56] [80], [81], [82] [87], [88], [89] [93], [94], [95]	[74]	
Expert Knowledge	[33], [39], [40]						[74], [96], [97] [101], [102], [103] [107]	[98], [99], [100] [104], [105], [106] [108], [109], [110]

TABLE 3
References Classified by Knowledge Representation and (Path to) Knowledge Integration

INTEGRAT.	REPRESENTATION							
	Algebraic Equations	Differential Equations	Simulation Results	Spatial Invariances	Logic Rules	Knowledge Graphs	Probabilistic Relations	Human Feedback
Training Data	[42]		[12], [18], [19] [59], [64], [65] [67]	[52], [77], [83]		[81]		[100], [105]
Hypothesis Set	[34], [38], [40]	[44], [47], [48] [49]	[60], [68], [69]	[50], [51], [73] [71], [72], [75] [76]	[53], [78], [79] [54], [91], [92] [90]	[14], [15], [16] [62], [63], [82] [80], [87], [95]	[74], [97], [101]	[105]
Learning Algorithm	[12], [13], [33] [35], [36], [37] [39]	[20], [43], [45] [46]	[66]		[10], [11], [13] [21], [85], [86] [84]	[55], [56], [89]	[57], [96], [101] [58], [102], [103]	[98], [99], [100] [104], [106], [110] [108], [109]
Final Hypothesis	[12], [41]		[19], [61], [66] [70]			[88], [93], [94]	[107]	

simulations. These are simplified simulations that approximate the overall behaviour of a system but ignore intricate details for the sake of computing speed.

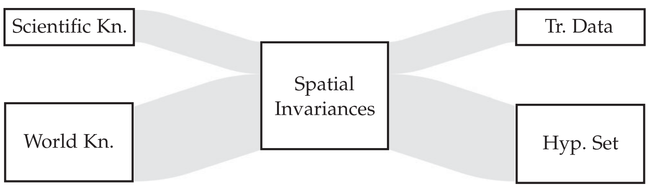
When building a machine learning model that reflects the actual, detailed behaviour of a system, low-fidelity simulation results or a response surface (a data-driven model of the simulation results) can be build into the architecture of a knowledge-based neural network (KBANN [53], see Insert 3), e.g. 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 or high-fidelity simulations [60], [69].

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 [66]. By means of reinforcement learning, this is even feasible if there are no gradients available from the simulation.

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 [19], [61], [66].

5.4 Spatial Invariances

Next, we describe informed machine learning approaches involving the representation type of spatial invariances. Their main path comes from world knowledge and goes to the hypothesis set.



5.4.1 (Paths From) Knowledge Source

We mainly found references using spatial invariances in the context of world knowledge or scientific knowledge.

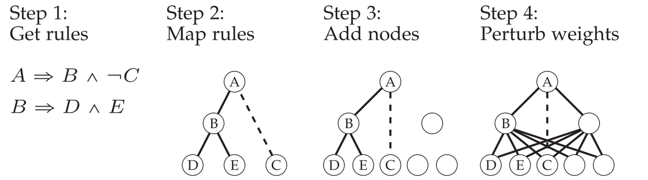


Fig. 6. Steps of rules-to-network translation [53]. Simple example for integrating rules into a KBANN.

Insert 3: Knowledge-Based Artificial Neural Networks (KBANNs)

Rules can be integrated into neural architectures by mapping the rule's components to the neurons and weights with these steps [53] (see Fig. 6):

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.

World Knowledge. Knowledge about invariances may fall into the category of world knowledge, for example when modeling facts about local or global pixel correlations in images [72]. 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.

Scientific Knowledge. 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 [52], [50]. For example, in equations modeling planetary motion, the angular momentum serves as such an invariant.

5.4.2 (Paths to) Knowledge Integration

In most references we found spatial invariances informing 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 [50], or the physical Minkowski metric that reflects mass invariance can be integrated via a Lorentz layer into a neural network [51].

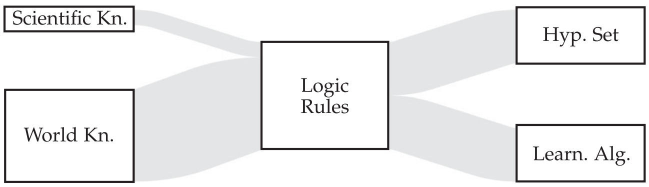
A recent trend is to integrate knowledge as spatial invariances into the architecture or layout of convolutional neural networks, which leads to so called geometric deep learning in [111]. A natural generalization of CNNs are group equivariant CNNs (G-CNNs) [70], [71], [74]. G-convolutions provide a 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 [75]. 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 [72]. 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 invariances into machine learning is the creation of virtual examples [76] 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 [82]. For instance, in turbulence modelling using the Reynolds stress tensor, a feature can be createad that is rotational, reflectional and Galilean invariant [52]. This is achieved by selecting features fulfilling rotational and Gallilean symmetries and augmenting the training data to ensure reflectional invariance.

5.5 Logic Rules

Logic Rules play an important role for the integration of prior knowledge into machine learning. In our literature survey, we mainly found the the source of world knowledge and the two integration paths into the hypothesis set and the learning algorithm.



5.5.1 (Path From) Knowledge Source

Logic rules can formalize knowledge from various sources, but the most frequent is world knowledge. Here we give some illustrative examples.

World Knowledge. Logic rules often describe knowledge about real-world objects [10], [11], [13], [77], [78] such as seen in images. This can focus on object properties, such as for animals x that $(\mathrm{FLY}(x) \wedge \mathrm{LAYEGGS}(x) \Rightarrow \mathrm{BIRD}(x))$ [10]. It can also focus on relations between objects such as the co-occurrence of characters in game scenes, e.g. $(\mathrm{PEACH} \Rightarrow \mathrm{MARIO})$ [13].

Another knowledge domain that can be well represented by logic rules is linguistics [83], [84], [85], [90], [91], [112], [113]. 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 [85]); 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 [83]).

Rules can also describe dependencies in social networks. For example, on a scientific research platform, it can be observed that authors citing each other tend to work in the same field $(\mathrm{Cite}(x, y) \wedge \mathrm{hasFieldA}(x) \Rightarrow \mathrm{hasFieldA}(y))$ [21].

