

REVIEW ARTICLE OPEN



Explainable machine learning in materials science

Xiaoting Zhong¹✉, Brian Gallagher², Shusen Liu², Bhavya Kailkhura², Anna Hiszpanski¹ and T. Yong-Jin Han¹✉

Machine learning models are increasingly used in materials studies because of their exceptional accuracy. However, the most accurate machine learning models are usually difficult to explain. Remedies to this problem lie in explainable artificial intelligence (XAI), an emerging research field that addresses the explainability of complicated machine learning models like deep neural networks (DNNs). This article attempts to provide an entry point to XAI for materials scientists. Concepts are defined to clarify what explain means in the context of materials science. Example works are reviewed to show how XAI helps materials science research. Challenges and opportunities are also discussed.

npj Computational Materials (2022)8:204; <https://doi.org/10.1038/s41524-022-00884-7>

INTRODUCTION

Traditional materials science studies depend heavily on the knowledge of individual experts. Expert knowledge is highly useful, especially for advancing physical understanding and generating new scientific hypotheses. However, the traditional expert knowledge approach has clear limitations in terms of prediction accuracy and efficiency. Progress in technology often requires new materials with specific properties, but it can take a long research cycle to accumulate enough knowledge about one material system. Advances in experimental and computational tools are producing data with increased volume, speed, and complexity^{1–3}, but the amount of information a human expert can process is limited. New research tools are needed to design better materials at a faster speed.

One promising tool that attracts an increasing amount of research interest is machine learning (ML). ML models are efficient compared to human individuals. They follow a data-driven approach and can analyze a large amount of data without requiring profound domain knowledge or ingenious domain insights. ML models are flexible. Both traditional materials state variables (e.g., temperature and pressure) and raw materials characterization data (e.g., spectrum and image) can serve as ML model inputs. ML models are also accurate. They have shown exceptional prediction accuracy for various material properties at different scales. Examples include atomic properties like potential energy^{4,5} and crystal structure^{6,7}, microscopic properties like strain distribution⁸, and macroscopic properties like mechanical compressive strength⁹, electronic conductivity¹⁰, and thermal stability¹¹.

Modern ML methods are accurate, but their great predictive power often comes at the price of explainability. There is usually a tradeoff between model accuracy and model explainability^{12,13}. The most accurate ML models (e.g., deep neural networks, or DNNs) are usually difficult to explain and are often known as black boxes. This lack of explainability has restrained the usability of ML models in general scientific tasks, like understanding the hidden causal relationship, gaining actionable information, and generating new scientific hypotheses. Many materials scientists also find black-box ML models difficult to trust. After all, recent ML studies show that even the most advanced ML models are not always logically reliable^{14,15} and may show poor extrapolation performance. There is a general desire for ML explainability in the

materials science community. For example, many materials science machine learning studies evaluate some version of feature importance using different methods (e.g., filter, wrapper, embedded methods)^{16,17}. Feature importance explanations are useful, but traditional feature importance explanations are limited to tabular input data and are often inaccurate or inefficient. There have been many new ML explanation techniques in recent years.

Explainable Artificial Intelligence (XAI) is an emerging field in which the explainability of advanced machine learning models is intensively studied. The term XAI is brought up by DARPA in 2017 and has become popular in many fields like healthcare, transportation, legal, finance, military, and scientific research¹². Limited by the advanced in general AI, the current focus of XAI is the interpretability of ML models. In other words, at the core of XAI is a rich set of model explanation techniques that achieve explainability from different perspectives. Note that explainability is a highly general concept and means different things to different audiences. Concept definition is thus an important topic in XAI as proper definitions help establish the context of the discussion. Then XAI also addresses explanation evaluation since model explanations can be misleading.

There is an increasing interest to apply XAI in scientific studies^{18,19}. Some XAI techniques are also being applied in materials ML studies^{20–22} but XAI as a comprehensive field remains unfamiliar to the mainstream materials science community. Oviedo et al.²³ recently presented a materials science and chemistry XAI review, but their examples are heavy on chemistry and model evaluation is not discussed in detail. Here we present a systematic review of important XAI concepts and useful techniques with representative materials science application examples. By doing so, we hope to provide an entry point for materials scientists who desire explainability in addition to prediction accuracy when building ML models for material applications.

The paper is organized as follows. The background and concepts section discusses important XAI concepts and defines what constitutes a model explanation within the context of materials science. The machine learning explainability in materials science section reviews XAI techniques with recent materials science application examples. Many of the model explanation techniques we discuss work with image data and convolution neural networks (CNNs), but other data types (e.g., tabular data, spectral, and crystal graphs) and other ML models (e.g., multi-layer

¹Materials Science Division, Lawrence Livermore National Laboratory, Livermore, CA, USA. ²Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA, USA. ✉email: zhong2@llnl.gov; han5@llnl.gov

perceptrons, graph neural networks, and general ML models) are also covered. The explanation evaluation section discusses the necessity of the criteria for explanation evaluation. Finally, the challenges and opportunities section identifies some challenges and opportunities in applying XAI techniques for materials science understanding.

BACKGROUND AND CONCEPTS

Machine learning models have different explainability. Simple models like linear/logistic regression, decision trees, k-nearest neighbors, Bayesian models, rule-based learners, and general additive models are usually considered transparent and can be examined directly^{24,25}. On the other hand, complicated models like tree ensembles, supported vector machines, and neural networks are often considered as black boxes and are difficult to explain^{24,26}. Indeed, there is usually a trade-off between model complexity and model explainability^{25,27}. While high complexity is usually necessary for ML models to achieve high accuracy on difficult problems, the same complexity also poses challenges in explainability. Most examples discussed in this paper concern deep neural networks (DNNs), but the general principles described here apply to different ML models and explanation techniques.

Clarifying the scope of an explanation is important. Many modern ML models (e.g., DNNs) are often so complicated that a simple crisp explanation for the entire ML model is not possible with current technology. Luckily, one can aim to explain part of an ML model. Lipton²⁸ introduced three levels of model explainability: simulatability, decomposability, and algorithm explainability. Simulatability is achieved if the entire model (e.g., a simple linear regression model) is easily comprehensible to a human user. Decomposability is achieved if part of the model (e.g., model parameter) is explainable. Algorithm explainability is satisfied if the learning algorithm is simple to understand and always converges to the same unique solution.

An explanation can be post-hoc or ante-hoc. Post-hoc explanations provide decision-level explanations by referring to external data or proxy models. Post-hoc explanations usually take a practical perspective. Ante-hoc explanations address the overall working logic on a model level and usually take a theoretical perspective. In other words, ante-hoc explainability is intrinsic to the model of interest while post-hoc explainability depends on tools extrinsic to the model of interest (e.g., input data or surrogate models). Most explanation techniques we discuss in the next section (e.g., salience maps, feature importance, explanation by example, surrogate models, concept visualization, and transfer learning) belong to post-hoc explanations. Both ante-hoc and post-hoc explainability can be achieved on different levels: globally for the entire model and full input space, or locally for part of the model (e.g., functional form, parameters, calculations) or part of the input space (e.g., a given data instance).

We summarize different scopes of explainability in Fig. 1 and propose to define some basic XAI concepts for materials science applications in the following way. A model is *transparent* if all model components are readily understandable. A model is *intrinsically explainable* if part of the model (e.g., functional form, parameters, calculations) is explicitly understandable or physically grounded. One can *explain* a non-transparent model extrinsically by simplifying its working logic or providing (data) evidence to support its reasoning. In other words, explanations do not have to address the entire model as a whole and can focus on part of the model. Moreover, explanations do not have to address the original black-box model itself. Explanations can focus on other components in the same learning pipeline (e.g., input and learning process) or a simpler proxy model that behaves similarly to the original black-box model, though such explanations are not intrinsic to the original black box.

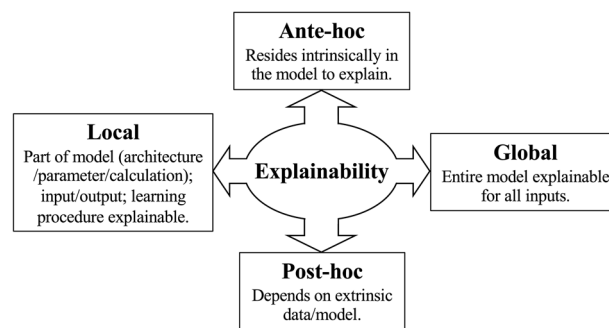


Fig. 1 Different scopes and aspects of explainability. Explainability is not a binary property. In other words, not all explanations address the ante-hoc global explainability of the entire model. Post-doc/local explanations are also acceptable.

The same concepts (transparent, explainable, explain) are sometimes defined differently in the literature. Indeed, no definition is accepted by all communities because explanations are abstract and mean different things to different users. Some XAI researchers differentiate interpret (interpretability) from explain (explainability) but we propose not to do so to avoid unnecessary jargon for materials scientists. Some alternative concept definitions (of explain, explainability, interpret, interpretability, transparent) are summarized in the Supplementary Material for readers who find the topic interesting.

Finally, it is probably essential to understand what characterizes a good explanation in general. Miller²⁹ summarized four important characteristics of good explanations, including (1) contrastive, meaning that counterfactual explanations which address not only why the model made decision X but also why the model did not make decision Y are useful; (2) selectivity, meaning that a good explanation should be simple and reveal only the main causes; (3) causal, meaning that probabilities are not as effective as causal links though probabilities may allow more accurate predictions; and (4) social, meaning that an explanation is a social interaction, in which the social convention and the people in interaction are important. Specifically for explanations in ML, Alvarez-Melis et al.³⁰ defined three requirements for explanations, including (1) fidelity, meaning that interpretable representations should preserve important information within the original data/model; (2) diversity, meaning that interpretable representations should be constructed from a few non-overlapping concepts; and (3) grounding, meaning that what claims to be interpretable should be readily human-understandable.

MACHINE LEARNING EXPLAINABILITY IN MATERIALS SCIENCE

This section introduces some representative XAI techniques with recent materials science application examples. An overview is presented in Fig. 2. The techniques and examples are organized according to whether they provide post-hoc explainability (explaining DNNs) or ante-hoc explainability (explainable DNNs). Post-hoc explanations are usually more handy as there are many off-the-shelf choices. Ante-hoc explainability often requires more significant model design efforts.

Figure 2 organization is inspired by the machine learning explainability taxonomy defined by Gilpin³¹, and is motivated by two fundamental questions: (1) 'Why does this particular input lead to that particular output?' (addressed by explaining model processing), and (2) 'What information does the network contain?' (addressed by explaining model representations)³¹. Note that different perspectives of explainability are not orthogonal and one explanation technique can belong to different leaves in Fig. 2^{24,25}. Nevertheless, we try to discuss the different perspectives separately following the map in Fig. 2.

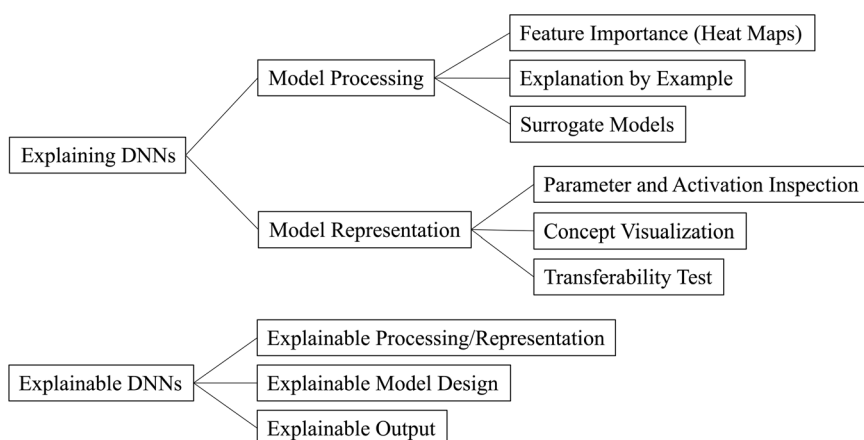


Fig. 2 Overview of approaches to achieving explainability within DNNs. This framework is extended from the machine learning explainability taxonomy presented by Gilpin et al.³¹ and the format is inspired by the work of Arrieta et al.²⁵.

Most models we discuss belong to the deep neural network (DNN) family, such as convolution neural networks (CNNs) and graph neural networks (GNNs), though some XAI model explaining techniques are model agnostic and can be applied to all ML models. The examined input data types include 2D images, 3D images, spectral, tabular data, and molecular graphs.

Explaining DNN model processing

Feature importance (heat map). Feature importance is a highly popular model processing explaining approach within the materials science community^{16,17,32–34}. Feature importance techniques explain how a model processes its input data by answering questions like ‘What distinctive features of this input make it representative of that output?’¹³. Such answers are usually presented as weights of the input data features. The larger the weight, the more important the corresponding feature is from the perspective of the model. In complicated or novel materials systems where the physics is poorly understood, feature importance weights can help materials scientists understand the system and its underlying physics. In materials systems where the physics is already understood, feature importance weights can help materials scientists gain trust in the model prediction (if the feature weights agree with physical laws) or improve model performance (if the feature weights reveal reasons for why the model fails).

Feature importance weights are associated with input features. In traditional ML models where the inputs are tabular data of relatively small dimension (10^1 – 10^2), feature importance weights are often presented as bar plots^{16,35}. In DNNs where the inputs are images or pseudo-images, feature importance weights are often presented as heat maps (also known as saliency maps)^{13,36}.

An example that uses heat maps to understand semantic features from microstructure images is the work of Kondo et al.²¹. The authors predict the ionic conductivity of a solid electrolyte ceramic material (yttria-stabilized zirconia, or YSZ) from image quality (IQ) maps using a CNN. They generate heat maps following a technique similar to class activation map (CAM)³⁶ and gradient-weighted class activation mapping (Grad-CAM)³⁷. Specifically, they design their CNN in a way that the top convolution layer is connected to the first fully connected layer via global average pooling (GAP)^{36,38}, which allows clear one-to-one correspondences between the top convolution filters and the first fully connected layer (**g**) neurons (Fig. 3a). The vector **g** (and its corresponding convolution filter output) is approximately linearly correlated with the prediction target y (or ionic conductivity in this task) with a weight vector **a**. The authors group the convolution filters according to the sign of its weight a_i . A weight $a_i < 0$ means that the filter contributes to low ionic conductivity and a weight $a_i > 0$

means that the filter contributes to high ionic conductivity. A mask is then computed for each filter group (i.e., low/high ionic conductivity group) by averaging filter outputs within the group and thresholding the averaged filter outputs. Finally, the filter group masks are resized to match the input image size via bicubic interpolation and overlaid with the input image (Fig. 3b). These masks are a special kind of heat map. They show that the CNN considers voids as a signature for low ionic conductivity and flattened defect-free areas as a signature for high ionic conductivity (Fig. 3b). These heat map explanations are consistent with experimental evidence, which shows that ionic conductivity decreases with decreasing sintered density (and increasing voids)^{39,40}. The heat maps thus help the authors confirm that their CNN model captures physically reasonable features and increase their confidence in the CNN model. The heat maps also help the authors investigate the optimal representative volume element (RVE) size for their material, which provides guidance for further materials characterization and data collection experiments.

