

Machine learning in materials design: Algorithm and application

Zhilong Song(宋志龙), Xiwen Chen(陈曦雯), Fanbin Meng(孟繁斌), Guanjian Cheng(程观剑), Chen Wang(王陈), Zhongti Sun(孙中体), Wan-Jian Yin(尹万健)

Citation: Chin. Phys. B .2020, 29 (**11**): 116103. **doi:**10.1088/1674-1056/abc0e3
Journal homepage: http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

What follows is a list of articles you may be interested in

---

*t*P40 carbon:A novel superhard carbon allotrope

Heng Liu(刘恒), Qing-Yang Fan(樊庆扬), Fang Yang(杨放), Xin-Hai Yu(于新海), Wei Zhang(张伟), Si-Ning Yun(云斯宁)
Chin. Phys. B . 2020, 29(10): 106102 . **doi:** 10.1088/1674-1056/ab9c01

Fundamental band gap and alignment of two-dimensional semiconductors explored by machine learning

Zhen Zhu(朱震), Baojuan Dong(董宝娟), Huaihong Guo(郭怀红), Teng Yang(杨腾), Zhidong Zhang(张志东)
Chin. Phys. B . 2020, 29(4): 046101 . **doi:** 10.1088/1674-1056/ab75d5

Pressure-induced new chemistry

Jianyan Lin(蔺健妍), Xin Du(杜鑫), Guochun Yang(杨国春)
Chin. Phys. B . 2019, 28(10): 106106 . **doi:** 10.1088/1674-1056/ab3f91

The CALYPSO methodology for structure prediction

Qunchao Tong(童群超), Jian Lv(吕健), Pengyue Gao(高朋越), Yanchao Wang(王彦超)
Chin. Phys. B . 2019, 28(10): 106105 . **doi:** 10.1088/1674-1056/ab4174

Discovery of superhard materials via CALYPSO methodology

Shuangshuang Zhang(张爽爽), Julong He(何巨龙), Zhisheng Zhao(赵智胜), Dongli Yu(于栋利), Yongjun Tian(田永君)
Chin. Phys. B . 2019, 28(10): 106104 . **doi:** 10.1088/1674-1056/ab4179

TOPICAL REVIEW — Machine learning in condensed matter physics

Machine learning in materials design: Algorithm and application*

Zhilong Song(宋志龙), Xiwen Chen(陈曦雯), Fanbin Meng(孟繁斌), Guanjian Cheng(程观剑), Chen Wang(王陈), Zhongti Sun(孙中体)†, and Wan-Jian Yin(尹万健)‡

College of Energy, Soochow Institute for Energy and Materials InnovationS (SIEMIS), and Jiangsu Provincial Key Laboratory for Advanced Carbon Materials and Wearable Energy Technologies, Soochow University, Suzhou 215006, *China*

(Received 6 July 2020; revised manuscript received 24 August 2020; accepted manuscript online 14 October 2020)

Traditional materials discovery is in 'trial-and-error' mode, leading to the issues of low-efficiency, high-cost, and unsustainability in materials design. Meanwhile, numerous experimental and computational trials accumulate enormous quantities of data with multi-dimensionality and complexity, which might bury critical 'structure–properties' rules yet unfortunately not well explored. Machine learning (ML), as a burgeoning approach in materials science, may dig out the hidden structure–properties relationship from materials bigdata, therefore, has recently garnered much attention in materials science. In this review, we try to shortly summarize recent research progress in this field, following the ML paradigm: (i) data acquisition → (ii) feature engineering → (iii) algorithm → (iv) ML model → (v) model evaluation → (vi) application. In section of application, we summarize recent work by following the 'material science tetrahedron': (i) structure and composition → (ii) property → (iii) synthesis → (iv) characterization, in order to reveal the quantitative structure–property relationship and provide inverse design countermeasures. In addition, the concurrent challenges encompassing data quality and quantity, model interpretability and generalizability, have also been discussed. This review intends to provide a preliminary overview of ML from basic algorithms to applications.

Keywords: machine learning, materials design, structure–property relationship, active learning

PACS: 61.50.Ah, 81.20.−n, 07.05.Mh **DOI:** 10.1088/1674-1056/abc0e3

1. Introduction

Discovery of the hidden rules behind the data is not new in the history of science. In 1500 s, Tyco Brahe carried out precise and extensive measurements of the positions of planets in our solar system and obtained a large quantity of data. Johannes Kepler discovered the rules and summarized his famous laws to describe the motion of the planets from Brahe's data.[1] Isaac Newton pondered these laws from an *ab initio* view and concluded his gravitational law via advanced mathematics in his era. Five hundred years later, machine learning (ML) follows the same type of research paradigm. The difference is that, data acquisition is becoming easier and data quantities could be huge ("big data"), owing to the development of computational algorithm and facilities and experimental techniques. In this case, the rules behind the data is not easily observed by human intuition. ML methods are playing increasingly critical roles and trained to learn rules from data, which automate the data analysis step like Kepler's work. In materials science, ML had been applied to accelerate material research including structure prediction,[2–8] synthesis,[9–13] characterization recognition,[14–18] properties prediction,[19–29] and devices evaluation,[30,31] etc.

The typical workflow of ML methods in materials science research is shown in Fig. 1. In light of the type and amount of data, ML model can be normally classified into supervised learning, unsupervised learning, semi-supervised learning. Supervised learning is a popular and useful model which can point out the quantitative structure property/activity relationship (QSPR/QSAR) attributing to explicit input-output labeled data using solid blue line in Fig. 1. Feature engineering was applied to transform the raw information from collected data into certain fingerprint that will be deemed as inputs for the algorithm selection. ML models are then trained to predict the outcome of remaining data. If the model is not good upon validation, the previous steps are repeated iteratively with improved data cleaning, feature re-selection, or algorithm translating. In this review, some commonly used supervised models in materials science will be introduced in section 3. For unsupervised models, there are mainly two types: clustering models aiming to distinguish different categories of unlabeled data such as K-means clustering, mean-shift clustering, and density-based spatial clustering of applications with noise (DBSCAN), and dimension reduction models like PCA, Locally Linear Embedding (LLE) and t-SNE. The t-SNE is usually utilized to draw figures of high dimensional data such as reference.[32,33]

*Project support by the National Natural Science Foundation of China (Grant Nos. 11674237 and 51602211), the National Key Research and Development Program of China (Grant No. 2016YFB0700700), the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD), China, and China Post-doctoral Foundation (Grant No. 7131705619).

†Corresponding author. E-mail: ztsun@suda.edu.cn

‡Corresponding author. E-mail: wjyin@suda.edu.cn

© 2020 Chinese Physical Society and IOP Publishing Ltd

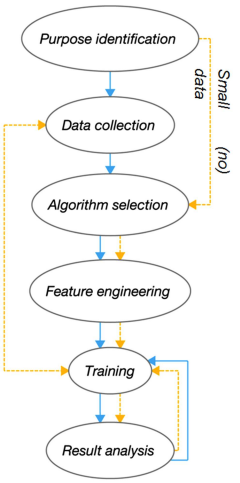


Fig. 1. The main workflow of traditional supervised learning and active learning.

For limited data quantity ("small data"), active learning, also known as adaptive design, is normally employed to expedite material discovery.[34] An ML model is acquired on the basis of a given data set and it will recommend the next data to add into the original data set, then updating the ML model. It is an iterative process so that the model performance is continuously improved, as shown in Fig. 1. In this review, we firstly introduce several recent methods and progress to obtain and represent data for ML algorithms in section 2. Next, we concisely discuss the mechanisms and implementation of different categories of ML algorithms. The algorithms that can produce analytic expressions and algorithms for active learning are introduced in separate sections. In section 4, recent applications of ML in materials science, in terms of structure, property, synthesis and characterization, are reviewed generally in logical sequence to show the development of research concerns and algorithms. Finally, we discuss the perspectives and conclusions of ML in material research.

2. Data and features

2.1. Data

Data is of central importance in the ML. To obtain data, materials scientists can either explore public databases or generate data by themselves through computations or experiments. Usually, the ML algorithm for materials science is supervised learning based on a large quantity of data. Therefore, the approaches to obtain plentiful data like high-throughput computation or experiment are required. Table 1 enumerates prevalent tools for high-throughput computation that could help to realize this step.[35–43] The majority of these tools are based on python language, the most popular programming language for ML currently.

Table 1. HT tools.

Name	Function	URL
Pymatgen	Robust, open-source python library for materials analysis	https://pymatgen.org
AFLOW π	A minimalist framework for high-throughput first principles calculations	http://aflowlib.org/src/aflowpi
FireWorks	An open-source code for defining, managing, and executing calculation workflows	https://materialsproject.github.io/fireworks
AiiDA	A workflow to automate complex numerical procedures of calculation	http://www.aiida.net
Pymatflow	A workflow simplifier for research on materials science by means of *ab initio* simulation	http://pymatflow.readthedocs.org
ASE	Setting up, steering, and analyzing atomistic simulations	https://wiki.fysik.dtu.dk/ase
Atomate	Built on top of state-of-the-art open-source libraries: pymatgen, custodian, and FireWorks	https://atomate.org
Custodian	A simple, robust, and flexible just-in-time (JIT) job management framework	https://pypi.org/project/custodian
MPInterfaces	A python tool that enables high throughput analysis of interfaces using VASP, VASPsol, and MP tools	http://henniggroup.github.io/MPInterfaces
Imeall	A database framework for the calculation of the atomistic properties of grain boundaries	https://github.com/Montmorency/imeall
Pylada	A modular python framework to control physics simulations	http://pylada.github.io/pylada
Pyiron	An integrated development environment (IDE) for computational materials science	https://pyiron.github.io

Owing to the rapid development of Materials Genomes, materials databases, such as Materials Project (MP),[44] Inorganic Crystal Structure Database (ICSD),[45] Open Quantum Materials Database (OQMD),[46] and other databases,[36,47–68] are becoming the main venue to store the structures and properties of materials. A plethora of those databases contain tons of materials information. Some of them focus on specific materials like Supercon (a database for superconducting materials) and TEDesignLab (a database for thermoelectric materials). Table 2 summarizes the information about part of these databases. Some of these databases in Table 2 were developed based on other databases in order to obtain better features or be more domain-specific. For instance, MaterialsWeb derived from MP and Pearson's Crystal Data (PCD) originated from the Linus Pauling File (LPF).

Table 2. Material databases.

Name	Data type	URL	Free
Materials Project	Multiple	https://materialsproject.org	√
ICSD	Inorganic & Experimental	https://icsd.fiz-karlsruhe.de	×
AFLOWLIB	Inorganic & Computational	http://aflowlib.org	√
COD	Multiple & Experimental	http://crystallography.net	√
QM9	Organic molecules	http://quantum-machine.org/datasets/	√
OMDB-GAP1	Organic crystals	https://omdb.mathub.io/dataset	√
OQMD	Multiple & Computational	http://oqmd.org	√
NOMAD	Multiple	https://nomad-repository.eu	√
Materials Cloud	Multiple (3D, 2D)	https://www.materialscloud.org	√
NREL Materials	Computational & Renewable	https://materials.nrel.gov	√
Clean Energy Project	Solar cell	https://cepdb.molecularspace.org	×
TEDesignLab	Thermoelectric	http://www.tedesignlab.org	√
HTEM	Inorganic	https://htem.nrel.gov/	√
Supercon	Superconducting	https://supercon.nims.go.jp	√
MaterialsWeb	2D	https://www.materialsweb.org	√
C2DB	2D	https://cmr.fysik.dtu.dk/c2db/c2db.html	√
CSD	Multiple	https://www.ccdc.cam.ac.uk	×
CMR	Multiple (3D, 2D)	https://cmr.fysik.dtu.dk	√
Citrination	Multiple	https://citrination.com	√
JARVIS-DFT	2D	https://www.ctcms.nist.gov/∼ knc6/JVASP.html	√
MatNavi	Multiple	https://mits.nims.go.jp	√
MatWeb	Engineering	http://matweb.com	√
GDB	Small organic molecules	http://gdb.unibe.ch	√
ZINC	Compounds	https://zinc15.docking.org	√
ChEMBL	Bioactive molecules	https://www.ebi.ac.uk/chembl	√
ChemSpider	Multiple	http://www.chemspider.com	√
Materials Commons	Computational	https://materialscommons.org	√
AiiDA	Alloy Phase Diagram	http://www.aiida.net	√
ASM	Inorganic	https://www.asminternational.org	×
LPF	Multiple	https://paulingfile.com	×
PCD	Multiple	http://www.crystalimpact.com/pcd	×
Nano-HUB	Nanomaterials	http://nanohub.org	√
EELS Data Base	Spectra	https://eelsdb.eu	√
XAFS database	Spectra	https://www.cat.hokudai.ac.jp/catdb/	√

Many of these databases offer official application programming interface (API) for users to obtain or batch download data. For example, Pymatgen[35] is an open-source python library that functions as the API of materials project and a powerful toolkit for materials analysis. AFLOW π is an efficient high-throughput calculation framework and serves as the API of the AFLOW database. Nevertheless, some commercial databases require a license to access such as ICSD, ChemSpider, etc.

Apart from materials database, a large quantity of data are buried in numerous publications. Text mining or natural language processing (NLP) can be used to extract information from them. Vahe *et al.* implemented unsupervised word embeddings to mine knowledge from materials science publications. They demonstrated that the knowledge in literature could be encoded as information-dense word embeddings.[69] Amangeldi *et al.* wrote a review about Li-O_2 batteries through the assistance of a text mining program that explored over 1800 literatures.[70] Matthew *et al.* developed an effective toolkit named ChemDataExtractor to extract chemical information from literature automatically.[71] Text mining or NLP methods were also successfully applied to obtain information about chemistry, materials synthesis, nanoparticle properties, etc.[13,72–75]

Data quality and quantity are crucial prerequisites for ML. In general, the more data there are, the more likely there are erroneous and problematic data. However, a reliable ML model requires both high quality and large quantity for training data.[61] To overcome this dilemma, many efforts have been taken to improve the reproducibility and reduce the error in materials databases. Fourches *et al.* proposed some principles and guidelines for quantitative structure-activity/property relationship (QSA/PR) modeling in Cheminformatics.[76] The FAIR (findability, accessibility, interoperability, and reusability) data principles,[77] put forward by Mark *et al.* are imperative for database establishment. These principles would help researchers develop better databases and improve the performance of the ML model trained based on these databases.

Meanwhile, the unfavorable data, e.g., data from failed experiments, which are usually abandoned in databases or publications, could be meaningful as well. For instance, Paul *et al.* used data from failed syntheses to predict conditions for certain materials formation.[78] In terms of quantity, if the data size used for training is considered too small and difficult to enlarge, a simple method is to perform data operations like cutting and rotation on initial data to create more. This method is called data augmentation and it could help to improve the variety of training data and reduce overfitting. Examples of this method in materials science can be seen in reference.[9,79–83]

2.2. Features

The raw data often suffer from many issues that make them unsuitable as direct use for ML algorithms. For instance, if the raw data were not representative and could not capture enough distinctive information, it would be difficult to distinguish different structures. Another issue is that the training data may not be in the same order of magnitude, which would significantly impact the performance of ML algorithms.

Therefore, the next step after data acquisition is feature engineering, transforming raw data into more representational forms. Better features lead to better model performance. For example, crystal structure can be described by the atomic coordinates and cell shape. However, such kind of description is usually not good for ML, since it is extremely difficult for ML algorithm to link them with materials properties. Feature engineering for molecular and crystal structures has been active fields for decades. Two prevalent examples are the SMILES (simplified molecular-input line-entry system)[84] proposed in 1987 and the Coulomb matrix (CM)[85] first proposed in 2012. Also, the Ewald sum matrix and sine matrix proposed by Felix *et al.*[86] and the partial radial distribution function (PRDF) proposed by Schütt *et al.*[87] are extended from the Coulomb matrix. Other features include the London matrix,[88] bag of bonds (BOB),[89] bonds, angles, machine learning representation (BAML),[88] extended-connectivity fingerprints (ECFP4),[90] many-body tensor representation (MBTR),[91] smooth overlap of atomic positions (SOAP),[92] atom-centered symmetry functions (ACSFs),[93] classical force-field-inspired descriptors (CFID),[94] etc.

The above mentioned approaches are all based on domain knowledge. Recently, following the idea of graph features,[95] graph neural network (GNN)[96] has been proposed to describe crystal structures and molecules. In GNN, the atoms and bonds are represented as nodes and edges in graphs, respectively. For instance, Xie *et al.* implemented crystal graph convolutional neural networks (CGCNN) to interpretably predict material properties.[97] Another strategy is implementing dimension reduction methods such as the classic PCA algorithm[98] that projects high-dimensional complicated data to low-dimensional properties-relevant features. What all those approaches do is transforming crystal or molecule structures to an algorithm-readable and -implementable format.

Feature engineering in common approach can be done by available tools, which is convenient for materials scientists. Table 3 enumerates the information about these tools[97,99–107] and most of them are based on python language. Prevalent python libraries for data science and ML such as pandas,[108] sklearn,[109] tensorflow,[110] and pytorch,[111] also offer affiliated codes to cope with feature engineering. Recently, Liu *et al.*[112] developed a data-driven multi-layer feature selection tool named DML-FS$_{\mathrm{dek}}$ to automate the feature engineering process for materials science. This tool can also incorporate domain knowledge to select features with more interpretability, and models trained on these features can achieve similar accuracy compared to that without domain knowledge.

Table 3. Feature tools.

Name	Description	URL
QML	A python toolkit for representation learning of properties of molecules and solids	https://www.qmlcode.org
AMP	A modular approach to machine learning in atomistic simulations	https://amp.readthedocs.io
Magpie	Materials-agnostic platform for informatics and exploration	https://github.com/amarkrishna/demo1
RDkit	A collection of cheminformatics and machine-learning software written in C++ and Python	https://github.com/rdkit/rdkit
ChemML	A ML program suite for the analysis, mining, and modeling of chemical and materials data	https://pypi.org/project/chemml
DScribe	Library of descriptors for machine learning in materials science	https://singroup.github.io/dscribe
Matminer	A Python library for data mining the properties of materials	https://hackingmaterials.lbl.gov/matminer
SchNet	A deep learning architecture for quantum chemistry	https://github.com/atomistic-machine-learning/SchNet
DeepChem	Deep-learning models for drug discovery and quantum chemistry	http://github.com/deepchem/deepchem
MEGNet	An implementation of DeepMind's graph networks for universal machine learning in materials science	https://github.com/materialsvirtuallab/megnet
CGCNN	Implement crystal graph convolutional neural networks to arbitrary crystal structures	https://github.com/txie-93/cgcnn

3. Algorithms

3.1. Introduction of algorithms

In this section, we firstly introduce conventional ML methods[113] widely used in materials science research, such as linear-based, kernel-based, tree-based algorithms, and basic neural network, followed by the introduction of deep learning models. Then, we introduce a class of unique supervised ML approach that is able to distill analytical formula (descriptor) between features and properties, making hidden ML model interpretable. Last but not least, algorithms for small data, such as adaptive or active machine learning, are introduced.

3.1.1. Conventional algorithms and deep learning

For linear-based, kernel-based algorithms and basic neural network methods, there are mainly two critical parts: the loss function and the way for coefficients update.

Linear model[114] predicts the target by a linear combination of features, which is the most basic and widely used model. Considering an n-dimensional data sample after data collection and feature engineering $\boldsymbol{x} = [x_1,\ x_2,\ \ldots,\ x_n]^{\mathrm{T}}$, the objective function y under linear combination is

$$y = f(\boldsymbol{x};\boldsymbol{w},b) = \sum_i w_i x_i + b = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b = \hat{\boldsymbol{w}}^{\mathrm{T}}\hat{\boldsymbol{x}}, \quad (1)$$

where $\boldsymbol{w} = [w_1,\ w_2,\ \ldots,\ x_n]^{\mathrm{T}}$ is called the weight vector, b refers to the bias, $\hat{\boldsymbol{w}}$ and $\hat{\boldsymbol{x}}$ are the augmented matrix of the weight vector and data sample, respectively.

The loss function measures the difference between real and predicted target values. Suppose a set of training data is (x^n, y^n), the commonly used loss function for regression problem is the mean square error (MSE) [Eq. (2)], the mean absolute error (MAE), the root mean square (RMSE), etc. Loss functions for classification will be discussed in Subsection 3.2.1. The linear regression model using MSE is also called ordinary least squares regression (OLSR).

$$L(f) = \frac{1}{n}\sum_n (f(x^n) - y^n)^2. \quad (2)$$

Both the weight vector and the bias are coefficients that need to be learned, which can be achieved by optimizing the loss function using the least square method or the gradient descent method [Eq. (3)] to predict the value of the target y,

$$w_i \leftarrow w_i - \eta \frac{\partial L(f)}{\partial w_i}, \quad (3)$$

where η is the learning rate and w_i is the i-th coefficient. By continuously calculating the gradient of the loss function, the parameters are continuously updated in the negative direction of the gradient, and the learning rate reflects the step size of each update.

To mitigate over fitting, which is an important issue in ML (details in Subsection 3.2.2), some regularization terms that relate to all the coefficients could be added to the loss function. For instance, the least absolute shrinkage and selection operator (LASSO) algorithm[115] adds a regularization term about the absolute value of coefficients, also known as the L_1-norm [Eq. (4)],

$$L(f) = L_{\mathrm{o}}(f) + \frac{\lambda}{n}\sum_w |w|, \quad (4)$$

where $L_{\mathrm{o}}(f)$ stands for the original loss function and λ is the regularization coefficient that can be adjusted through validation (Subsection 3.2.1). Similarly, the ridge regression[116] has a regularization term about the square value of coefficients L_2-norm and the elastic net method[117] has both L_1-norm and L_2-norm.