5.5.2 (Path to) Knowledge Integration

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

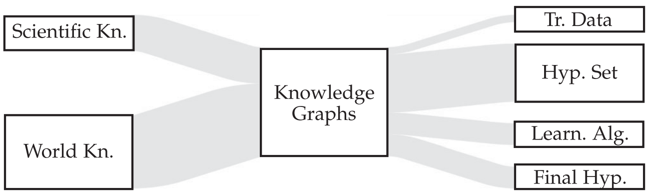
Hypothesis Set. Integration into the hypothesis set comprises both deterministic and probabilistic approaches. The former include neural-symbolic systems, which use rules as the basis for the model structure [53], [54], [89]. In Knowledge-Based Artificial Neural Networks (KBANNs), the architecture is constructed from symbolic rules by mapping the components of propositional rules to network components [53] as further explained in Insert 3. Extensions are available that also output a revised rule set [54] or also consider first-order logic [89]. A recent survey about neural-symbolic computing [114] summarizes further methods.

Integrating logic rules into the hypothesis set in a probabilistic manner is yet another approach [77], [78], [90], [91]. These belong to the research direction of statistical relational learning [115]. 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 [77], [90] and probabilistic soft logic [78], [91], 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.

Learning Algorithm. The integration of logic rules into the learning algorithm is often accomplished via additional, semantic loss terms [10], [11], [13], [21], [83], [84], [85]. 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 [10]. Semantic loss functions can also be derived from first principles using a set of axioms [11]. As a specific approach for student-teacher architectures, the rules can be first integrated in a teacher network and can then be used by a student network that is trained by minimizing a semantic loss term that measures the imitation of the teacher network [84], [85].

5.6 Knowledge Graphs

The taxonomy paths we observed in our literature survey that are related to knowledge representation are illustrated in the following graphic.



5.6.1 (Paths From) Knowledge Source

Since graphs are very versatile modeling tools, they can represent various kinds of structured knowledge. Typically, they are constructed from databases, however, the most frequent source we found in informed machine learning papers is world knowledge.

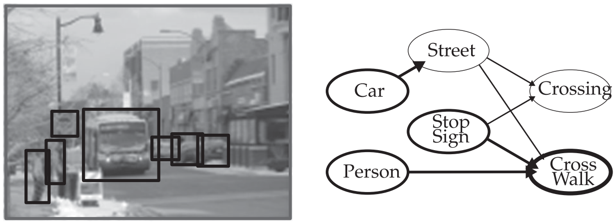


Fig. 7. Illustrative application example of using neural networks and knowledge graphs for image classification, similar as in [15]. The image (from the COCO dataset) shows a pedestrian cross walk.

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 [15] (see Fig. 7). During the detection, the network graph is propagated, starting with detected nodes and then expanding to neighbors [24].

World Knowledge. Since humans perceive the world as composed of entities, graphs are often used to represent relations between visual entities. For example, the Visual Genome knowledge graph is build from human annotations of object attributes and relations between objects in natural images [15], [16]. Similarly, the MIT ConceptNet [116] encompasses concepts of everyday life and their relations automatically built from text data. In natural language processing, knowledge graphs often represent knowledge about relations among concepts, which can be referred to by words. For example, WordNet [117] 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 [118].

Scientific Knowledge. In physics, graphs can immediately describe physical systems such as spring-coupled masses [14]. In medicine, networks of gene-protein interactions describe biological pathway information [55] and the hierarchical nature of medical diagnoses is captured by classification systems such as the International Classification of Diseases (ICD) [56], [63].

5.6.2 (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 [24]. A recent survey [24] 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 [14], [62] or object detection [16].

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 [15] (see Insert 4). More generally, a graph reasoning layer can be inserted into any neural network [81]. 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 [79]. Similarly, graph-based attention mechanism are used to counteract too few data points by using more general categories [63]. Also, attention on related knowledge graph embedding can support the training of word embeddings like ERNIE [86], which are fed into language models like BERT [94], [119].

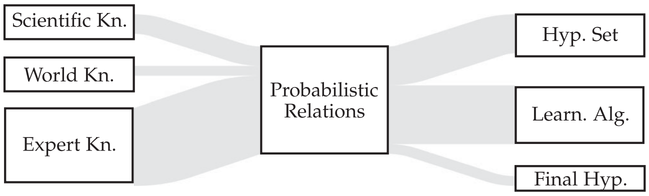
Training Data. Another prominent approach is distant supervision where information in a graph is used to automatically annotate texts to train natural language processing systems. This was originally done naïvely by considering each sentence that matches related entities in a graph as a training sample [80]; however, recently attention-based networks have been used to reduce the influence of noisy training samples [120].

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 [55], [56]. 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 [88] in word2vec training [121].

Final Hypothesis. Finally, graph can also be used to improve or validate final hypotheses or trained models. For instance, a recent development is to post-process word embeddings based on information from knowledge graphs [87], [92]. Furthermore, semantic segmentation in autonomous driving can be validated using knowledge graphs of street maps [110], or in object detection, predicted probabilities of a learning system can be refined using semantic consistency measures [93] derived form knowledge graphs. In both cases, the knowledge graphs are used to indicate whether the prediction is consistent with available knowledge.

5.7 Probabilistic Relations

The most frequent paths probabilistic relations found in our literature survey comes from expert knowledge and goes to the hypothesis set or the learning algorithm.



5.7.1 (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. In the car insurance, driver features like age relate to risk aversion [95]. Another examples is computer expertise for troubleshooting, i.e relating a device status to observations [90].

Scientific Knowledge. Correlation structures can also be obtained from natural sciences knowledge. For example, correlations between genes can be obtained from gene interaction networks [122] or from a gene ontology [57].

5.7.2 (Paths to) Knowledge Integration

We generally observe the integration of 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, extra nodes can be added to a Bayesian network thus altering the hypothesis set [96], or the structure of a probabilistic model can be chosen in accordance to given spatio-temporal structures [124]. In other hybrid approaches, the parameters of the conditional distribution of the Bayesian network are either learned from data or obtained from knowledge [73], [100].

Learning Algorithm. Human knowledge can also be used to define an informative prior [100], [125], which affects the learning algorithm as is has a regularizing effect. Structural constraints can alter score functions or the selection policies of conditional independence test, informing the search for the network structure [95]. 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 [102]. Causal network inference can make use of ontologies to select the tested interventions

[57]. Furthermore, prior causal knowledge can be used to constrain the direction of links in a Bayesian network [58].