An example of using heat map explanations to diagnose model mistakes is the work of Oviedo et al.²². The authors predict the space group and the crystallographic dimensionality of thin-film materials from X-ray diffraction (XRD) spectral inputs using an in-house CNN. The CNN architecture consists of three 1D convolution layers followed by a global average pooling (GAP) layer and a final dense layer with softmax activation. This CNN handles XRD diffraction pattern inputs as 1D pseudo image inputs. Its design idea is similar to the idea of regular image CNNs, although spectrum data seems very different from image data. The authors generate heat maps following a CAM technique³⁶, which is conceptually similar to that used by Kondo et al.²¹ but with different design details (e.g., value grouping and thresholding). They first generate heat maps for individual XRD patterns and then average heat maps within each space group class to get the heat map of the class. With these heat maps, the authors investigate the causes of model misclassification by comparing heat maps of individual input patterns and average heat maps of space group classes. Two examples are shown in Fig. 4. The heat map of a correctly classified input (Fig. 4c) is similar to the average heat map of its space group class (Fig. 4a). The heat map of an incorrectly classified input (Fig. 4d) is more similar to the average heat map of its predicted space group class (Fig. 4a) than to the average heat map of its true space group class (Fig. 4b). These explanations help the authors confirm the lack of discriminative features in some input data. The authors then further examine the experiment samples and the collected additional data. Turns out that the number of data points in different space-group classes is imbalanced and there is a mixture of phases in the sample. Based on these identified root causes, the authors propose to improve

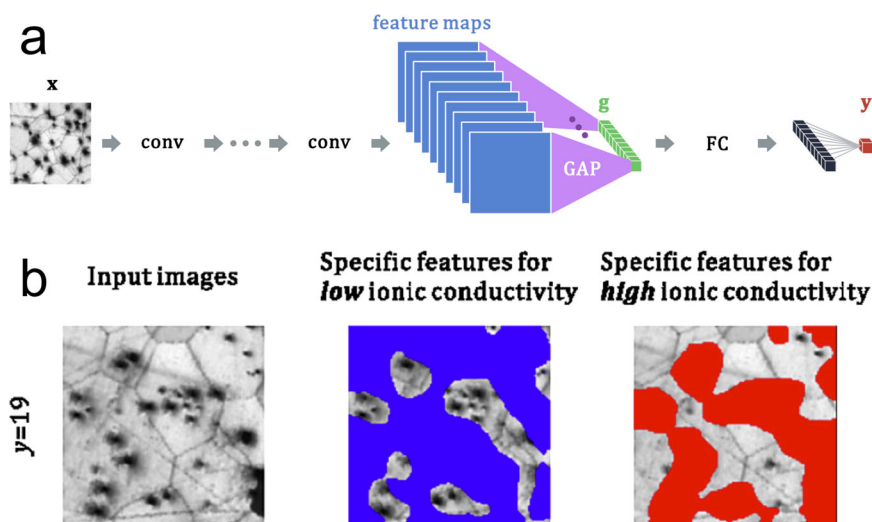


Fig. 3 Example heat map explanations for a CNN with image inputs. The task is to predict the ionic conductivity of a ceramic material from its image quality maps. **a** CNN model architecture. The last convolution layer is connected with the first fully connected layer via global average pooling (GAP), which allows the tracking of implicit attentive response weights from the top fully connected layers to pixel locations on the original image. **b** Heat map masked input images. The blue and red regions hide image features that were ignored by the CNN when predicting low and high ionic conductivities. Figure reprinted from ref. ²¹ with permission. Copyright 2017 Elsevier.

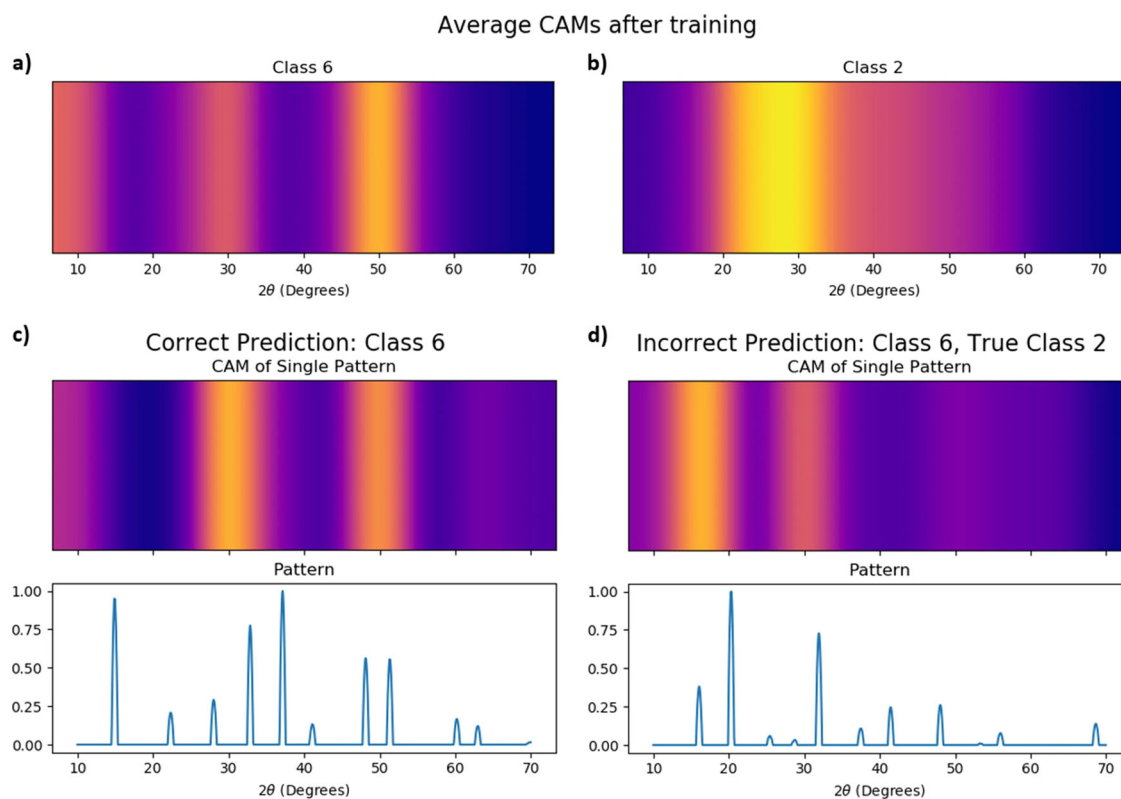


Fig. 4 Example heat map explanations for a CNN with spectral input. The task is to predict sample space groups from XRD spectral. **a, b** Averaged CAM heat maps of space group Class 2 ($P2_1/a$) and Class 6 ($Pm\bar{3}m$). **c, d** CAM heat map and input spectrum of a correctly classified sample and an incorrectly classified sample. Comparing the heat map explanation of each prediction (**a, b**) to the average heat map explanation of each class (**c, d**) helped the authors identify the root cause of model mistakes. Figure reprinted from ref. ²² under the CC BY 4.0 license ¹⁵⁶.

the data collection process by increasing the phase purity during the sample preparation stage and by collecting more data for the poorly classified space-group classes. This work of Oviedo et al.²² is an example of explaining model processing to achieve better

performance. It shows that sometimes poor model performance is rooted in the data, not the model. Heat map explanations can help identify the root cause of model mistakes, which is the first step toward improving the model performance.

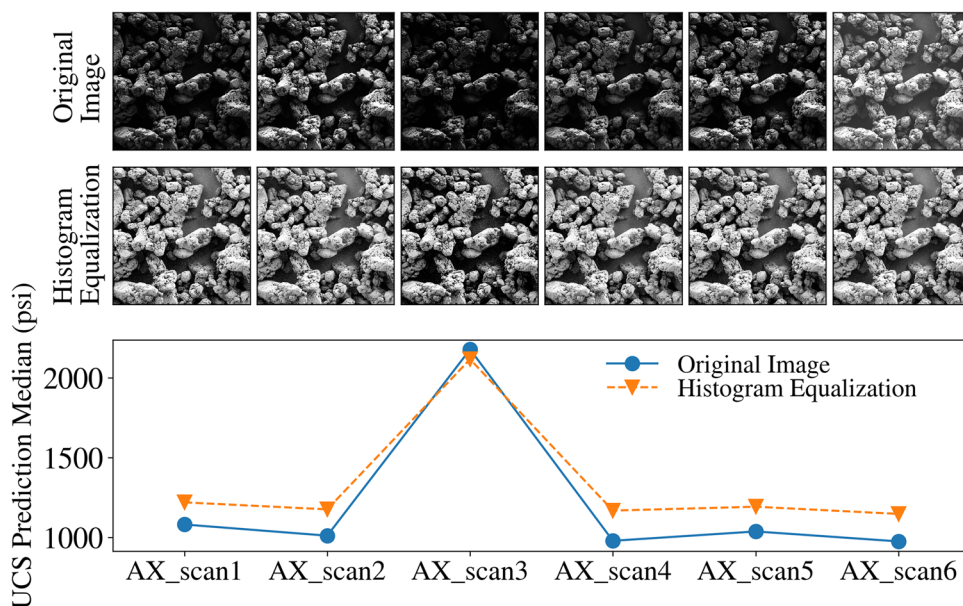


Fig. 5 Example of data distribution shift resulted machine learning model mistakes. The task is to predict material mechanical strength from feedstock material SEM images. The authors take different scans of the same microstructure using different microscope settings. Ideally, ML model predictions should only depend on the microstructure content, not the microscope settings. However, results show that darker images are consistently predicted to have bigger ultimate compressive strength (UCS) values, even with image normalization⁶¹. The x-axis shows experiment id. The first row shows example images from the given experiment. The second row illustrates the effect of one image normalization method (i.e., histogram equalization).

Heat maps can explain a wide range of materials ML problems. In the two examples we discussed, CAM³⁶ based techniques are applied to explain CNNs with image and spectrum inputs. Heat map explanations can also handle other input data types (e.g., diffraction images⁴¹ and molecule graphs^{42–44}) and other DNN models (e.g., graph convolutional neural networks and recurrent neural networks, or GCNNs⁴⁵ and RNNs⁴⁶). Except for CAM, there are also other techniques that can generate heat maps for DNNs, such as guided backpropagation (GBP)⁴⁷, integrated gradient-s(IG)⁴⁸, layer-wise relevance propagation (LRP)⁴⁹, and many more^{50–52}. The main logic of these heat map techniques is usually either maximum activation (i.e., what directly contributes to the output) or maximum sensitivity (i.e., what changes affect the output the most)³¹ though each technique takes a different path. Note that none of these heat map explanation techniques is guaranteed to give the best explanation so it may be interesting to try several techniques and compare the results with domain knowledge. Also, there are important caveats about applying DNN heat map explanation techniques, which will be discussed in more detail in the next section (explanation evaluation).

Finally, general model agnostic feature importance explanation techniques can also be applied to explain DNNs. For example, Local interpretable model-agnostic explanations (LIME)⁵³ is a feature importance explanation technique popular in the machine learning community. It explains model predictions locally by generating a locally perturbed dataset and building linear surrogate models using the perturbed dataset. SHapley Additive exPlanations (SHAP)⁵⁴ is another famous model agnostic explanation technique with good theoretical bounds. LIME and SHAP have been compared empirically in some financial prediction tasks^{55,56}. SHAP does not always show better empirical performance despite its better theoretical bounds^{55,56}. LIME and SHAP have been applied to explain materials problems recently^{35,57,58} but have not been thoroughly compared in materials contexts.

Explanation by example. The second approach to explain DNN model reasoning is by showing illustrative data examples. Studies

of human reasoning show that the use of examples is essential for strategic decision-making⁵⁹. A few examples are sometimes more expressive than a long paragraph of description or a complicated set of predefined equations and coefficients^{12,60}. One common use case of data example explanations is the evaluation of ML model trustworthiness. The trustworthiness of ML models is usually evaluated by test accuracy, which reflects the model performance on new data instances not shown to the model during training. However, test accuracy is not perfect because not all new data instances are the same. For example, new data instances can be classified into in-distribution instances, which follow the same distribution as the training data, and out-of-distribution instances, which do not follow the training data distribution. ML models usually perform very differently on new in-distribution and out-of-distribution instances. This problem is known as distribution shift and is very common in all kinds of real-world ML problems. For example, Zhong et al.⁶¹ recently showed that instrument-induced intensity variations within scanning electron microscopy (SEM) images can deviate ML predictions of material ultimate compressive strength (UCS). The authors evaluated five popular ML models and four image intensity normalization methods on hundreds of well-controlled experimental SEM images of a molecular solid material. Six example SEM images, one image normalization method (histogram equalization), and one machine learning model prediction results are shown in Fig. 5. The results show that darker images are consistently associated with larger predictions. The machine learning model in Fig. 5 is a random forest and the image featurization is binarized statistical image features (BSIF). Other machine learning models and image normalization methods show similar results⁶¹. These results signify the limitation of ML model robustness. Accuracy numbers only carry a limited amount of information. It is important to really understand the data instances and explanation by example can help in this direction.

One can avoid over estimating model accuracy by explaining model predictions with data examples. If the new data instance is reasonably similar to a considerable amount of training data examples, then the model prediction on this new data instance is

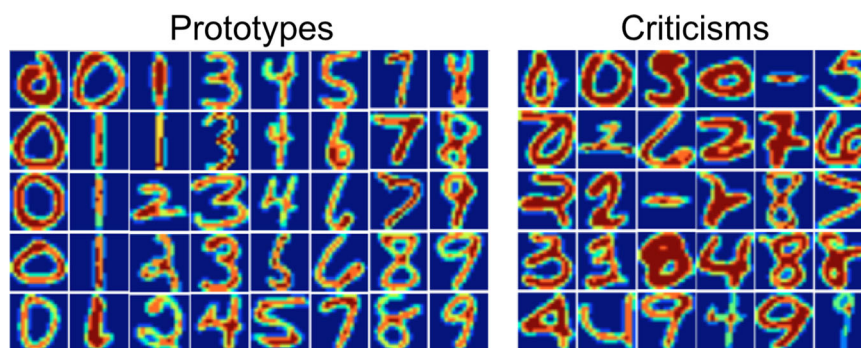


Fig. 6 Example of prototype and criticism explanations. Kim et al.⁶² found random subsets of prototypes and criticisms of the USPS handwritten digits data set⁵⁷ by applying the MMD-critic technique⁶². Prototypes and criticisms are interesting because prototypes can benefit data understanding while criticisms can benefit data cleaning. Figure reprinted from ref.⁶² with permission.

likely reliable. Following this idea, Kailkhura et al.²⁶ design a general-purpose explainable ML framework for materials predictions. The authors explain a previously unseen compound by referring to training compound data instances that are highly similar to the new compound in a customized feature space.

