SCHOOL OF COMPUTATION,
INFORMATION AND TECHNOLOGY

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics

Machine Learning in Materials Science: A Review

Jonas Hägele

SCHOOL OF COMPUTATION,
INFORMATION AND TECHNOLOGY

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics

Machine Learning in Materials Science: A Review

Author:	Jonas Hägele
Examiner:	Prof. Dr. Felix Dietrich
Submission Date:	15.07.2025

I confirm that this bachelor's thesis is my own work and I have documented all sources and material used.

Jonas Hägele

Rosenheim, 15.07.2025 Jonas Hägele

Abstract

For different purposes, different materials are needed with specific property requirements. Finding the best-suited materials fulfilling these properties comes with its own set of challenges and limitations, and forms the field of materials science. One of the limiting factors is the time and resource investment required, whether by experimental methods or computational simulations, to search the material space for good candidates. Experimental methods heavily rely on domain expertise and intuition and can take multiple years for results. Simulations, like Density Functional Theory (DFT) and Molecular Dyncamics (MD) simulations, are faster, but can still take a long time and are limited by available computing power. However, in the field of computational materials science, Machine Learning (ML) based methods are on the rise. They have shown to deliver good results at a fraction of the time cost that the aforementioned methods come with, and build on the data gathered from traditional methods in the past. ML models can classify materials and predict their properties, detect defects, systematically search for good candidates, and even directly generate potential materials fulfilling property requirements. This review will give an introduction to ML algorithms and how they work, and cover different ways the technology is utilized for different materials. Superconductors, perovskites, polymers, fibre-reinforced polymers (FRPs), and Metal-Organic Frameworks (MOFs) specifically will get a more detailed coverage. The current state of the art, as well as the direction research could head in the future, will be presented, giving a good overall picture of ML in materials science.

Contents

1 Introduction

Materials science, the discipline of finding materials well suited for their purpose, was traditionally based on experiments and trial-and-error. This process can span many years and takes substantial resources [87], as chemical or physical relationships between different material properties are not always known in a closed form [51]. To help alleviate this problem, different computational tools, like DFT and MD simulations, are in use. These tools can simulate and predict the behaviour of materials and chemical compounds under different conditions, predicting their properties [83, 64, 87]. While they reduce the need for experiments, they still require significant time and computational power [40]. To further reduce the time requirements of this process, ML, a subfield of Artificial Intelligence (AI), has seen increasing use in recent years. ML is a field of computer algorithms based on statistics that aim to automatically recognize patterns and relationships in data. This can then be used to quickly and accurately predict the properties of different materials [18]. Using this technology, it is possible to classify materials, search through databases for those exhibiting desired characteristics, and generate potential candidates directly. However, they first require a certain amount of training data before they can be put to use. The quality, amount, representativeness, and accuracy of this training data directly influence the performance of the final model. To help facilitate this process, multiple databases, like the Materials Project database [28], or the NOMAD database [63], exist, both to train ML models and to search through them using a trained model. Additionally, certain material classes have their own databanks, containing different materials of this class, like the Supercon databank, which contains superconductors. Different models are used for different purposes for multiple material types, like superconductors, perovskites, or polymers. Supervised learning models can find relationships between material properties, unsupervised models can classify data, and generative models can directly generate potential compounds.

In section 2, a short introduction to ML algorithms, how they are classified and evaluated, will be given, as well as a short introduction to materials science. Additionally, past reviews about ML in materials science and what they cover will also be introduced. Section 3 will first explain the methodology behind creating this review. Then, section 3.2 will go into different material classes and how ML is used for them, and what models are commonly used. Different efforts to improve the predictive power of these

models will also be covered. After that, an explanation of commonly used models and how they work will also be given in section 3.3.

2 State of the Art

2.1 Machine Learning

ML encompasses a set of algorithms that enable automatic recognition of patterns in data [87]. Then, it is possible to make inferences about unseen data. This way, it is possible to avoid lengthy trial-and-error methods where typical methods, like using physical laws, are not available. However, utilizing these algorithms is not as simple as running them and getting results. The steps of a general ML workflow will be shortly described here.