Final Hypothesis. Finally, predictions obtained from a Bayesian network can be judged by probabilistic relational knowledge in order to refine the model [106].

5.8 Human Feedback

Finally, we look at informed machine learning approaches belonging to the representation type of human feedback. The most common path begins with expert knowledge and ends at the learning algorithm.

5.8.1 (Paths From) Knowledge Source

Compared to other categories in our taxonomy, knowledge representation via human feedback is less formalized and mainly stems from expert knowledge.

Expert Knowledge. Examples of knowledge that fall into this category include knowledge about topics in text documents [97], agent behaviors [98], [99], [103], [104], and data patterns and hierarchies [97], [105], [109]. Knowledge is often provided in form of relevance or preference feedback and humans in the loop can integrate their intuitive knowledge into the system without providing an explanation for their decision. For example, in object recognition, users can provide their corrective feedback about object boundaries via brush strokes [107]. As another example, in Game AI, an expert user can give spoken instructions for an agent in an Atari game [99].

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 algorithm itself. This often occurs in areas of reinforcement learning, or interactive learning combined with visual analytics.

Learning Algorithm. In reinforcement learning, an agent observes an unknown environment and learns to act based on reward signals. The TAMER framework [98] provides the agent with human feedback rather than (predefined) rewards. This way, the agent learns from observations and human knowledge alike. 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 [103]. This enables the learning of reward functions. Expert knowledge can also be incorporated through natural language interfaces[99]. Here, a human provides instructions and agents receive rewards upon completing these instructions.

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 [126]. 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.

Visual analytics combines analysis techniques and interactive visual interfaces to enable exploration of –and inference from– data [127]. Machine learning is increasingly combined with visual analytics. For example, visual analytics systems allow users to drag similar data points closer in order to learn distance functions [105], provide corrective feedback in object recognition [107], or even to alter correctly identified instances where the interpretation is not in line with human explanations [108], [109].

Lastly, various tools exist for text analysis, in particular for topic modeling [97] where users can create, merge and refine topics or change keyword weights. They thus impart 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 [104]. 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.

6 Historical Background

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 [128] 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 used together with training data to obtain knowledge-based artificial neural networks [53]. 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. Especially the discipline of data assimilation deals with techniques that combine statistical and mechanistic models to improve prediction accuracy [129], [130].

7 Discussion of Challenges and Directions

Our findings about the main approaches of informed machine learning are summarized in Table 4. It gives for each approach the taxonomy path, its main motivation, the central approach idea, remarks to potential challenges, and our viewpoint on current or future directions. For further details on the methods themselves and the corresponding papers, we refer to Section 5. In the following, we discuss

TABLE 4
Main Approaches of Informed Machine Learning

Taxonomy Path			Main Motivation	Central Approach Idea	Potential Challenge	Current / Future Directions
Source	Represent.	Integration				
Scientific Knowl.	Algebraic Equations (See Sec. 5.1)	Learning Algor.	Less data, Knowl. conform.	Knowledge-based loss terms from constraints (see Insert 1)	Weighting supervision from data labels vs. knowledge	Hyperparameter setting, Novel learning algorithms, Extension of learning theory
	Differential Equations (See Sec. 5.2)	Learning Algor.	Knowl. conform., Less data	Physics-informed neural networks with derivatives in loss function	Solution robustness, Real-time data generation and integration	Uncertainty quantification, Numerical solver comparison, Online learning, data assimilation
	Simulation Results (See Sec. 5.3)	Training Data	Less data	Synthetic data generation or data augmentation (see Insert 2)	Sim-to-real gap, i.e. mismatch between real and simulated data	Adversarial domain adaptation, Domain randomization; Hybrid systems
World Knowl.	Spatial Invariances (See Sec. 5.4)	Hypoth. Set	Performance (Small models)	Models with invariant characteristics, e.g. group equivariant DNNs/CNNs	Invariance specification, expensive geometric evaluations	Geometric-based representation learning, Adaptaion to complex scenarios
	Logic Rules (See Sec. 5.5)	Hypoth. Set	Performance	KBANNs (see Insert 3); SRL (e.g., Markov logic networks, prob. soft logic)	Feasibility for deep neural networks; Acquisition of rules	Automated integration interface, Neuro-symbolic systems; Structure learning
	Knowl. Graphs (See Sec. 5.6)	Hypoth. Set	Performance, Less data	Gr. propagation (see Insert 4), attention; Gr. neural networks (relational inductive bias)	Comparability with custom graphs, Getting the graph, Entity linking	Standardized graph data pool, Combine graph using and learning, Neuro-symbolic systems
Expert Knowl.	Probabilistic Relations (See Sec. 5.7)	Hypoth. Set	Less data	Informed structure of prob. graphical models, informative priors	High computational effort, Formalization of knowledge	Variational methods combining prob. models with numerical opt., Probabilistic neural networks
	Human Feedback (See Sec. 5.8)	Learning Algor.	Less data, Performance, Interpretability	HITL Reinforcement learning; Explanation alignment via Visual anal./interactive ml	Feedback latency; Formalization of intuition, Evaluation methods	Reward estimation from logs; Representation transformation, Utilization for interpretability

The approaches are sorted by taxonomy path and knowledge representation. Methodical details can be found in Section 5. Challenges and directions are discussed in Section 7.

the challenges and directions for these main approaches, sorted by the integrated knowledge representations.

Prior knowledge in the form of algebraic equations can be integrated as constraints via knowledge-based loss terms (e.g., [12], [13], [35]). Here, we see a potential challenge in finding the right weights for supervision from knowledge versus data labels. Currently, this is solved by setting the hyperparameters for the individual loss terms [12]. However, we think that strategies from more recently developed learning algorithms, such as self-supervised [131] or few-shot learning [132], could also advance the supervision from prior knowledge. Moreover, we suggest further research on theoretical concepts based on the existing generalization bounds from statistical learning theory [133], [134] and the connection between regularization and effective hypothesis space [135].