Data example explanations can also help examine data sets. Kim et al.⁶² design a technique (MMD-critic) to efficiently identify prototypes and criticisms within a data set. Prototypes are representative data instance examples, and criticisms are abnormal data instances that do not quite fit the ML model⁶². The authors demonstrated the effectiveness of the technique with some relatively simple images of handwritten digits (Fig. 6). The identified criticisms in this hand-written digit set tend to have bolder lines and irregular shapes while the identified prototypes tend to have regular shapes, which align with human intuitions. Criticisms can be helpful for cleaning the data and prototypes can be helpful for simplifying and understanding the data. Though the hand-written digit task is simple, the same technique can be applied to materials science problems and help build high-quality materials data sets with the minimum human inspection.

Surrogate models. The third approach to explain the data processing of black-box models is by building simple surrogate models that behave similarly to the original black box but are easier to explain. Note that the first two data processing explanation approaches (heat maps and data examples) aim at explaining specific decisions of the original black box. The surrogate model approach takes a slightly different perspective and uses the original black box only to train the surrogate model. In other words, explanations are derived from the surrogate model rather than from the original black box. The implicit assumption is that similar behaviors of the original model and the surrogate model suggest that they have learned a similar set of knowledge.

One interesting example that generates new scientific hypotheses by inspecting a decision tree surrogate model is the work of Raccuglia et al.²⁰. The task is to predict the formation reaction success of inorganic-organic hybrid materials from tabular experimental notes and material atomic/molecular properties. The authors first build an SVM model and achieved a reaction recommendation success rate notably higher than human intuition (89% vs. 78%). The SVM is difficult to explain, so the authors build a surrogate decision tree to explain the original SVM model results. Specifically, they train a C4.5 decision tree^{63,64} using SVM model predictions (Fig. 7). When interpreting the surrogate decision tree, the authors first identify tree leaves that contain the majority of successful reactions and then track upward along the decision tree path to identify reaction conditions/parameters that facilitate formation successes, as shown in Fig. 7. Inspection of these promising reaction conditions/parameters reveals previously unknown chemical insights and generates testable new chemical

hypotheses about amine properties and crystal formation success. Such new scientific insights and scientific hypotheses are valuable and go beyond the specific model and data set.

Decision trees are suitable for building surrogate models because of their high flexibility and natural compatibility with human-readable rules. The reverse decision tree inference approach is general and has been followed to form new scientific hypotheses for other materials systems⁶⁵. Surrogate decision trees can be built and visualized easily with modern ML libraries like scikit-learn⁶⁶ and XGBoost⁶⁷. There are also several techniques that specialize in building relatively simple surrogate decision trees for DNNs^{63,68–70}. However, DNN surrogate trees may still end up being large and complicated due to the extremely large model capacity of DNNs. Note that many subtrees are excised even in the SVM surrogate decision tree of Raccuglia et al. (Fig. 7 triangles). If the surrogate tree is too complicated, users may need to trade off accuracy for explainability. In other words, users can either simplify the original model (e.g., perform feature selection) or simplify the surrogate model (e.g., limit the size of the surrogate tree) to ensure the explainability of the surrogate decision tree. Finally, rule extraction techniques^{71,72} can also be used to build surrogate models and extract knowledge but are less commonly used.

Many materials scientists are interested in deriving simple analytical equations^{73–79}, which can also be considered as one kind of surrogate model. Simple analytical equations are usually less accurate than their black-box counterparts. Nevertheless, simplicity is valuable as suggested by Occam's razor principle⁸⁰. Many scientists believe that fundamental natural laws are simple, though their exact forms are unknown. Moreover, simple analytical equations usually behave robustly on different material systems and are easy to refer to in scientific discussions and education activities. One can derive analytical equations by trying to fit the results of a black-box ML model or apply ML to assist the equation derivation directly. Deriving an analytical equation takes three steps: (1) identify the relevant state variables, (2) decide the equation functional form (e.g., linear or exponential), and (3) determine the coefficients. ML can help with each of the three steps. State variables can be found by performing feature selection or dimension reduction on the original data^{73–76,81}. Functional forms can be screened automatically using tools like Eureqa^{78,79} and SISO^{32,82}. Variable coefficients can be predicted directly from an ML model or estimated by performing a least square fit^{73–75}.

One example of machine learning-assisted analytical equation design is the work of Rovinelli et al.⁷⁸. The objective is to describe the growth of small cracks in terms of the direction and speed of crack propagation. The authors collect a multi-fidelity dataset from crystal plasticity simulations and an in situ crack propagation experiment. The dataset can be described by multiple micromechanical and

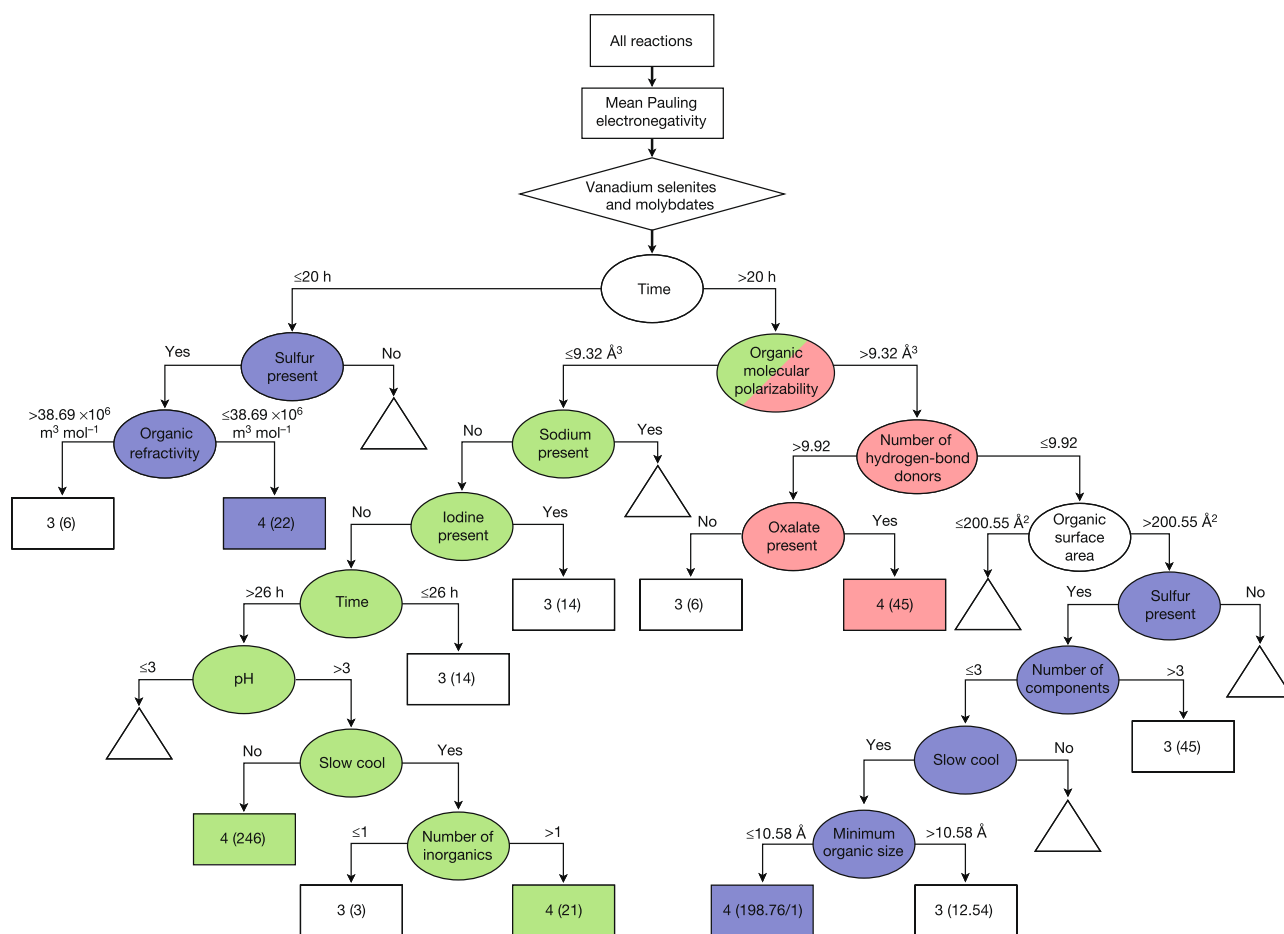


Fig. 7 Example surrogate decision tree model. The task is to predict reaction successes from tabular input data, for which a SVM model achieves good accuracy. A surrogate tree is then built using original SVM model predictions to help understand the original model. The surrogate tree is shown above. Ovals represent decision nodes. Rectangles represent reaction-outcome bins. Triangles represent excised subtrees. Each reaction-outcome bin contains a reaction-outcome value (3 or 4) and the number of reactions assigned to that bin (shown in parenthesis). Bins containing the most successful reactions and their associated synthesis paths are identified and colored. The authors generate the following new testable chemical hypotheses by inspecting the green, blue and red subtrees: (1) Small, low-polarizability amines require the absence of competing Na^+ cations and longer reaction times. (2) Spherical, low projection-size amines require V^{4+} -containing reagents such as VO_2 . (3) Long tri- and tetramines require oxalate reactants. Figure reprinted from ref. ²⁰ with permission. Copyright 2016 Springer Nature.

microstructure features but the quality of these domain knowledge tabular features have not been thoroughly evaluated. The authors build a Bayesian network (BN) for the prediction task and identify relevant features by examining correlations produced by the BN. The identified relevant features serve as variables in a new analytical equation. The functional form of the analytical equation is determined by searching a large number of possible functional forms automatically using Eureqa⁷⁹. This new analytical equation is reasonably simple and performs better than traditional fatigue metrics from the literature (Fig. 8).

Explaining DNN model representations

DNNs usually follow highly organized structures. For example, the vast number of neurons in a CNN are organized first as channels and then as layers. The input data transforms systematically in a hierarchical way as it passes along the network and different CNN components tend to learn different information^{83,84}. Two examples are shown in Fig. 9. Figure 9a shows that different network components (neuron, channel, and layer) tend to learn different concepts. Figure 9b shows that the complexity of CNN filter (channel) representations increases gradually as filter positions move deeper into the network. Because of the hierarchical

representation in CNN components, the convolution layers of a CNN are sometimes considered to perform feature learning and the final fully connected layers are considered to perform prediction⁸⁵. This section summarizes approaches for understanding the knowledge (e.g., data representation) learned in DNN models and the functionality of intermediate DNN components. Example approaches include parameter and activation inspection, concept visualization, and transferability test. They can help users examine the information contained in a well-learned network and answer questions like 'What information does the network contain?'³¹ or 'What abstract concepts did the network learn from the input data?'.

Parameter and activation inspection. A DNN learns by optimizing its parameters with respect to the training data. Do the network parameters contain useful information? This question can be answered by interpreting network parameters directly within the context of materials science domain knowledge. For example, Cecen et al.⁸⁶ train a 3D CNN to predict the structure-property relationship in a synthetic 3-D high-contrast material dataset. The prediction targets are material effective elastic properties (e.g., the C_{11}^* component of the stiffness tensor) and the target data labels are obtained by running finite element (FE) simulations. The 3D

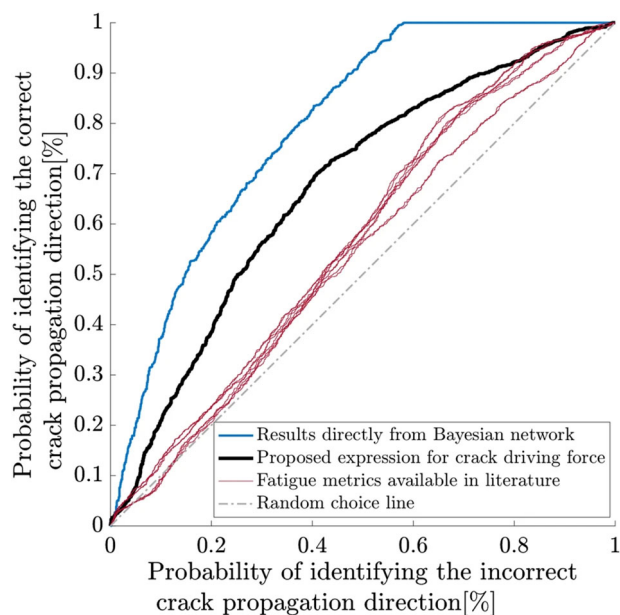


Fig. 8 Performance comparison for different models. The task is to predict crack growth directions. Performances of a Bayesian network model, a machine learning designed analytical equation surrogate model, and some traditional analytical fatigue metrics are given in receiver operative characteristic (ROC) curves. Curves further away from the diagonal indicate better performances. Figure reprinted from ref. ⁷⁸ under the CC BY 4.0 license ¹⁵⁶.

CNN contains a convolution layer of 32 convolution filters, a rectified layer, an average pooling layer, and a linear regression layer. The authors analyze convolution filters of the 3D CNN from three different perspectives. First, from a theoretical perspective, they show that CNN convolution filters are connected to the classic n -point spatial correlation theory^{87,88}. A 2-point and a 1000-point statistics are derived to illustrate the point. Then, they argue that CNN filters describe a weighted set of important material structural patterns. Visual evidence is provided to demonstrate the point (Fig. 10). Finally, they report that some important filters are learned stably across different folds of a 10-fold cross-validation experiment, which confirms that these filters are not random. These analyses show that CNN convolution filters can learn physically sound information about the underlying microstructure. In a different study, Yang et al.⁸ also argue that CNN convolution filters are intrinsically suitable for learning microstructure neighborhood details and CNNs are suitable for microstructure localization problems.