Nevertheless, many feature–target relationships are intrinsically nonlinear, therefore a linear-based algorithm is unsuitable. To simulate nonlinear targets, a commonly used method is solving the optimization problem of loss function in a higher dimension which leads to the kernel-based algorithms. However, these algorithms do not actually transform the input data into a higher dimension. Instead, only the so-called kernel function values are calculated. The kernel function measures the relationships between every pair of data sample points as if they are in a higher dimension. This type of method could significantly improve the efficiency, so they are called kernel trick. Two kernel examples are the Gaussian kernel (radial basis function kernel) [Eq. (5)] and the polynomial kernel [Eq. (6)],

$$K(x_1, x_2) = \mathrm{e}^{-\frac{|x_1 - x_2|^2}{2\sigma^2}}, \quad (5)$$

$$K(x_1, x_2) = (x_1 x_2 + r)^d, \quad (6)$$

where x_1 and x_2 refer to two different data points. The ridge regression method using kernel is called kernel ridge regression (KRR).[118] Another example is the support vector regression (SVR)[119] based on the concept of support vector machines (SVMs),[120] it also uses the kernel function but uses an ε-insensitive loss function (7),

$$L(f) = \begin{cases} 0, & \varepsilon > |y - f(x)|, \\ |y - f(x)| - \varepsilon, & \text{other}. \end{cases} \quad (7)$$

Another way to fit nonlinear targets is to let the output of the linear model pass through an activation function. The activation function could be the logistic function [Eq. (8)], one kind of sigmoid functions, then the linear model changes to a logistic regression model,

$$\sigma(z) = \frac{1}{1 + \mathrm{e}^{-z}}. \quad (8)$$

There also exist many other activation functions like the tanh function, rectified linear unit (ReLU), leaky ReLU, parametric ReLU (PReLU), exponential linear unit (ELU), softout function, maxout unit, etc. The combination of a linear

model and an activation function forms a neuron. The $f(\boldsymbol{x}; \boldsymbol{w}, b)$ found through optimizing the loss function by one neuron may not perform well, so multiple neurons could be connected. A group of neurons that cooperate with each other through some connection methods and different information transmission methods is an artificial neural network (ANN). Neural networks could also update coefficients through gradient descent method (3), only the loss function is different in (3) compared to linear regression. At present, the basic structures of neural networks commonly used are feedforward networks, memory networks, and graph networks.

Feedforward networks include fully connected networks and convolutional networks (CNN),[121] the information only spreads in one direction in this kind of network. The layers between the input layer and output layer are called the hidden layers, figure 2 shows a simple fully connected network with two hidden layers. In the hidden layers of 2D CNN, the input 2D data are operated with some matrices in order to distill the local information. A fixed-size matrix is used to perform dot products or selecting the maximum on each area of the 2D data from the last layer, and these two operations are called convolution and max-pooling, respectively. Figure 3 illustrates a 2D CNN with two max-pooling layers and one convolution layer, the 2D data is flattened into 1D data to match the shape of targets at last. Recurrent neural network (RNN) (Fig. 4) is another kind of networks, after the network receives the input x_t at time t, the value of the hidden layer is s_t and the output value is o_t. The key point is that the value of s_t depends not only on x_t, but also on s_{t-1}, which makes them have some memory to some extent and suitable for sequential data. However, sequential data is rare in material science so the RNN is not widely used. The spirit of deep learning based on these networks is to increase the number of hidden layers. The more hidden layers there are, the deeper the model.[122] Although it can be proven that a neural network with only one hidden layer can theoretically represent all functions, the efficiency of training and ultimate performance are far behind deep neural networks.[123]

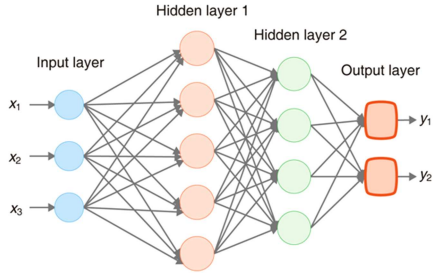


Fig. 2. A simple fully connected neural network structure with two hidden layers.

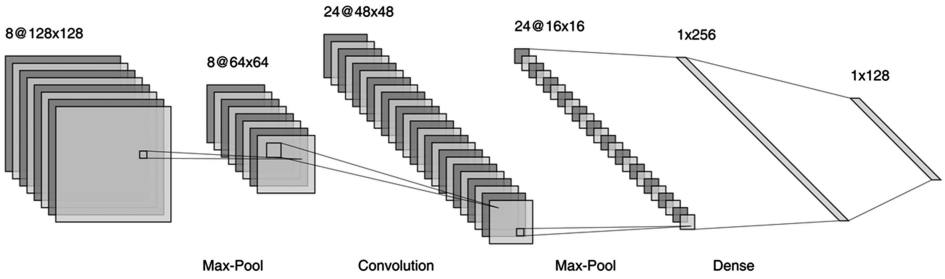


Fig. 3. A 2D CNN structure with two max-pooling layers and one convolution layer.

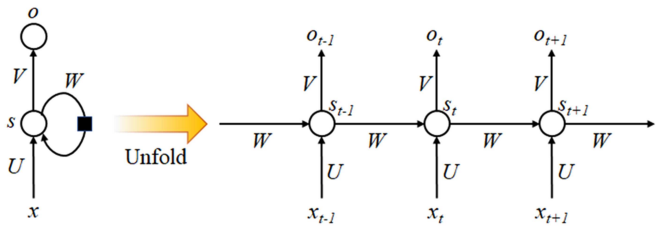


Fig. 4. The sketch of the methodology of RNN.

Tree-based models such as decision tree[124] are pure nonlinear model, which use a tree structure with every leaf containing criteria to classify or regress targets. Tree-based models are usually applied through ensemble methods which mainly include two categories: boosting and bagging. The boosting methods such as CatBoost,[125] light gradient boosting machine (LightGBM),[126] AdaBoost,[127] XGBoost,[128] and gradient boosting decision tree (GBDT),[129] combine weak models to improve the fitting accuracy. The bagging methods like random forest (RF),[130] which is the ensemble version of decision tree, combine relatively strong models to reduce over fitting.

So far, all the algorithms mentioned above are discriminative models which focus on the relationship between input and target, or the law or distribution that the corresponding target should meet given a certain input data point. Another category of models is the generative models that try to describe the joint distribution of input and target. Such as the Naïve Bayes method, Gaussian process regression (GPR), Bayesian optimization (details in Subsection 3.1.3), hidden Markov model (HMM),[131] etc. Variational autoencoders (VAE)[132] and generative adversarial network (GAN)[133] that are developed on neural network are also generative models, their applications will be discussed in Section 4.

3.1.2. Algorithms for descriptors

Most ML models are "black box", which means that one does not know the explicit forms of the rules like Kepler's analytical expression, i.e., Kepler's three laws, behind data. Black box model is not easily accessed by many researchers, particularly by experimentalists unfamiliar with theoretical algorithms.[134] To be specific, researchers without resources or experience of the codes/software, could not adopted ML approach. Therefore, the ML algorithms that could obtain straightforward analytical forms of the rules, also known as descriptors, are critical in materials science.

Symbolic regression (SR)[135] is a supervised machine learning technique that seeks to find or determine the best mathematical formula (descriptor) to represent a relationship between the features and target. Taking the genetic algorithm (GA) based SR as an example, the main idea is to employ GA to simulate the evolution of analytical formulas by the natural rule of "survival of the fittest" and the science of heredity like Mendel's law. A comprehensive introduction about SR in materials science can be found in the review of Wang *et al.*[136]

The GA based SR (Fig. 5) usually contains two loops. Taking grid search of hyperparameters as examples, in the outer loop of the program, the grid search of hyperparameters is used to optimize the performance of the whole program. And in the inner loop, the program will firstly carry out the process called 'building block', which focuses on the setup of nodes like 'plus(+), minus(−), multiply(×), divide(÷)' and variables or constants to crate analytical forms. Then, the error (fitness) between the predicted value and target value will be calculated. The best form will be stored in the 'pool' which contains the winner of each generation. Finally, according to the termination criteria, SR can determine whether it should stop here and carry on the hyperparameters optimizing process or execute genetic operations and add descendants to the individual form and repeat the process above. When the hyperparameters optimize process also finishes, the program will output the best form in the pool.

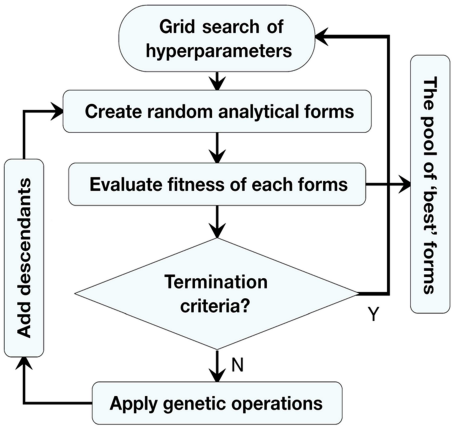


Fig. 5. The workflow of the SR program. Modified based on Fig. 2 of Ref. [355].

AI Feynman[137] (Fig. 6) is another powerful SR method inspired by physics. In contrast to general problems, scientific formulas representing physical rules usually have the properties of symmetric, separable, or dimensional consistency of variables and results. Based on that, AI Feynman algorithm uses serval sub algorithms to simplify the corresponding SR procedure.

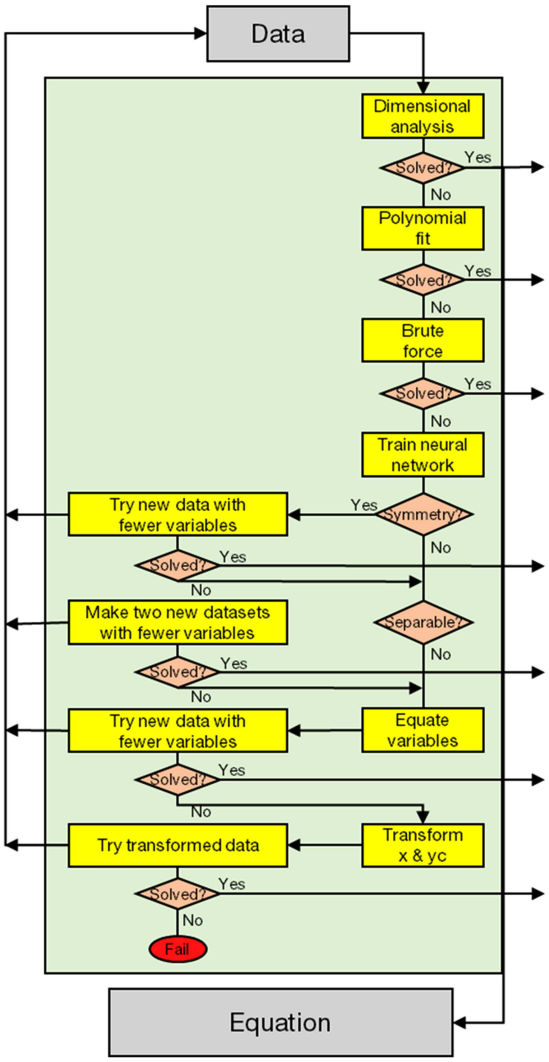


Fig. 6. The structure of the AI Feynman. Reprinted with permission from Ref. [137].

The sub algorithm contains (1) Dimensional analysis: converting the physical quantities into dimensionless quantities. (2) Polynomial fit: using a specific module to check whether the function can be solved by low order polynomials, and get the coefficients of the polynomials if it is. (3) Brute force: using common SR method to obtain the analytical formula. (4) Train a neural network: training a neural network to predict the output given its input which are not in the database. (5) Translational symmetry and generalizations: using the returning result to check if any two variables have the property of symmetry, and transform them to single one variable. (6) Separability: testing the additive and multiplicatively separability of the any two variables and separate it if we could, and it will

help us solve equations with fewer variables (7) Equate variables: using neural network to check the effect of setting any two variables, and try to solve the equation with only one of the two variables, which helps us get rid of the denominator. (8) Transform data: transforming the original data by functions like square root, log, exp, etc. Recent research shows that 100 equations from the Feynman Lectures on physics can be obtained by AI Feynman.[137] In comparison, only 71 of 100 was obtained by previous software (Eureqa)[138] for SR.

The sure independence screening (SIS),[139] compressed sensing[140] method along with the LASSO and PCA that mentioned before can also be used for extracting descriptors. Luca *et al.* developed the LASSO $+\ell_0$ methodology combining LASSO with compressed sensing for learning physical descriptors.[141] Praveen *et al.* proposed the bootstrapped projected gradient descent (BoPGD)[142] method for material scientists to distill descriptors from data. Recently, the sure independence screening and sparsifying operator (SISSO)[134] proposed by Ouyang *et al.* that integrates the SIS, sparsifying operators like LASSO, and the work from Ref. [141] is reliable and powerful for identifying descriptors.

3.1.3. Algorithms for small data and iteratively design

In general, ML algorithms require a quantity of data, as large as hundreds, thousands, or even millions to avoid overfitting. Though we are in a data-drenched era, obtaining reliable and comparable data from certain experiments or calculations could be extremely costly in materials science. Therefore, the algorithms that could recommend the optimal target based on a small amount of data, have a considerable impetus for materials research. This kind of methodology is usually called active learning, also known as surrogate-based optimization or adaptive design.

Sometimes, conventional ML algorithms are adopted into a material design loop for recommendation of next target in small data problems.[6,24,143–147] ML python libraries that can automatically select ML algorithms and optimize hyper parameters (introduced in Subsection 3.2.2) could be utilized. For instance, Tran *et al.*[145] developed an active learning framework to screen electrocatalysts for CO_2 reduction and H_2 evolution. Their framework combines DFT calculations with an auto ML tool named Tpot.[148] Recently, the same framework was employed to explore promising CO_2 electrocatalysts, and Cu–Al alloys were predicted and experimentally validated to have superior performance.[149]

In additional to conventional algorithms, specific algorithms for small data and recommendation problem have been developed. Here, we introduce two of them: Bayesian optimization (BO) and Monte Carlo tree search (MCTS). BO[150] is based on Bayesian modeling and has advantages in finding the maximum (or minimum) of unknown functions. The forecasts of BO rely on both the mean value and uncertainty. Figure 7 is a typical illustration of BO, with the x-coordinate axis and the y-coordinate axis as the features (material candidates) and the target (figure of merit), respectively. As shown in Fig. 7(a), the red line is supposed to be the true function. Based on the four sampling points, BO can predict the mean value (the blue line) and the confidence interval (the gray region). If we only considered mean value, the predicted maximum value was sampling point 1. By combining confidence interval, the maximum value was predicted to be the point 5 (Fig. 7(b)). One then obtains the true value of the fifth point from either experiments or computation and adds the true value to the training data to iteratively optimize the BO model. In the cycle of prediction, verification, and optimization, the prediction function gradually approaches the true function, and the prediction result is more reliable.

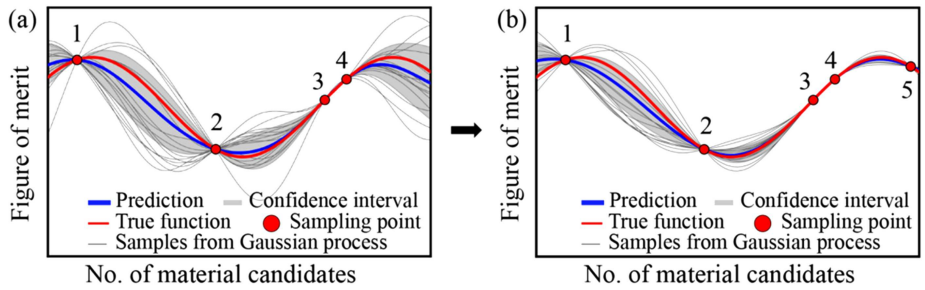


Fig. 7. The mechanism of the BO method. Reprinted with permission from Ref. [150]. Copyright (2020) Springer Nature.

MCTS[151] is a tree-based search method, which represents the search space as a tree structure, and then makes the tree grow from the root node to the most promising direction iteratively. Each iteration of MCTS is composed of four parts: selection, expansion, simulation, and backpropagation (Fig. 8). In the selection part, the node with the highest upper confidence bounds (UCB) value is selected among all the nodes that existed. Then children nodes will be expanded under this node in the expansion part. In the simulation part, a complete path determined by the optimization problem is created randomly. In other words, the tree selected one random node at each level until it reaches the deepest level, i.e., the

bottom of tree. This process called rollout or playout is to measure how promising the chosen node is, it could be implemented for many times and take the average or use Bayesian method to obtain reliable measurement.[152] Finally, the visit times and the scores of each node are updated in the backpropagation part. The ultimate result is obtained when the predetermined number of playouts or the bottom of tree is reached. MCTS behaves more efficiently in some optimization problems compared to Bayesian optimization.[151–153]

Recently, the popular and evolving reinforcement learning (RL) methods have also been used to guide materials design, such as Q-learning, deep Q network (DQN), policy gradients, etc. These algorithms are trained to take the best action (e.g., choosing the optimal experiment conditions) under a certain environment they meet through maximizing the reward function.[154] Specifically, for instance, the reward function is one part of the loss function in the policy gradients algorithms, so optimizing the loss function by gradient descend similar to the neural network is the actual process it carries out. The application in material science of RL methods will be discussed in Section 4.

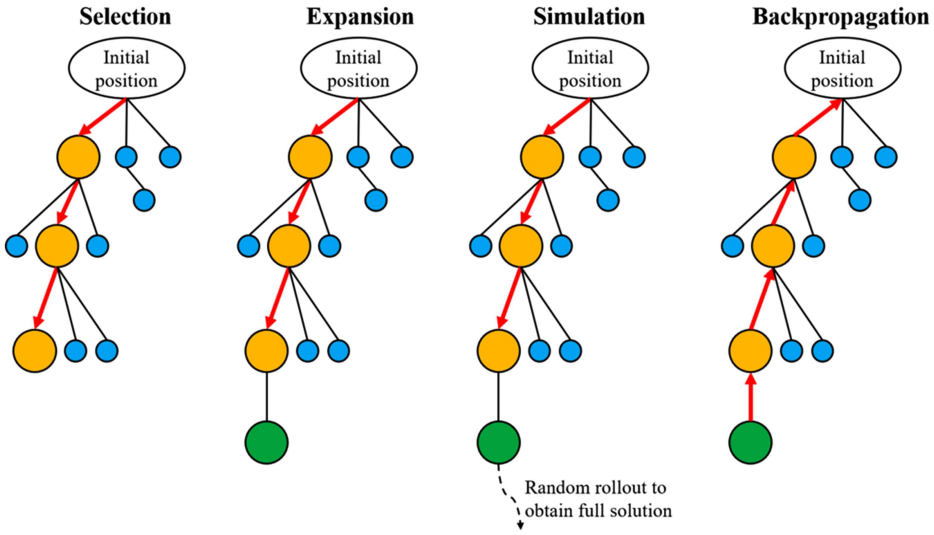


Fig. 8. Four main parts of MCTS. Modified based on Fig. 2 of Ref. [151].

3.2. Implementation of algorithms

3.2.1. Validation and evaluation

Before the training of an ML model, the data is usually divided into three parts: training, validation, and testing sets if the data is sufficient. The testing set should be opted out at first and never be seen by ML algorithms throughout the training process because only the unseen data could evaluate the predictability of algorithms. Based on this data division, a classic method to evaluate the performance of ML model is the k-fold cross-validation (CV). It separates dataset into k subsets of the same size and uses one of these k subsets for testing while others for training, then calculates the average result if all the subsets have been used for testing. Bryce *et al.* proposed a method named leave-one-cluster-out cross-validation (LOCO CV) that could better estimate the performance of ML algorithms for materials discovery.[155] Notably, such kind of data division may not be suitable for small data sets, e.g., smaller than one hundred data points,[156] because the noise in the data could play an important role when calculating errors and the results could fluctuate a lot.

MAE, MSE, RMSE introduced in Subsection 3.1.1 are usually chosen as the fitness function to evaluate the performance of regression model. Nevertheless, sometimes these functions are very sensitive to data, especially the abnormal data. In this sense, the Pearson, Kendall, and Spearman correlation coefficients are alternative kinds of functions for evaluation that could mitigate data sensitivity. For example, the Pearson correlation coefficients can be expressed as

$$r_{ij} = \frac{\sum_{k=1}^{n}(X_{ki}-\bar{X})(Y_{ki}-\bar{Y})}{\sqrt{\sum_{k=1}^{n}(X_{ki}-\bar{X})^2}\sqrt{\sum_{n=1}^{n}(Y_{ki}-\bar{Y})^2}}, \tag{9}$$

where $\bar{X}$ and $\bar{Y}$ are the average values. The abnormal data also enlarge the denominator so their influence on the overall result could be greatly reduced. Meanwhile, these three correlation coefficients normalize the regression evaluation into the $(-1, 1)$ interval so that a relatively uniform evaluation criteria could be obtained. These features make them advisable for material science in some cases because of the variety of data quality and scale used by researchers. The commonly used R^2 criterion is the square value of Pearson correlation coefficients.