Differential equations can be integrated similarly, but with a specific focus on physics-informed neural networks that constrain the model derivatives by the underlying differential equation (e.g., [20], [45], [46]). A potential challenge is the robustness of the solution, which is the subject of current research. One approach is to investigate the the model quality by a suitable quanitification of its uncertainty [43], [46]. We think, a more in-depth comparison with existing numerical solvers [136] would also be helpful. Another challenge of physical systems is the generation and integration of sensor data in real-time. This is currently tackled by online learning methods [48]. Furthermore, we think that techniques from data assimilation [130] could also be helpful to combine modelling from knowledge and data.

Simulation results can be used for synthetic data generation or augmentation (e.g., [18], [19], [59]), but this can bring up the challenge of a mismatch between real and simulated data. A promising direction to close the gap is domain adaptation, especially adversarial training [67], [137], or domain randomization [138]. Moreover, for future work we see further potential in the development of new hybrid systems that combine machine learning and simulation in more sophisticated ways [139].

The utilization of spatial invariances through model architectures with invariant characteristics, such as group equivariant or convolutional networks, diminish the model search space (e.g., [70], [71], [75]). Here, a potential challenge is the proper invariance specification and implementation [75] or expensive evaluations on more complex geometries [111]. Therefore, we think that the efficient adaptation of invariant-based models to further scenarios can further improve geometric-based representation learning [111].

Logic rules can be encoded in the architecture of knowledge-based neural networks (KBANNs), (e.g., [53], [54], [89]). Since this idea was already developed when neural networks had only a few layers, a question is, if it is still feasible for deep neural networks. In order to improve the practicality, we suggest to develop automated interfaces for knowledge integration. A future direction could be the development of new neuro-symbolic systems. Although the combination of connectionist and symbolic systems into hybrid systems is a longtime idea [140], [141], it is currently getting more attention [142], [143]. Another challenge, especially in statistical relational learning (SRL), such as Markov logic networks or probabilistic soft logic (e.g., [78], [91], [144]). is the aquisition of rules when they are not yet given. An ongoing research topic to this end is the learning of rules from data, which is called structure learning [145].

Knowledge graphs can be integrated into learning systems either explicitly via graph propagation and attention mechanisms, or implicitly via graph neural networks with relational inductive bias (e.g., [14], [15], [16]). A challenge is the comparability between different methods, because authors often use template like ConceptNet [79] or VisualGenome [15], [16] and customize the graphs in to improve running time and performance. Since the choice of graph can have high influence [81], we suggest a pool of standardized graphs in order to improve comparability, or even to establish benchmarks. Another interesting direction is to combine graph using and graph learning. A requirement here is the need for good entity linking models in approaches such as KnowBERT [94] and ERNIE [86] and the continuous embedding of new facts in the graph.

Probabilistic Relations can be integrated as prior knowledge in terms of a-priori probability distributions that are refined with additional observations (e.g., [73], [96], [100]). The main challenges are the large computational effort and the formalization of knowledge in terms of inductive priors. Directions responding to this are variational methods with origins in optimization theory and functional analysis [146] and variational neural networks [147]. Besides scaling issues, an explicit treatment of causality is becoming more important in machine learning and closely related to graphical probabilistic models [148].

Human feedback can be integrated into the learning algorithm by human-in-the-loop (HITL) reinforcement learning (e.g., [98], [103]), or by explanation alignment through interactive learning combined with visual analytics (e.g., [108], [109]). However, the exploration of human feedback can be very expensive due to its latency in real systems. Exploratory actions could hamper user experience [149], [150], so that online reinforcement learning is generally avoided. A promising approach is learning a reward estimator [151], [152] from collected logs, which then provides unlimited feedback for unseen instances that do not have any human judgments. Another challenge is that human feedback is often intuitive and not formalized and thus difficult to incorporate into machine learning systems. Also human-gorunded evaluation is very costly, especially compared to functionally-grounded evaluation [153]. Therefore we suggest to further study representation transformations to formalize intuitive knowledge, e.g., from human feedback to logical rules. Furthermore, we found that improved interpretability still only is a minor goal for knowledge integration (see Fig. 4). This, too, suggests opportunities for future work.

Even if these directions are motivated by specific approaches, we think that they are generally relevant and can advance the whole field of informed machine learning.

8 Conclusion

In this paper, we presented a unified classification framework for the explicit integration of additional prior knowledge into machine learning, which we described using the umbrella term of *informed machine learning*. Our main contribution is the development of a taxonomy that allows a structured categorization of approaches and the uncovering of main paths. Moreover, we presented a conceptual clarification of informed machine learning, as well as a systematic and comprehensive research survey. This helps current and future users of informed machine learning to identify the right methods to use their prior knowledge, for example, to deal with insufficient training data or to make their models more robust.

Acknowledgments

The authors would like to thank Dorina Weichert, Daniel Paurat, Lars Hillebrand, Theresa Bick, and Nico Piatkowski for helpful discussions. This work was a joint effort of the Fraunhofer Research Center for Machine Learning (RCML) within the Fraunhofer Cluster of Excellence Cognitive Internet Technologies (CCIT) and the Competence Center for Machine Learning Rhine Ruhr (ML2R). This work was supported by the Federal Ministry of Education and Research of Germany under Grant 01|S18038B. All authors are with the Fraunhofer Center for Machine Learning.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Neural Inf. Process. Syst.*, 2012, pp. 84–90.

[2] G. Hinton *et al.*, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

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

[4] D. Silver *et al.*, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 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, pp. 547–555, 2018.

[6] T. Ching *et al.*, "Opportunities and obstacles for deep learning in biology and medicine," *J. Roy. Soc. Interface*, vol. 15, no. 141, 2018, Art. no. 20170387.

[7] J. N. Kutz, "Deep learning in fluid dynamics," *J. Fluid Mechanics*, vol. 814, pp. 1–4, 2017.

[8] M. Brundage *et al.*, "Toward trustworthy AI development: Mechanisms for supporting verifiable claims," 2020, *arXiv:2004.07213*.

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

[10] M. Diligenti, S. Roychowdhury, and M. Gori, "Integrating prior knowledge into deep learning," in *Proc. Int. Conf. Mach. Learn. Appl.* 2017, pp. 920–923.

[11] 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*.

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

[13] R. Stewart and S. Ermon, "Label-free supervision of neural networks with physics and domain knowledge," in *Proc. Conf. Artif. Intell.*, 2017, pp. 2576–2582.

[14] P. Battaglia *et al.*, "Interaction networks for learning about objects, relations and physics," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 4509–4517.