Another common practice is to inspect network activations (or intermediate outputs of specific network components). One activation inspection example is the work of Jha et al.⁸⁹. Jha et al. build an in-house DNN (ElemNet) to predict material properties from elemental composition inputs⁸⁹. ElemNet consists of 17 layers, which include fully connected layers (with ReLU activation) and dropout layers. The prediction target is the DFT computed lowest formation enthalpy and a mean absolute error (MAE) of 0.050 ± 0.0007 eV/atom (or 9% mean absolute deviation) is achieved on a test dataset containing 2×10^4 compounds (training data contains 2×10^5 compounds). The performance of ElemNet is better than its physical-attributes-based conventional ML counterparts even in terms of generalization. In two generalization tests, ElemNet achieves generalization MAEs of 0.138 and 0.122 eV/atom, while the best physical-attributes-based conventional ML model (a random forest) only achieves 0.198 and 0.179 eV/atom MAEs. The authors examine intermediate data representations within the ElemNet to explain its good performance. They provide various inputs to the network and measure intermediate

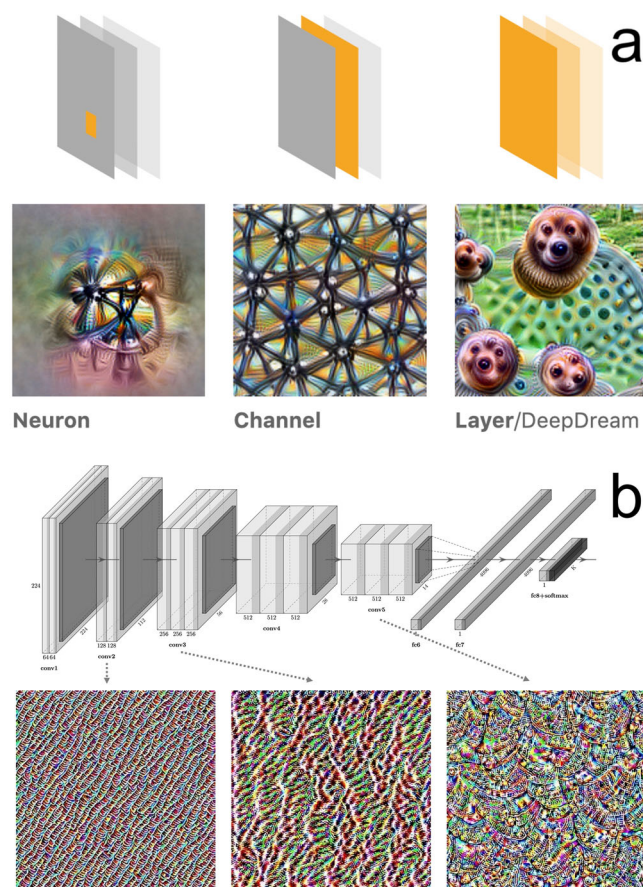


Fig. 9 Illustration of the hierarchical feature representation within CNNs. **a** Comparison of concepts contained in different network levels (neuron, channel, and layer)¹⁵⁸. Figure reprinted from ref. ¹⁵⁸ under the CC BY 4.0 license ¹⁵⁶. **b** Visualization of CNN filters from different layers of an imageNet⁹⁶ pretrained VGG16 network²⁷. Note that the filter representations become more and more complicated as filter positions move deep along the network. The early layer filter representations are relatively primitive, while the top layer filter representations are highly complicated. The VGG16 architecture is plotted using the plot neural net¹⁵⁹ library. Filter representations are visualized using the convolutional neural network visualizations¹⁵⁰ library.

network layer outputs (or activations) associated with these inputs. For easy interpretability, the authors further compress the activations via principal component analysis (PCA) and plot the first two principal components (PCs) (Fig. 11). Figure 11 shows that the network layers can learn to capture essential chemical information from raw composition element inputs. Moreover, successive layers learn the information incrementally. The early layers tend to learn information directly based on the input data (i.e., the presence of certain elements). The later layers tend to learn more complex interactions between the input elements (e.g., charge balance). This kind of simple to complicated data representation within ElemNet is comparable to the hierarchical learning observed in CNNs (Fig. 9b), though ElemNet is not a CNN and consists of only fully connected layers and drop-out layers. More recently, Wang et al.⁹⁰ propose a Transformer⁹¹ based network, Compositionally Restricted Attention-Based network (CrabNet), which shows better prediction accuracy than ElemNet for structure-agnostic materials properties on 28 datasets. CrabNet also provides heat map style explanations and is discussed in detail in the explainable processing/representation section.

If the network architecture follows an encoder-decoder style (e.g., auto-encoders) then the network bottleneck layer activations

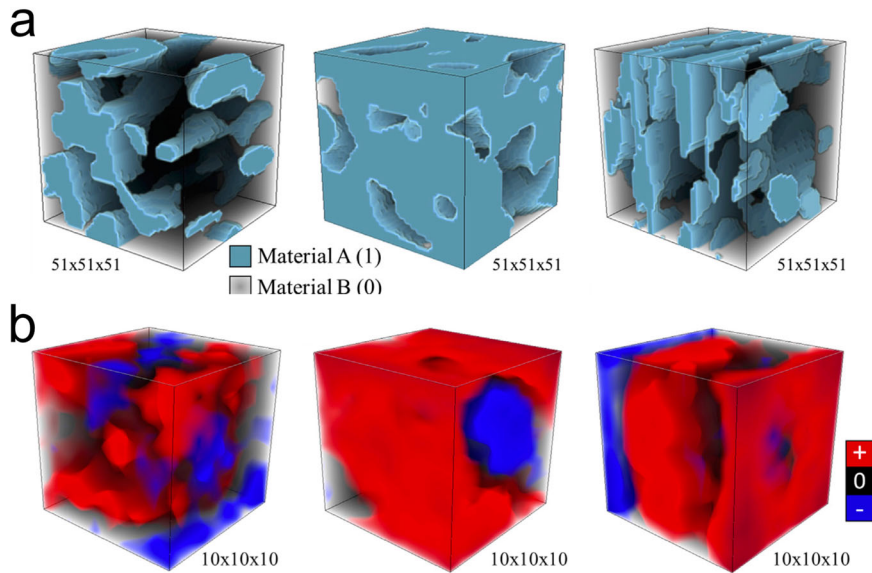


Fig. 10 CNN model parameter direct inspection example. The authors design a simple 3D CNN to predict effective elastic properties of high contrast composites from synthetic microstructures. **a** 3D microstructure data, colored by material phase. **b** CNN filter weights, colored according to their sign (near zero values are rounded to zero). Positive weights in the CNN filters (red) suggest a preference for a structural pattern and negative weights (blue) penalty deviations. A comparison between input microstructures and CNN filter weights shows that CNN filters can learn simplified characteristics of the input microstructure. Figure reprinted from ref. ⁸⁶ with permission. Copyright 2018 Elsevier.

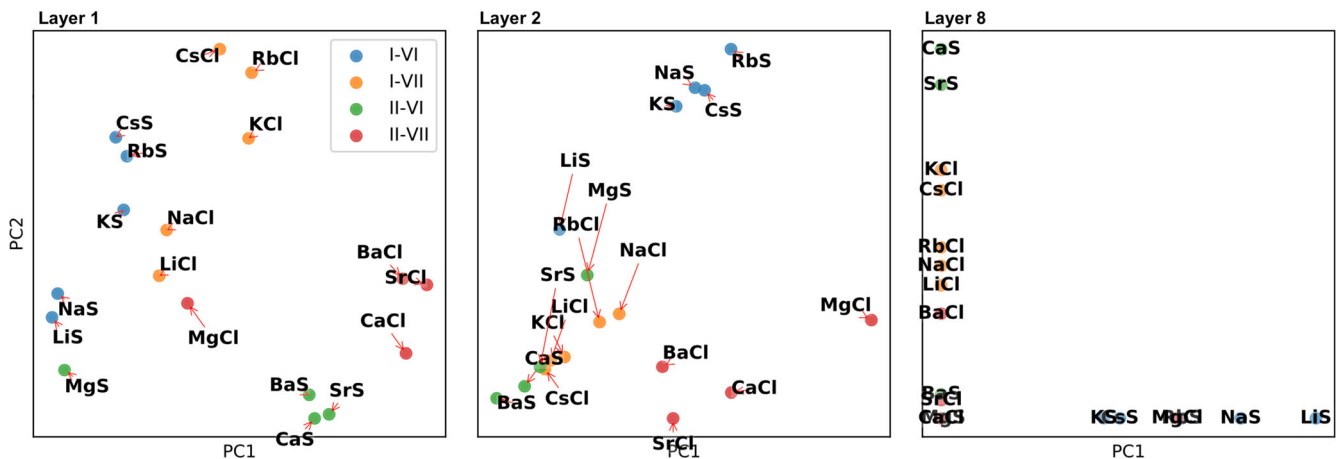


Fig. 11 Network intermediate output (activation) inspection example. The authors predict material property from elemental composition inputs with a fully connected neural network. They then examine network activations by compressing the activations of different materials with PCA and plotting the first two PCs. Results show that intermediate network layers can learn essential chemical information. Early network layers tend to learn the presence of elements and later network layers tend to learn the interaction between elements (e.g., charge balance). Figure reprinted from ref. ⁸⁹ under the CC-BY 4.0 license ¹⁵⁶.

are sometimes referred to as latent data representations. DNN latent representations attract research interests because of their low dimensionality and high expressiveness. DNN latent data representations sometimes outperform their traditional linear dimension reduction counterparts^{92,93}, though not for all problems⁹⁴. The optimal compression method depends on the data of interest. DNN compression usually works the best with highly non-linear data while traditional linear dimension reduction methods (e.g., principal component analysis, or PCA) usually perform better on relatively simple data. Kadeethum et al.⁹⁴ recently proposes that a visual comparison between PCA and t-SNE compressed data representations can indicate the type of compression that works the best for a given data set. This kind of visual prescreening is likely helpful since DNNs are generally computationally expensive to try.

Concept visualization. One can also visualize different concepts learned in individual network components. The idea is that the concept learned in a network component can be represented by a prototypical input that highly activates this network component. One interesting technique to find such prototypical inputs is activation maximization⁸³, which starts with a random input and optimizes the input iteratively with gradient descent until the network component of interest is highly activated. Activation maximization is taken by Ling et al.⁹⁵ to explain important CNN filters in a microstructure classification task. The authors perform the classification task in two steps. Step one is to pass input microstructure images through an ImageNet⁹⁶ pretrained CNN (VGG16⁹⁷) and compute mean texture features by averaging CNN filter activations. Step two is to predict the material class with these mean texture features and a random forest classifier. The

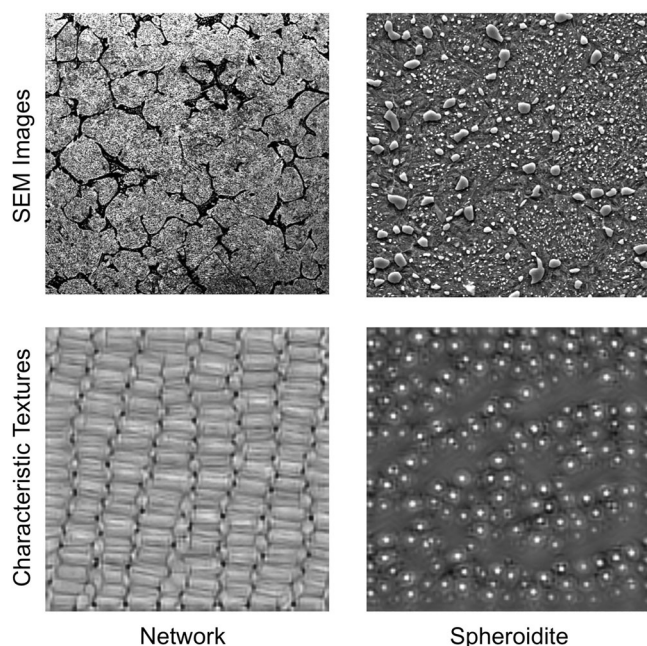


Fig. 12 Concept visualization example. The first row shows SEM images of two different steels and the second row shows their characteristic texture patterns generated from a CNN. Specifically, these synthetic texture patterns illustrate concepts learned in CNN filters through activation maximization. Results suggest that important synthetic texture patterns show interesting abstractions of the corresponding material microstructure. Figure reprinted from ref. ⁹⁵ with permission. Copyright 2017 Elsevier.

authors then examine the mean texture features and their corresponding CNN filters to explain the predictions. They first identify the most important CNN filters by examining the random forest model feature importance. Prototypical images are then generated for the important CNN filters by performing activation maximization. Results show that synthetic prototypical images returned by activation maximization provide interesting abstractions for the corresponding input microstructures (Fig. 12). These synthetic prototypical images are interesting because they may disentangle concepts (e.g., textures) in complicated microstructures and shed light on their roles in the structure-property relationship. Other techniques to generate similar synthetic prototypical images include modified activation maximization^{98–100} and generative adversarial network (GAN)¹⁰¹.

Note that prototypical images from concept visualization are related to heat maps as both of them can be generated from network activations. The difference is that heat maps aim to find discriminative features within the inputs, while concept visualization aims to find representative inputs. Concept visualization is also closely related to explanation by example. However, examples in concept visualization can be generic (rather than real data examples), and the purpose of the concept examples is not to explain the model decision but to understand the model component functionality or a data prototype. CNN filters are especially interesting to materials scientists as they usually correspond to textures¹⁰², which correspond naturally to microstructure textures¹⁰³. Nevertheless, prototypical images can also be generated for other CNN components like neurons and layers (Fig. 9a).

Transferability test. Finally, a third way to interpret network component representations is to test their functionality with a different task. For example, CNN layers trained in one task can be used to generate features for a different task. This reusing of well-trained CNN parameters is known as transfer learning^{104,105}. The

microstructure classification research by Ling et al.⁹⁵ is an example of transfer learning. Remember that Ling et al. compute image features using an ImageNet pretrained CNN. The feature extraction CNN is not trained on microstructure images but can generate high-quality features for various classes of microstructure images (titanium, steel, and powder). Specifically, Ling et al. computed image features from convolution layer activations of a VGG16 network⁹⁷. In a different work, Kitahara et al.¹⁰⁶ computed image features from fully connected layer activations of the same VGG16 and also achieved remarkable accuracy (98.3% and 88.4%) in two unsupervised microstructure classification tasks.

The reasoning for the explainability of transfer learning is similar to that of zero knowledge proof¹⁰⁷. Although we are not sure about what information a network layer contains, the fact that it performs well in multiple different tasks suggests that it must have learned a way to capture key characteristics of the input data. Transfer learning is an important approach in materials science machine learning problems. In addition to the explainability it enables, a more important benefit is that it can greatly reduce the amount of required training data¹⁰⁸. Both Ling and Kitahara examined VGG16⁹⁷, but there is no constraint on the network choice and transfer learning applies to most CNNs. Also, note that neither Ling nor Kitahara used the raw CNN activations directly. There are no rules about how CNN activations should be used and further featurization can often help reduce the feature size and improve the model performance.