For a classification problem, a conventional loss function

is the cross-entropy

$$L(f) = -\frac{1}{n}\sum_{i}\sum_{c=1}^{m} y_{ic}\log(p_{ic}), \tag{10}$$

where m is the number of categories, y_{ic} equals 1 if category c is the same as the category of sample i otherwise 0, p_{ic} is the prediction probability that sample i belongs to the category c. The results of the model on the testing set can be divided into four cases for each category: true positive (TP), false negative (FN), false positive (FP), and true negative (TN). Take double perovskite category as an example, a double perovskite is correctly classified means TP, other types of perovskite (e.g., single perovskite) are incorrectly classified as double perovskite means FP. The TP, FN, FP, and TN are usually put together to form a so-called confusion matrix which is widely used in classification problem. Other than the simplest evaluation called accuracy, i.e., the ratio of data correctly classified, the precision, recall, and F score are three important indicators. The precision and recall are defined as TP/(TP + FP) and TP/(TP + FN), respectively. Based on these concepts, more comprehensive evaluations can be obtained by adjusting the threshold of classification models such as precision–recall (PR) curve, receiver operating characteristic (ROC) curve, area under curve (AUC), etc.

3.2.2. Hyperparameters and overfitting

In ML, the optimization process can be divided into parameter optimization and hyperparameter optimization. The $\hat{\boldsymbol{w}}$ in model $f(\boldsymbol{x};\hat{\boldsymbol{w}})$ is parameters of models that can be learned by optimizing the loss function. In addition to learnable parameters within a fixed model structure, there is another class of parameters called hyperparameters that define a model structure or an optimization strategy such as regularization coefficient, learning rate, etc. A primitive but robust way of hyperparameter optimization is the grid search method that acquires an advisable set of hyperparameter configurations by trying all the combinations of hyperparameters. However, in the case of several learning algorithms on several datasets, random search behaves more efficiently than grid search.[157] In terms of efficiency, the Bayesian optimization and MCTS introduced in Subsection 3.1.3 perform better in principle. The neural architecture search (NAS) proposed by Zoph *et al.* is a promising method that automates the design of neural network architecture through reinforcement learning.[158] Notably, a variety of tools could automate the hyperparameter optimization such as the Tune,[159] Tpot,[148] Nni,[160] Hyperas,[161] etc. In material science, the hyperparameter optimization is integrated in the materials simulation toolkit for machine learning (MAST-ML) developed by Jacobs *et al.*[162] These methods and tools would help materials researchers develop ML models faster and thus boost the development of material science.

Overfitting and underfitting can be described with the concept of bias and variance. The underfitting means high bias because the model is too far from the targets. The overfitting means high variance because the model fits the training data so precisely that a slight variation of data could cause a considerable variation of the model parameters. Thus, the overfitting model would perform poorly on unseen data which means the lack of generalization capability. For instance, a model that fits the transition temperature of superconducting materials perfectly on training data may be incapable of predicting new materials.

Several measures to reduce overfitting have been discussed previously like data augmentation, adding regularization in loss function, ensemble methods, and appropriate datasets partition and evaluation. Appropriately adjusting hyperparameters is an efficient method to mitigate overfitting. For neural networks, a classic and efficient way is the dropout method[163] that randomly discards some neurons in training, this method is widely used in most types of deep neural networks. On the other hand, as for the small datasets with considerable noise, the Occam's razor principle could be applied to refine models: a simple model is less likely to be overfitting. In general, a robust and reliable way to measure the overfitting is the learning curve (Fig. 9) that illustrates the loss function values on training and validation sets versus the size of training sets. For example, a model for predicting the transition temperature of superconducting materials achieves lower MSE on testing set compared to training set is a good result. However, without learning curve, the better performance on testing set than training or validation set is not hard evidence of no overfitting, it is probably just an instant moment of an oscillating learning curve and the model still lacks generalization capability.

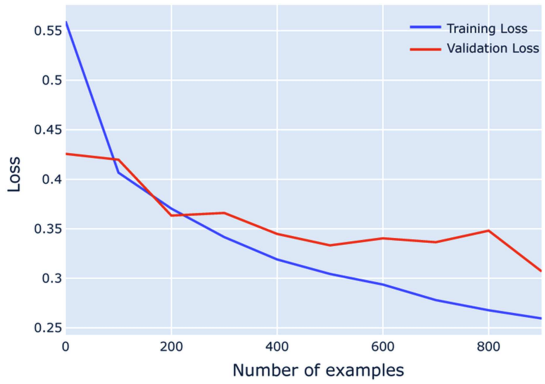


Fig. 9. Typical learning curve, y axis refers to the value of loss function, and x axis is the number of examples.

4. Application

4.1. Crystal structures

Generally, there are mainly two kinds of ML application related to crystal structures: classification and searching. The

former utilized ML algorithms to distinguish different structures from existing information like chemical composition and XRD patterns. The later aims to explore the optimal structures which meet certain conditions (e.g. minimum free energy, proper bandgap).

4.1.1. Structures classification

Oliynyk *et al.*[164] developed two ML models (partial least-squares discriminant analysis (PLS-DA) and SVM) based only on elemental information for the classification of seven types of binary AB compounds: CsCl, NaCl, ZnS, CuAu, TlI, β-FeB, and NiAs. They collected the data of 706 compounds from the PCD and ASM (automatic storage management) databases and the PLS-DA and SVM model achieved an accuracy of 77.1% and 93.2% on validation set, respectively. A hypothetical structure of RhCd was predicted to be CsCl-type by SVM model and further validated by experiment, demonstrating the predictability of their model. They[165] extended their previous work to 1556 ternary equiatomic ABC compounds in PCD. The SVM model was trained for classifying these structures into seven types: TiNiS, ZrNiAl, PbFCl, LiGaGe, YPtAs, UGeTe, LaPtSi. After achieving accuracy, sensitivity, and specificity of 96.9%, 97.3%, and 93.9%, respectively, the SVM model was utilized to make a prediction for 98769 possible ternary compounds in order to show a structural preference for most ternary composition. For some compounds, their model indicated a confused region where multiple structure types exist under certain experimental conditions. This region was validated by the synthesis of TiFeP, existing in both TiNiSi- and ZrNiAl-type structures.

Xu *et al.*[7] collected the structural data of 3354 single and double perovskite candidates from MP database. Sixteen kinds of ML models were then trained to predict the formability of perovskites (i.e., classifying perovskite and non-perovskite). All these ML models achieved the CV accuracy about 90%. Notably, there were some unreported perovskites predicted to be perovskite by ML models, which deserve further experimental validation. For example, the Ba_2NbCrO_6 was predicted to be perovskite by 13 of 16 ML models. ML model for classifying structures into perovskites and non-perovskites was also served as the first step to search for high-temperature ferroelectric perovskites in the work of Balachandran *et al.*,[6] which will be further discussed in Subsection 4.2.5. These ML works mentioned above classified crystal structures into a few groups (i.e., perovskites and non-perovskites), all of which have abundant structures. In some cases, the imbalanced data distribution at each group is a crucial issue: some groups may only contain very few data which makes ML difficult to classify. Graser *et al.*[166] proposed one solution to overcome this problem, they grouped materials types with less than a certain number (cutoff number) of data into one type called others. The RF model they developed was trained on 24913 data extracted from PCD and achieved an accuracy of 97% and 85% with a cutoff number of 150 and 10, respectively. Recently, Zhao *et al.*[2] trained RF and neural network models based on compositional features to predict crystal systems and space groups of material samples in MP database. A method named synthetic minority over-sampling technique (SMOTE)[167] was utilized to mitigate the imbalance of data, and RF with features calculated by Magpie tool achieved the best performance.

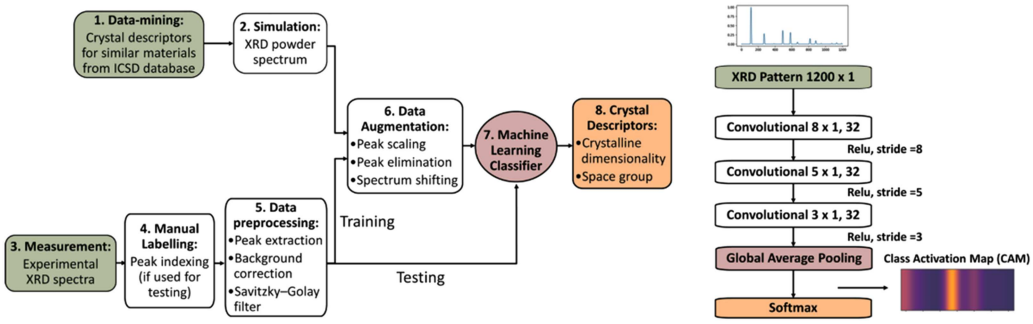


Fig. 10. (a) The workflow of classification of XRD data with data augmentation method. (b) The structure of CNN model they used. Reprinted with permission from Ref. [2].

Other than using features based only on chemical compositions to classify materials structures, x-ray diffraction (XRD) patterns could also be utilized to train ML models. Park *et al.*[168] prepared 150000 XRD patterns calculated from structural solution data in ICSD. Then CNN models were trained for space group, extinction group, and crystal system classification and achieved an accuracy of 81.14%, 83.83%, and 94.99%, respectively. Notably, two experimental powder XRD patterns of novel structures outside the ICSD database were correctly recognized by CNN, which showed the predictability of their model. Ziletti *et al.*[4] also utilized CNN based on simulated XRD patterns to classify crystal structures. Over 100000 crystal structures were effectively classified by their CNN model into eight crystal classes, and the attentive response maps method for improving the interpretability of CNN was used to reveal what the CNN leant. Naturally, ex-

perimental XRD patterns can also serve as the input for ML model of crystal classification. However, the relatively high cost of XRD experiments leads to the scarcity of data, thus limiting its ML application. To mitigate this issue, Oviedo *et al.*[2] performed data augmentation strategy by combining simulated XRD data and small number of real XRD data. Figure 10(a) shows the main workflow of their work. CNN (architecture can be seen in Fig. 10(b)) was trained on data after data augmentation to predict the crystallographic dimensionality and space group of thin-film materials and reached an accuracy of 93% and 89%, respectively.

4.1.2. Structures searching

Searching for materials structures with expected properties is another main direction for ML methods application. For example, seeking the most stable structure configuration of certain composition at certain temperature and pressure via as few searching trials as possible, this task is usually integrated with DFT calculation tools like VASP and QE for the estimation of formation energy or total energy.

First, take the genetic algorithm (GA) as an example. Oganov *et al.*[169,170] developed a tool named universal structure predictor: evolutionary xtallography (USPEX) for this task. This tool was based on the GA which has been discussed in Subsection 3.1.2 and nearly 100% successfully found the most stable configuration of dozens of prototypes with up to 20 atoms in the unit cell. Lyakhov *et al.* improved the efficiency of USPEX and enabled the application of larger search space (up to 200 atoms in the unit cell), they also adapted the USPEX for searching cluster structures.[171] Wu *et al.*[172] developed their GA method for structure search and applied to study complex binary and ternary intermetallic alloys. The crystal structure analysis by particle swarm optimization (CALYPSO) developed by Wang *et al.*,[173,174] was built on another classic search algorithm called particle swarm optimization (PSO). CALYPSO had been successfully applied in the design of various materials structures such as 3D crystals,[174] clusters,[175] and 2D materials.[176] Other algorithms that have been applied in materials structures search includes the *ab initio* random structure searching (AIRSS),[177] simulated annealing,[178–180] basin hopping,[181–183] and minima hopping.[184,185]

Recently, Bayesian optimization (BO) and Monte Carlo tree search (MCTS) have been increasingly applied in structures search. Yamashita *et al.*[186] applied BO for the most stable structure search in systems like NaCl and Y_2Co_{17}, saving 30%–40% computational cost compared to random search. Seko *et al.*[187] utilized BO in combination with cluster expansion method to seek stable oxygen-deficient perovskite structures. Not only existing structures in literature were successfully rediscovered, but their method also recommended some new structures, demonstrating the prospect of BO in application of atom-deficient structures. Dieb *et al.*[152] applied MCTS to search for the most stable structure configurations of boron-doped graphene with different concentrations. Bayesian rollout was integrated into the simulation part of MCTS in order to improve the performance, as a result, only less than 2% search space was explored to find the optimal configuration and their method performed better than sole BO in large search space.

4.2. Materials properties

At present, most of ML works about materials science, summarized in Table 4, focus on predicting materials properties based on structures and compositions. In this review, because of the relatively intensive ML works in these fields, we focus on five aspects including thermodynamic stability, band gap, superconductivity, thermal conductivity, and Curie temperature. Basically, these works are introduced in chronological order so that the update of research concerns and the improvement of ML methods and performance could be shown.

Table 4. ML application for some materials properties.

	Conventional	Deep learning	Tree-based	Descriptor	Active	Unsupervised
Thermodynamic stability	[86,188–193]	[26]	[188–190,194–196]	[195]		[86]
Band gap	[31,193,197–204]	[20,203,205]	[31,196,201,204]		[33]	[206]
Superconductivity	[207–210]	[211,212]	[19,213,214]	[215]		[207,210,213]
Thermal conductivity	[79,216–221]	[79,218,222,223]	[79,224–227]		[216,222,228–230]	[216]
Curie temperature	[6,231–236]		[235–237]		[6]	[233,236]
Bulk and shear moduli	[238–242]	[97,102,243]	[25,240,244–246]	[246]		
Debye temperature and heat capacity	[239,242,247,248]		[25,248]			
Density of states	[87,249,250]	[251]				[249,252]
Dielectric breakdown strength	[23,253–255]		[255]	[253]		
grain boundary structure and properties	[256–259]	[260]	[257,258]		[261,262]	[263]
Lattice parameter	[264–266]		[266]			
Lithium ion batteries SOC and conduction	[22,267–274]	[275–277]	[22,273,278]			
melting temperature	[221,279–282]	[279]				[279]

Next, we summary some applications using universal ML frameworks for predicting multiple materials properties, many of these universal methods are compared in a fair environment so the difference between performance is clear.

4.2.1. Thermodynamic stability

Thermodynamic stability, quantitatively described by the formation energy or the energy above hull, is usually the first criterion for screening novel materials. The ML models for thermodynamic stability are able to accelerate screen stable materials without costly DFT calculations.

In 2015, Faber *et al.*[86] proposed the Ewald sum matrix, extended Coulomb-like matrix, and sine matrix developed on the Coulomb matrix to represent crystal structures. Using the aforementioned matrices as features, a KRR model was trained on a dataset of 3938 crystal structures extracted from the MP database to predict the formation energies of crystal structures. The MAE on testing set was reduced when increasing the size of training set, and it reached 0.49 eV/atom, 0.64 eV/atom, and 0.37 eV/atom on a training set of 3000 structures using three matrices, respectively.

In 2016, Faber *et al.*[191] developed a KRR model about the formation energies of elpasolite (ABC_2D_6) crystals. The model was trained on a training set of 10^4 crystals and yielded a MAE of 0.1 eV/atom which was comparable to the accuracy of DFT calculations. Then, this model was applied to predict the formation energies of around 2×10^6 elpasolite crystals constructed by main-group elements. Finally, 128 new structures with 90 unique stoichiometries were predicted on the convex hull and added to the MP database.

Ward *et al.*[189] extracted the formation energies of 435000 crystal structures from OQMD and used random forests (RF) algorithm to train the model mapping structure features and formation energies. A total of 271 attributes constructed by Voronoi tessellation and the structure composition calculated by the Magpie and Vassal tools were used as features to represent crystalline compounds. A KRR model was also trained along with the RF model to compare the performance of the features they used with other works. The MAE from cross-validation reached 80 meV/atom which was superior to the performance using the CM and the partial radial distribution function (PRDF). Moreover, the difference in performance between their features and CM and PRDF became larger with the enlargement of training set size when using RF.

Schmidt *et al.*[190] established a dataset containing around 250000 cubic perovskites through DFT calculations. Afterward, the ridge regression, random forests, extremely randomized trees with AdaBoost, and neural network models were applied to fit the distance to the convex hull utilizing this dataset. The extremely randomized trees with AdaBoost performed best among the four methods and its MAE was 121 meV/atom on training and testing sets containing 20000 and 230000 structures, respectively. Furthermore, they found that the ML models had relatively poor performance when including elements on the first row of periodic table or relating to magnetism. Following the conclusions in Ref. [190], Schmidt *et al.*[194] used a model based on extremely randomized trees in combination with DFT calculations to explore the stability of ternary intermetallic structures including two prototypes: the tI10-$CeAl_2Ga_2$ and the tP10-$FeMo_2B_2$. A total of 1893 new structures were predicted to be thermodynamically stable and the use of ML models saved about 75% computational time.

Li *et al.*[188] carried out two different tasks: classifying whether perovskite phase is stable at given chemical formula and regressing the energy above the convex hull (E_{hull}) values, 791 features of elemental property were generated by Magpie tools. In classification task, the logistic regression, SVM, decision tree, extra trees classifier, and ANN methods were trained on a dataset of 1929 structures from the work of Jacobs Jacobs *et al.*[283] Among them, the extra trees classifier performed best, which had an accuracy, F_1 score, and AUC of 0.93, 0.88, and 0.98, respectively. Meanwhile, linear regression, KRR, decision tree, extra trees regressor, and ANN were used for regression. The results showed that the best performing KRR model achieved the RMSE and R^2 of 28.5 meV/atom and 0.89, respectively, using leave-out 20% cross-validation.

The works discussed so far used features of high complexity or large quantity, Ye *et al.*[26] trained deep neural networks to predict the formation energies of ABO_3 perovskites and $C_3A_2D_3O_{12}$ garnets using only two features: the Pauling electronegativity and ionic radii. The ANN models were trained on a 64:16:20 random split dataset generated by DFT calculations and achieved the MAE of 7–10 meV/atom and 20–34 meV/atom on perovskites and garnets, respectively.

Li *et al.*[192] calculated the formation energies of 354 halide perovskites by DFT, used them as training set, and predicted the formability of 14190 halide double perovskite candidates $A_2B(\text{I})B(\text{III})X_6$. Three ML models KRR, KNN, and SVM were then trained to establish a map between the formation energies and the features of ionic radii and electronegativity. The best performing KRR model was then validated by 246 unseen experimental data of the perovskite formability (Fig. 11). The F_1 score reached 95.9% which showed the generalization ability of ML model and better performance than the traditional descriptor tolerance factor t.[284]

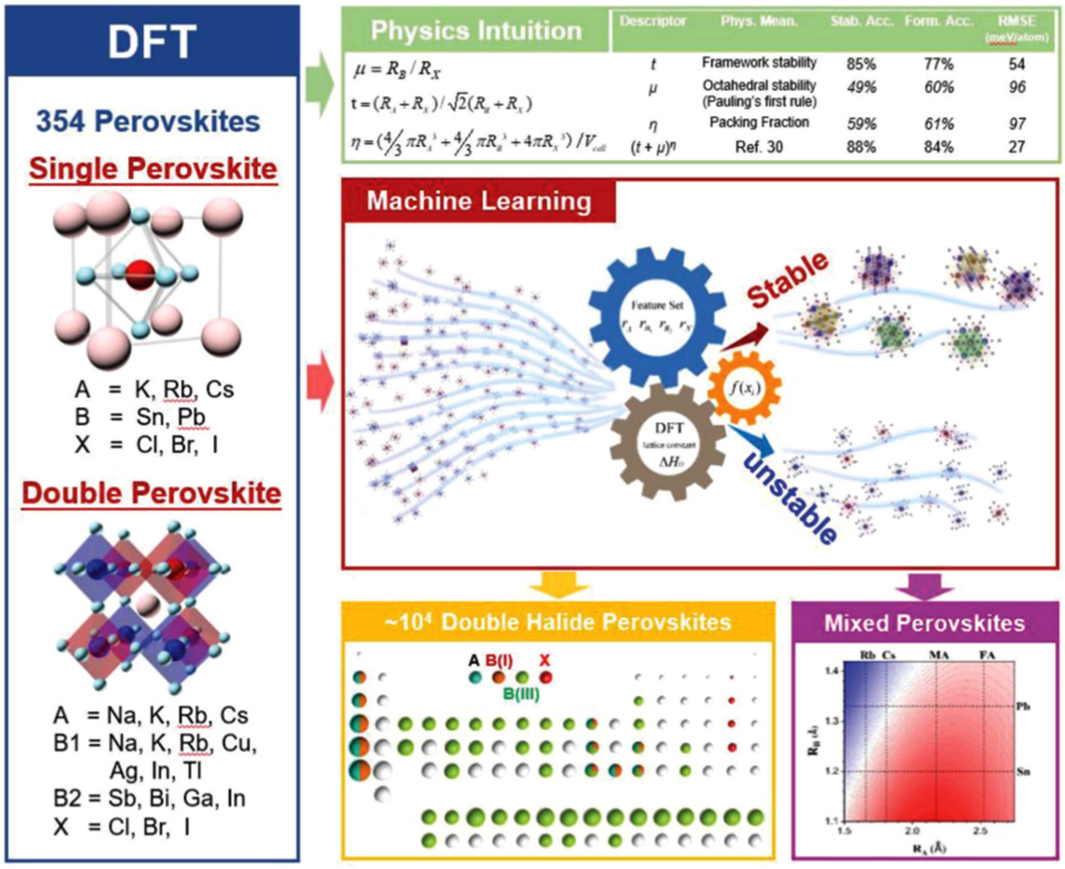


Fig. 11. The main workflow of screening stable halide perovskites via ML in combination with DFT calculations. Reprinted with permission from Ref. [192]. Copyright (2019) John Wiley and Sons.