[15] K. Marino, R. Salakhutdinov, and A. Gupta, "The more you know: Using knowledge graphs for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*. 2017, pp. 20–28.

[16] C. Jiang, H. Xu, X. Liang, and L. Lin, "Hybrid knowledge routed modules for large-scale object detection," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2018, 1559–1570.

[17] A. Cully, J. Clune, D. Tarapore, and J.-B. Mouret, "Robots that can adapt like animals," *Nature*, vol. 521, no. 7553, pp. 503–507, 2015.

[18] K.-H. Lee, J. Li, A. Gaidon, and G. Ros, "Spigan: Privileged adversarial learning from simulation," in *Proc. Int. Conf. Learn. Representations*, 2019.

[19] 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, pp. 426–431, 2018.

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

[21] M. Diligenti, M. Gori, and C. Sacca, "Semantic-based regularization for learning and inference," *Artif. Intell.*, vol. 244, pp. 143–165, 2017.

[22] A. Karpatne *et al.*, "Theory-guided data science: A new paradigm for scientific discovery from data," *Trans. Knowl. Data Eng.*, vol. 29, no. 10, pp. 2318–2331, 2017.

[23] F. Lauer and G. Bloch, "Incorporating prior knowledge in support vector machines for classification: A review," *Neurocomputing*, vol. 71, no. 7–9, pp. 1578–1594, 2008.

[24] P. W. Battaglia *et al.*, "Relational inductive biases, deep learning, and graph networks," 2018, *arXiv:1806.01261*.

[25] M. Steup, "Epistemiology," in *The Stanford Encyclopedia of Philosophy (Winter 2012 Edition)*, Edward N. Zalta (ed.), 2018. [Online]. Available: https://stanford.library.sydney.edu.au/archives/win2018/entries/epistemology/.

[26] L. Zagzebski, *What is Knowledge?* Hoboken, NJ, USA: Wiley, 2017.

[27] P. Machamer and M. Silberstein, *The Blackwell Guide to the Philosophy of Science*. Hoboken, NJ, USA: Wiley, 2008, vol. 19.

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

[29] D. Kahneman, *Thinking, Fast and Slow*. New York, NY, USA: Macmillan, 2011.

[30] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman, "Building machines that learn and think like people," *Behav. Brain Sci.*, vol. 40, 2017.

[31] H. G. Gauch, *Scientific Method in Practice*. Cambridge, U.K.: Cambridge University Press, 2003.

[32] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin, *Learning From Data*. USA: AMLBook, 2012.

[33] N. Muralidhar, M. R. Islam, M. Marwah, A. Karpatne, and N. Ramakrishnan, "Incorporating prior domain knowledge into deep neural networks," in *Proc. Int. Conf. Big Data*, 2018, pp. 36–45.

[34] Y. Lu, M. Rajora, P. Zou, and S. Liang, "Physics-embedded machine learning: Case study with electrochemical micromachining," *Machines*, vol. 5, no. 1, 2017, Art. no. 4.

[35] R. Heese, M. Walczak, L. Morand, D. Helm, and M. Bortz, "The good, the bad and the ugly: Augmenting a black-box model with expert knowledge," in *Proc. Int. Conf. Artif. Neural Netw.*, 2019, pp. 391–395.

[36] G. M. Fung, O. L. Mangasarian, and J. W. Shavlik, "Knowledge-based support vector machine classifiers," in *Proc. 15th Int. Conf. Neural Inf. Process. Syst.*, 2003, pp. 537–544.

[37] O. L. Mangasarian and E. W. Wild, "Nonlinear knowledge-based classification," *IEEE Trans. Neural Netw.*, vol. 19, no. 10, pp. 1826–1832, Oct. 2008.

[38] R. Ramamurthy, C. Bauckhage, R. Sifa, J. Schücker, and S. Wrobel, "Leveraging domain knowledge for reinforcement learning using MMC architectures," in *Proc. Int. Conf. Artifi. Neural Netw.*, 2019, 595–607.

[39] M. von Kurnatowski, J. Schmid, P. Link, R. Zache, L. Morand, T. Kraft, I. Schmidt, and A. Stoll, "Compensating data shortages in manufacturing with monotonicity knowledge," 2020, *arXiv:2010.15955*.

[40] C. Bauckhage, C. Ojeda, J. Schücker, R. Sifa, and S. Wrobel, "Informed machine learning through functional composition." *In LWDA*, pp. 33–37, 2018.

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

[42] S. Jeong, B. Solenthaler, M. Pollefeys, M. Gross *et al.*, "Data-driven fluid simulations using regression forests," *ACM Trans. Graph.*, vol. 34, no. 6, 2015, Art. no. 199.

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

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

[45] I. E. Lagaris, A. Likas, and D. I. Fotiadis, "Artificial neural networks for solving ordinary and partial differential equations," *IEEE Trans. Neural Netw.*, vol. 9, no. 5, pp. 987–1000, Sep. 1998.

[46] 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. Comput. Phys.*, vol. 394, pp. 56–81, 2019.

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

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

[49] 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 *Proc. Neural Inf. Process. Syst.*, 2018, pp. 7178–7189.

[50] J. Ling, A. Kurzawski, and J. Templeton, "Reynolds averaged turbulence modelling using deep neural networks with embedded invariance," *J. Fluid Mechanics*, vol. 807, pp. 155 –166, 2016.

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

[52] J.-L. Wu, H. Xiao, and E. Paterson, "Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework," *Phys. Rev. Fluids*, vol. 3, no. 7, 2018, Art. no. 074602.

[53] G. G. Towell and J. W. Shavlik, "Knowledge-based artificial neural networks," *Artif. Intell.*, vol. 70, no. 1–2, pp. 119–165, 1994.

[54] A. S. d. Garcez and G. Zaverucha, "The connectionist inductive learning and logic programming system," *Appl. Intell.*, vol. 11, no. 1, pp. 59–77, 1999.

[55] T. Ma and A. Zhang, "Multi-view factorization autoencoder with network constraints for multi-omic integrative analysis," in *Proc. Int. Conf. Bioinform. Biomed.* 2018, pp. 702–707.

[56] Z. Che, D. Kale, W. Li, M. T. Bahadori, and Y. Liu, "Deep computational phenotyping," in *Proc. Int. Conf. Knowl. Discov. Data Mining*, 2015, pp. 507–516.