Firstly, one needs to acquire training data. This is the data that the model will recognize patterns in, apply these patterns to other unseen data, and make predictions. In the field of Materials Science specifically, multiple databases containing information about different compounds exist and can be used as a source for training data. Experiments and computer simulations are additional methods of data acquisition.

Secondly, the features for the models need to be selected. Features, also called descriptors [18], are the properties the model bases its predictions on in the case of material science. This step is called feature engineering, and also includes modeling or encoding the features in a way so that they are usable for ML algorithms. [84].

Thirdly, one must select a model to use. The choice of algorithm depends on the specific training data available and the use case. On one hand, they are classified into supervised, unsupervised, and reinforcement learning, based on their learning type [19]:

- Supervised learning requires labeled training data, which consists of input-output pairs. The final model predicts the output on new data, given only the inputs.
- Unsupervised learning utilizes unlabeled data and seeks patterns in it. Thus, it lends itself to different tasks than supervised learning, like clustering. This method is less common.
- Reinforcement learning gives positive or negative stimuli to the algorithm to influence its behavior as it interacts with the environment. This learning type is rarely used for materials science.

On the other hand, the models are also classified by their function. The most relevant are regression, classification, clustering, and generative models:

- Regression predicts one or multiple properties given a set of predictors. This requires supervised learning [83].
- Classification is similar to regression, but instead assigns datapoints to predetermined classes during inference [83].
- During clustering, the algorithm assigns datapoints to different clusters and determines what each cluster is characterized by. The clusters are not predetermined, unlike the classes in classification. Clustering is one of the applications of unsupervised learning.
- Generative models follow an inverse design principle. While regressive models predict properties given a material, generative models generate a material given a set of properties it should exhibit.

Lastly, after selecting a model, it can be trained and evaluated. Since it is hard to predict which specific model performs best on a given task, it is also possible to train multiple different ones and compare their performance to find the best-suited one.

This performance evaluation is the next and last step of the workflow. Generally, not all available data is used for training, but part of it is withheld to test the model later. Different metrics are used to judge and compare model performance. First, error metrics measure how far away a model's prediction is from the actual result. Common ones include mean absolute error (MAE), mean square error (MSE), and root mean square error (RMSE), whose formulas are

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right|, \tag{2.1}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2, \tag{2.2}$$

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}, \tag{2.3}$$

where n is the number of observations, y_i is the actual result for observation i, and $\hat{y}_i$ is the predicted result for observation i. Lower error metrics mean that the model's predictions are closer to reality and better. Another important metric is the coefficient of determination, or R^2-value, given by this formula:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \overline{y}\right)^2}. \tag{2.4}$$

A R^2-value closer to one indicates that the model explains more of the variance in the predicted variable from the predictors and thus has higher predictive power [83].

2.2 What is Materials Science?

2.2.1 Experimental Materials Science

The field of materials science includes the study and discovery of materials. The specific materials vary wildly depending on the area of application, like FRPs in aerospace applications [41] or superconductors for particle accelerators and nuclear fusion reactors [10, 11]. Finding new materials fulfilling the desired properties for their respective area of application is one of the main goals of this field [5]. Traditionally, this experimental approach was a costly and lengthy process, encompassing researchers conducting experiments and slowly optimizing parameters and material properties [6, 64], requiring human intuition or even luck [64]. Iteratively, new candidates would be synthesized and characterized until a satisfying result is achieved [12]. For instance, searching for new superconductors, materials without electrical resistance, experimentally, is just testing materials for this property in a trial-and-error manner, with a low success rate [88]. One method for this is high-throughput experimentation (HTE). Synthesis robots can automatically synthesize a large number of materials, albeit these robots are fairly expensive [65]. The design space for potential materials is very vast, involving multiple predictive variables with a lot of possible combinations. Furthermore, the relationship between predictive and target properties is not necessarily known in a closed form, and can instead only be evaluated point-wise [51]. This experiment-based approach is also known as the first paradigm of science, while science based on physical and chemical laws is the second paradigm [40, 84].