Schleder *et al.*[195] utilized the XGBoost and SISSO method to predict the thermodynamic stability of 2D materials using the computational 2D materials database (C2DB). 2D materials were first classified into three classes: low, medium, and high stability. The features they used were produced through SISSO which combining atomic and structural properties. Then a regression model that achieved a 5-fold cross-validation RMSE of 0.165–0.245 eV was trained to predict the formation energies and E_{hull}. Moreover, 10 potential 2D materials for photoelectrocatalytic water splitting were found by HT screening.

Chen *et al.*[150] used BO based on a unified figure of merit for accelerating screening of halide perovskite photovoltaic among 432 candidates. The flowchart is shown in Fig. 12. Initially, the bandgap and decomposition energies of 20 random candidates were calculated. In order to quantitatively describe the potential of a material as a solar cell absorber, the decomposition energy and band gap were combined into a unified figure of merit, and BO was used to search for perovskite with stable and suitable band gap ($\Delta H_{\text{d}} > -29$ meV/atom, E_{g}, PBE in 0.6–1.2 eV). As shown in Fig. 12(b), BO is able to screen optimal candidates more efficiently than an exhaustive search via multiple parameters. This work opens a viable route for the search of optimal solar cell absorbers from a large amount of material candidates with less computational cost.

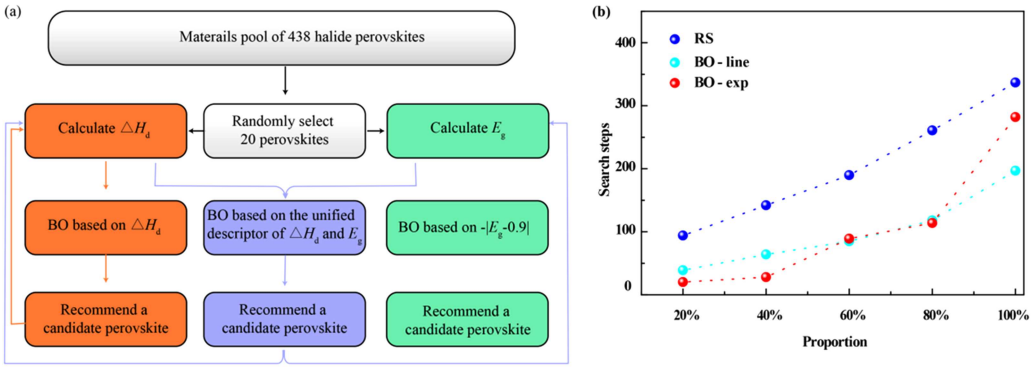


Fig. 12. (a) The search progress for the halide perovskites with ideal decomposition energy and band gap. (b) The performance of BO. Reprinted with permission from Ref. [150]. Copyright (2020) Springer Nature.

4.2.2. Band gap

Bandgap is of significant importance in the application of optoelectronic devices, such as photocatalysis, photovoltaic, and luminescene. Meanwhile, bandgap is a common and relatively easily-obtained materials property either by DFT calculations or experiments. Therefore, there is a large quantity of bandgap data stored in materials database. ML models for the band gap prediction without carrying out time-consuming experiment or DFT calculations could boost the discovery of novel optoelectronic materials.

Pilania *et al.*[197] trained a KRR model to predict the bandgaps of double perovskites using the data in the computational materials repository (CMR) through 5-fold cross-validation. The RMSE by using 4 selected primary atomic features and 16 compound features filtered by LASSO was 0.5 eV and 0.36 eV, respectively. Among all the features, the lowest occupied Kohn–Sham (KS) levels of A-site and elemental electronegativities of B-site are two crucial ones having influence on the bandgaps. The predictability of the model was further validated on the unseen data of 53 single perovskites. Considering the well-known bandgap underestimation of DFT-PBE, Lee *et al.*[198] predicted G_0W_0 bandgaps through OLSR, LASSO, and SVR methods. Computationally efficient PBE gaps were utilized as features, together with cohesive energy, crystalline volume per atom, and basic elemental information. The optimal SVR model was optimized by 10-fold cross-validation and RMSE of 0.24 eV was achieved on a randomly split testing set.

An ML model with high predictability for materials properties usually requires a broad exploration of diverse structures as training set. However, sometimes the exploration had a high computational cost like the calculation of vast PBE band gaps in Ref. [198]. The multi-fidelity co-kriging approach based on Gaussian process used by Lee *et al.*[199] could mitigate this issue, the low-fidelity band gaps (e.g., PBE band gaps) were utilized but not directly as features to predict high-fidelity targets such as HSE06 band gaps. The input low-fidelity data for the complete set of targets was not required using this method and it had a good performance on a dataset of 640 double perovskites.

Without involving PBE band gaps as a feature, Rajan *et al.*[200] predicted the GW band gaps of MXenes (2D transition metal carbides and nitrides) using the KRR, SVR, GPR, and bootstrap bagging methods. The workflow of their work is shown in Fig. 13. 47 easily accessible features such as melting points and atomic radii were reduced to 15 by LASSO, and the GPR model which yielded a RMSE of 0.14 eV performed best.

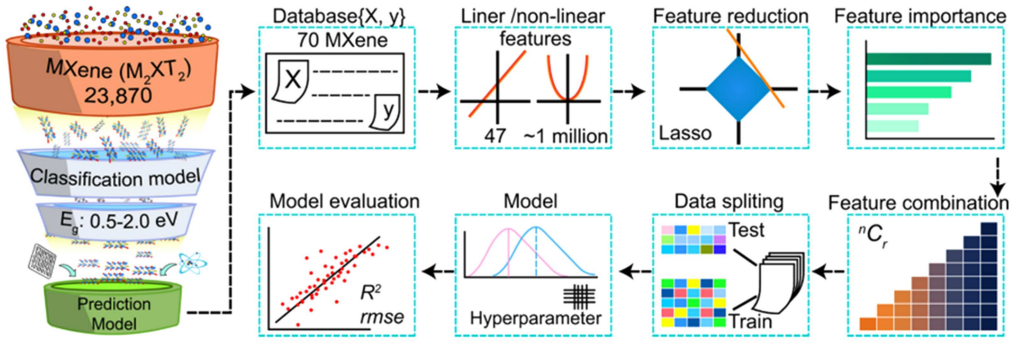


Fig. 13. Workflow of screening semiconductors from the MXene database and predicting band gaps. Reprinted with permission from Ref. [200]. Copyright (2018) American Chemical Society.

Weston *et al.*[201] first trained linear regression, linear and kernel SVR, decision tree, RF, and boosted regression tree models to regress the HSE band gaps of Kesterite I_2–II–IV–V_4 semiconductors which are promising materials in photovoltaic (PV) application. The kernel SVR obtained the smallest 10-fold cross-validation RMSE of 283 meV and R^2 of 0.957 on 184 HSE-calculated band gaps. Then the logistic regression model which achieved a F_1 of 0.89 was trained to classify between the indirect and direct band gaps. Finally, 25 promising structures were found to be stable and had appropriate band gaps.

All the aforementioned works trained ML models on computational data. In the work of Zhuo *et al.*,[202] they firstly used SVC method to classify inorganic solids into metals and nonmetals using 3896 experimental band gaps extracted from literature and got the accuracy and AUC of 92% and 0.97, respectively. Then an SVR model (RMSE: 0.45 eV) was trained to regress the band gaps of nonmetals. These models were employed to estimate the band gaps of 94095 materials in PCD database.

Dong *et al.*[20] constructed three kinds of CNN models to predict the band gaps of hybridized graphene and boron nitride. The 4×4 and 5×5 hybridized graphene supercell systems were described by 2D matrix (Fig. 14(a)) which served as the input of CNN models (Fig. 14(b)). All three CNN models achieved the prediction accuracy over 90% using computational data by quantum espresso. SVM method was also applied but had relatively poor performance. Then the band gaps

of 6×6 supercell systems were predicted by transfer learning and reached similar MSE and RMSE value. Olsthoorn *et al.*[203] adopted two methods to predict band gaps of large organic crystal structures: the KRR model with SOAP kernel and the SchNet model which is a deep learning architecture for materials science developed by Schütt *et al.*[104] The SOAP kernel defining the similarity between two structures had performed greatly in previous works,[285,286] thus it could skip some feature engineering works. The ensemble model achieved a MAE of 0.388 eV on 12500 crystal structures from OMDB-GAP1 database presented by themselves and this model was further applied to predict the band gaps of 260092 structures in COD database.

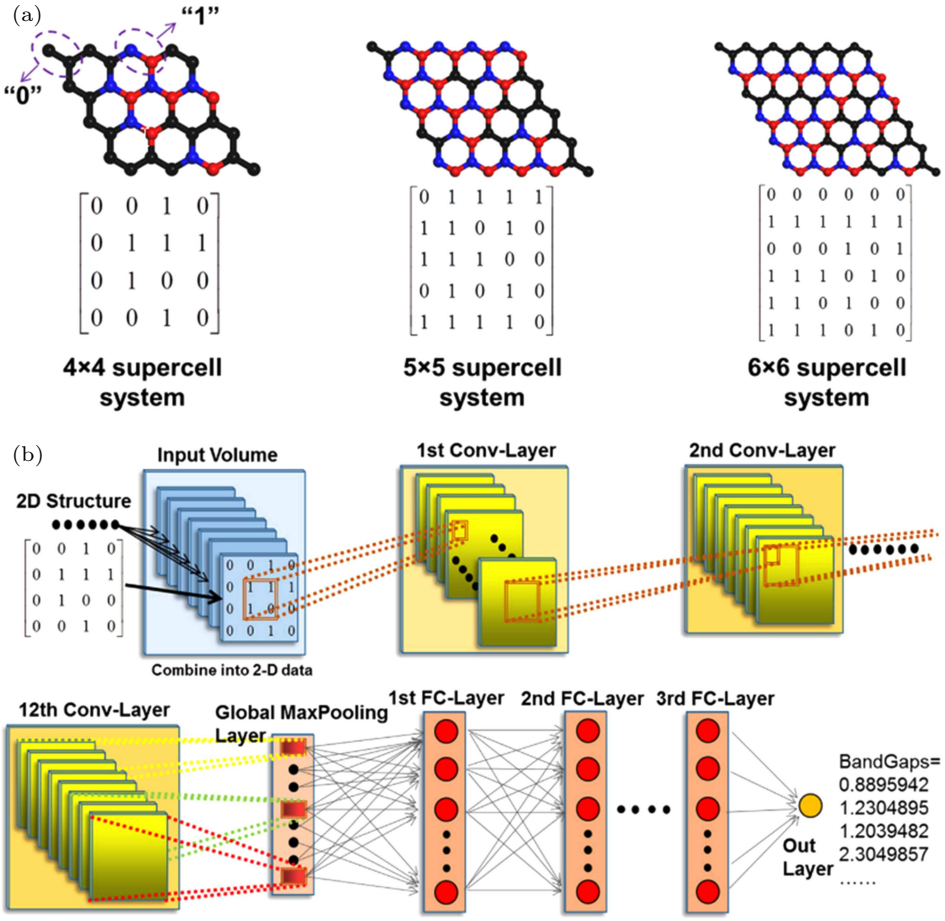


Fig. 14. (a) The representation matrix of doped graphene supercell systems. (b) One of the CNN structures to predict band gaps. Reprinted with permission from Ref. [20].

4.2.3. Superconductivity

Modeling the superconducting transition temperature (T_c) and the search for the unconventional high-T_c superconductors are currently challenging tasks. Since the theoretical model for unconventional superconductors has not been well-established, the ML models learning from data might provide different perspectives. Owolabi *et al.* applied SVM to predict the T_c of Fe-based superconductors[208] and doped MgB_2 system[209] using crystal lattice parameters as features. The RMSE on testing set for the former was 1.104 K and the percentage deviations of the latter reached 5.99%–10.46%. Stanev *et al.*[19] modeled the T_c for more than 12000 potential superconductors from the SuperCon database via the RF method. They firstly used chemical and structural features calculated by Magpie tool to train a classification model that identified materials with T_c below and above 10 K, this model achieved an accuracy of about 92% on unseen data. Then three separate T_c-predicted regression models for cuprate, iron-based, and low-T_c superconductors with $T_c > 10$ K were trained and got an R^2 of 0.88. Finally, these models were combined into a pipeline to search for new superconductors in ICSD and 35 promising materials were identified. Nevertheless, the models trained on one group of data did not have enough transferability on other groups (Fig. 15), this was later discussed by Meredig *et al.* who proposed the LOOCV.[155]

The PCA, particle swarm optimization (PSO), and SVR algorithms were combined by Liu *et al.*[210] to predict the T_c. Using datasets extracted from literature and dividing data by LOOCV, the PCA–PSO–SVR model with a corresponding MAE of 5.34 K performed better than the model without PCA and the neural network model they trained.

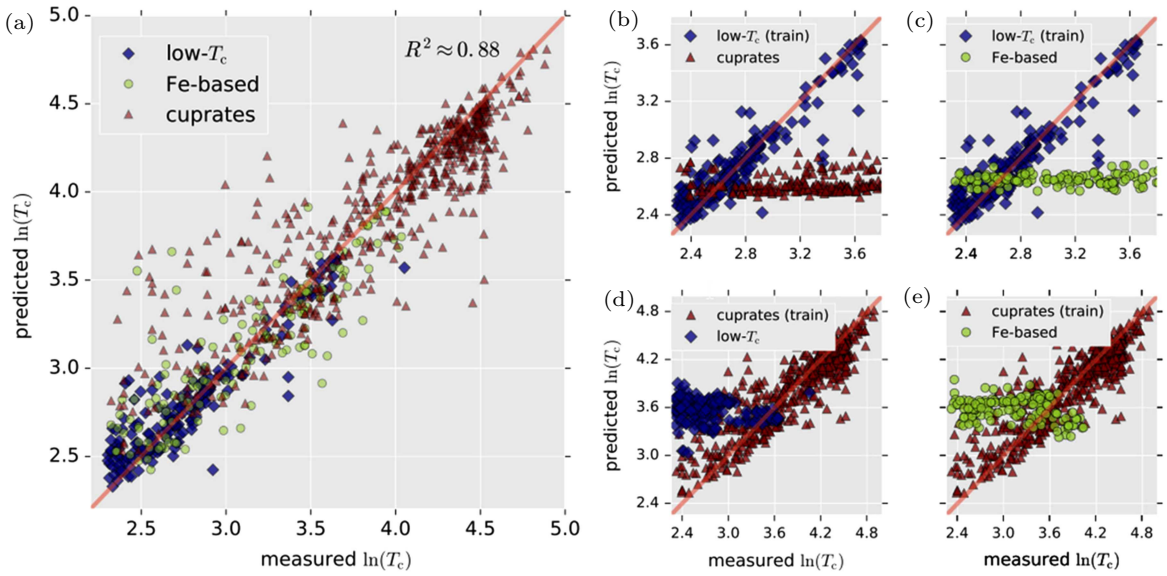


Fig. 15. (a) Prediction on testing set of low-T_c, iron-based, and cuprate superconductors. (b)–(c) Prediction from model trained on data only containing low-T_c materials. (d)–(e) Prediction from model trained on data only containing cuprate materials. Reprinted with permission from Ref. [19].

Another T_c-predicted work was carried out by Hamidieh *et al.*[213] using the SuperCon database. The XGBoost model with the out-of-sample RMSE of 9.5 K was trained and further applied to predict T_c of promising materials identified by Stanev *et al.*[19] According to feature importance analysis, the thermal conductivity, atomic radius, valence, electron affinity, and atomic mass affect the ML performance mostly.

A similar work based on the SuperCon database was carried out by Matsumoto *et al.*[214] Multiple ML models were used and it was found that the RF model which had the R^2 of 0.92 on testing data performed best, and the T_c of Fe-based superconductors outside the dataset were also well predicted.

Konno *et al.*[212] trained the first deep learning (DL) model for predicting T_c and achieved an R^2 value of 0.92. The DL model used only component information of materials as features through representing elements as one-hot vectors. They discovered the $CaBi_2$[287] superconductor outside of the SuperCon and COD databases, and the Fe-based high-temperature superconductors[288] (discovered in the year of 2008) based on training data before 2008.

Xie *et al.*[215] proposed simple analytical expressions for T_c using the SISSO method, the first expression was obtained from the data of 29 superconducting materials from the work of Allen and Dynes[289] and achieved the leave-one-out cross-validation RMSE of 0.26 K using only three features. This expression was slightly better the Allen–Dynes equation and could reasonably predict the H_3S[290] superconductor discovered recently at high pressure. Other analytical expressions with higher complexity and lower error were also discovered.

Recently, Le *et al.*[211] developed a generative ML model for predicting T_c in SuperCon database by means of the variational Bayesian neural network (VBNN). This method had more interpretability and reached a high R^2 value of 0.94. Notably, the RMSE of the model reached 3.83 K which outperformed all the previous works using SuperCon database.

4.2.4. Thermal conductivity

The low lattice thermal conductivity (LTC, κ_ω) is a key point for seeking thermoelectric materials with high conversion efficiency. Carrete *et al.*[225] estimated the LTC of half-Heusler semiconductors using theoretical descriptors and RF method. After exploring 79000 structures in the AFLOWLIB database, the structures with large atomic radii on two structural positions were found to have low κ_ω and three potential semiconductors with low κ_ω were proposed via HT screening. Seko *et al.*[291] applied the BO approach to search for LTC compounds in the MP databases containing 54779 materials based on 101 initial DFT-calculated data. A total of 221 new LTC materials were discovered and two of them with bandgap below 1.0 eV were proposed to be promising candidates for thermoelectric applications.

About 400 cubic oxide and fluoride perovskites at high temperatures were explored by Roekeghem *et al.*[216] via calculating the force constants of structures recommended from PCA iteratively. The thermal conductivities of the selected stable materials were calculated and 36 of them were considered as new potential stable perovskites. They also found that the thermal conductivities of oxides were much larger than those of fluorides.

Ju *et al.*[228] combined the atomistic Green's function and Bayesian optimization to seek the optimal Si–Ge composite interfacial structures which make the interfacial thermal conductance (ITC) across Si–Si and Si–Ge interfaces largest or smallest. Only several hundreds of structures (total search space containing 60000 structures) were calculated to find the optimal structure, which saved considerable computational

time.

The regression tree ensembles of LSBoost were used by Wu *et al.*[226] for (lattice thermal resistance) ITR prediction in nanoscale. Bi/Si system was selected from 2025 kinds of interfaces. The used data originated from the work of Zhan *et al.*,[217] where ML models with a best R value of 0.96 for predicting thermal boundary resistance (TBR) were built using GPR, SVR, and linear regression in combination with LASSO. The LSBoost model (Fig. 16) achieved the R value and RMSE of 0.93 and 11.9×10^{-9} $m^2 \cdot K \cdot W^{-1}$ and selected the optimal Bi/Si system for further analysis. Using the same ML methods, Wu *et al.*[227] trained ITR-predicting models via physical, chemical, and material features in their later work. The best model obtained a R value of 0.96 and these models were then used to screen 80282 material systems. As a result, 25 material systems with high ITR including Bi/Si system were recommended by more than two ML models.

Monte Carlo tree search, which is an alternative search method for BO, was implemented by Ju *et al.*[229] to study the effect of different interface roughness on interfacial heat conduction between Si–Ge double-layer nanofilms. Figure 17 shows the main workflow of this work. Possible roughness configurations constituted over 10^6 candidates in the searching space, while only a few hundred candidates were calculated to obtain approximate maximum and minimum interfacial thermal conductivity. For 2D materials, Hong *et al.*[79] carried out an ML work for the prediction of the interfacial thermal resistance between graphene and hexagonal boron nitride. The linear regression, decision tree, RF, and deep neural network models were trained only using the system temperature, coupling strength, and tensile strains as features. They trained models based on data generated via HT molecular dynamics (MD) simulation and the best deep neural networks had the MSE of 0.045×10^{-7} $K \cdot m^2/W$.

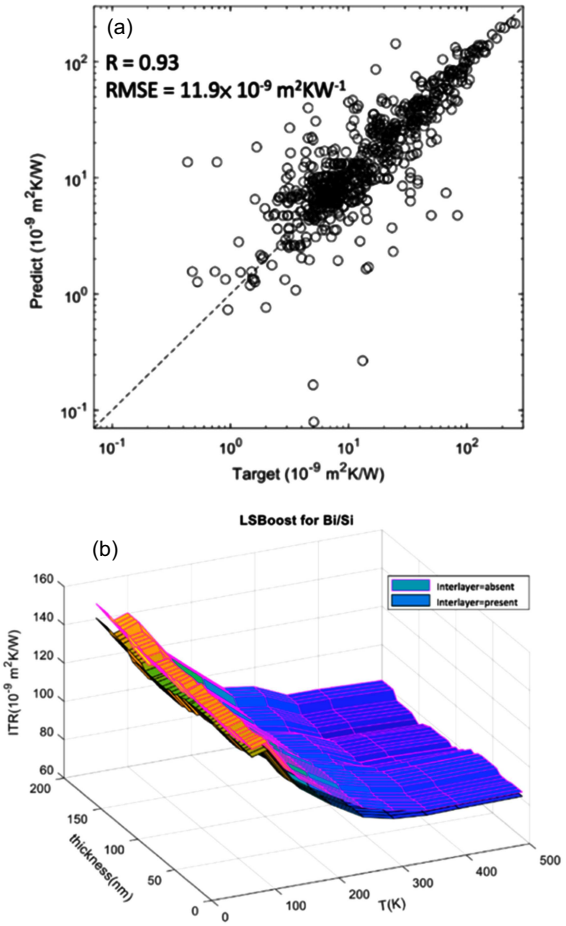


Fig. 16. (a) The prediction result for ITR from LSBoost. (b) The relation between ITR prediction and the thickness and temperature. Reprinted with permission from Ref. [226]. Copyright (2018) American Chemical Society.