Designing explainable DNNs

The techniques discussed until now explain black-box DNNs in a post-hoc manner. It is also possible to achieve intrinsic ante-hoc explainability for DNNs to some extent. Based on existing literature we identified three ways to achieve intrinsic explainability. A DNN is explainable to some extent if (1) the way it processes data is explainable or its internal data representation is explainable, or; (2) its design choice roots in domain knowledge, or; (3) it produces explainable output.

Explainable processing/representation. Transformer⁹¹ is a deep neural network designed from an explainable self-attention mechanism and has shown remarkable success in many different fields in recent years^{109–113}. Transformer is probably underused in materials science problems currently so we introduce its main idea here in detail. At the core of Transformer is an explainable data processing design called self-attention^{114,115}. The intuition of self-attention is that the network is likely to succeed if it can search for relevant information efficiently. Modern search processes usually work with three key components: query, key, and value. For example, to search for a paper about machine learning in materials science, we would enter a query 'machine learning in materials science' in a search engine. The search engine compares our query to keys of all papers and returns a relevance ranking. We then check the paper details (values) based on the relevance ranking. Of course, in DNN prediction problems we only know the prediction input and output, not the appropriate queries, keys, or values. The ML solution is to implement three numerical matrices that correspond to queries, keys, and values and then let the network learn these matrices automatically. One attention block from the Compositionally Restricted Attention-Based network (CrabNet), a Transformer based material property prediction network, is illustrated in Fig. 13. Transformer outperforms LSTM and many other sequential data models for two reasons. First, the self-attention mechanism allows better capture of long-range dependencies. Second, Transformer allows easy parallelization of sequence data due to a positional encoding design. Moreover, the position-aware feature design allows Transformer to provide heat map style explanations in addition to its explainable data processing. A detailed example is presented in the next paragraph

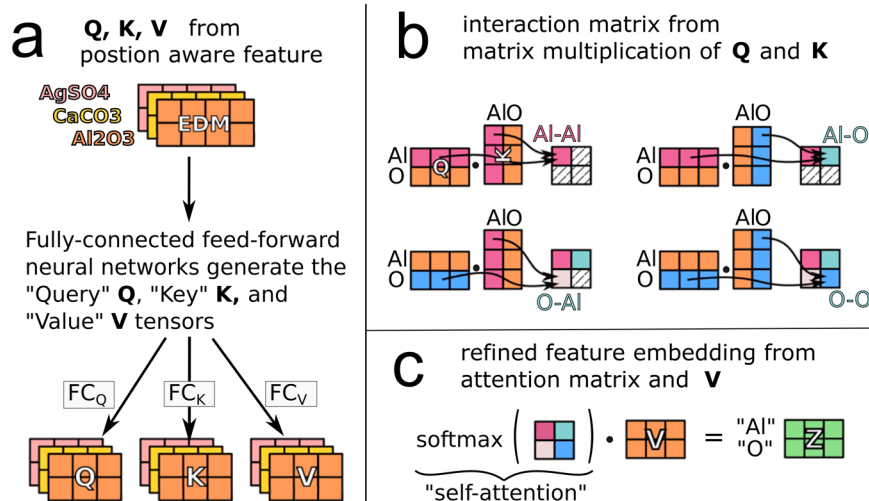


Fig. 13 Illustration of an attention block from CrabNet⁹⁰. **a** The input position-aware features (i.e., EDM) are first mapped to query (**Q**), key (**K**), and value (**V**) matrices by the corresponding fully connected layers. **b** **Q** and **K** are multiplied to give the interaction matrix. This matrix multiplication operation is inspired by the fact that dot product shows the distance between vectors. **c** The interaction matrix is scaled and normalized to give the self-attention matrix, which is then combined with **V** to generate the refined feature representation **Z**. **Z** is further refined by passing through a shallow feed-forward network, which is not shown here. Figure reprinted from ref. ⁹⁰ under the CC BY 4.0 license¹⁵⁶.

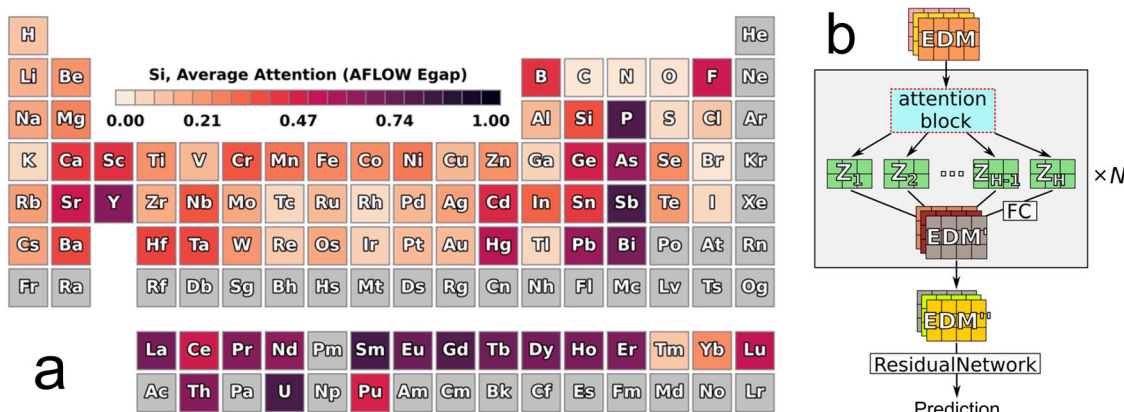


Fig. 14 CrabNet heat map explanation example and model architecture. **a** An element importance heat map from CrabNet. The colors indicate the average amount of attention that Si pays to other elements. This heat map shows that its corresponding attention head (i.e., the first attention head from the first attention layer of CrabNet) attends to common n-type dopants. Here CrabNet is trained with the aflow_Egap dataset¹⁶⁰ in which many compounds contain Si. **b** Illustration of the CrabNet architecture, which includes the EDM embedding, the encoder (shown in the gray square), and the predictor (ResidualNetwork). The attention block design is shown in Fig. 13. One attention block has multiple attention heads. Z_1, \dots, Z_H are generated from different attention heads in the same attention layer. FC stands for fully connected layer, which corresponds to a small feed-forward network. $\times N$ means that the attention layer is repeated N times to form the encoder network. Figure reprinted from ref. ⁹⁰ under the CC BY 4.0 license¹⁵⁶.

to illustrate the positional encoding and the heat map explanation. We highly recommend readers to check the Transformer paper⁹¹ because the self-attention idea is powerful and goes beyond sequential data. Recent research shows that the self-attention mechanism can also be combined with, or even replace, convolution operations in image-based tasks^{116,117}. The attention mechanism may be useful for materials tasks where the truly important features are sparse or long-term interactions play an important role.

CrabNet is a recent Transformer style network that achieves highly accurate structure-free material property predictions and provides heat map style explanations (Fig. 14a). CrabNet predicts material properties based solely on the material chemical formula. One chemical formula is separated into a vector of atomic numbers and a vector of atomic fractions (e.g., [8, 13] and [0.4, 0.6] for Al_2O_3). The atomic number vector and the atomic fraction

vector are padded with zero to the required size, embedded (or featurized) separately, and then combined to generate the full feature matrix, which is called element-derived matrix (EDM) in CrabNet. The separation between element embeddings (i.e., atomic numbers) and positional embeddings (i.e., atomic fractions) allows a unique chemical-environment-aware embedding for each element. EDM is then passed into the CrabNet for further refinement and material property predictions. The CrabNet architecture consists of an encoder and a predictor (Fig. 14b). The CrabNet encoder improves EDMs based on other elements in the current chemical formula and other training chemical formulas. The refined EDM (or EDM' in Fig. 14b) is then passed into a residual network (i.e., the predictor) to make material property predictions. CrabNet is highly accurate. Wang et al. evaluated CrabNet on 28 structure-agnostic material property datasets and CrabNet consistently shows better accuracy than the



Fig. 15 Illustration of the interpretable CNN designed by Zhang et al.¹¹⁸. The top four rows show example filter feature maps from the interpretable CNN and the bottom two rows show example filter feature maps from an ordinary CNN. Note that feature maps from the interpretable CNN filters tend to be meaningful (animal heads in this example) while feature maps from ordinary CNN filters are usually meaningless. These filter feature maps are computed from filter receptive fields following a technique proposed by Zhou et al.¹⁶¹. Figure reprinted from ref. ¹¹⁸ with permission. Copyright 2018 IEEE.

benchmarks (i.e., random forest and ElemNet⁸⁹). The explanations provided by CrabNet also give chemical insights (Fig. 14a).

CNNs with explainable data processing/representation are not yet widely used in materials science research. An example from natural image prediction tasks is the interpretable CNN designed by Zhang et al.¹¹⁸. The architecture of the interpretable CNN is different from mainstream CNNs as there is an additional loss for each convolution filter. These additional filter losses encourage individual filters to encode distinct object parts (e.g., animal faces), as shown in Fig. 15. In other words, the interpretable CNN contains disentangled object representations within its internal filters and predicts image classes by referring to these disentangled representations. The network thus becomes explainable due to its explainable representation. Similar disentangled representation ideas are also seen in other explainable CNNs (e.g., ProtoPNet¹¹⁹) with different technical details. These explainable CNNs are interesting because the possibility of representation disentanglement is exciting. However, one common problem with applying explainable CNNs to materials science data is that currently available explainable CNNs usually reason based on prototypical parts of real objects (e.g., animal face in Fig. 15). In materials data, isolated objects are usually not meaningful and statistical distributions are often important^{88,120}. For the explainable CNNs to be useful for materials science applications, they need to reason based on statistical distributions rather than individual objects. This is likely an interesting research opportunity.

Explainable model design. A black-box DNN can have some explainable design choices. Just like physics-aware kernels¹²¹ are sometimes considered to be more explainable than default linear/Gaussian kernels¹⁹, DNNs with materials science domain

knowledge inspired design choices are more explainable than off-the-shelf DNNs designed for general purposes.

An example of domain knowledge-inspired DNN architecture design is SchNet¹²². SchNet is a DNN customized to predict quantum properties and potential energies for atomistic material systems. The authors are inspired by the great success of DNNs in commercial applications and intend to achieve high-accuracy high-efficiency computation for atomistic systems with DNNs. They summarize a few rules of thumb from materials domain knowledge and designed the DNN (i.e., SchNet) accordingly. First, the fundamental building blocks of atomistic systems are atoms, so the network input is a matrix of atom-type features and the first network layer is an atom embedding layer, which constructs a latent representation of the original atomic system within the network. Second, the interaction between atoms is important, so the following network layers are interaction blocks, which gradually improve the latent representation. The key design in the interaction blocks is a filter-generating network, which takes in atomic position inputs and generates continuous convolution filters that work on the latent representation. Note that the continuous convolutions in the interaction blocks of SchNet are generalizations of discrete convolutions in normal CNNs. This idea may look surprising at first glance since atomic position inputs and image inputs seem vastly different. However, it becomes reasonable once one realizes that the essence of a CNN is the capture of local neighborhood information at different scales, which makes sense in both pixel neighborhoods of images and atomic neighborhoods of molecules/crystals. Note that other researchers working on microstructure analysis problems also comment that CNNs are suitable for microstructure analysis for a similar reason (i.e., the capture of neighborhood information at different scales)^{8,86}. However, while image pixels sit on discrete pixel grids, atom positions are arbitrary and continuous. That is why SchNet adapts the filter-generating network design rather than the normal CNN filters. These domain knowledge-inspired architecture designs (i.e., embedded layer followed by interaction blocks of continuous convolution) help SchNet achieve supreme accuracy on many different atomic scale prediction tasks for various material systems^{123–125} and are explainable to some materials scientists.

Another successful domain knowledge-inspired network example is physics-informed neural networks (PINNs)¹²⁶. More related works can be found in a recent review by Karniadakis et al.¹²⁷, which discusses physics-informed machine learning with a focus on partial differential equation solutions.

Explainable output. Finally, some DNNs can produce explainable outcomes despite highly complicated data processing within the network itself. These DNNs can help address the lack of ground truth challenge in materials science problems. For example, in a study of ferroelectric domain wall transportation behavior, Holstad et al.¹²⁸ apply a sparse Long Short-Term Memory (LSTM) autoencoder to disentangle I(V)-spectroscopy measurements into three components. The authors find that the three disentangled components correspond to two intrinsic signals from domain wall transport behavior and an extrinsic signal from tip-sample contact conduction contribution. These separated output signals allow the authors to better analyze the physical problem of interest though the network architecture and its internal data processing are difficult to understand.

Another example is the work of Liu et al.¹²⁹. The authors trained an attribute editing GAN (attGAN)¹³⁰ that can modify scanning electron microscopy (SEM) images based on desired microstructure characteristics. The model can generate synthetic images of smaller/larger particle sizes or more/fewer pores based on an experimental SEM image. Some examples are shown in the first row of Fig. 16, in which the model modifies the average particle size within the image by removing small particles. The authors also trained a material strength (i.e., peak stress) prediction model

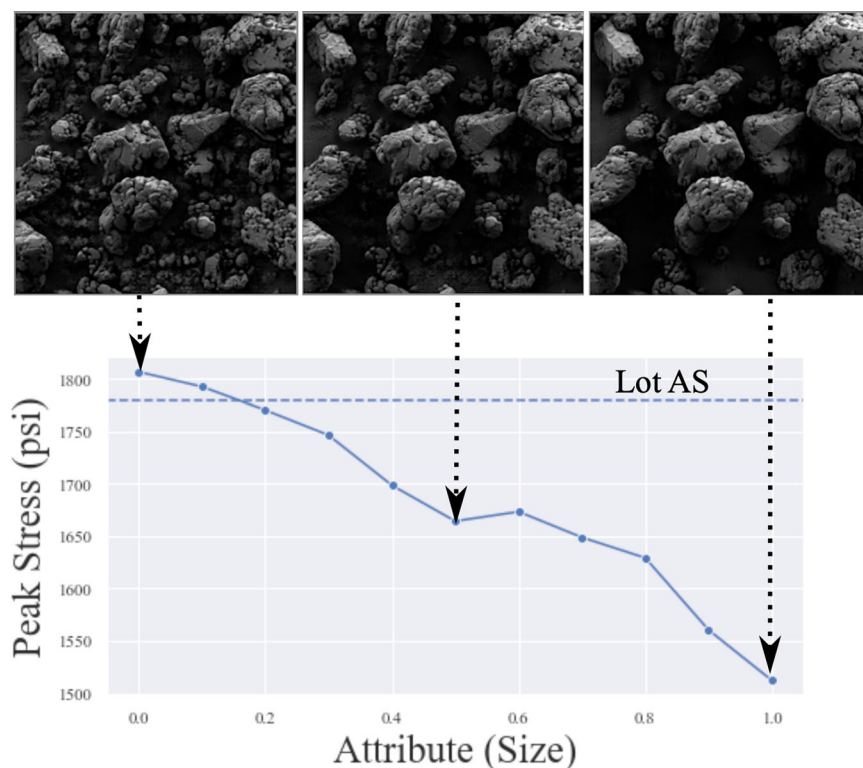


Fig. 16 Example explanation generating network. Material attribute aware visual explanations (i.e., synthetic images) are generated from an image editing GAN¹²⁹. The top row shows synthetic SEM images generated with increasing average particle size input. The lower row shows predictions from an experimental data trained material strength prediction model, in which the x-axis shows the normalized size attribute value. These prediction results agree with materials domain knowledge (i.e., the Hall-Petch rule). The material attribute aware image generation model together with the forward property prediction model form a general framework that can explain the effect of abstract material attributes. Figure reprinted from ref. ¹²⁹ with permission. Copyright 2022 ACS.

using experimental SEM images and strength measurements. When synthetic SEM images from the attGAN are fed into the (experimental data trained) strength prediction model, results agree with domain knowledge (i.e., an inverse correlation between particle size and material strength, which is also known as the Hall-Petch rule^{131–133}). The attGAN is interesting as it provides a new approach to understand abstract but important materials science concepts. The authors showcase their model with particle size and porosity, which are relatively well-understood in the materials science community. Nevertheless, this framework itself is general and can handle less-understood abstract materials concepts. Also, note that Liu et al. trained the model with only 30 unique labels for each concept, which is attainable for many material problems.