Besides the synthesis of new materials, analysis of them is also important, for instance, to find defects or signs of wear. Non-Destructive Evaluation (NDE) is the process of finding such defects in components or materials without causing any damage to them. Ultrasonic testing is an example of an NDE method, relying on ultrasonic waves interacting with the material to get an idea of what the inside looks like [78]. Another example is infrared thermography (IRT), which uses external heat sources to heat up the material, and changes in the surface temperature are measured and interpreted [61]. These methods then however, rely on humans interpreting the data, causing potential bottlenecks, as this process takes time, and human errors [78].

2.2.2 Computational Materials Science

With the emergence of computational materials science, the efficiency of this process was greatly improved. Computational methods like DFT, MD [83, 64], and Monte Carlo simulations [64] form the third paradigm of science [40] and can predict a compound's properties, reducing the need for costly experiments. These methods can be categorized into quantum-mechanical simulations based on DFT, which solve Schrödinger's equation, and methods based on inter-atomic potentials or force-fields, like MD simulations. MD simulations allow for a larger scale and are faster than quantum-mechanical methods, but are less accurate and not as easily transferable [17]. Additionally, both the performance of these simulations and available computing power constantly increased over time. They allow for high-throughput screening of a large number of candidate materials, and find the best candidates for experiments [64]. To help researchers in their efforts, multiple large databases of known materials and their properties exist. One such database is the Materials Project database, which is a collaboration between different scientists to create an open archive of lots of varied materials and compounds [28]. This data is based on theoretical calculations and simulations. Another similar database is the NOMAD database [63].

Still, these computational methods require significant time and resources, limiting their usability for long-term simulations or larger systems [40]. Additionally, screening-based methods are still limited to already known materials [90]. The fourth paradigm of science [12] utilizes ML to find patterns in the training data [38], instead of calculating a compound's behavior. This allows for faster results and lower resource requirements compared to the simulation methods mentioned above [18]. These methods and the aforementioned databases laid important groundwork for the rise of ML in materials science. The algorithms require vast amounts of training data first, and this data often stems from simulations and datasets [6], like the Materials Project. Additionally, there is great interest in a workflow based on an inverse design principle. This means that one starts with the properties the material needs to exhibit, and then creates it from there. Using ML, this approach is possible [90]. Utilizing ML, the aforementioned synthesis robots can also be improved. ML models can guide the synthesis process and help accelerate it and improve its efficiency [92]. Other models can also interpret data from NDE methods, to support humans in their evaluations of the testing data [78]. Interatomic potential required for simulations can also be obtained via ML [17]. In conclusion, there are many different ways ML can improve common workflows in computational materials science.

2.3 Existing Reviews

Of course, multiple other reviews covering the use of ML in materials science exist. They either give a general overview of how AI is used in this field for multiple materials or focus on one class in particular. In either case, they usually also give an introduction to ML and the workflow behind it, as well as an explanation of different algorithms used.

2.3.1 General Reviews

B. Yu, L. Zhang, et al. (2025) give an overview of how ML is used in materials science for property and phase stability prediction, crystal structure prediction, materials discovery, and design, determining structure-property relationships and materials characterization and analysis. Additionally, they give some problems and future challenges this technology faces, including the need for a lot of quality data and the need to share it between different industries or organizations.

Bai and X. Zhang (2025) also describe different ML algorithms, as well as databases used for training and what compound types they focus on. The materials and application fields covered include carbon nanotubes and organic light-emitting diodes (OLEDs) in material discovery, property prediction for multimetallic and intermetallic compounds, and optimization and design of gold/silver nanoparticles and hybrid materials. Finally, the importance of databases, feature engineering, and considerations concerning the uncertainty of ML are described as important challenges.

2.3.2 Material Specific Reviews

Tran, Dam, et al. (2024) focus on superconductors. They explain the physical theory behind superconductivity and list different superconductors, like sulfur hydride H_3S or lanthanum hydride LaH_{10}, found using computational methods. Notably, this is not referring to ML, but instead direct calculations or simulations. Then they cover how ML is used in this field. Specifically, how it is used to predict important properties like T_c from chemical formulas or atomic structures and to find high T_c superconductors. A cyclical approach is employed to find new superconductors directly, where outputs generated by the model are verified using simulations or DFT calculations and then used again to train the model further. Using symbolic ML techniques, empirical formulas for T_c can also be refined.