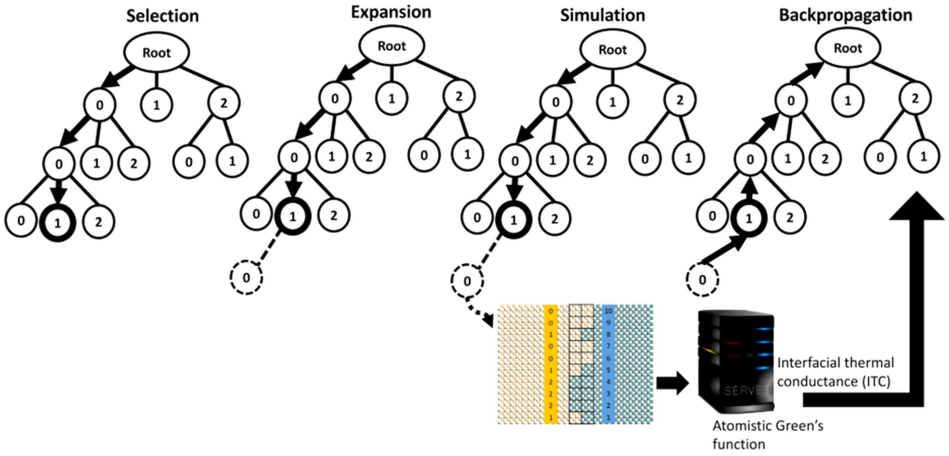


Fig. 17. The workflow of MCTS for optimizing the surface roughness. Reprinted with permission from Ref. [229].

Wei *et al.*[218] utilized the CNN along with the SVR and GPR methods to estimate the effective thermal conductivities of composite materials and porous media. The quartet structure generation set (QSGS)[292] was used to generate composite materials structures and the thermal conductivities of these structures were calculated by the lattice Boltzmann method (LBM).[293] The percentage RMSE and R^2 of the CNN model reached 1.9% and 0.986, respectively, and

the above three ML models outperformed existing domain-knowledge models developed in the field. Recent studies show that the nanoporous graphene structures containing disordered pores have lower thermal conductivity than those containing periodic pores.[294–296] In contrast, Wei *et al.*[230] utilized a search method based on genetic algorithm and successfully discovered the disordered configurations with higher thermal conductivity than the periodic ones. To optimize pores configuration for minimal and maximal thermal conductivity of graphene, Wan *et al.*[222] trained a CNN model that had R^2 of 0.97 on testing data and adopted the model to inversely design the optimal structures. This method was successful and efficient, only about 10^3 of the total 10^6 structures were calculated.

4.2.5. Curie temperature

Curie temperature (T_c) is the transition temperature of ferroelectric and paramagnetic materials, or ferromagnetic and paramagnetic materials. Following their prevenient work that studied the Curie temperature (T_c) of $BaTiO_3$-based ferroelectric perovskites assisted by SVR models,[232] Balachandran *et al.*[6] carried out an active learning work for the design of xBi[$Me'_yMe''_{1-y}$]O_3–$(1-x)$$PbTiO_3$-based perovskites with high T_c using data extracted from literature. The SVC model for classifying perovskites and non-perovskites and the SVR model for regressing the T_c were trained and integrated into an iterative experimental design loop (Fig. 18(a)). Finally, six perovskites including three new ones were successfully synthesized and their best T_c reached 898 K.

Dam *et al.*[233] trained 3.3×10^7 KRR models with different combinations of features to predict the urie temperatures of the rare-earth transition-metal stoichiometric alloys. They performed subgroup relevance analysis for features and the best model with 8 descriptors achieved an R^2 of 0.954. Aiming for estimating the T_c of the same type of materials, Nguyen *et al.*[234] developed five KRR models with different kernels based on 101 structures collected from the Atomwork database of NIMS. The rare-earth concentration was demonstrated to be the most important feature. The model with Gaussian kernel had an R^2 of 0.982 and an ensemble model was used to predict the T_c of $NdFe_{12}$, $GdFe_{12}$, and $DyFe_{12}$.

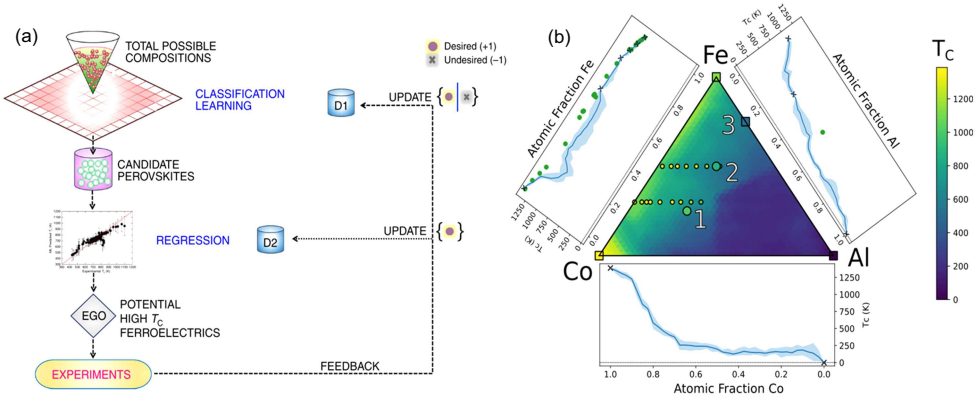


Fig. 18. (a) The design loop for searching high-temperature ferroelectric perovskites. Reprinted from Ref. [6]. (b) The relation between the T_c and chemical composition for the ternary system Al–Co–Fe. Reprinted with permission from Ref. [236]. Copyright (2019) American Physical Society.

Zhai *et al.*[235] performed a regulatory ML work to predict the T_c of perovskites. They extracted 47 perovskites from 9 publications and the genetic algorithm and greedy feature selection method were applied to select features. The SVM, relevance vector machine (RVM), and RF models were employed. SVM model owing R of 0.8549 and RMSE of 28.6659 K performed best. Furthermore, the model predicted $La_{0.66}Sr_{0.3}Ba_{0.04}MnO_3$ to have higher T_c than the materials in the dataset.

The T_c and the maximum value of magnetic entropy change ($(\Delta S_M)_{max}$) are two critical properties for magnetocaloric effect (MCE) materials. Zhang *et al.*[237] trained GBRT models for estimating these two properties of $La(Fe,Si/Al)_{13}$-based materials. Their models based on chemical composition features and the T_c-predicting model achieved an R^2 of 0.98 and 0.96 on training set and testing set, respectively.

Also using features based on chemical composition, Nelson *et al.*[236] studied the T_c of ferromagnets via ridge regression, KRR, RF, and ANN. In their work, a relatively big dataset of 767 compounds was constructed by combining four different data sources. Their best model had a MAE of 57 K on testing set and the generalization ability was further proven by analyzing the relation between the T_c and chemical composition for the ternary system Al–Co–Fe (Fig. 18(b)).

Recently, Balachandran[231] studied the relation between the T_c and B20 alloys structures though the SVR model, which was trained and tested on the experimental data of 19 B20 compositions extracted from literature. In combination with DFT calculation performed by quantum espresso, the Sn-substituted alloys achieving higher T_c in Fe(Ge, Sn) compounds were predicted by his model and validated by DFT

calculation.

4.2.6. Universal properties prediction

In general, there are two mainstream models that were applied for universal properties prediction: the conventional ML or deep neural networks methods using fixed features designed by domain knowledge (like CM, BOB, HDAD, etc.) and graph-based neural networks, such models represent material structure as graph which can be learned directly from data. The former can also be called fingerprint-based models and can outperform the graph-based models on small databases ($\sim$ 1000 data), but on large databases, the latter is more likely to be superior to the former.[297] In this section, we focus on applications using models with learnable features.

A comprehensive comparison of different features and ML models for 13 electronic properties of organic molecules was carried out by Faber *et al.*[298] Multiple features for molecules were studied, including the CM, BOB, BAML, and ECFP4 mentioned in Subsection 2.2 and the molecular atomic radial angular distribution (MARAD), histogram of distances (HD) histogram of distances and angles (HDA), and histogram of distances, angles, and dihedral angles (HDAD). The elastic net (EN), Bayesian ridge regression (BR), RF, KRR, graph convolutions (GC),[299] and gated graph networks (GG)[300] models were trained to predict 13 targets based on the QM9 database containing about 118 thousand molecules. The GC, GG, and KRR model had the lowest out-of-sample errors across 12 properties and the best model of all properties reached or went beyond the performance of DFT (B3LYP level).

Gilmer *et al.*[80] generalized some ML models based on graphs (such as the GC, GG, deep tensor neural networks (DTNN),[301] etc.) into a concept named the message passing neural networks (MPNNs). Then they tried to improve the MPNNs based on the same database and properties in the work of Faber *et al.*[298] Using the GC, GG, DTNN and serval methods utilizing the BAML, BOB, CM, ECFP4 and HDAD to represent molecules as baselines, their best model (enn-s2s) and its ensemble version (enn-s2s-ens5) outperformed all previous methods. In order to fairly compare the performance of different methods on various databases, Wu *et al.*[302] built a platform named MoleculeNet, it integrated all the ML methods mentioned above in this section into one part of the open source python package called DeepChem.

The SchNet proposed by Schütt *et al.*[104] was developed on DTNN and outperformed the enn-s2s method for learning molecules properties on QM9 database. When learning the formation energies of materials in materials project (MP) database, the SchNet had a lower MAE than using features extended from CM proposed by Faber *et al.*[86] Moreover, the SchNet was used to learn the potential energy surfaces and force field for molecular dynamics (MD) simulations, it achieved great accuracy on the MD17 baseline set[303] and was further applied to perform the MD of C_{20}-fullerene at the PBE + vdW^{TS} level which was impracticable for conventional DFT calculation.

Gradually, graph-based neural network models with better performance for general molecular or crystal properties prediction were developed. Jørgensen *et al.* extended the SchNet with edge updates networks, their method reduced the MAE by 23% on the QM9 database for learning 12 molecular properties, 27% and 46% on the MP and OQMD database for learning the formation energies, respectively. The multilevel graph convolutional neural network (MGCN) proposed by Lu *et al.*[304] had better predictive accuracy than the SchNet and DTNN on the QM9 and ANI-1[305] database (MAE reduced by $\sim$ 30%). Following the idea of many-body extension, instead of using graphs, the hierarchically interacting particle neural network (HIP-NN) proposed by Lubbers *et al.*[306] represented molecules as hierarchical terms to fit their properties, their model outperformed the MPNNs, DTNN, and SchNet on QM9 database and energies (MAE: 0.26 kcal/mol) from MD trajectories data. Enlightened by the SchNet and HIP-NN, Unke *et al.*[306] presented the PhysNet that improved the performance again compared to previous methods on the QM9, MD17, and ISO17[104] databases. Chen *et al.*[307] developed the path-augmented graph transformer networks (PAGTN), each node was related to all other nodes in the graph in order to better capture structures of large graphs. The PAGTN showed better performance than the MoleculeNet and conventional GC methods for the prediction of seven molecular properties on multiple databases. Yang *et al.*[297] carried out an all-round comparison of general ML models for learning molecular properties on 35 (19 public) material databases and proposed their own model: directed MPNN (D-MPNN), which outperformed all the baseline models on 12 public databases.

Meanwhile, for learning crystal properties, Xie *et al.*[97] developed the crystal graph convolutional neural networks (CGCNN) and tested the performance on seven properties (formation and total energies, Fermi energy, band gap, bulk and shear moduli, Poisson ratio) in the MP database. The CGCNN achieved the MAE comparable or superior to DFT accuracy and was further utilized to extract useful information about perovskites, which show its interpretability. Then Sanyal *et al.*[308] improved the CGCNN by combining the multitask (MT) learning based methods. Their method named MT-CGCNN reduced the MAE by over 8% compared to CGCNN for predicting the formation energies and band gaps, and it still performed better than the CGCNN using 90% training data. On the OQMD database, Yamamoto[309] developed the ensemble CGNN models without using bond distances to successfully regress the formation energies, band gaps, unit

cell volumes, and total magnetization.

To learn the properties of both molecules and crystals, Chen *et al.*[102] proposed a general framework called the materials graph network (MEGNet). On the QM9 database, the MEGNet presented lower MAE on 11 (13 in all) molecular properties than the enn-s2s and SchNet. On the MP database, the MEGNet was applied to predict the formation energies, elastic moduli, and band gaps, it also outperformed the SchNet and CGCNN. Moreover, they proved the feasibility of the transfer learning of models between large databases and small databases utilizing the MEGNet.

Recently, Klicpera *et al.*[310] developed the directional message passing neural network (DimeNet) considering the spatial direction between atoms which was usually ignored in previous methods. The performance of DimeNet went beyond all the prevenient methods again including the MEGNet, on average, the MAE was reduced by 31% and 76% on the QM9 and MD17 database, respectively.

4.3. Synthesis condition and characterization recognitions

4.3.1. Synthesis condition

Materials synthesis and characterization are indispensable to materials design and development. Traditional experimental synthesis is trial-and-error process based on the domain knowledge of experimentalists, thus time-consuming and economically costly. Recently, ML has been successfully applied in organic chemistry synthesis to assist with synthetic route design via retrosynthetic analysis or predict the products of chemical reactions given a set of precursors and specified reaction conditions.[311–317] Thanks to the high-throughput experimentation (HTE) in organic chemistry synthesis, it is a powerful tool to generate a crowd of comparable data, acting as a foundation of ML for reaction optimization over the past decades.[318,319] Ahneman *et al.* predicted the reaction performance of a palladium-catalyzed cross-coupling of aryl halides with 4-methylaniline using RF algorithm based on data produced by HTE, revealing the value of RF model in promoting the employment of synthetic methodology.[320] Segler *et al.* applied a retrosynthesis strategy to plan the syntheses of small organic molecules, the precursors of target molecules using MCTS algorithm combined with deep neural networks and symbolic artificial intelligence (AI). Even though ML algorithm can effectively boost the prediction fidelity of organic synthesis reaction and precursors of target molecules by inverse design, it is also necessary to provide an automatic platform to promote chemical synthesis, relieving time and effort investment from expert chemist.[9] Coley *et al.* took a step toward a paradigm of chemical synthesis that combines AI-driven synthesis planning and robotically controlled executions. The strategy for computer-aided organic synthesis was demonstrated for 15 medicinally relevant small molecules with high yields.[10]

Besides organic chemistry synthesis, in the controllable synthesis of inorganic materials, hydrothermal and solvothermal syntheses were deemed as a popular and facile approach to generate a mass of new materials.[12,321] Based on the above mentioned method in the crystallization of templated vanadium selenites, Raccuglia[78] *et al.* utilized unsuccessful hydrothermal synthesis databases which was gathered from archived laboratory notebooks of their laboratory, added the corresponding physicochemical property feature and reaction conditions to raw notebook using cheminformatics tools. They provided an SVM algorithm to train a machine-learning model to predict the outcome of reaction success, as indicated in Fig. 19. This inverse design method possesses much more excellent performance to forecast successful synthetic conditions with 89% accuracy than traditional human chemists by success rate of 78% for the organically templated inorganic product formation.

Based on the experimental data in a full-text publisher, Kim *et al.* proposed an automatic platform (http://www.synthesisproject.org/) that retrieves tens of thousands of articles and then extracts and codifies the materials synthesis recipes via natural language processing (NLP).[13] In light of this, they trained an ML model via a decision tree along with a linear classifier to excavate the underlying correlation between the synthesis condition feature and the specifically desired materials without prior knowledge of experimentalists. Through the above reverse design strategy, they further predicted the critical parameter of synthesized titania nanotubes via hydrothermal methods.

Post-treatment in experiment is also a facile and effective strategy to modify material property in synthesis processing.[322,323] As known, inorganic–organic halide perovskites have attracted immense concern in photovoltaics and light-emitting diode devices in the past few years, attributing to high photoluminescence efficiency and low fabrication cost.[324–326] However, challenges like stabilities and toxicities need to be surmounted before large-scale fabrication of their devices can be developed. To improve the stability of methylammonium lead iodide ($MAPbI_3$) films, a post-treatment approach with different types of amines by dip-coating or spin-coating craft is a simple but facile method.[327,328] Combined with experimental data, Yu *et al.* employed a series of ML algorithms to analyze their stability trend based on their compatibility on the $MAPbI_3$ films. The model has reached 86% accuracy on predicting the outcome of whether the perovskite films are kept after post-treatment. Amines with fewer hydrogen bond donors and acceptors, more steric bulk, a plethora of alternatives on the nitrogen site, and pyridine derivatives tend to have high compatibility with perovskites films.[329]

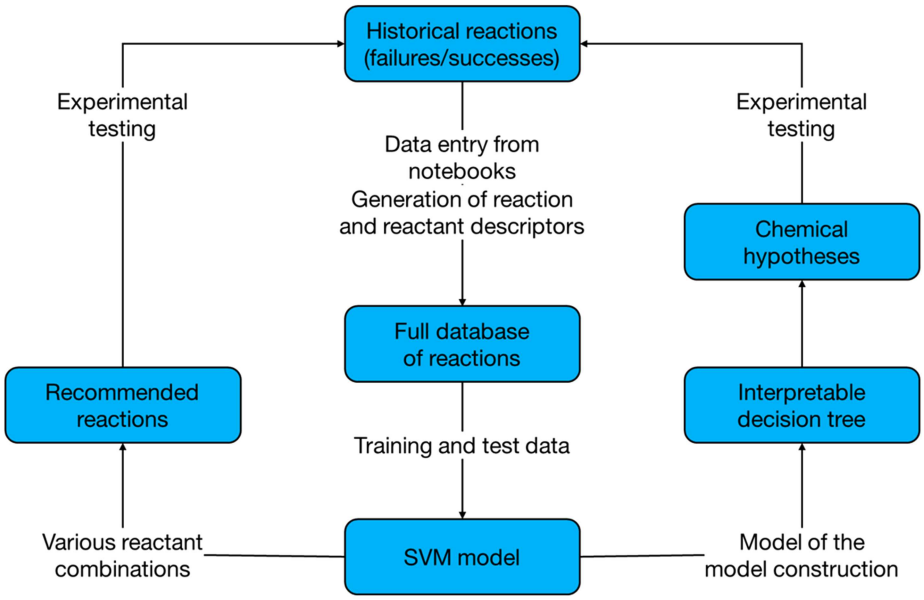


Fig. 19. The workflow of the work of Raccuglia[78] *et al.* Copyright (2016) Springer Nature.

Compared to conventional bulk materials, low-dimensional materials have attracted much attention for academic and industrial institutions due to the ample physical and chemical properties induced by the deduction of dimensionality since the emergence of graphene.[330,331] Transition metal dichalcogenides (TMDs) with a sandwiched structure, as a representative two-dimensional (2D) material, have recently spotlighted attributing to widespread application in nanoelectronics and spintronics. The reduction of lateral scale to one-dimension (1D) can further induce greater research interest due to the quantum confinement effect.[332–334] Chemical vapor deposition (CVD) method in experiment is an effective and controllable approach to synthesize high-quality low-dimensional materials, especially for TMDs. However, it is difficult to coordinate every growth condition with large regulation ranges which include carrier gas flow rate, precursor source ratio, reaction temperature, pressure, deposition time, ramp time, etc. ML is also accounted as a promising capability conducting experimental synthesis via effectively learning from the existing data then making recommendations as a retrosynthesis design strategy. Liu *et al.*[335] adopted an XGBoost classifier (XGBoost-C) together with a 10-fold nested cross validation algorithm to predict the optimal CVD synthesis condition of 1D few-layers WTe_2 nanoribbons based on 255 experimental data encompassing 141 successfully synthesized experiments and 114 negative results. H_2 gas flow rate mainly makes the evolution of 2D to 1D WTe_2, the ratio of Te/W control nanoribbon morphology with different length-width ratio. They provided a facile and effective approach to infer the optimal synthesis condition of low-dimensional nanostructures, opening up a new insight for intelligent material development. Another general synthesis means of 2D material is mechanical or chemical exfoliation method, such as graphene, TMDs, and MXene. Based on the theoretical simulations, a myriad of potential 2D materials were predicted, but a few dozens had been experimentally realized in a controllable way. Complementary for the experimental synthesis and high-throughput DFT (HTD) prediction, Frey *et al.*[335] provided a positive and unlabeled (PU) ML approach to predict the synthesizability of 2D MXene using elementary information and data from HTD calculations as inputs. They also identified 18 MXene compounds that are promising candidates to synthesize using chemical exfoliation means from MAX precursors.

4.3.2. Characterization recognitions

The characterization analyses such as spectroscopy and microscopy are essential to reveal the structural, optoelectronic, and other materials properties in experiments. Spectroscopy is a conventional mean to probe bonding information of material via the interactions of light ray and substance. Microscopy can explore the surface and morphology characterizations of materials through focused and accelerated electrons casting on the matter. Traditional experimental and simulated characterizations have accumulated abundant data to acquire material properties, but some structural and electronic information cannot be conveniently attained from the above mentioned data. ML provides a new paradigm to gain material properties on the existing data, overcoming time-consuming conundrum.