[57] M. B. Messaoud, P. Leray, and N. B. Amor, "Integrating ontological knowledge for iterative causal discovery and visualization," in *Proc. Eur. Conf. Symbolic Quantitative Approaches Reasoning Uncertainty*. Springer, 2009, pp. 168–179.

[58] G. Borboudakis and I. Tsamardinos, "Incorporating causal prior knowledge as path-constraints in Bayesian networks and maximal ancestral graphs," 2021, *arXiv:1206.6390*.

[59] T. Deist, A. Patti, Z. Wang, D. Krane, T. Sorenson, and D. Craft, "Simulation assisted machine learning." *Bioinformatics*, vol. 35, no. 20, pp. 4072–4080, 2019.

[60] 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. Mach. Tools Manuf.*, vol. 47, no. 2, pp. 211–222, 2007.

[61] 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," *Chem. Mater.*, vol. 22, no. 12, pp. 3762–3767, 2010.

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

[63] 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. Knowl. Discov. Data Mining*, 2017, pp. 787–795.

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

[65] A. Rai, R. Antonova, F. Meier, and C. G. Atkeson, "Using simulation to improve sample-efficiency of Bayesian optimization for bipedal robots." *J. Machine Learn. Res.*, vol. 20, no. 49, pp. 1–24, 2019.

[66] Y. Du *et al.*, "Learning to exploit stability for 3D scene parsing," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 1733–1743.

[67] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proc IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2242–2251.

[68] F. Wang and Q.-J. Zhang, "Knowledge-based neural models for microwave design," *Trans. Microwave Theory Techn.*, vol. 45, no. 12, pp. 2333–2343, Dec. 1997.

[69] S. J. Leary, A. Bhaskar, and A. J. Keane, "A knowledge-based approach to response surface modelling in multifidelity optimization," *J. Global Optim.*, vol. 26, no. 3, pp. 297–319, 2003.

[70] T. S. Cohen and M. Welling, "Group equivariant convolutional networks," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 2990–2999.

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

[72] B. Schölkopf, P. Simard, A. J. Smola, and V. Vapnik, "Prior knowledge in support vector kernels," in *Proc. Conf. Adv. Neural Inf. Process. Syst.*, 1998, 640–646.

[73] 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. Biomed. Inform.*, vol. 52, pp. 373–385, 2014.

[74] J. Li, Z. Yang, H. Liu, and D. Cai, "Deep rotation equivariant network," *Neurocomputing*, vol. 290, pp. 26–33, 2018.

[75] D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow, "Harmonic networks: Deep translation and rotation equivariance," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5028–5037.

[76] P. Niyogi, F. Girosi, and T. Poggio, "Incorporating prior information in machine learning by creating virtual examples," *Proc. IEEE*, vol. 86, no. 11, pp. 2196–2209, Nov. 1998.

[77] M. Schiegg, M. Neumann, and K. Kersting, "Markov logic mixtures of gaussian processes: Towards machines reading regression data," in *Proc. Artif. Intell. Statist.*, 2012, pp. 1002–1011.

[78] 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 *Neural Inf. Process. Syst.*, 2018, pp. 140–151.

[79] 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. Artif. Intell.*, 2018, pp. 4623–4629.

[80] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," in *Proc. Assoc. Comput. Linguistics, Int. Joint Conf. Natural Lang. Process.*, 2009, pp. 1003–1011.

[81] X. Liang, Z. Hu, H. Zhang, L. Lin, and E. P. Xing, "Symbolic graph reasoning meets convolutions," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 1858–1868.

[82] D. L. Bergman, "Symmetry constrained machine learning," in *Proc. SAI Intelligent Systems Conf.* 2019, pp 501–512.

[83] M.-W. Chang, L. Ratinov, and D. Roth, "Guiding semi-supervision with constraint-driven learning," in *Association for Computat. Linguistics*, 2007, pp. 280–287.

[84] Z. Hu, Z. Yang, R. Salakhutdinov, and E. Xing, "Deep neural networks with massive learned knowledge," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 1670–1679.

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

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

[87] N. Mrk ši *et al.*, "Counter-fitting word vectors to linguistic constraints," 2016, *arXiv:1603.00892*.

[88] J. Bian, B. Gao, and T.-Y. Liu, "Knowledge-powered deep learning for word embedding," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, 2014, pp. 132–148.

[89] M. V. França, G. Zaverucha, and A. S. d. Garcez, "Fast relational learning using bottom clause propositionalization with artificial neural networks," *Mach. Learn.*, vol. 94, no. 1, pp. 81–104, 2014.

[90] M. Richardson and P. Domingos, "Markov logic networks," *Mach. Learn.*, vol. 62, no. 1–2, pp. 107–136, 2006.

[91] A. Kimmig, S. Bach, M. Broecheler, B. Huang, and L. Getoor, "A short introduction to probabilistic soft logic," in *Proc. NIPS Workshop Probabilistic Program: Found. Appl.*, 2012, pp. 1–4.

[92] G. Glavaš and I. Vulić, "Explicit retrofitting of distributional word vectors," in *Proc. Assoc. Comput. Linguistics*, 2018, 34–45.

[93] Y. Fang, K. Kuan, J. Lin, C. Tan, and V. Chandrasekhar, "Object detection meets knowledge graphs," *Int. Joint Conf. Artif. Intell.* 2017, pp. 1661–1667.

[94] M. E. Peters *et al.*, "Knowledge enhanced contextual word representations," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP), Int. Joint Conf. Nat. Lang. Process.*, 2019, pp. 43–54.

[95] L. M. de Campos and J. G. Castellano, "Bayesian network learning algorithms using structural restrictions," *Int. J. Approx. Reasoning*, vol. 45, no. 2, pp. 233–254, 2007.

[96] 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 Syst. Appl.*, vol. 56, pp. 197–208, 2016.

[97] J. Choo, C. Lee, C. K. Reddy, and H. Park, "Utopian: User-driven topic modeling based on interactive nonnegative matrix factorization," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 1992–2001, Dec. 2013.

[98] W. B. Knox and P. Stone, "Interactively shaping agents via human reinforcement: The tamer framework," in *Proc. Int. Conf. Knowl. Capture (K-CAP)*2009, pp. 9–16.