EXPLANATION EVALUATION

Explanations should not be blindly trusted. Some explanations, especially post-hoc explanations, may not be faithful to their initial design purpose. For example, Adebayo et al.¹³⁴ showed that several popular CNN heat map explanation techniques do not depend on the parameters of the model being explained. The authors conducted a simple model randomization experiment in which heat maps produced by a well-trained CNN are compared to those produced by a randomly initialized CNN of the same architecture. The results show that some heat maps (e.g., guided GradCAM) are basically unaffected by the model parameter randomization. The authors then conducted another data randomization experiment in which a CNN is trained and tested on a normal dataset, and then trained and tested on the same dataset again but with permuted data labels. On the normal dataset, the model achieved high training and high test accuracy,

suggesting that the model was able to learn the underlying structure of the data. On the label-randomized data set, the model achieved over 95% training accuracy but no better than random guess test accuracy. In other words, the model memorized the training labels of the randomized dataset without being able to truly exploit the underlying structure of the data. Interestingly, some heat maps from the two models are highly similar though the second model was not able to truly learn the underlying structure of the data (Fig. 17). These results suggest that the model architecture alone imposes a heavy prior on the learned network representation and some heat maps are mainly dominated by this model architecture prior. Such heat maps act as edge detectors and do not carry information about the training status of the model being explained. As a result, these heat maps cannot be used for purposes like model debugging. While they can still convey useful information, their limitations must be kept in mind to avoid over interpretation.

Before diving into model evaluation, one should first follow proper model training and evaluation processes to ensure that the model being explained has high quality. One common caveat of model training is overfitting. Overfitting is seen in the randomized data label experiment of Adebayo et al.¹³⁴, in which the model simply memorized the training data to achieve high training accuracy (95%) and failed to generalize well on the test data (i.e., no better than random guessing). To identify overfitting, it is important to follow an appropriate model evaluation convention (e.g., vanilla train/test split, k-fold cross validation, leave-one-out cross validation) and make sure that no data leakage happened during the training. Another important caveat is the multiplicity of good models¹³⁵ and the statistical significance of an explanation. If there are many different ways to model the problem with similar accuracy and the explanation is valid only under specific

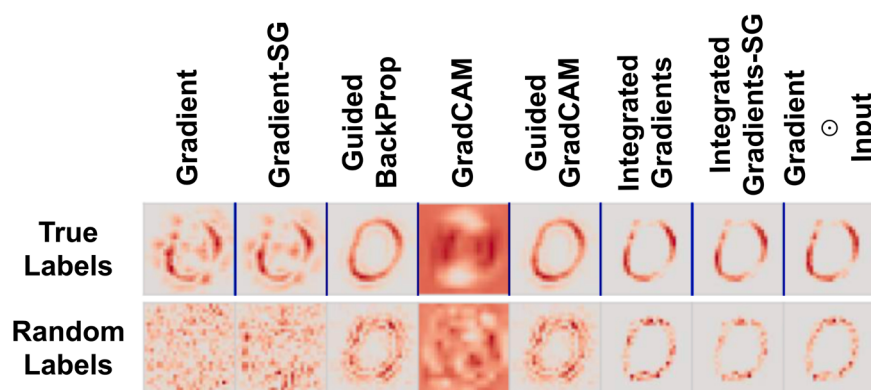


Fig. 17 Illustration for explanation invariance to data label randomization. A CNN is trained on two different datasets separately. The first dataset is normal. The second dataset contains the same data entries as the first dataset but the data labels are permuted randomly. The true-labels dataset trained model achieves high training and high test accuracy. The random-labels dataset trained model achieves high training and no better than random accuracy. Heat maps generated from the two CNNs are shown above. The training dataset is indicated in the row label and the explanation technique is indicated in the column label. Surprisingly, heat maps from the two models can be highly similar despite the dramatically different model generalization (i.e., test) performance. Figure reprinted from ref. ¹³⁴ with permission.

circumstances (e.g., specific hyperparameters), then the validity of the explanation needs to be checked. One example is the recent research of Griffin et al.¹³⁶, in which the authors highlighted the importance of proper physical constraints and unbiased model hyperparameter choice. They show that some ferroelectric switching experiments data in $\text{PbZr}_{0.2}\text{Ti}_{0.8}\text{O}_3$ thin films can lead one to conclude either exotic mechanisms or classic ferroelectric switching mechanisms depending on the ML model being explained. The exotic mechanisms should be double checked in this case. After all, explanation quality depends on the model and serious scientific conclusions should not be drawn by explaining a poor model.

Evaluating different explanations on the same ground, as done by Abedayo et al.¹³⁴, is desirable but very difficult to achieve in general. Explanations are motivated by different purposes (e.g., trust, causation, discovery), function from different perspectives (e.g., ante-hoc/post-hoc, global/local), and exist in different forms (e.g., visual, numerical, and rules). It is difficult to compare explanations that serve different purposes with different approaches. Moreover, as discussed in the previous paragraph, explanations depend on the model being explained and the available training data. Different explanation techniques are rarely compared to each other due to the above reasons. One exception is heat map explanations, which are highly popular for explaining CNNs. Several studies have compared different heat map explanations side by side^{134,137–141}.

Though there is no magic evaluation recipe that works for all explanations, a few core principles should be considered when designing evaluations for materials science explanations. We summarized the principles and present them in Fig. 18. First, an explanation should always be evaluated on a simplified task whenever possible. This principle is simple but highly useful. Take the work of Abedayo et al.¹³⁴ (Fig. 17) as an example. The authors compared different heat map explanation techniques using simple images of hand-written images and natural objects. The simplicity of the input images allowed the authors to focus on the explanation techniques. The same conclusion would be less clear if the authors had started with some complicated materials images (e.g., those in Fig. 16). Then when it comes to the actual evaluation of explanations, we identified four basic attributes of good explanations: (1) usefulness, (2) robustness, (3) sensitivity, and (4) simplicity. The work of Doshi et al.¹⁴², Alvarez et al.³⁰, and Montavon et al.¹³ were referred to when summarizing these four attributes.

Usefulness means that an explanation should be checked with respect to its intended goal. For example, if an explanation is made to identify relevant features in the input data, then the

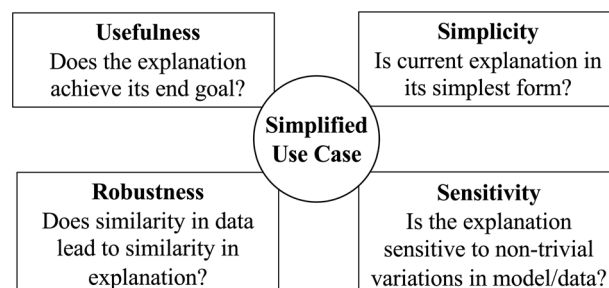


Fig. 18 Graphic summary of explanation evaluation core ideas. An explanation should first be evaluated with a simple use case where the ground truth is clear and the explanation quality can be easily determined. A good explanation should be useful, robust, simple, and sensitive.

determined important features should actually affect model predictions. This can be tested by removing/replacing important features gradually and observing how the model prediction changes, as in the works of Samek et al.^{140,143}. Robustness means that explanations should be robust with respect to small noises in data. In other words, if two input data instances are highly similar and have similar (or identical) labels, then explanations for these two instances should also be similar. This requirement can be tested via input perturbation. For example, several researchers perturbed input images, by adding small noises^{137,141} or by introducing small shift vectors^{13,138}, and measured the change/similarity in resulting explanations to test the explanation robustness (also referred to as continuity or reliability). Sensitivity means that explanations should be sensitive to non-trivial perturbations in both the data under investigation and the model being explained. The work of Abedayo et al.¹³⁴ on explanations independent of model parameter randomization and data label randomization is an example of undesirable invariance that violates the sensitivity attribute. Simplicity means that simpler explanations should be reasonably favored when multiple explanations are available. Too much information is distracting. Good explanations should be selective²⁹ and reveal only the relevant information. The simplicity of different explanations can sometimes be compared quantitatively. One example is the research of Samek et al.¹⁴³, in which the simplicity of visual explanations is measured by the explanation image file size and the image entropy.

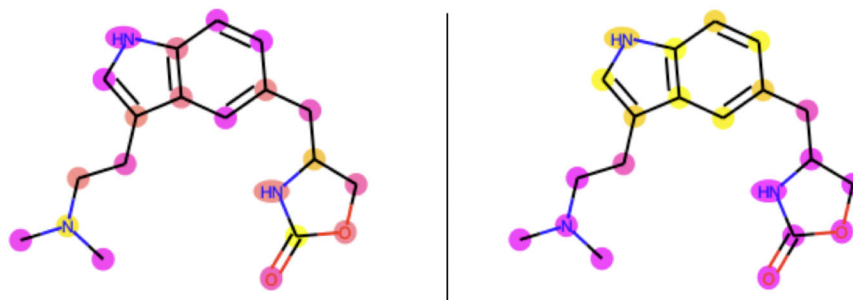


Fig. 19 Example of improved explanation quality due to adversarial training. Molecule attribution heat maps are generated using the vanilla gradients technique. Yellow indicates higher importance and pink indicates lower importance. Note that the molecular importance explanations are sparse before adversarial training (left) and become more compact and physically meaningful after adversarial training (right). Figure reprinted from ref. ⁴⁴ with permission.

CHALLENGES AND OPPORTUNITIES

Following the recent success of ML, there has been an increasing interest in applying XAI techniques to address real-world challenges in various fields (e.g., healthcare, finance, transportation, legal). This rising research interest is distilling into the materials science community. We have presented several recent materials science research examples that applied XAI techniques to understand the underlying physics, generate new scientific hypotheses, and ensure trust in the predictive ML model. Many of the materials XAI examples we found deal with offline prediction tasks using image data. Applying/Designing XAI techniques for other input data types and other application fields would be a research opportunity. After all, XAI is still in its infancy. Many challenges and opportunities exist.

One common challenge in applying current XAI techniques for materials science problems is the lack of clear ground truth. Many XAI techniques were initially designed for natural data, which contain explicit ground truth (e.g., animal classes). Such XAI techniques usually have an implicit assumption of data explicitness/clarity, and model explanations are derived accordingly. One example is the interpretable CNN¹¹⁸ in the explainable processing/representation section. The interpretable CNN can disentangle data representations learned by different filters, but its interpretability is based on the fact that natural objects are composed of easily identifiable parts (e.g., all birds have heads, bodies, wings, and claws). Materials data rarely enjoy the same level of data explicitness. As a result, many existing XAI techniques do not work well for materials science problems. Remedies to this lack of ground truth challenge probably lie in materials science domain knowledge. Experimental materials scientists, computational materials scientists, and machine learning scientists need to collaborate to consolidate abstract materials science domain knowledge into explicit forms (e.g., numerical equations, simple rules, or visual signatures), and then approach explainability by making use of such explicit knowledge. One example study in this direction is the counterfactual image editing GAN¹²⁹ in the explainable output section (Fig. 16), in which high-quality synthetic visual example images are generated by referring domain knowledge (i.e., relevant microstructure attributes of interest).

Another challenge is the evaluation of explanations. As of now, many materials XAI explanations are not carefully evaluated. This lack of evaluation is understandable since explanation evaluation can be highly difficult. Nevertheless, evaluation is an important component of XAI and should not be ignored. More research efforts are needed in three directions: (1) formalizing sanity checks to compare explanations of similar kinds, like the evaluations of heat maps^{134,137–141}; (2) designing new customized explanation evaluation pipelines, like the network representation disentanglement evaluation framework by Bau et al.¹⁴⁴; and (3) incorporating more human interactions to maximize the flexibility of explanation

evaluations. Vilone et al.¹⁴⁵ recently summarized the various ways to achieve human-involved explanation evaluation in a detailed XAI review, which can be useful for designing materials science explanation evaluations.

A third challenge is addressing the misbehavior of ML models and XAI explanations. Note that ML models and XAI explanations can behave illogically even when proper data cleaning and model training procedures are followed. For example, many DNNs of natural images are brittle (i.e., sensitive to small perturbations) due to the pervasive existence of non-robust features (i.e., features are that meaningless to humans and illogically sensitive to small noises)¹⁵. This model performance problem, and its associated explanation problems, can be solved with advanced training techniques like adversarial training¹⁵. This has been shown in the recent work of Loveland et al.⁴⁴. The authors applied adversarial training to train a graph neural network (GNN) for a molecule classification task (i.e., classify explosive and pharmaceutical molecules). They found that adversarial training forced the GNN to make stronger use of the model's learned representations and improved the quality of heat maps explanations (Fig. 19). Moreover, the improvements in model explanation quality do not result in a significant degradation in prediction accuracy (within 2% accuracy difference).

Apart from various theoretical and technical challenges, practical aspects like dissemination and implementation are also important. XAI is a relatively new field to the materials science community. We expect more systematic tutorials with hands-on materials examples will benefit the community. It would also be highly rewarding to build a comprehensive code library and gather different XAI techniques that are most relevant to materials science problems. There are already several general-purpose XAI code libraries (e.g., AI Explainability 360¹⁴⁶, XAI¹⁴⁷, explainerdash-board¹⁴⁸, eli5¹⁴⁹, and convolutional neural network visualizations¹⁵⁰). These libraries are useful but each of them only covers a limited number of XAI techniques for now. Moreover, these libraries are not tailed for materials science use cases. For a code library to benefit the materials science community the most, its user interface needs to be simple since most users with materials science problems do not have strong coding backgrounds. The library should also allow for easy incorporation of domain knowledge and user/instrument inputs for maximum design flexibility. Such a library will greatly promote the usage of XAI and the understanding of materials machine learning problems.