Raman spectroscopy is a spectroscopy technique used to study the vibration, rotation, and other low-frequency patterns of lattices and molecules. The covalent-modification of pristine multi-walled carbon nanotubes (MCNTs) does not lead to obvious changes in D/G ratio of Raman spectra. Therefore, it is not an easy task to quantify the Raman spectral changes

manually owing to tiny difference with and without functionalized MCNTs. Sato-Berrú *et al.*[336] used the principal component analysis (PCA) method to distinguish the Raman spectra of pristine and functionalized MCNTs.

X-ray free-electron lasers (XFELs) owning ultra-short high-brightness pulse have great potentials for a widespread application on physics, chemistry, biology, and material science to unveil the structural dynamics of matter. Attributing to inherent fluctuations, free-electron lasers usually lead to property measurement with low-repetition rate. Sanchez-Gonzalez *et al.*[17] optimized four ML models including a linear model, a quadratic model, an SVR, and an ANN to investigate the non-trivial hidden correlations between slow, complex diagnostics and fast diagnostics and make predictions of complex diagnostics using simple diagnostics as input with high accuracy of 97% for the spectral shape predictions. This approach opens a new opportunity for next-generation high-repetition rate XFELs characterization. Ghosh *et al.*[18] applied deep learning algorithm to predict molecular excitation spectra. Compared to traditional instrumental measurements and computational simulations, it is an effective paradigm shift avoiding time-consuming conundrum to accelerate spectroscopic analysis of materials, make predictions for novel and hitherto uncharacterized materials, and discover new molecules or materials. They adopted 10000 molecules of diastereomers datasets to predict their own spectra which helps us to point out interesting molecules for further analysis.

In catalytic reactions, the identification of active sites is a critical issue so as to decipher the reaction mechanism and design novel catalysts with high activity, selectivity, and stability. X-ray absorption specta (XAS) was accounted as an effective approach to accurately character geometric and electronic properties of catalyst. Despite the rich structure-specific information stored in the XAS data, extracting useful information from the spectra is challenging, especially in the condition of low metal weight loading, nanoscale dimension, and composition distribution. Timoshenko and Frankel[337] gave a perspective about the development on the XAS data analysis using supervised and unsupervised ML to reveal the relationship between spectra and properties based on the labeled data, and discover patterns in large sets of experimental data without any labels. This inverse design strategy can be extended to other experimental and computational spectra data analysis.

Microscopy characterization is also of particular importance to probe the structural and morphologic characters of materials. Attributing to digital data acquisition and storage, large amount of data can be gathered as microscopic images in the form of digital data from a simple sample. Analysis of microscopic images is a tedious work that needs practice and patience, usually done manually by microscopist after data collections. The results of analysis might be biased by human who hardly performs systematic and entire identifications, further leading to the results lacking consistency and reproducibility. Hitherto, alternative approaches are developed to automate the identifications of local structures on the microscopic images and show ability to obtain high-fidelity result as good as manual analysis.

A typical field for automating microscopy recognition is 2D materials. Since the first exfoliation of graphene in 2004,[330] the atomically thin 2D materials, such as hexagonal boron nitride, transition metal dichalcogenides (TMD), and MXene have attracted great attention attributing to their unique and exotic properties. Their fantastic functionalities are correlated with their layer counts and a tiny change of which may result in a significant difference in their performance.[330,338,339] In this sense, facile and accurate prediction of layer count for 2D materials on a large scale is of particular interest. Masubuchi and Machida[340] adopted data-driven clustering analysis to automatically identify the position, shape, and layer counts of exfoliated graphene flakes from optical microscopic images on the SiO_2/Si substrates. This analysis method can also be applied to a wide range of 2D materials and substrates. Additionally, based on the identification of graphene thicknesses using ML method, the study of graphene nanomechanics and other 2D materials has similarly led to exciting new properties in 2D crystals, such as their remarkable in-plane stiffness and out of plane flexibility.[341,342] Recently, an ultra-stiff phase, named as diamene, was found by force-induced chemical transformation of a few-layer thick graphene, which was grown in epitaxial and exfoliated graphene films on SiC and SiO_2 substrates, respectively.[343] Cellini *et al.*[344] proposed a spectral cluster analysis method to reveal the formation origin of diamene phase which exclusively derived from one-layer plus buffer layer epitaxial graphene using friction force microscopy (FFM) and atomic force microscopy (AFM) topography data. They established a novel relationship between mechanical behavior and morphological properties, such as number of layers and type of graphene–substrate interface.

ML techniques can also open up a new venue to identify and classify defects and vacancies, further to trace phase transformation and chemical reaction processes using electron microscopy data in experiment.[14–16,345,346] In recent years, advanced characterization methods, such as scanning transmission electron (STEM) and scanning probe (SPM) microscopies, have generated enormous amounts of high-veracity experimental data with multi-dimensional structure and morphology information, especially for 2D materials, like graphene and TMDs.[347,348] Traditional data analysis suffers from high laborious investment and poor accuracy. Dan *et al.*[346] reviewed the recent progress on the prime effect of defect topology and density on the properties of 2D materi-

als using STEM data in the field of ML and computer vision. Ziatdinov *et al.*[345] developed a "weakly supervised" model based on deep neural network algorithm of atomically resolved STEM images. They used limited prior information about the location of atom species and type of defects in a sample to identify a rich variety of defect structures that were not in training set. They further applied this approach to track the structural and chemical transformations of multiple defects based on the silicon implanted into graphene. Maksov *et al.*[15] built a deep learning framework to automatically analyze the electron beam-induced solid state reactions and chemical transformations of Mo-doped WS_2 using STEM data. This approach is universal and its application to beam-induced reactions allows mapping structural transformation pathways in solids at the atomic level.

5. Conclusion and perspective

5.1. Summary

Machine learning approaches have been applied to various aspects of materials science embraced by structure, synthesis, characterization, property, and design, in order to accelerate research and development of new materials and reduce cost. ML method may identify the targeted crystal structures, like *AB* alloy, perovskite ABO_3, design materials with specific material properties such as superconductivity, thermal conductivity, and Curie temperature, guide chemical synthesis, and help characterization analysis. Since at its early stage, there are many issues that machine learning should be developed for its application in materials science.

5.2. Interpretability

As discussed in Subsection 3.1.2, other than predictability, ML models with good interpretability would promote the fundamental understanding and development in materials research. Several methods had been proposed to improve the predictability of regular ML models in materials science[19,196,211,349,350] and Molnar wrote a comprehensive guide for this propose.[351] And algorithms that can distill analytical forms from data like SR and SISSO have already been applied in the fields of physics and materials. For example, Michael Schmidt *et al.*[352] rediscovered the law of conservation of energy and conservation of momentum using SR trained on the experimental data of double pendulum and air spring trolleys; Hernandez *et al.*[353] utilized SR to quickly and accurately find multiple multibody interatomic potentials that are more concise than the traditional Lennard–Jones potential and Sutton–Chen potential, and reached comparable accuracy when applied to the simulation of elastic constants in MD. An unprecedentedly simple descriptor for perovskite catalysts, μ/t, was discovered by Weng *et al.*[354] via SR, this descriptor was validated to linearly correlates the overpotential of perovskites and guided the discovery of Cs-based oxide perovskites with improved OER activity. Also, SISSO had been used to identify the descriptor for classifying metals and non-metals,[134] Gibbs energies of stoichiometric inorganic compounds,[355] the stability of perovskite,[27] superconducting critical temperature,[215] etc. More applications and impact of interpretable ML are expected in materials science by using those algorithms or more powerful algorithms to be developed beyond SR and SISSO.

5.3. Active learning

The mechanisms and some applications of active learning algorithms, such as BO and MCTS, have been introduced in Subsections 3.1.3 and 4. BO had shown significant efficiency in predicting LTC compounds,[291] designing nanostructures for phonon transport,[228] predicting interfacial thermal resistance,[227] screening halide perovskites,[150] etc. For MCTS, AlphaGo[356] efficiently combines MCTS with two deep neural networks (strategic network and valuation network) to conquer Go. MCTS served as the main framework to select chess moves. In material science, Dieb *et al.*[152] utilized MCTS to quickly and accurately find the most stable structure of graphene doped with B in different proportions, searching only 2% of the search space. Cao *et al.*[357] used MCTS in combination with first-principles calculations to effectively obtain the optimum band gap of graphene nanoflakes (GNF). One of the significant advantages of MCTS is that the searching time increases almost linearly with the exponent of the search space.[151] Therefore, compared to BO, MCTS could be more suitable for problems with immense search space and can be represented as a tree structure. Active learning algorithms like BO and MCTS would be increasingly applied in material design requiring high computational or experimental costs.

5.4. Advanced algorithms and applications

Owing to the rapid development of computer science, modern ML algorithms like VAE and GAN that had achieved great success in other fields are very likely to become a considerable impetus for material science research. To our knowledge, the VAE and GAN have not been widely applied in material research recently. The GAN contains a generative model for generating data, then input the data into a discriminative model for estimating the probability that its input is from real world or generative model. These two models are trained together in GAN to let the generative model produce approximately real data. Li *et al.*[358] proposed a methodology combining a GAN for generating microstructures with BO to design microstructure materials. Nouira *et al.*[359] developed an architecture named CrystalGAN to produce new stable crys-

tallographic structures from existing data. Autoencoders contain a neural network with output layer containing much less neurons than input layer to encode data, then decode the data though another opposite neural network. The VAE is one kind of autoencoders trained with a method named stochastic gradient variational Bayes and can also be utilized to generate data. Ryan *et al.*[3] applied the VAE for generating structural representations in order to predict crystal structures from the ICSD and COD databases. The SciNet,[1] which is a modified version of VAE, was developed by Iten *et al.* and was utilized to rediscover physical laws such as the heliocentricity of the solar system. Another popular ML domain is the reinforcement learning (RL) that has conquered many fields like the Go game,[356] some video games, etc. RL methods such as the basic deep Q network (DQN) and policy gradients attempt to optimize the reward after taking certain actions in certain environments, RL would choose the actions with the highest reward after training. RL might be increasingly applied to guide materials design directly or integrated into other ML algorithms. For example, Sanchez-Lengeling *et al.*[360] developed a framework named objective-reinforced generative adversarial networks (ORGAN) that combined GAN with RL to inversely design materials with desired chemical properties.

References

[1] Iten R, Metger T, Wilming H, Del Rio L and Renner R 2020 *Phys. Rev. Lett.* **124** 010508
[2] Oviedo F, Ren Z, Sun S, Settens C, Liu Z, Hartono N T P, Ramasamy S, DeCost B L, Tian S I P, Romano G, Gilad Kusne A and Buonassisi T 2019 *npj Comput. Mater.* **5** 60
[3] Ryan K, Lengyel J and Shatruk M 2018 *J. Am. Chem. Soc.* **140** 10158
[4] Ziletti A, Kumar D, Scheffler M and Ghiringhelli L M 2018 *Nat. Commun.* **9** 2775
[5] Podryabinkin E V , Tikhonov E V , Shapeev A V and Oganov A R 2019 *Phys. Rev. B* **99** 064114
[6] Balachandran P V , Kowalski B, Sehirlioglu A and Lookman T 2018 *Nat. Commun.* **9** 1668
[7] Xu Q, Li Z, Liu M and Yin W J 2018 *J. Phys. Chem. Lett.* **9** 6948
[8] Fischer C C, Tibbetts K J, Morgan D and Ceder G 2006 *Nat. Mater.* **5** 641
[9] Segler M H S, Preuss M and Waller M P 2018 *Nature* **555** 604
[10] Coley C W, Thomas D A, Lummiss J A M, Jaworski J N, Breen C P, Schultz V, Hart T, Fishman J S, Rogers L, Gao H, Hicklin R W, Plehiers P P, Byington J, Piotti J S, Green W H, John Hart A, Jamison T F and Jensen K F 2019 *Science* **365** eaax1566
[11] Frey N C, Wang J, Vega Bellido G I, Anasori B, Gogotsi Y and Shenoy V B 2019 *ACS Nano* **13** 3031
[12] Rao C N R, Natarajan S and Neeraj S 2000 *J. Am. Chem. Soc.* **122** 2810
[13] Kim E, Huang K, Saunders A, McCallum A, Ceder G and Olivetti E 2017 *Chem. Mater.* **29** 9436
[14] Vasudevan R K, Laanait N, Ferragut E M, Wang K, Geohegan D B, Xiao K, Ziatdinov M, Jesse S, Dyck O and Kalinin S V 2018 *npj Comput. Mater.* **4** 30
[15] Maksov A, Dyck O, Wang K, Xiao K, Geohegan D B, Sumpter B G, Vasudevan R K, Jesse S, Kalinin S V and Ziatdinov M 2019 *npj Comput. Mater.* **5** 12
[16] Li W, Field K G and Morgan D 2018 *npj Comput. Mater.* **4** 36
[17] Sanchez-Gonzalez A, Micaelli P, Olivier C, Barillot T R, Ilchen M, Lutman A A, Marinelli A, Maxwell T, Achner A, Agåker M, Berrah N, Bostedt C, Bozek J D, Buck J, Bucksbaum P H, Montero S C, Cooper B, Cryan J P, Dong M, Feifel R, Frasinski L J, Fukuzawa H, Galler A, Hartmann G, Hartmann N, Helml W, Johnson A S, Knie A, Lindahl A O, Liu J, Motomura K, Mucke M, O'Grady C, Rubensson J E, Simpson E R, Squibb R J, Såthe C, Ueda K, Vacher M, Walke D J, Zhaunerchyk V, Coffee R N and Marangos J P 2017 *Nat. Commun.* **8** 15461
[18] Ghosh K, Stuke A, Todorović M, Jørgensen P B, Schmidt M N, Vehtari A and Rinke P 2019 *Adv. Sci.* **6** 1801367
[19] Stanev V, Oses C, Kusne A G, Rodriguez E, Paglione J, Curtarolo S and Takeuchi I 2018 *npj Comput. Mater.* **4** 29
[20] Dong Y, Wu C, Zhang C, Liu Y, Cheng J and Lin J 2019 *npj Comput. Mater.* **5** 26
[21] Liu Y, Wu J, Wang Z, Lu X G , Avdeev M, Shi S, Wang C and Yu T 2020 *Acta Mater.* **195** 454
[22] Ng M F , Zhao J, Yan Q, Conduit G J and Seh Z W 2020 *Nat. Mach. Intell.* **2** 161
[23] Shen Z H, Wang J J, Jiang J Y, Huang S X, Lin Y H, Nan C W, Chen L Q and Shen Y 2019 *Nat. Commun.* **10** 1843
[24] Xue D, Balachandran P V , Hogden J, Theiler J, Xue D and Lookman T 2016 *Nat. Commun.* **7** 11241
[25] Isayev O, Oses C, Toher C, Gossett E, Curtarolo S and Tropsha A 2017 *Nat. Commun.* **8** 15679
[26] Ye W, Chen C, Wang Z, Chu I H and Ong S P 2018 *Nat. Commun.* **9** 3800
[27] Bartel C J, Sutton C, Goldsmith B R, Ouyang R, Musgrave C B, Ghiringhelli L M and Scheffler M 2019 *Sci. Adv.* **5** eaav0693
[28] Liu Y, Wu J, Yang G, Zhao T and Shi S 2019 *Sci. Bull.* **64** 1195
[29] Liu Y, Zhao T, Yang G, Ju W and Shi S 2017 *Comput. Mater. Sci.* **140** 315
[30] Cao B, Adutwum L A, Oliynyk A O, Luber E J, Olsen B C, Mar A and Buriak J M 2018 *ACS Nano* **12** 7434
[31] Li J, Pradhan B, Gaur S and Thomas J 2019 *Adv. Energy Mater.* **9** 1970181
[32] Zubatyuk R, Smith J S, Leszczynski J and Isayev O 2019 *Sci. Adv.* **5** eaav6490
[33] Ling J, Hutchinson M, Antono E, Paradiso S and Meredig B 2017 *Integr. Mater. Manuf. Innov.* **6** 207
[34] Settles B 2012 *Synth. Lect. Artif. Intell. Mach. Learn.* **6** 1
[35] Ong S P, Richards W D, Jain A, Hautier G, Kocher M, Cholia S, Gunter D, Chevrier V L, Persson K A and Ceder G 2013 *Comput. Mater. Sci.* **68** 314
[36] Pizzi G, Cepellotti A, Sabatini R, Marzari N and Kozinsky B 2016 *Comput. Mater. Sci.* **111** 218
[37] Jain A, Ong S P, Chen W, Medasani B, Qu X, Kocher M, Brafman M, Petretto G, Rignanese G M, Hautier G, Gunter D and Persson K A 2015 *Concurr. Comput.* **27** 5037
[38] Lambert H, Fekete A, Kermode J R and De A 2018 *Comput. Phys. Commun.* **232** 256
[39] Mathew K, Singh A K, Gabriel J J, Choudhary K, Sinnott S B, Davydov A V , Tavazza F and Hennig R G 2016 *Comput. Mater. Sci.* **122** 183
[40] Mathew K, Montoya J H, Faghaninia A, Dwarakanath S, Aykol M, Tang H, heng I, Smidt T, Bocklund B, Horton M, Dagdelen J, Wood B, Liu Z K, Neaton J, Ong S P, Persson K and Jain A 2017 *Comput. Mater. Sci.* **139** 140
[41] Supka A R, Lyons T E, Liyanage L, D'Amico P, Al Rahal Al Orabi R, Mahatara S, Gopal P, Toher C, Ceresoli D, Calzolari A, Curtarolo S, Nardelli M B and Fornari M 2017 *Comput. Mater. Sci.* **136** 76
[42] Hjorth Larsen A, Jørgen Mortensen J, Blomqvist J, Castelli I E, Christensen R, Dułak M, Friis J, Groves M N, Hammer B, Hargus C, Hermes E D, Jennings P C, Bjerre Jensen P, Kermode J, Kitchin J R, Leonhard Kolsbjerg E, Kubal J, Kaasbjerg K, Lysgaard S, Bergmann Maronsson J, Maxson T, Olsen T, Pastewka L, Peterson A, Rostgaard C, Schiøtz J, Schütt O, Strange M, Thygesen K S, Vegge T, Vilhelmsen L, Walter M, Zeng Z and Jacobsen K W 2017 *J. Phys. Condens. Matter* **29** 273002
[43] Janssen J, Surendralal S, Lysogorskiy Y, Todorova M, Hickel T, Drautz R and Neugebauer J 2019 *Comput. Mater. Sci.* **163** 24
[44] Jain A, Ong S P, Hautier G, Chen W, Richards W D, Dacek S, Cholia S, Gunter D, Skinner D, Ceder G and Persson K A 2013 *APL Mater.* **1** 011002
[45] Hellenbrandt M 2004 *Crystallogr. Rev.* **10** 17
[46] Kirklin S, Saal J E, Meredig B, Thompson A, Doak J W, Aykol M, Rühl S and Wolverton C 2015 *npj Comput. Mater.* **1** 15010
[47] Haastrup S, Strange M, Pandey M, Deilmann T, Schmidt P S, Hinsche N F, Gjerding M N, Torelli D, Larsen P M, Riis-Jensen A C, Gath J, Jacobsen K W, Mortensen J J, Olsen T and Thygesen K S 2018 *2D Mater.* **5** 042002