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

[100] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning Bayesian networks: The combination of knowledge and statistical data," *Mach. Learn.*, vol. 20, no. 3, pp. 197–243, 1995.

[101] M. Richardson and P. Domingos, "Learning with knowledge from multiple experts," in *Proc. Int. Conf. Mach. Learn.*, 2003, pp. 624–631.

[102] A. Feelders and L. C. Van der Gaag, "Learning Bayesian network parameters under order constraints," *Int. J. Approx. Reasoning*, vol. 42, no. 1–2, 2006, pp. 37–53.

[103] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," in *Proc. Neural Inf. Process. Syst.*, 2017, pp. 4302–4310.

[104] T. Hester *et al.*, "Deep q-learning from demonstrations," in *Proc. Conf. Artif. Intell.*, 2018, PP. 3223–3230.

[105] E. T. Brown, J. Liu, C. E. Brodley, and R. Chang, "Dis-function: Learning distance functions interactively," in *Proc. Conf. Visual Analytics Sci. Technol.*, 2012, pp. 83–92.

[106] B. Yet, Z. Perkins, N. Fenton, N. Tai, and W. Marsh, "Not just data: A method for improving prediction with knowledge," *J. Biomed. Inf.*, vol. 48, 2014, pp. 28–37.

[107] J. A. Fails and D. R. Olsen Jr, "Interactive machine learning," in *Proc. Int. Conf. Intell. User Interfaces*, 2003, pp. 39–45.

[108] L. Rieger, C. Singh, W. J. Murdoch, and B. Yu, "Interpretations are useful: Penalizing explanations to align neural networks with prior knowledge," 2019, *arXiv:1909.13584*.

[109] P. Schramowski *et al.*, "Right for the wrong scientific reasons: Revising deep networks by interacting with their explanations," 2020, *arXiv:2001.05371*.

[110] L. von Rueden, T. Wirtz, F. Hueger, J. D. Schneider, N. Piatkowski, and C. Bauckhage, "Street-map based validation of semantic segmentation in autonomous driving," in *Proc. Int. Conf. Pattern Recognit.*, 2021, pp. 10203–10210.

[111] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: Going beyond euclidean data," *IEEE Signal Process. Mag.*, vol. 34, no. 4, 2017, pp. 18–42.

[112] M.-W. Chang, L. Ratinov, and D. Roth, "Structured learning with constrained conditional models," *Mach. Learn.*, vol. 88, no. 3, 2012, pp. 399–431.

[113] D. Sridhar, J. Foulds, B. Huang, L. Getoor, and M. Walker, "Joint models of disagreement and stance in online debate," in *Proc. Assoc. Comput. Linguistics Int. Joint Conf. Nat. Lang. Process.*, 2015, pp. 116–125.

[114] A. S. 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," 2019, *arXiv:1905.06088*.

[115] L. D. Raedt, K. Kersting, and S. Natarajan, *Statistical Relational Artificial Intelligence: Logic, Probability, and Computation*. San Rafael, CA, USA: Morgan & Claypool, 2016.

[116] R. Speer and C. Havasi, "Conceptnet 5: A large semantic network for relational knowledge," in *Proc. People's Web Meets NLP*, 2013, pp. 161–176.
[117] G. A. Miller, "WordNet: A lexical database for english," *Communications ACM*, vol. 38, no. 11, 1995, pp. 39–41.
[118] T. Mitchell *et al.*, "Never-ending learning," *Communications ACM*, vol. 61, no. 5, 2018, pp. 103–115.
[119] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[120] Z.-X. Ye and Z.-H. Ling, "Distant supervision relation extraction with intra-bag and inter-bag attentions," 2019, *arXiv:1904.00143*.
[121] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, *arXiv:1301.3781*.
[122] M. S. Massa, M. Chiogna, and C. Romualdi, "Gene set analysis exploiting the topology of a pathway," *BMC Syst. Biol.*, vol. 4, no. 1, 2010, Art. no. 121.
[123] N. Angelopoulos and J. Cussens, "Bayesian learning of Bayesian networks with informative priors," *Ann. Math. Artif. Intell.*, vol. 54, no. 1–3, 2008, pp. 53–98.
[124] N. Piatkowski, S. Lee, and K. Morik, "Spatio-temporal random fields: Compressible representation and distributed estimation," *Mach. Learn.*, vol. 93, no. 1, 2013, pp. 115–139.
[125] R. Fischer, N. Piatkowski, C. Pelletier, G. I. Webb, F. Petitjean, and K. Morik, "No cloud on the horizon: Probabilistic gap filling in satellite image series," in *Proc. Int. Conf. Data Sci. Adv. Anal.*, 2020, pp. 546–555.
[126] B. Settles, "Active learning literature survey," Comput. Sci., Univ. Wisconsin–Madison, Madison, WI, USA, Tech. Rep. 1648, 2009.
[127] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon, "Visual analytics: Definition, process, and challenges," in *Proc. Inf. Visual.*, 2008, pp. 154–175.
[128] M. L. Minsky, "Logical versus analogical or symbolic versus connectionist or neat versus scruffy," *AI Mag.*, vol. 12, no. 2, pp. 34–51, 1991.
[129] E. Kalnay, *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge, U.K: Cambridge University Press, 2003.
[130] S. Reich and C. Cotter, *Probabilistic Forecasting and Bayesian Data Assimilation*. Cambridge, U.K: Cambridge Univ. Press, 2015.
[131] M. Janner, J. Wu, T. D. Kulkarni, I. Yildirim, and J. B. Tenenbaum, "Self-supervised intrinsic image decomposition," in *Proc. Neural Inf. Process. Syst.*, 2017, pp. 5938–5948.
[132] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Comput. Surveys*, vol. 53, no. 3, 2020, Art no. 63.
[133] F. Cucker and D. X. Zhou, *Learning Theory: An Approximation Theory Viewpoint*. Cambridge, U.K.: Cambridge Univ. Press, 2007.
[134] I. Steinwart and A. Christmann, *Support Vector Machines*. Germany: Springer, 2008.
[135] F. Cucker and S. Smale, "Best choices for regularization parameters in learning theory: On the bias-variance problem," *Proc. Found. Comput. Math.*, vol. 2, no. 4, 2002, pp. 413–428.
[136] L. Lapidus and G. F. Pinder, *Numerical Solution of Partial Differential Equations in Science and Engineering*. Hoboken, NJ, USA: Wiley, 2011.
[137] M. Wulfmeier, A. Bewley, and I. Posner, "Addressing appearance change in outdoor robotics with adversarial domain adaptation," in *Proc. Int. Conf. Intell. Robots Syst.*, 2017, pp. 1551–1558.
[138] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 3803–03810.
[139] L. von Rueden, S. Mayer, R. Sifa, C. Bauckhage, and J. Garcke, "Combining machine learning and simulation to a hybrid modelling approach: Current and future directions," in *Proc. Int. Symp. Intell. Data Anal.*, 2020, pp. 548–560.
[140] K. McGarry, S. Wermter, and J. MacIntyre, "Hybrid neural systems: From simple coupling to fully integrated neural networks," *Neural Comput. Surv.*, vol. 2, no. 1, 1999, pp. 62–93.
[141] R. Sun, "Connectionist implementationalism and hybrid systems," Encyclopedia of Cognitive Science. Hoboken, NJ, USA: Wiley, 2006.
[142] A. S. d. Garcez and L. C. Lamb, "Neurosymbolic AI: The 3rd wave," 2020, *arXiv:2012.05876*.
[143] T. Dong *et al.* "Imposing category trees onto word-embeddings using a geometric construction," in *Proc. Int. Conf. Learn. Representations*, 2018.
[144] S. H. Bach, M. Broecheler, B. Huang, and L. Getoor, "Hinge-loss Markov random fields and probabilistic soft logic," 2015, *arXiv:1505.04406*.
[145] V. Embar, D. Sridhar, G. Farnadi, and L. Getoor, "Scalable structure learning for probabilistic soft logic," 2018, *arXiv:1807.00973*.
[146] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, "Variational inference: A review for statisticians," *J. Amer. Stat. Assoc.*, vol. 112, no. 518, 2017, pp. 859–877.
[147] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," 2019, *arXiv:1906.02691*.
[148] J. Pearl, *Causality*. Cambridge, U.K.: Cambridge Univ. Press, 2009.
[149] J. Kreutzer, S. Riezler, and C. Lawrence, "Learning from human feedback: Challenges for real-world reinforcement learning in NLP," 2020, *arXiv:2011.02511*.
[150] G. Dulac-Arnold, D. Mankowitz, and T. Hester, "Challenges of real-world reinforcement learning," 2019, *arXiv:1904.12901*.
[151] J. Kreutzer, J. Uyheng, and S. Riezler, "Reliability and learnability of human bandit feedback for sequence-to-sequence reinforcement learning," in *Proc. Assoc. Computat. Linguistics*, 2018, pp. 1777–1788.
[152] Y. Gao, C. M. Meyer, and I. Gurevych, "April: Interactively learning to summarise by combining active preference learning and reinforcement learning," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 4120–4130.
[153] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," 2017, *arXiv:1702.08608*.