Finally, XAI is not the only useful ML approach for materials science. For example, materials science domain knowledge/insight can boost ML model explainability and performance in materials problems, as we have seen in the SchNet¹²² example. Physics-informed machine learning is a blooming research field¹²⁷. Related to the topic are general informed machine learning¹⁵¹ and theory-guided data science¹⁵². Uncertainty quantification is another interesting field that can help ensure trust in ML model

predictions. A common reason that a well-trained ML model behaves poorly on new test data is distribution shift¹⁵³. If the distribution of the test data does not match the distribution of the training data, then the model is likely to fail. One solution to this problem is to detect distribution shifts before trying to predict. Zhang et al.¹⁵⁴ recently presented an interesting work in this direction. They leveraged predictive uncertainty from deep neural networks to detect real-world shifts in materials data (e.g. material synthesis condition changes and SEM imaging condition drifts). This technique can be applied to raise warnings in case of confusing samples and prevent ML models from making unreliable predictions. Last but not least, visualization (of the data or the ML model) is an important approach to achieve explainability. Several explanation techniques we discussed make heavy use of visualization (e.g., heat maps and concept visualization). Note that visualization by itself is an active research field and there are many interesting visualization techniques that we do not have the time to discuss in this article. Readers can refer to review articles about visual analytics and information visualization for more information¹⁵⁵.

DATA AVAILABILITY

The raw/processed data can be found from the referenced original sources.

Received: 9 April 2022; Accepted: 30 August 2022;

Published online: 22 September 2022

REFERENCES

- Xu, Y. et al. High-throughput calculations of magnetic topological materials. *Nature* **586**, 702–707 (2020).
- Ophus, C. Four-dimensional scanning transmission electron microscopy (4d-stem): From scanning nanodiffraction to ptychography and beyond. *Microsc. Microanal.* **25**, 563–582 (2019).
- Ludwig, A. Discovery of new materials using combinatorial synthesis and high-throughput characterization of thin-film materials libraries combined with computational methods. *npj Comput. Mater.* **5**, 70 (2019).
- Behler, J. Perspective: machine learning potentials for atomistic simulations. *J. Chem. Phys.* **145**, 170901 (2016).
- Li, Z., Kermode, J. R. & De Vita, A. Molecular dynamics with on-the-fly machine learning of quantum-mechanical forces. *Phys. Rev. Lett.* **114**, 096405 (2015).
- Schmidt, J., Marques, M. R., Botti, S. & Marques, M. A. Recent advances and applications of machine learning in solid-state materials science. *npj Comput. Mater.* **5**, 83 (2019).
- Schütt, K. T. et al. How to represent crystal structures for machine learning: Towards fast prediction of electronic properties. *Phys. Rev. B* **89**, 205118 (2014).
- Yang, Z. et al. Establishing structure-property localization linkages for elastic deformation of three-dimensional high contrast composites using deep learning approaches. *Acta Mater.* **166**, 335–345 (2019).
- Gallagher, B. et al. Predicting compressive strength of consolidated molecular solids using computer vision and deep learning. *Mater. Des.* **190**, 108541 (2020).
- Zhang, H. et al. Dramatically enhanced combination of ultimate tensile strength and electric conductivity of alloys via machine learning screening. *Acta Mater.* **200**, 803–810 (2020).
- Lu, Z. et al. Interpretable machine-learning strategy for soft-magnetic property and thermal stability in Fe-based metallic glasses. *npj Comput. Mater.* **6**, 187 (2020).
- Adadi, A. & Berrada, M. Peeking inside the black-box: a survey on explainable artificial intelligence (xai). *IEEE Access* **6**, 52138–52160 (2018).
- Montavon, G., Samek, W. & Müller, K.-R. Methods for interpreting and understanding deep neural networks. *Digit. Signal Process.* **73**, 1–15 (2018).
- Nguyen, A., Yosinski, J. & Clune, J. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 427–436 (2015).
- Ilyas, A. et al. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems (NIPS)*, vol. 32 (2019).
- Liu, Y., Wu, J.-M., Avdeev, M. & Shi, S.-Q. Multi-layer feature selection incorporating weighted score-based expert knowledge toward modeling materials with targeted properties. *Adv. Theory Simul.* **3**, 1900215 (2020).
- Mangal, A. & Holm, E. A. A comparative study of feature selection methods for stress hotspot classification in materials. *Integr. Mater. Manuf. Innov.* **7**, 87–95 (2018).
- Pion-Tonachini, L. et al. Learning from learning machines: a new generation of ai technology to meet the needs of science. Preprint at <https://arxiv.org/abs/2111.13786> (2021).
- Roscher, R., Bohn, B., Duarte, M. F. & Garcke, J. Explainable machine learning for scientific insights and discoveries. *IEEE Access* **8**, 42200–42216 (2020).
- Raccuglia, P. et al. Machine-learning-assisted materials discovery using failed experiments. *Nature* **533**, 73–76 (2016).
- Kondo, R., Yamakawa, S., Masuoka, Y., Tajima, S. & Asahi, R. Microstructure recognition using convolutional neural networks for prediction of ionic conductivity in ceramics. *Acta Mater.* **141**, 29–38 (2017).
- Oviedo, F. et al. Fast and interpretable classification of small x-ray diffraction datasets using data augmentation and deep neural networks. *npj Comput. Mater.* **5**, 60 (2019).
- Oviedo, F., Ferres, J. L., Buonassisi, T. & Butler, K. T. Interpretable and explainable machine learning for materials science and chemistry. *Acc. Mater. Res.* **3**, 597–607 (2022).
- Guidotti, R. et al. A survey of methods for explaining black box models. *ACM Comput. Surv.* **51**, 1–42 (2018).
- Arrieta, A. B. et al. Explainable artificial intelligence (xai): concepts, taxonomies, opportunities and challenges toward responsible ai. *Inf. Fusion* **58**, 82–115 (2020).
- Kailkhura, B., Gallagher, B., Kim, S., Hiszpanski, A. & Han, T. Y.-J. Reliable and explainable machine-learning methods for accelerated material discovery. *npj Comput. Mater.* **5**, 108 (2019).
- Pilania, G. Machine learning in materials science: From explainable predictions to autonomous design. *Comput. Mater. Sci.* **193**, 110360 (2021).
- Lipton, Z. C. The mythos of model interpretability: in machine learning, the concept of interpretability is both important and slippery. *Queue* **16**, 31–57 (2018).
- Miller, T. Explanation in artificial intelligence: Insights from the social sciences. *Artif. Intell.* **267**, 1–38 (2019).
- Alvarez-Melis, D. & Jaakkola, T. S. Towards robust interpretability with self-explaining neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, vol. 31 (2018).
- Gilpin, L. H. et al. Explaining explanations: an overview of interpretability of machine learning. In *Proc. IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, 80–89 (2018).
- Ouyang, R., Curtarolo, S., Ahmetcik, E., Scheffler, M. & Ghiringhelli, L. M. Sisso: a compressed-sensing method for identifying the best low-dimensional descriptor in an immensity of offered candidates. *Phys. Rev. Mater.* **2**, 083802 (2018).
- Pankajakshan, P. et al. Machine learning and statistical analysis for materials science: stability and transferability of fingerprint descriptors and chemical insights. *Chem. Mater.* **29**, 4190–4201 (2017).
- Gola, J. et al. Advanced microstructure classification by data mining methods. *Comput. Mater. Sci.* **148**, 324–335 (2018).
- Yang, C. et al. A machine learning-based alloy design system to facilitate the rational design of high entropy alloys with enhanced hardness. *Acta Mater.* **222**, 117431 (2022).
- Zhou, B., Khosla, A., Lapedriza, A., Oliva, A. & Torralba, A. Learning deep features for discriminative localization. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2921–2929 (2016).
- Selvaraju, R. R. et al. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 618–626 (2017).
- Lin, M., Chen, Q. & Yan, S. Network in network. In *Proc. International Conference on Learning Representations (ICLR)* (2014).
- Gibson, I., Dransfield, G. & Irvine, J. Sinterability of commercial 8 mol% zirconia powders and the effect of sintered density on the ionic conductivity. *J. Mater. Sci.* **33**, 4297–4305 (1998).
- Chen, X., Khor, K., Chan, S. & Yu, L. Influence of microstructure on the ionic conductivity of yttria-stabilized zirconia electrolyte. *Mater. Sci. Eng.: A* **335**, 246–252 (2002).
- Ziletti, A., Kumar, D., Scheffler, M. & Ghiringhelli, L. M. Insightful classification of crystal structures using deep learning. *Nat. Commun.* **9**, 1–10 (2018).
- Schütt, K. T., Arbabzadah, F., Chmiela, S., Müller, K. R. & Tkatchenko, A. Quantum-chemical insights from deep tensor neural networks. *Nat. Commun.* **8**, 1–8 (2017).
- Hsu, T. et al. Efficient and interpretable graph network representation for angle-dependent properties applied to optical spectroscopy. *npj Comput. Mater.* **8**, 151 (2022).
- Loveland, D., Liu, S., Kailkhura, B., Hiszpanski, A. & Han, Y. Reliable graph neural network explanations through adversarial training. In *International Conference on Machine Learning (ICML) Workshop on Theoretic Foundation, Criticism, and Application Trend of Explainable AI* (2021).

45. Pope, P. E., Kolouri, S., Rostami, M., Martin, C. E. & Hoffmann, H. Explainability methods for graph convolutional neural networks. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10764–10773 (2019).
46. Jung, Y.-J., Han, S.-H. & Choi, H.-J. Explaining cnn and rnn using selective layer-wise relevance propagation. *IEEE Access* **9**, 18670–18681 (2021).
47. Springenberg, J. T., Dosovitskiy, A., Brox, T. & Riedmiller, M. Striving for simplicity: the all convolutional net. In *International Conference on Learning Representations (ICLR) Workshop* (2015).
48. Sundararajan, M., Taly, A. & Yan, Q. Axiomatic attribution for deep networks. In *Proc. International Conference on Machine Learning (ICML)*, 3319–3328 (2017).
49. Bach, S. et al. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE* **10**, e0130140 (2015).
50. Shrikumar, A., Greenside, P. & Kundaje, A. Learning important features through propagating activation differences. In *Proc. International Conference on Machine Learning (ICML)*, 3145–3153 (2017).
51. Smilkov, D., Thorat, N., Kim, B., Viégas, F. & Wattenberg, M. Smoothgrad: removing noise by adding noise. In *International Conference on Machine Learning (ICML) Workshop on Visualization for Deep Learning* (2017).
52. Ancona, M., Ceolini, E., Öztireli, C. & Gross, M. Towards better understanding of gradient-based attribution methods for deep neural networks. In *Proc. International Conference on Learning Representations (ICLR)* (2018).
53. Ribeiro, M. T., Singh, S. & Guestrin, C. “Why should i trust you?” explaining the predictions of any classifier. In *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144 (2016).
54. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems (NIPS)*, vol. 30 (2017).
55. Gramegna, A. & Giudici, P. SHAP and LIME: an evaluation of discriminative power in credit risk. *Front. Artif. Intell.* **4**, 752558 (2021).
56. Man, X. & Chan, E. P. The best way to select features? comparing mda, lime, and shap. *J. Financ. Data Sci.* **3**, 127–139 (2021).
57. lquebal, A. S., Pandagare, S. & Bukkapatnam, S. Learning acoustic emission signatures from a nanoindentation-based lithography process: towards rapid microstructure characterization. *Tribol. Int.* **143**, 106074 (2020).
58. Giles, S. A., Sengupta, D., Broderick, S. R. & Rajan, K. Machine-learning-based intelligent framework for discovering refractory high-entropy alloys with improved high-temperature yield strength. Preprint at <https://arxiv.org/abs/2112.02587> (2021).
59. Newell, A. et al. *Human Problem Solving*, vol. 104 (Prentice-hall Englewood Cliffs, NJ, 1972).
60. Aamodt, A. & Plaza, E. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Commun.* **7**, 39–59 (1994).
61. Zhong, X. et al. A study of real-world micrograph data quality and machine learning model robustness. *npj Comput. Mater.* **7**, 161 (2021).
62. Kim, B., Khanna, R. & Koyejo, O. O. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems (NIPS)*, vol. 29 (2016).
63. Quinlan, J. R. *C4.5: Programs for Machine Learning* (Elsevier, 2014).
64. Hall, M. et al. The weka data mining software: an update. *SIGKDD Explor., NewsL.* **11**, 10–18 (2009).
65. Picklum, M. & Beetz, M. Matcalo: knowledge-enabled machine learning in materials science. *Comput. Mater. Sci.* **163**, 50–62 (2019).
66. Pedregosa, F. et al. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
67. Chen, T. & Guestrin, C. Xgboost: A scalable tree boosting system. In *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794 (2016).
68. Zilke, J. R., Mencía, E. L. & Janssen, F. Deepred – rule extraction from deep neural networks. In *Proc. International Conference on Discovery Science*, 457–473 (2016).
69. Schmitz, G. P., Aldrich, C. & Gouws, F. S. Ann-dt: an algorithm for extraction of decision trees from artificial neural networks. *IEEE Trans. Neural Netw.* **10**, 1392–1401 (1999).
70. Augasta, M. G. & Kathirvalavakumar, T. Reverse engineering the neural networks for rule extraction in classification problems. *Neural Process. Lett.* **35**, 131–150 (2012).
71. Andrews, R., Diederich, J. & Tickle, A. B. Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowl.-based Syst.* **8**, 373–389 (1995).
72. Hailesilassie, T. Rule extraction algorithm for deep neural networks: a review. *Int. J. Comput. Sci. Inf. Technol.* **14**, 376–381 (2016).
73. Ghiringhelli, L. M., Vybiral, J., Levchenko, S. V., Draxl, C. & Scheffler, M. Big data of materials science: critical role of the descriptor. *Phys. Rev. Lett.* **114**, 105503 (2015).
74. Pagan, D. C., Schmidt, G. H., Borum, A. D., Long, T. J. & Beaudoin, A. J. Informing mechanical model development using lower-dimensional descriptions of lattice distortion. *Integr. Mater. Manuf. Innov.* **9**, 459–471 (2020).
75. Iwasaki, Y. et al. Machine-learning guided discovery of a new thermoelectric material. *Sci. Rep.* **9**, 1–7 (2019).
76. Grigo, C. & Koutsourelakis, P.-S. A physics-aware, probabilistic machine learning framework for coarse-graining high-dimensional systems in the small data regime. *J. Comput. Phys.* **397**, 108842 (2019).
77. Iwasaki, Y. et al. Identification of advanced spin-driven thermoelectric materials via interpretable machine learning. *npj Comput. Mater.* **5**, 103 (2019).
78. Rovinelli, A., Sangid, M. D., Proudhon, H. & Ludwig, W. Using machine learning and a data-driven approach to identify the small fatigue crack driving force in polycrystalline materials. *npj Comput. Mater.* **4**, 35 (2018).
79. Schmidt, M. & Lipson, H. Distilling free-form natural laws from experimental data. *Science* **324**, 81–85 (2009).
80. Blumer, A., Ehrenfeucht, A., Haussler, D. & Warmuth, M. K. Occam’s razor. *Inf. Process. Lett.* **24**, 377–380 (1987).
81. Pei, Z., Yin, J., Hawk, J. A., Alman, D. E. & Gao, M. C. Machine-learning informed prediction of high-entropy solid solution formation: beyond the hume-rothery rules. *npj Comput. Mater.* **6**, 50 (2020).
82. Ouyang, R. Sisso. <https://github.com/rouyang2017/SISSO>.
83. Erhan, D., Bengio, Y., Courville, A. & Vincent, P. Visualizing higher-layer features of a deep network. *Univ. Montr.* **1341**, 1 (2009).
84. Simonyan, K., Vedaldi, A. & Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Proc. International Conference on Learning Representations (ICLR) Workshop* (2014).
85. Holm, E. A. et al. Overview: computer vision and machine learning for microstructural characterization and analysis. *Metall. Mater. Trans.* A1–15 (2020).
86. Cecen, A., Dai, H., Yabansu, Y. C., Kalidindi, S. R. & Song, L. Material structure-property linkages using three-dimensional convolutional neural networks. *Acta Mater.* **146**, 76–84 (2018).
87. Jiao, Y., Stillinger, F. & Torquato, S. Modeling heterogeneous materials via two-point correlation functions: basic principles. *Phys. Rev.* **76**, 031110 (2007).
88. Fullwood, D. T., Niezgodza, S. R. & Kalidindi, S. R. Microstructure reconstructions from 2-point statistics using phase-recovery algorithms. *Acta Mater.* **56**, 942–948 (2008).
89. Jha, D. et al. Elemnet: deep learning the chemistry of materials from only elemental composition. *Sci. Rep.* **8**, 1–13 (2018).
90. Wang, A. Y.-T., Kauwe, S. K., Murdock, R. J. & Sparks, T. D. Compositionally restricted attention-based network for materials property predictions. *npj Comput. Mater.* **7**, 77 (2021).
91. Vaswani, A. et al. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS)*, vol. 30 (2017).
92. Kim, Y., Choi, Y., Widemann, D. & Zohdi, T. Efficient nonlinear manifold reduced order model. In *Conference on Neural Information Processing Systems (NIPS) Workshop on Machine Learning for Engineering Modeling, Simulation and Design* (2020).
93. Kim, Y., Choi, Y., Widemann, D. & Zohdi, T. A fast and accurate physics-informed neural network reduced order model with shallow masked autoencoder. *J. Comput. Phys.* **451**, 110841 (2022).
94. Kadeethum, T. et al. Non-intrusive reduced order modeling of natural convection in porous media using convolutional autoencoders: comparison with linear subspace techniques. *Adv. Water Resour.* **160**, 104098 (2022).
95. Ling, J. et al. Building data-driven models with microstructural images: Generalization and interpretability. *Mater. Discov.* **10**, 19–28 (2017).
96. Deng, J. et al. Imagenet: A large-scale hierarchical image database. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 248–255 (2009).
97. Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. In *Proc. 3rd International Conference on Learning Representations (ICLR)* (2015).
98. Nguyen, A., Clune, J., Bengio, Y., Dosovitskiy, A. & Yosinski, J. Plug & play generative networks: conditional iterative generation of images in latent space. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4467–4477 (2017).
99. Mahendran, A. & Vedaldi, A. Visualizing deep convolutional neural networks using natural pre-images. *Int. J. Comput. Vis.* **120**, 233–255 (2016).
100. Yosinski, J., Clune, J., Nguyen, A., Fuchs, T. & Lipson, H. Understanding neural networks through deep visualization. In *Deep Learning Workshop at the 32nd International Conference on Machine Learning (ICML)* (2015).
101. Nguyen, A., Dosovitskiy, A., Yosinski, J., Brox, T. & Clune, J. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *Advances in Neural Information Processing Systems (NIPS)*, vol. 29 (2016).
102. Cimpoi, M., Maji, S. & Vedaldi, A. Deep filter banks for texture recognition and segmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015).
103. Bunge, H.-J. *Texture Analysis in Materials Science: Mathematical Methods* (Elsevier, 2013).
104. Sharif Razavian, A., Azizpour, H., Sullivan, J. & Carlsson, S. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 512–519 (2014).