[48] Stevanović V, Lany S, Zhang X and Zunger A 2012 *Phys. Rev. B - Condens. Matter Mater. Phys.* **85** 115104
[49] Graulis S, Chateigner D, Downs R T, Yokochi A F T, Quirós M, Lutterotti L, Manakova E, Butkus J, Moeck P and Le Bail A 2009 *J. Appl. Crystallogr.* **42** 726
[50] Saal J E, Kirklin S, Aykol M, Meredig B and Wolverton C 2013 *Jom* **65** 1501
[51] Borysov S S, Geilhufe R M and Balatsky A V 2017 *PLoS One* **12** e0171501
[52] Ashton M, Paul J, Sinnott S B and Hennig R G 2017 *Phys. Rev. Lett.* **118** 106101
[53] Williams A 2008 *Chem. Int.* **30**
[54] Choudhary K, Cheon G, Reed E and Tavazza F 2018 *Phys. Rev. B* **98** 014107
[55] Groom C R, Bruno I J, Lightfoot M P and Ward S C 2016 *Acta Crystallogr. Sect. B Struct. Sci. Cryst. Eng. Mater.* **72** 171
[56] Sterling T and Irwin J J 2015 *J. Chem. Inf. Model.* **55** 2324
[57] Landis D D, Hummelshøj J S, Nestorov S, Greeley J, Dułłak M, Bligaard T, Nørskov J K and Jacobsen K W 2012 *Comput. Sci. Eng.* **14** 51
[58] Gorai P, Gao D, Ortiz B, Miller S, Barnett S A, Mason T, Lv Q, Stevanović V and Toberer E S 2016 *Comput. Mater. Sci.* **112** 368
[59] Curtarolo S, Setyawan W, Wang S, Xue J, Yang K, Taylor R H, Nelson L J, Hart G L W, Sanvito S, Buongiorno-Nardelli M, Mingo N and Levy O 2012 *Comput. Mater. Sci.* **58** 227
[60] Gaulton A, Hersey A, Nowotka M L, Patricia Bento A, Chambers J, Mendez D, Mutowo P, Atkinson F, Bellis L J, Cibrian-Uhalte E, Davies M, Dedman N, Karlsson A, Magarinos M P, Overington J P, Papadatos G, Smit I and Leach A R 2017 *Nucleic Acids Res.* **45** D945
[61] Zakutayev A, Wunder N, Schwarting M, Perkins J D, White R, Munch K, Tumas W and Phillips C 2018 *Sci. Data* **5** 180053
[62] Ramakrishnan R, Dral P O, Rupp M and Von Lilienfeld O A 2014 *Sci. Data* **1** 140022
[63] Hachmann J, Olivares-Amaya R, Atahan-Evrenk S, Amador-Bedolla C, Sánchez-Carrera R S, Gold-Parker A, Vogt L, Brockway A M and Aspuru-Guzik A 2011 *J. Phys. Chem. Lett.* **2** 2241
[64] Hill J, Mannodi-Kanakkithodi A, Ramprasad R and Meredig B 2018 *Computational Materials System Design* (Cham: Springer International Publishing) p. 193
[65] Fink T and Reymond J 2007 *J. Chem. Inf. Model.* **47** 342
[66] Glasser L 2016 *J. Chem. Educ.* **93** 542
[67] Puchala B, Tarcea G, Marquis E A, Hedstrom M, Jagadish H V and Allison J E 2016 *JOM* **68** 2035
[68] Villars P, Onodera N and Iwata S 1998 *J. Alloys Compd.* **279** 1
[69] Tshitoyan V, Dagdelen J, Weston L, Dunn A, Rong Z, Kononova O, Persson K A, Ceder G and Jain A 2019 *Nature* **571** 95
[70] Torayev A, Magusin P C M M, Grey C P, Merlet C and Franco A A 2019 *J. Phys. Mater.* **2** 044004
[71] Swain M C and Cole J M 2016 *J. Chem. Inf. Model.* **56** 1894
[72] Jessop D M, Adams S E and Murray-Rust P 2011 *J. Cheminform.* **3** 40
[73] Jones D E, Igo S, Hurdle J and Facelli J C 2014 *PLoS One* **9** e83932
[74] Krallinger M, Rabal O, Lourenço A, Oyarzabal J and Valencia A 2017 *Chem. Rev.* **117** 7673
[75] Jensen Z, Kim E, Kwon S, Gani T Z H, Román-Leshkov Y, Moliner M, Corma A and Olivetti E 2019 *ACS Cent. Sci.* **5** 892
[76] Fourches D, Muratov E and Tropsha A 2010 *J. Chem. Inf. Model.* **50** 1189
[77] Wilkinson M D, Dumontier M, Aalbersberg Ij J, Appleton G, Axton M, Baak A, Blomberg N, Boiten J W, da Silva Santos L B, Bourne P E, Bouwman J, Brookes A J, Clark T, Crosas M, Dillo I, Dumon O, Edmunds S, Evelo C T, Finkers R, Gonzalez-Beltran A, Gray A J G, Groth P, Goble C, Grethe J S, Heringa J, t Hoen P A C, Hooft R, Kuhn T, Kok R, Kok J, Lusher S J, Martone M E, Mons A, Packer A L, Persson B, Rocca-Serra P, Roos M, van Schaik R, Sansone S A, Schultes E, Sengstag T, Slater T, Strawn G, Swertz M A, Thompson M, Van Der Lei J, Van Mulligen E, Velterop J, Waagmeester A, Wittenburg P, Wolstencroft K, Zhao J and Mons B 2016 *Sci. Data* **3** 160018
[78] Raccuglia P, Elbert K C, Adler P D F, Falk C, Wenny M B, Mollo A, Zeller M, Friedler S A, Schrier J and Norquist A J 2016 *Nature* **533** 73
[79] Yang H, Zhang Z, Zhang J and Zeng X C 2018 *Nanoscale* **10** 19092
[80] Gilmer J, Schoenholz S S, Riley P F, Vinyals O and Dahl G E 2017 *34th Int. Conf. Mach. Learn. ICML 2017* **3** 2053
[81] Balachandran P V , Emery A A, Gubernatis J E, Lookman T, Wolverton C and Zunger A 2018 *Phys. Rev. Mater.* **2** 043802
[82] Kajita S, Ohba N, Jinnouchi R and Asahi R 2017 *Sci. Rep.* **7** 16991
[83] Hoffmann J, Maestrati L, Sawada Y, Tang J, Sellier J M and Bengio Y 2019 arXiv:1909.00949 [cs.LG]
[84] Weininger D 1988 *J. Chem. Inf. Comput. Sci.* **28** 31
[85] Rupp M, Tkatchenko A, Müller K R and Von Lilienfeld O A 2012 *Phys. Rev. Lett.* **108** 058301
[86] Faber F, Lindmaa A, Von Lilienfeld O A and Armiento R 2015 *Int. J. Quantum Chem.* **115** 1094
[87] Schütt K T, Glawe H, Brockherde F, Sanna A, Müller K R and Gross E K U 2014 *Phys. Rev. B* **89** 205118
[88] Huang B and Von Lilienfeld O A 2016 *J. Chem. Phys.* **145** 161102
[89] Hansen K, Biegler F, Ramakrishnan R, Pronobis W, Von Lilienfeld O A, Müller K R and Tkatchenko A 2015 *J. Phys. Chem. Lett.* **6** 2326
[90] Rogers D and Hahn M 2010 *J. Chem. Inf. Model.* **50** 742
[91] Huo H and Rupp M 2017 arXiv:1704.06439 [physics.chem-ph]
[92] Bartók A P, Kondor R and Csányi G 2013 *Phys. Rev. B* **87** 184115
[93] Behler J 2011 *J. Chem. Phys.* **134** 074106
[94] Choudhary K, Decost B and Tavazza F 2018 *Phys. Rev. Mater.* **2** 083801
[95] Mezey, Paul G , R Stephen Berry J I B 1994 *Graph Theoretical Approaches to Chemical Reactivity*, ed. D Bonchev and O Mekenyan (Dordrecht: Springer Netherlands)
[96] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner G M 2009 *IEEE Trans. Neural Networks* **20** 61
[97] Xie T and Grossman J C 2018 *Phys. Rev. Lett.* **120** 145301
[98] F R S K P 1901 *London Edinburgh Dublin Philos. Mag. J. Sci.* **2** 559
[99] Khorshidi A and Peterson A A 2016 *Comput. Phys. Commun.* **207** 310
[100] Ramsundar B, Eastman P, Walters P, Pande V, Leswing K and Wu Z 2019 *Deep Learning for the Life Sciences* (Sebastopol: O'Reilly Media)
[101] Ward L, Dunn A, Faghaninia A, Zimmermann N E R, Bajaj S, Wang Q, Montoya J, Chen J, Bystrom K, Dylla M, Chard K, Asta M, Persson K A, Snyder G J, Foster I and Jain A 2018 *Comput. Mater. Sci.* **152** 60
[102] Chen C, Ye W, Zuo Y, Zheng C and Ong S P 2019 *Chem. Mater.* **31** 3564
[103] Himanen L, Jäger M O J, Morooka E V , Federici Canova F, Ranawat Y S, Gao D Z, Rinke P and Foster A S 2020 *Comput. Phys. Commun.* **247** 106949
[104] Schütt K T, Sauceda H E, Kindermans P J , Tkatchenko A and Müller K R 2018 *J. Chem. Phys.* **148** 241722
[105] Landrum G 2016 rdkit: open source cheminformatics software
[106] Ward L, Agrawal A, Choudhary A and Wolverton C 2016 *npj Comput. Mater.* **2** 16028
[107] Haghighatlari M, Vishwakarma G, Altarawy D, Subramanian R, Kota B U, Sonpal A, Setlur S and Hachmann J 2020 *WIREs Comput. Mol. Sci.* **10** e1458
[108] McKinney W 2011 *Python High Perform. Sci. Comput.* **14**
[109] Baranwal A, Bagwe B R and M V 2011 *J. Mach. Learn. Res.* **12** 128
[110] Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M and others 2016 *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16)*. pp 265
[111] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Kopf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J and Chintala S 2019 *Advances in Neural Information Processing Systems 32* ed H Wallach, H Larochelle, A Beygelzimer, F Alché-Buc, E Fox and R Garnett (Curran Associates, Inc.) p. 8024
[112] Liu Y, Wu J, Avdeev M and Shi S 2020 *Adv. Theory Simul.* **3** 1900215
[113] Chauhan N K and Singh K 2019 *2018 International Conference on Computing, Power and Communication Technologies, GUCON 2018* (IEEE) p.347
[114] Schneider A, Hommel G and Blettner M 2010 *Dtsch. Arztebl.* **107** 776
[115] Tibshirani R 1996 *J. R. Stat. Soc. Ser. B* **58** 267
[116] Hoerl A E and Kennard R W 1970 *Technometrics* **12** 55
[117] Zou H and Hastie T 2005 *J. R. Stat. Soc. Ser. B* **67** 301
[118] Murphy K P 2012 *Machine Learning: A Probabilistic Perspective* (Cambridge: The MIT Press)
[119] A Smola B S 2004 *Stat. Comput.* **14** 199
[120] Cortes C and Vapnik V 1995 *Mach. Learn.* **20** 273
[121] O'Shea K and Nash R 2015 arXiv:1511.08458 [cs.NE]
[122] Lecun Y, Bengio Y and Hinton G 2015 *Nature* **521** 436

[123] Nielsen M A 2018 *Neural Networks and Deep Learning* (Determination Press)
[124] Quinlan J R 1987 *Int. J. Man. Mach. Stud.* **27** 221
[125] Prokhorenkova L, Gusev G, Vorobev A, Dorogush A V and Gulin A 2018 *Advances in neural information processing systems* p. 6638
[126] Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q and Liu T Y 2017 *Advances in Neural Information Processing Systems 30* ed I Guyon, U V Luxburg, S Bengio, H Wallach, R Fergus, S Vishwanathan and R Garnett (Curran Associates, Inc.) p. 3146
[127] Freund Y and Schapire R E 1996 *Proc. 13th Int. Conf. Mach. Learn.* **148**
[128] Chen T and Guestrin C 2016 *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* KDD '16 (New York, NY, USA: ACM) p. 785
[129] Xuan P, Sun C, Zhang T, Ye Y, Shen T and Dong Y 2019 *Front. Genet.* **10** 459
[130] Liaw A and Wiener M 2002 *R News* **2** 18
[131] Rabiner L and Juang B 1986 *IEEE ASSP Mag.* **3** 4
[132] Kingma D P and Welling M 2019 *Found. Trends®Mach. Learn.* **12** 307
[133] Creswell A, White T, Dumoulin V, Arulkumaran K, Sengupta B and Bharath A A 2018 *IEEE Signal Process. Mag.* **35** 53
[134] Ouyang R, Curtarolo S, Ahmetcik E, Scheffler M and Ghiringhelli L M 2018 *Phys. Rev. Mater.* **2** 83802
[135] Augusto D A and Barbosa H J C 2000 *Proceedings. Vol. 1. Sixth Brazilian Symposium on Neural Networks* vol 2000-Janua (IEEE Comput. Soc) p. 173
[136] Wang Y, Wagner N and Rondinelli J M 2019 *MRS Commun.* **9** 793
[137] Udrescu S M and Tegmark M 2020 *Sci. Adv.* **6** eaay2631
[138] Dubčáková R 2011 *Genet. Program. Evolvable Mach.* **12** 173
[139] Fan J and Lv J 2008 *J. R. Stat. Soc. Ser. B* **70** 849
[140] Eldar Y C and Kutyniok G 2012 *Compressed Sensing: Theory and Applications* (Cambridge: Cambridge University Press)
[141] Ghiringhelli L M, Vybiral J, Ahmetcik E, Ouyang R, Levchenko S V, Draxl C and Scheffler M 2017 *New J. Phys.* **19** 023017
[142] Pankajakshan P, Sanyal S, De Noord O E, Bhattacharya I, Bhattacharyya A and Waghmare U 2017 *Chem. Mater.* **29** 4190
[143] Settles B 2012 *Active Learning* (San Rafael: Morgan & Claypool Publishers)
[144] Gubaev K, Podryabinkin E V and Shapeev A V 2018 *J. Chem. Phys.* **148** 241727
[145] Tran K and Ulissi Z W 2018 *Nat. Catal.* **1** 696
[146] Yuan R, Liu Z, Balachandran P V , Xue D, Zhou Y, Ding X, Sun J, Xue D and Lookman T 2018 *Adv. Mater.* **30** 1702884
[147] Botu V and Ramprasad R 2015 *Int. J. Quantum Chem.* **115** 1074
[148] Le T T, Fu W and Moore J H 2020 *Bioinformatics* **36** 250
[149] Zhong M, Tran K, Min Y, Wang C, Wang Z, Dinh C T, De Luna P, Yu Z, Rasouli A S, Brodersen P, Sun S, Voznyy O, Tan C S, Askerka M, Che F, Liu M, Seifitokaldani A, Pang Y, Lo S C, Ip A, Ulissi Z and Sargent E H 2020 *Nature* **581** 178
[150] Chen X, Wang C, Li Z, Hou Z and Yin W J 2020 *Sci. Chin. Mater.* **63** 1024
[151] Dieb T M, Ju S, Shiomi J and Tsuda K 2019 *MRS Commun.* **9** 532
[152] Dieb T M, Hou Z and Tsuda K 2018 *J. Chem. Phys.* **148** 241716
[153] M Dieb T, Ju S, Yoshizoe K, Hou Z, Shiomi J and Tsuda K 2017 *Sci. Technol. Adv. Mater.* **18** 498
[154] Mnih V, Kavukcuoglu K, Silver D, Rusu A A, Veness J, Bellemare M G, Graves A, Riedmiller M, Fidjeland A K, Ostrovski G, Petersen S, Beattie C, Sadik A, Antonoglou I, King H, Kumaran D, Wierstra D, Legg S and Hassabis D 2015 *Nature* **518** 529
[155] Meredig B, Antono E, Church C, Hutchinson M, Ling J, Paradiso S, Blaiszik B, Foster I, Gibbons B, Hattrick-Simpers J, Mehta A and Ward L 2018 *Mol. Syst. Eng.* **3** 819
[156] Butler K T, Davies D W, Cartwright H, Isayev O and Walsh A 2018 *Nature* **559** 547
[157] Bergstra J and Bengio Y 2012 *J. Mach. Learn. Res.* **13** 281
[158] Gao Y, Yang H, Zhang P, Zhou C and Hu Y 2019 *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings* pp 1
[159] Liaw R, Liang E, Nishihara R, Moritz P, Gonzalez J E and Stoica I 2018 arXiv:1807.05118 [cs.LG]
[160] Microsoft 2020 NNI (Neural Network Intelligence)
[161] Bergstra J, Komer B, Eliasmith C, Yamins D and Cox D D 2015 *Comput. Sci. Discov.* **8** 014008
[162] Jacobs R, Mayeshiba T, Afflerbach B, Miles L, Williams M, Turner M, Finkel R and Morgan D 2020 *Comput. Mater. Sci.* **176** 109544
[163] Srivastava N, Hinton G, Krizhevsky A, Sutskever I and Salakhutdinov R 2014 *J. Mach. Learn. Res.* **15** 1929
[164] Oliynyk A O, Adutwum L A, Harynuk J J and Mar A 2016 *Chem. Mater.* **28** 6672
[165] Oliynyk A O, Adutwum L A, Rudyk B W, Pisavadia H, Lotfi S, Hlukhyy V, Harynuk J J, Mar A and Brgoch J 2017 *J. Am. Chem. Soc.* **139** 17870
[166] Graser J, Kauwe S K and Sparks T D 2018 *Chem. Mater.* **30** 3601
[167] Chawla N V, Bowyer K W, Hall L O and Kegelmeyer W P 2002 *J. Artif. Int. Res.* **16** 321
[168] Park W B, Chung J, Jung J, Sohn K, Singh S P, Pyo M, Shin N and Sohn K S 2017 *IUCrJ.* **4** 486
[169] Oganov A R and Glass C W 2006 *J. Chem. Phys.* **124** 244704
[170] Oganov A R, Lyakhov A O and Valle M 2011 *Acc. Chem. Res.* **44** 227
[171] Lyakhov A O, Oganov A R, Stokes H T and Zhu Q 2013 *Comput. Phys. Commun.* **184** 1172
[172] Wu S Q, Ji M, Wang C Z, Nguyen M C, Zhao X, Umemoto K, Wentzcovitch R M and Ho K M 2014 *J. Phys.: Condens. Matter* **26** 035402
[173] Wang Y, Lv J, Zhu L and Ma Y 2010 *Phys. Rev. B* **82** 094116
[174] Wang Y, Lv J, Zhu L and Ma Y 2012 *Comput. Phys. Commun.* **183** 2063
[175] Tian Y, Sun W, Chen B, Jin Y and Lu C 2019 *Chin. Phys. B* **28** 103104
[176] Tang C, Kour G and Du A 2019 *Chin. Phys. B* **28** 107306
[177] Pickard C J and Needs R J 2011 *J. Phys.: Condens. Matter* **23** 053201
[178] Doll K, Schön J C and Jansen M 2007 *Phys. Chem. Chem. Phys.* **9** 6128
[179] Kirkpatrick S, Gelatt C D and Vecchi M P 1983 *Science* **220** 671
[180] Wille L T 1987 *Nature* **325** 374
[181] Wales D J and Doye J P K 1997 *J. Phys. Chem. A* **101** 5111
[182] Heiles S and Johnston R L 2013 *Int. J. Quantum Chem.* **113** 2091
[183] Burnham C J and English N J 2019 *J. Chem. Theory Comput.* **15** 3889
[184] Amsler M and Goedecker S 2010 *J. Chem. Phys.* **133** 224104
[185] Goedecker S 2004 *J. Chem. Phys.* **120** 9911
[186] Yamashita T, Sato N, Kino H, Miyake T, Tsuda K and Oguchi T 2018 *Phys. Rev. Mater.* **2** 013803
[187] Seko A and Ishiwata S 2020 *Phys. Rev. B* **101** 134101
[188] Li W, Jacobs R and Morgan D 2018 *Comput. Mater. Sci.* **150** 454
[189] Ward L, Liu R, Krishna A, Hegde V I, Agrawal A, Choudhary A and Wolverton C 2017 *Phys. Rev. B* **96** 024104
[190] Schmidt J, Shi J, Borlido P, Chen L, Botti S and Marques M A L 2017 *Chem. Mater.* **29** 5090
[191] Faber F A, Lindmaa A, Von Lilienfeld O A and Armiento R 2016 *Phys. Rev. Lett.* **117** 135502
[192] Li Z, Xu Q, Sun Q, Hou Z and Yin W J 2019 *Adv. Funct. Mater.* **29** 1807280
[193] Stanley J C, Mayr F and Gagliardi A 2020 *Adv. Theory Simul.* **3** 1900178
[194] Schmidt J, Chen L, Botti S and Marques M A L 2018 *J. Chem. Phys.* **148** 241728
[195] Schleder G R, Acosta C M and Fazzio A 2020 *ACS Appl. Mater. Interfaces* **12** 20149
[196] Kailkhura B, Gallagher B, Kim S, Hiszpanski A and Han T Y J 2019 *npj Comput. Mater.* **5** 108
[197] Pilania G, Mannodi-Kanakkithodi A, Uberuaga B P, Ramprasad R, Gubernatis J E and Lookman T 2016 *Sci. Rep.* **6** 19375
[198] Lee J, Seko A, Shitara K, Nakayama K and Tanaka I 2016 *Phys. Rev. B* **93** 115104
[199] Pilania G, Gubernatis J E and Lookman T 2017 *Comput. Mater. Sci.* **129** 156
[200] Rajan A C, Mishra A, Satsangi S, Vaish R, Mizuseki H, Lee K R and Singh A K 2018 *Chem. Mater.* **30** 4031
[201] Weston L and Stampfl C 2018 *Phys. Rev. Mater.* **2** 085407
[202] Zhuo Y, Mansouri Tehrani A and Brgoch J 2018 *J. Phys. Chem. Lett.* **9** 1668
[203] Olsthoorn B, Geilhufe R M, Borysov S S and Balatsky A V 2019 *Adv. Quantum Technol.* **2** 1900023
[204] Lu S, Zhou Q, Ouyang Y, Guo Y, Li Q and Wang J 2018 *Nat. Commun.* **9** 3405