Laura von Rueden received the BSc degree in physics and the MSc degree in simulation sciences in 2015 from RWTH Aachen University. She was a data scientist with Capgemini. Since 2018, she has been a research scientist with Fraunhofer IAIS. She is currently working toward the PhD degree in computer science with the Universität Bonn. Her research interests include machine learning and especially the combination of data and knowledge-based modeling.

Sebastian Mayer received the diploma degree inmathematics from TU Darmstadt, in 2011, and the PhD degree in mathematics from University Bonn, in 2018. Since 2017, he has been a research scientist with Fraunhofer SCAI. His research interests include machine learning and biologically-inspired algorithms in the context of cyberphysical systems.

Katharina Beckh received the MSc degree in human-computer interaction from the Julius Maximilian University of Wuerzburg in 2019. Since 2019, she has been a research scientist with 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 the PhD degree in mathematics from Max-Planck-Institute and Bonn University in 2018. Since 2018, he has been a research scientist with Fraunhofer IAIS. His current research interests include aspects of learning theory such as generalization or compression bounds, geometric learning, and quantum computing.

Sven Giesselbach received the MSc degree in computer science from the University of Bonn in 2012. Since 2015, he has been a data scientist with Fraunhofer IAIS and is also lead of the team natural language understanding with the department knowledge discovery. His research interest includes the use of external knowledge in natural language processing.

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

Birgit Kirsch received the MSc degree in business informatics from Hochschule Trier in 2017. Since 2017, she has been a research scientist with Fraunhofer IAIS. Her research interests include natural language processing and statistical relational learning.

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

Julius Pfrommer received the PhD degree in computer science from the Karlsruhe Institute of Technology in 2019. Since 2018, he has been the head of a research group with 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 the MSc. degree in computer science from the University of Bonn in 2018. Since 2019, she has been a data scientist with Fraunhofer IAIS. Her research interests include learning from healthcare data and pattern mining.

Rajkumar Ramamurthy received the MSc degree in media informatics from RWTH Aachen University in 2016. Since 2018, he has been a data scientist with Fraunhofer IAIS. He is currently working toward the Phd degree with the University of Bonn. His research interests include reinforcement learning and natural language processing.

Jochen Garcke received the diploma and PhD degrees in mathematics from the Universität Bonn, in 1999 and 2004, respectively. From 2004 to 2006, he was a postdoctoral fellow with the Australian National University. He was a postdoctoral researcher from 2006 to 2008 and a Junior Research Group leader from 2008 to 2011, with the Technical University Berlin. Since 2011, he has been professor of numerics with the University of Bonn and department head with Fraunhofer SCAI, Sankt Augustin. His research interests include machine learning, scientific computing, reinforcement learning, and highdimensional approximation. He is currently a member of DMV, GAMM, and SIAM. He is currently a reviewer for the *IEEE Transactions on Industrial Informatics*, *the IEEE Transactions on Neural Networks*, and *the IEEE Transactions on Pattern Analysis and Machine Intelligence.*

Christian Bauckhage (Member, IEEE) received the MSc and PhD degrees in computer science from Bielefeld University, in 1998 and 2002, respectively. Since 2008, he has been a professor of computer science with the University of Bonn and lead scientist for machine learning with Fraunhofer IAIS. He was with the Centre for Vision Research, Toronto, Canada, and a senior research scientist with Deutsche Telekom Laboratories, Berlin. His research interests include theory and practice of learning systems and next generation computing. He is currently reviewer 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.* He is currently an associate editor for *the IEEE Transactions on Games*.

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