105. Yosinski, J., Clune, J., Bengio, Y. & Lipson, H. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems (NIPS)*, vol. 27 (2014).
106. Kitahara, A. R. & Holm, E. A. Microstructure cluster analysis with transfer learning and unsupervised learning. *Integr. Mater. Manuf. Innov.* **7**, 148–156 (2018).
107. Goldreich, O. & Oren, Y. Definitions and properties of zero-knowledge proof systems. *J. Cryptol.* **7**, 1–32 (1994).
108. Tan, C. et al. A survey on deep transfer learning. In *Proc. International conference on artificial neural networks*, 270–279 (2018).
109. Jumper, J. et al. Highly accurate protein structure prediction with alphafold. *Nature* **596**, 583–589 (2021).
110. Schwaller, P. et al. Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. *ACS Cent. Sci.* **5**, 1572–1583 (2019).
111. Schwaller, P. et al. Mapping the space of chemical reactions using attention-based neural networks. *Nat. Mach. Intell.* **3**, 144–152 (2021).
112. Tetko, I. V., Karpov, P., Van Deursen, R. & Godin, G. State-of-the-art augmented nlp transformer models for direct and single-step retrosynthesis. *Nat. Commun.* **11**, 1–11 (2020).
113. Kovács, D. P., McCorkindale, W. & Lee, A. A. Quantitative interpretation explains machine learning models for chemical reaction prediction and uncovers bias. *Nat. Commun.* **12**, 1–9 (2021).
114. Parikh, A. P., Täckström, O., Das, D. & Uszkoreit, J. A decomposable attention model for natural language inference. In *Proc. 2016 Conference on Empirical Methods in Natural Language Processing*, 2249–2255 (2016).
115. Lin, Z. et al. A structured self-attentive sentence embedding. In *Proc. International Conference on Learning Representations (ICLR)* (2017).
116. Bello, I., Zoph, B., Vaswani, A., Shlens, J. & Le, Q. V. Attention augmented convolutional networks. In *Proc. IEEE/CVF International Conference on Computer Vision (ICCV)*, 3286–3295 (2019).
117. Dosovitskiy, A. et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. International Conference on Learning Representations (ICLR)* (2021).
118. Zhang, Q., Wu, Y. N. & Zhu, S.-C. Interpretable convolutional neural networks. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8827–8836 (2018).
119. Chen, C. et al. This looks like that: deep learning for interpretable image recognition. In *Advances in Neural Information Processing Systems (NIPS)*, vol. 32 (2018).
120. Randle, V. Twinning-related grain boundary engineering. *Acta Mater.* **52**, 4067–4081 (2004).
121. Camps-Valls, G. et al. Physics-aware gaussian processes in remote sensing. *Appl. Soft Comput.* **68**, 69–82 (2018).
122. Schütt, K. T., Saucedo, H. E., Kindermans, P.-J., Tkatchenko, A. & Müller, K.-R. SchNet—a deep learning architecture for molecules and materials. *J. Chem. Phys.* **148**, 241722 (2018).
123. Schutt, K. et al. Schnetpack: a deep learning toolbox for atomistic systems. *J. Chem. Theory Comput.* **15**, 448–455 (2018).
124. Lu, C. et al. Deep learning for optoelectronic properties of organic semiconductors. *J. Phys. Chem. C* **124**, 7048–7060 (2020).
125. Schütt, K. T., Gastegger, M., Tkatchenko, A. & Müller, K.-R. Quantum-chemical insights from interpretable atomistic neural networks. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, 311–330 (Springer, 2019).
126. Raisi, M., Perdikaris, P. & Karniadakis, G. E. Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* **378**, 686–707 (2019).
127. Karniadakis, G. E. et al. Physics-informed machine learning. *Nat. Rev. Phys.* **3**, 422–440 (2021).
128. Holstad, T. S. et al. Application of a long short-term memory for deconvoluting conductance contributions at charged ferroelectric domain walls. *npj Comput. Mater.* **6**, 163 (2020).
129. Liu, S. et al. Attribution-driven explanation of the deep neural network model via conditional microstructure image synthesis. *ACS Omega* **7**, 2624–2637 (2022).
130. He, Z., Zuo, W., Kan, M., Shan, S. & Chen, X. Attgan: Facial attribute editing by only changing what you want. *IEEE Trans. Image Process.* **28**, 5464–5478 (2019).
131. Petch, N. The cleavage strength of polycrystals. *J. Iron Steel Inst.* **174**, 25–28 (1953).
132. Hall, E. The deformation and ageing of mild steel: iii discussion of results. *Proc. Phys. Soc., B* **64**, 747 (1951).
133. Yu, H., Xin, Y., Wang, M. & Liu, Q. Hall-petch relationship in mg alloys: a review. *J. Mater. Sci. Technol.* **34**, 248–256 (2018).
134. Adebayo, J. et al. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems (NIPS)*, vol. 31 (2018).
135. Breiman, L. Statistical modeling: the two cultures (with comments and a rejoinder by the author). *Stat. Sci.* **16**, 199–231 (2001).
136. Griffin, L. A., Gaponenko, I. & Bassiri-Gharb, N. Better, faster, and less biased machine learning: electromechanical switching in ferroelectric thin films. *Adv. Mater.* **32**, 2002425 (2020).
137. Alvarez-Melis, D. & Jaakkola, T. S. On the robustness of interpretability methods. In *International Conference on Machine Learning (ICML) Workshop on Human Interpretability in Machine Learning (WHI)* (2018).
138. Kindermans, P.-J. et al. The (un) reliability of saliency methods. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, 267–280 (Springer, 2019).
139. Adebayo, J., Gilmer, J., Goodfellow, I. & Kim, B. Local explanation methods for deep neural networks lack sensitivity to parameter values. In *International Conference on Learning Representations (ICLR) Workshop* (2018).
140. Samek, W., Binder, A., Montavon, G., Lapuschkin, S. & Müller, K.-R. Evaluating the visualization of what a deep neural network has learned. *IEEE Trans. Neural Netw. Learn. Syst.* **28**, 2660–2673 (2016).
141. Ghorbani, A., Abid, A. & Zou, J. Interpretation of neural networks is fragile. *Proc. AAAI Conf. Artif. Intell.* **33**, 3681–3688 (2019).
142. Doshi-Velez, F. & Kim, B. Towards a rigorous science of interpretable machine learning. Preprint at <https://arxiv.org/abs/1702.08608> (2017).
143. Samek, W., Wiegand, T. & Müller, K.-R. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *ITU J.: ICT Discov.* **1**, 39–48 (2017).
144. Bau, D., Zhou, B., Khosla, A., Oliva, A. & Torralba, A. Network dissection: quantifying interpretability of deep visual representations. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3319–3327 (2017).
145. Vilone, G. & Longo, L. Notions of explainability and evaluation approaches for explainable artificial intelligence. *Inf. Fusion.* **76**, 89–106 (2021).
146. Arya, V. et al. One explanation does not fit all: a toolkit and taxonomy of ai explainability techniques. Preprint at <https://arxiv.org/abs/1909.03012> (2019).
147. Saucedo, A., Iqbal, U. & Krishna, S. Xai - an explainability toolbox for machine learning. <https://github.com/EthicalML/xai>.
148. Dijk, O. explainerdashboard. <https://github.com/oegedijk/explainerdashboard>.
149. Korobov, M. et al. Eli5. <https://github.com/TeamHG-Memex/eli5>.
150. Ozbulak, U., Stoken, A., Wang, H., Geirhos, R. & Jiang, P. Convolutional neural network visualizations. <https://github.com/utkuozbulak/pytorch-cnn-visualizations>.
151. von Rueden, L. et al. Informed machine learning—a taxonomy and survey of integrating knowledge into learning systems. *IEEE Trans. Knowl. Data Eng.* **1-1** (2021).
152. Karpatne, A. et al. Theory-guided data science: a new paradigm for scientific discovery from data. *IEEE Trans. Knowl. Data Eng.* **29**, 2318–2331 (2017).
153. Ben-David, S. et al. A theory of learning from different domains. *Mach. Learn.* **79**, 151–175 (2010).
154. Zhang, J., Kailkhura, B. & Han, T. Y.-J. Leveraging uncertainty from deep learning for trustworthy material discovery workflows. *ACS Omega* **6**, 12711–12721 (2021).
155. Liu, S., Wang, X., Liu, M. & Zhu, J. Towards better analysis of machine learning models: a visual analytics perspective. *Vis. Inform.* **1**, 48–56 (2017).
156. Creative commons attribution 4.0 international license. <https://creativecommons.org/licenses/by/4.0/>.
157. Hull, J. J. A database for handwritten text recognition research. *IEEE Trans. Pattern Anal. Mach. Intell.* **16**, 550–554 (1994).
158. Olah, C., Mordvintsev, A. & Schubert, L. Feature visualization. <https://distill.pub/2017/feature-visualization>.
159. Iqbal, H. et al. Plotneuralnet. <https://github.com/HarisIqbal88/PlotNeuralNet>.
160. Clement, C. L., Kauwe, S. K. & Sparks, T. D. Benchmark aflow data sets for machine learning. *Integr. Mater. Manuf. Innov.* **9**, 153–156 (2020).
161. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A. & Torralba, A. Object detectors emerge in deep scene cnns. In *Proc. International Conference on Learning Representations (ICLR)* (2015).

ACKNOWLEDGEMENTS

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 and was supported by the LLNL-LDRD Program under Project No. 19-SI-001.

AUTHOR CONTRIBUTIONS

T.Y.-J.H. and B.G. conceived the project. X.Z. drafted the manuscript. All authors contributed to the discussion and writing of the manuscript.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41524-022-00884-7>.

Correspondence and requests for materials should be addressed to Xiaoting Zhong or T. Yong-Jin Han.

Reprints and permission information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2022