[205] Gómez-Bombarelli R, Wei J N, Duvenaud D, Hernández-Lobato J M, Sánchez-Lengeling B, Sheberla D, Aguilera-Iparraguirre J, Hirzel T D, Adams R P and Aspuru-Guzik A 2018 *ACS Cent. Sci.* **4** 268
[206] Dey P, Bible J, Datta S, Broderick S, Jasinski J, Sunkara M, Menon M and Rajan K 2014 *Comput. Mater. Sci.* **83** 185
[207] Ziatdinov M, Maksov A, Li L, Sefat A S, Maksymovych P and Kalinin S V 2016 *Nanotechnology* **27** 475706
[208] Owolabi T O, Akande K O and Olatunji S O 2014 *Adv. Phys. Theor. Appl.* **35** 12
[209] Owolabi T O, Akande K O and Olatunji S O 2015 *J. Supercond. Nov. Magn.* **28** 75
[210] Liu Y, Zhang H, Xu Y, Li S, Dai D, Li C, Ding G, Shen W and Qian Q 2018 *Mater. Tehnol.* **52** 639
[211] Le T D, Noumeir R, Quach H L, Kim J H, Kim J H and Kim H M 2020 *IEEE Trans. Appl. Supercond.* **30** 1
[212] Konno T, Kurokawa H, Nabeshima F, Sakishita Y, Ogawa R, Hosako I and Atsutaka 2018 arxiv: 1812.01995
[213] Hamidieh K 2018 *Comput. Mater. Sci.* **154** 346
[214] Matsumoto K and Horide T 2019 *Appl. Phys. Express* **12** 073003
[215] Xie S R, Stewart G R, Hamlin J J, Hirschfeld P J and Hennig R G 2019 *Phys. Rev. B* **100** 174513
[216] Roekeghem A, Carrete J, Oses C, Curtarolo S and Mingo N 2016 *Phys. Rev. X* **6** 041061
[217] Zhan T, Fang L and Xu Y 2017 *Sci. Rep.* **7** 7109
[218] Wei H, Zhao S, Rong Q and Bao H 2018 *Int. J. Heat Mass Transf.* **127** 908
[219] Qian X, Peng S, Li X, Wei Y and Yang R 2019 *Mater. Today Phys.* **10** 100140
[220] Sosso G C, Deringer V L, Elliott S R and Csányi G 2018 *Mol. Simul.* **44** 866
[221] Seko A, Hayashi H, Nakayama K, Takahashi A and Tanaka I 2017 *Phys. Rev. B* **95** 144110
[222] Wan J, Jiang J W and Park H S 2020 *Carbon N. Y.* **157** 262
[223] Li R, Lee E and Luo T 2020 *Mater. Today Phys.* **12** 100181
[224] Gaultois M W, Oliynyk A O, Mar A, Sparks T D, Mulholland G J and Meredig B 2016 *APL Mater.* **4** 053213
[225] Carrete J, Li W, Mingo N, Wang S and Curtarolo S 2014 *Phys. Rev. X* **4** 011019
[226] Wu Y J, Sasaki M, Goto M, Fang L and Xu Y 2018 *ACS Appl. Nano Mater.* **1** 3355
[227] Wu Y J, Fang L and Xu Y 2019 *npj Comput. Mater.* **5** 56
[228] Ju S, Shiga T, Feng L, Hou Z, Tsuda K and Shiomi J 2017 *Phys. Rev. X* **7** 021024
[229] Ju S, Tsuda K, Shiomi J and Dieb T M 2018 *32nd Conference on Neural Information Processing Systems (NIPS 2018)*
[230] Wei H, Bao H and Ruan X 2020 *Nano Energy* **71** 104619
[231] Balachandran P V 2020 *J. Mater. Res.* **35** 890
[232] Balachandran P V , Xue D and Lookman T 2016 *Phys. Rev. B* **93** 144111
[233] Dam H C, Nguyen V C, Pham T L, Nguyen A T, Terakura K, Miyake T and Kino H 2018 *J. Phys. Soc. Jpn.* **87** 113801
[234] Nguyen D N, Pham T L, Nguyen V C, Nguyen A T, Kino H, Miyake T and Dam H C 2019 *J. Phys. Conf. Ser.* **1290** 012009
[235] Zhai X, Chen M and Lu W 2018 *Comput. Mater. Sci.* **151** 41
[236] Nelson J and Sanvito S 2019 *Phys. Rev. Mater.* **3** 104405
[237] Zhang B, Zheng X Q, Zhao T Y, Hu F X, Sun J R and Shen B G 2018 *Chin. Phys. B* **27** 067503
[238] De Jong M, Chen W, Notestine R, Persson K, Ceder G, Jain A, Asta M and Gamst A 2016 *Sci. Rep.* **6** 34256
[239] Furmanchuk A, Agrawal A and Choudhary A 2016 *RSC Adv.* **6** 95246
[240] Mansouri Tehrani A, Oliynyk A O, Parry M, Rizvi Z, Couper S, Lin F, Miyagi L, Sparks T D and Brgoch J 2018 *J. Am. Chem. Soc.* **140** 9844
[241] Chapman J, Batra R and Ramprasad R 2020 *Comput. Mater. Sci.* **174** 109483
[242] Kauwe S K, Graser J, Murdock R and Sparks T D 2020 *Comput. Mater. Sci.* **174** 109498
[243] Zhao Y, Yuan K, Liu Y, Louis S Y , Hu M and Hu J 2020 arXiv:2003.13425 [cond-mat.mtrl-sci]
[244] Evans J D and Coudert F X 2017 *Chem. Mater.* **29** 7833
[245] Kim G, Diao H, Lee C, Samaei A T, Phan T, de Jong M, An K, Ma D, Liaw P K and Chen W 2019 *Acta Mater.* **181** 124
[246] Xiong J, Shi S Q and Zhang T Y 2020 *Materials & Design* **187** 108378
[247] Zhuo Y, Mansouri Tehrani A, Oliynyk A O, Duke A C and Brgoch J 2018 *Nat. Commun.* **9** 4377
[248] Kauwe S K, Graser J, Vazquez A and Sparks T D 2018 *Integr. Mater. Manuf. Innov.* **7** 43
[249] Yeo B C, Kim D, Kim C and Han S S 2019 *Sci. Rep.* **9** 5879
[250] Umeno Y and Kubo A 2019 *Comput. Mater. Sci.* **168** 164
[251] Chandrasekaran A, Kamal D, Batra R, Kim C, Chen L and Ramprasad R 2019 *npj Comput. Mater.* **5** 22
[252] Broderick S R, Aourag H and Rajan K 2011 *J. Am. Ceram. Soc.* **94** 2974
[253] Yuan F and Mueller T 2017 *Sci. Rep.* **7** 17594
[254] Kim C, Pilania G and Ramprasad R 2016 *J. Phys. Chem. C* **120** 14575
[255] Kim C, Pilania G and Ramprasad R 2016 *Chem. Mater.* **28** 1304
[256] Wu X, Wang Y X, He K N, Li X, Liu W, Zhang Y, Xu Y and Liu C 2020 *Mater. (Basel).* **13** 179
[257] Huber L, Hadian R, Grabowski B and Neugebauer J 2018 *npj Comput. Mater.* **4** 64
[258] Rosenbrock C W, Homer E R, Csányi G and Hart G L W 2017 *npj Comput. Mater.* **3** 29
[259] Kiyohara S, Oda H, Miyata T and Mizoguchi T 2016 *Sci. Adv.* **2** e1600746
[260] Homer E R, Hensley D M, Rosenbrock C W, Nguyen A H and Hart G L W 2019 *Front. Mater.* **6** 168
[261] Kiyohara S, Oda H, Tsuda K and Mizoguchi T 2016 *Jpn. J. Appl. Phys.* **55** 45502
[262] Kikuchi S, Oda H, Kiyohara S and Mizoguchi T 2018 *Phys. B Condens. Matter* **532** 24
[263] Zhu Q, Samanta A, Li B, Rudd R E and Frolov T 2018 *Nat. Commun.* **9** 467
[264] Alade I O, Olumegbon I A and Bagudu A 2020 *J. Appl. Phys.* **127** 015303
[265] Pilania G and Liu X Y 2018 *J. Mater. Sci.* **53** 6652
[266] Liu Y, Zhao T, Ju W and Shi S 2017 *J. Materiomics* **3** 159
[267] Hannan M A, Lipu M S H, Hussain A, Ker P J, Mahlia T M I, Mansor M, Ayob A, Saad M H and Dong Z Y 2020 *Sci. Rep.* **10** 4687
[268] Xu Y, Hu M, Zhou A, Li Y, Li S, Fu C and Gong C 2020 *Appl. Math. Model.* **77** 1255
[269] Shen Y 2020 *Electrochim. Acta* **336** 135664
[270] Wang C, Aoyagi K, Wisesa P and Mueller T 2020 *Chem. Mater.* **32** 3741
[271] Li S, Li J, He H and Wang H 2019 *Energy Procedia* **159** 168
[272] Sendek A D, Cubuk E D, Antoniuk E R, Cheon G, Cui Y and Reed E J 2019 *Chem. Mater.* **31** 342
[273] Attarian Shandiz M and Gauvin R 2016 *Comput. Mater. Sci.* **117** 270
[274] Shi S, Gao J, Liu Y, Zhao Y, Wu Q, Ju W, Ouyang C and Xiao R 2016 *Chin. Phys. B* **25** 018212
[275] Chemali E, Kollmeyer P J, Preindl M and Emadi A 2018 *J. Power Sources* **400** 242
[276] Bian C, He H and Yang S 2020 *Energy* **191** 116538
[277] Wang W, Brady N W, Liao C, Fahmy Y A, Chemali E, West A C and M 2019 arxiv: 1909.02448
[278] Sidhu M S, Ronanki D and Williamson S 2019 *IECON 2019 - 45th Annu. Conf. IEEE Ind. Electron. Soc.* **1** 2732
[279] Sivaraman G, Jackson N E, Sanchez-Lengeling B, Vázquez-Mayagoitia A, Aspuru-Guzik A, Vishwanath V and De Pablo J J 2019 *chemrxiv.9914378.v1*
[280] Qu N, Liu Y, Liao M, Lai Z, Zhou F, Cui P, Han T, Yang D and Zhu J 2019 *Ceram. Int.* **45** 18551
[281] Pilania G, Gubernatis J E and Lookman T 2015 *Phys. Rev. B* **91** 214302
[282] Seko A, Maekawa T, Tsuda K and Tanaka I 2014 *Phys. Rev. B* **89** 054303
[283] Jacobs R, Mayeshiba T, Booske J and Morgan D 2018 *Adv. Energy Mater.* **8** 1702708
[284] Sun Q and Yin W J 2017 *J. Am. Chem. Soc.* **139** 14905
[285] De S, Bartók A P, Csányi G and Ceriotti M 2016 *Phys. Chem. Chem. Phys.* **18** 13754
[286] Jäger M O J, Morooka E V , Federici Canova F, Himanen L and Foster A S 2018 *npj Comput. Mater.* **4** 37
[287] Winiarski M J, Wiendlocha B, Gołab S, Kushwaha S K, Wiśniewski P, Kaczorowski D, Thompson J D, Cava R J and Klimczuk T 2016 *Phys. Chem. Chem. Phys.* **18** 21737

[288] Kamihara Y, Watanabe T, Hirano M and Hosono H 2008 *J. Am. Chem. Soc.* **130** 3296

[289] Allen P B and Dynes R C 1975 *Phys. Rev. B* **12** 905

[290] Drozdov A P, Eremets M I, Troyan I A, Ksenofontov V and Shylin S I 2015 *Nature* **525** 73

[291] Seko A, Togo A, Hayashi H, Tsuda K, Chaput L and Tanaka I 2015 *Phys. Rev. Lett.* **115** 205901

[292] Wang M, Wang J, Pan N and Chen S 2007 *Phys. Rev. E* **75** 36702

[293] He X, Chen S and Doolen G D 1998 *J. Comput. Phys.* **146** 282

[294] Feng T and Ruan X 2016 *Carbon N. Y.* **101** 107

[295] Hu S, Zhang Z, Jiang P, Chen J, Volz S, Nomura M and Li B 2018 *J. Phys. Chem. Lett.* **9** 3959

[296] Hu S, Zhang Z, Jiang P, Ren W, Yu C, Shiomi J and Chen J 2019 *Nanoscale* **11** 11839

[297] Yang K, Swanson K, Jin W, Coley C, Eiden P, Gao H, Guzman-Perez A, Hopper T, Kelley B, Mathea M, Palmer A, Settels V, Jaakkola T, Jensen K and Barzilay R 2019 *J. Chem. Inf. Model.* **59** 3370

[298] Faber F A, Hutchison L, Huang B, Gilmer J, Schoenholz S S, Dahl G E, Vinyals O, Kearnes S, Riley P F and Von Lilienfeld O A 2017 *J. Chem. Theory Comput.* **13** 5255

[299] Kearnes S, McCloskey K, Berndl M, Pande V and Riley P 2016 *J. Comput. Aided. Mol.* **30** 595

[300] Li Y, Tarlow D, Brockschmidt M and Zemel R 2015 *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings* pp 1

[301] Schütt K T, Arbabzadah F, Chmiela S, Müller K R and Tkatchenko A 2017 *Nat. Commun.* **8** 13890

[302] Wu Z, Ramsundar B, Feinberg E N, Gomes J, Geniesse C, Pappu A S, Leswing K and Pande V 2018 *Chem. Sci.* **9** 513

[303] Chmiela S, Tkatchenko A, Sauceda H E, Poltavsky I, Schütt K T and Müller K R 2017 *Sci. Adv.* **3** e1603015

[304] Lu C, Liu Q, Wang C, Huang Z, Lin P and He L 2019 *Proceedings of the AAAI Conference on Artificial Intelligence* vol 33 p. 1052

[305] Smith J S, Isayev O and Roitberg A E 2017 *Sci. Data* **4** 170193

[306] Unke O T and Meuwly M 2019 *J. Chem. Theory Comput.* **15** 3678

[307] Chen B, Barzilay R and Jaakkola T 2019 arXiv:1905.12712 [cs.LG]

[308] Sanyal S, Balachandran J, Yadati N, Kumar A, Rajagopalan P, Sanyal S and Talukdar P 2018 arXiv:1811.05660 [cs.LG]

[309] Yamamoto T 2019 *Crystal Graph Neural Networks for Data Mining in Materials Science* (Yokohama: Research Institute for Mathematical and Computational Sciences, LLC)

[310] Klicpera J, Groß J and Günnemann S 2020 arXiv:2003.03123 [cs.LG]

[311] Houben C and Lapkin A A 2015 *Curr. Opin. Chem. Eng.* **9** 1

[312] Wicker J G P and Cooper R I 2015 *CrystEngComm* **17** 1927

[313] Kayala M A, Azencott C A, Chen J H and Baldi P 2011 *J. Chem. Inf. Model.* **51** 2209

[314] Liu B, Ramsundar B, Kawthekar P, Shi J, Gomes J, Luu Nguyen Q, Ho S, Sloane J, Wender P and Pande V 2017 *ACS Cent. Sci.* **3** 1103

[315] Coley C W, Barzilay R, Jaakkola T S, Green W H and Jensen K F 2017 *ACS Cent. Sci.* **3** 434

[316] Wei J N, Duvenaud D and Aspuru-Guzik A 2016 *ACS Cent. Sci.* **2** 725

[317] Houben C, Peremezhney N, Zubov A, Kosek J and Lapkin A A 2015 *Org. Process Res. Dev.* **19** 1049

[318] Collins K D, Gensch T and Glorius F 2014 *Nat. Chem.* **6** 859

[319] Santanilla A B, Regalado E L, Pereira T, Shevlin M, Bateman K, Campeau L, Schneeweis J, Berritt S, Shi Z, Nantermet P, Liu Y, Helmy R, Welch C J, Vachal P, Davies I W, Cernak T and Dreher S D 2015 *Science* **347** 49

[320] Boyarshinov V 1997 *Machine Learning* (Long Beach: McGraw-Hill)

[321] Cundy C S and Cox P A 2003 *Chem. Rev.* **103** 663

[322] Yang S, Wang Y, Liu P, Cheng Y B, Zhao H J and Yang H G 1992 *Nat. Energy* **1** 1

[323] Zhang H, Ren X, Chen X, Mao J, Cheng J, Zhao Y, Liu Y, Milic J, Yin W J, Grätzel M and Choy W C H 2018 *Energy Environ. Sci.* **11** 2253

[324] Correa-Baena J P, Saliba M, Buonassisi T, Grätzel M, Abate A, Tress W and Hagfeldt A 2017 *Science* **358** 739

[325] Jeon N J, Noh J H, Yang W S, Kim Y C, Ryu S, Seo J and Il S 2015 *Nature* **517** 476

[326] Kumawat N K, Gupta D and Kabra D 2017 *Energy Technol.* **5** 1734

[327] Li W, Zhang W, Van Reenen S, Sutton R J, Fan J, Haghighirad A A, Johnston M B, Wang L and Snaith H J 2016 *Energy Environ. Sci.* **9** 490

[328] Wang F, Geng W, Zhou Y, Fang H H, Tong C J, Loi M A, Liu L M and Zhao N 2016 *Adv. Mater.* **28** 9986

[329] Yu Y, Tan X, Ning S and Wu Y 2019 *ACS Energy Lett.* **4** 397

[330] Novoselov K S, Geim A K, Morozov S V, Jiang D, Zhang Y, Dubonos S V , Grigorieva I V and Firsov A A 2004 *Science* **306** 666

[331] Kwak J, Jo Y, Song S, Kim J H, Kim S Y, Lee J U, Lee S, Park J, Kim K, Do G, Yoo J W, Kim S Y, Kong Y M, Lee G H, Lee W G, Park J, Xu X, Cheong H, Yoon E, Lee Z and Kwon S Y 2018 *Adv. Mater.* **30** 1707260

[332] Li Y, Zhou Z, Zhang S and Chen Z 2008 *J. Am. Chem. Soc.* **130** 16739

[333] Zheng F, Cai C, Ge S, Zhang X, Liu X, Lu H, Zhang Y, Qiu J, Taniguchi T, Watanabe K, Jia S, Qi J, Chen J H, Sun D and Feng J 2016 *Adv. Mater.* **28** 4845

[334] Xu S Y, Ma Q, Shen H, Fatemi V, Wu S, Chang T R, Chang G, Valdivia A M M, Chan C K, Gibson Q D, Zhou J, Liu Z, Watanabe K, Taniguchi T, Lin H, Cava R J, Fu L, Gedik N and Jarillo-Herrero P 2018 *Nat. Phys.* **14** 900

[335] Xu M, Tang B, Zhu C, Lu Y, Zhu C, Zheng L, Zhang J, Han N, Guo Y, Di J, Song P, He Y, Kang L, Zhang Z, Zhao W, Guan C, Wang X and Liu Z 2019 arXiv:1910.04603 [cond-mat.mtrl-sci]

[336] Snežzana Miljanić Leo Frkanec, Tomislav Biljan 3 Zlatko Meić Mladen Žini ć 2007 *J. Raman Spectrosc.* **38** 1538

[337] Timoshenko J and Frenkel A I 2019 *ACS Catal.* **9** 10192

[338] Huang B, Clark G, Navarro-Moratalla E, Klein D R, Cheng R, Seyler K L, Zhong Di, Schmidgall E, McGuire M A, Cobden D H, Yao W, Xiao D, Jarillo-Herrero P and Xu X 2017 *Nature* **546** 270

[339] Geim A K and Novoselov K S 2007 *Nat. Mater.* **6** 183

[340] Li Y, Kong Y, Peng J, Yu C, Li Z, Li P, Liu Y, Gao C F and Wu R 2019 *J. Materiomics* **5** 413

[341] Akinwande D, Brennan C J, Bunch J S, Egberts P, Felts J R, Gao H, Huang R, Kim J S, Li T, Li Y, Liechti K M, Lu N, Park H S, Reed E J, Wang P, Yakobson B I, Zhang T, Zhang Y W, Zhou Y and Zhu Y 2017 *Extrem. Mech. Lett.* **13** 42

[342] De Heer W A, Berger C, Ruan M, Sprinkle M, Li X, Hu Y, Zhang B, Hankinson J and Conrad E 2011 *Proc. Natl. Acad. Sci. USA* **108** 16900

[343] Gao Y, Cao T, Cellini F, Berger C, De Heer W A, Tosatti E, Riedo E and Bongiorno A 2018 *Nat. Nanotechnol.* **13** 133

[344] Cellini F, Lavini F, Berger C, De Heer W and Riedo E 2019 *2D Mater.* **6** 035043

[345] Ziatdinov M, Dyck O, Maksov A, Li X, Sang X, Xiao K, Unocic R R, Vasudevan R, Jesse S and Kalinin S V 2017 *ACS Nano* **11** 12742

[346] Dan J, Zhao X and Pennycook S J 2019 *InfoMat* **1** 359

[347] Moore A M and Weiss P S 2008 *Annu. Rev. Anal. Chem.* **1** 857

[348] Muller D A 2009 *Nat. Mater.* **8** 263

[349] Wagner N and Rondinelli J M 2016 *Front. Mater.* **3** 28

[350] Okamoto Y and Kubo Y 2018 *ACS Omega* **3** 7868

[351] Molnar C 2019 *Interpretable Machine Learning* (Leanpub)

[352] Schmidt M and Lipson H 2009 *Science* **324** 81

[353] Hernandez A, Balasubramanian A, Yuan F, Mason S A M and Mueller T 2019 *npj Comput. Mater.* **5** 112

[354] Weng B, Song Z, Zhu R, Yan Q, Sun Q, Grice C G, Yan Y and Yin W J 2020 *Nat. Commun.* **11** 3513

[355] Bartel C J, Millican S L, Deml A M, Rumptz J R, Tumas W, Weimer A W, Lany S, Stevanović V, Musgrave C B and Holder A M 2018 *Nat. Commun.* **9** 4168

[356] Silver D, Huang A, Maddison C J, Guez A, Sifre L, Van Den Driessche G, Schrittwieser J, Antonoglou I, Panneershelvam V, Lanctot M, Dieleman S, Grewe D, Nham J, Kalchbrenner N, Sutskever I, Lillicrap T, Leach M, Kavukcuoglu K, Graepel T and Hassabis D 2016 *Nature* **529** 484

[357] Cao Z P, Zhao Y J, Liao J H and Yang X B 2017 *RSC Adv.* **7** 37881

[358] Li X, Yang Z, Catherine Brinson L, Choudhary A, Agrawal A and Chen W 2018 *Proceedings of the ASME Design Engineering Technical Conference* vol 2B-2018 p. 1

[359] Nouira A, Sokolovska N and Crivello J C 2019 *CEUR Workshop Proc.* Vol 2350

[360] Sanchez-Lengeling B, Outeiral C, Guimaraes G L and Aspuru-Guzik A 2017 *ChemRxiv.5309668.v3*