J. Wang, Lou, et al. (2025) lay more emphasis on the niche field of ionic liquids (IL) and its areas of application, e.g., for perovskite solar cells (PSCs). ILs are liquids consisting of anions (like $COOH^-$ or Cl^-) or large, inorganic cations (like ammonium

and imidazolium). Besides the general ML workflow, the authors also describe different molecular descriptors commonly used, those being molecular fingerprints, SMILES strings, potential energy functions, a graph with atom and bond weights, Coulomb matrix, bag of bonds/fragments, 3D geometry with associated atomic charges, and the electronic density. Selecting a descriptor is an important consideration regarding availability, training cost, model performance, and accuracy. While multiple model types, like Random Forest (RF), Extreme gradient boosting (XGBoost), and neural networks, are used for PSCs, they can not adequately handle the complex interactions between ILs and solar cells. However, graph neural networks (GNNs) have an advantage, as a graph structure can better describe spatial and relational information.

ML techniques are also of interest for perovskites directly. They have been used to confirm whether compounds can actually form the perovskite crystal structure, analyze bandgaps of perovskites, as well as optimize and predict the power conversion efficiency (PCE) of a PSC [70]. Additionally, optimizing reactions, determining structure-property relationships, and discovering new materials of this type are other applications of the technology [92]. ML can also be used to guide Autonomous Material and Device Acceleration Platforms (AMADAPs), which automatically conduct experiments and screen materials for their properties. It is an iterative process repeated multiple times, guided by ML until the target efficiency is reached. These methods could independently discover new materials for efficient photovoltaic cells with a long lifespan [92].

M. Chen, Yin, et al. (2024) summarize the use of ML to predict bandgaps, stability, and crystal structure of perovskites. Additionally, the use of symbolic learning to find relationships between intrinsic properties of the material is covered, as well as applications directly related to perovskite use for solar cells.

Liang, X. Wei, et al. (2025) review progress in the field of composite materials specifically. The authors summarize how past work utilized ML to elucidate material properties and behavior, predict material properties, and directly design new composite materials. Artificial Neural Networks (ANNs) play a huge role in this material class, as this model is used in many studies.

Pai, Shah, et al. (2025) cover polymers, specifically elastomers, thermoplasts, and thermosets. Different properties of these materials are predicted using various models, with ANNs being a common choice. Large Language Models (LLMs) are another promising class for polymer design. They use a mixture of natural language and chemical descriptors or interpret chemical structures as a language. This approach allows for equally accurate, but much faster results than others. Additionally, different feature engineering strategies are discussed.

Y. Hu, Q. Wang, and Ma (2024) cover the search for thermally conductive single polymer chains, amorphous polymers, and MOFs. They also describe the general ML workflow and potential databases for training data, including PoLyInfo, PI1M, QM9,

and Polymer Genome for polymers and the CoRE MOF and QMOF databases for MOFs. The importance of the selected database, descriptors, and algorithms is covered as well. As a final conclusion, the authors cite a lack of quantity and quality in the data present in the aforementioned databases related to thermal conductivity as a major challenge moving forward.

Du, Xin, et al. (2024) give a detailed description of the general ML workflow and multiple algorithms used. They then describe how these algorithms are used for MOFs. Classification algorithms are used to find MOFs with high gas separation performance, primarily CO_2 adsorption. With prediction algorithms, relationships between molecule structure and properties are found. Additionally, optimization algorithms, like BO and GA, are used to find optimal synthesis conditions.

Y. Liu, Dong, and H. Wu (2024) focus on MOFs. First, they cover data sources for different compounds. Both simulations and natural language processing to extract information from the literature are possible options. Additionally, databases for different MOF structures exist: hMOF, a database of computationally generated hypothetical MOFs, and CoRE MOF, created from experimental data and made to be computation-ready. Different algorithms, like Crystal Graph Convolutional Neural Network (CGCNN), ANN, Decision Tree (DT), and linear regression, are in use, often to predict gas-related properties, like gas uptake, separation, and hydrogen storage. Other predicted properties include performance as catalysts, stability under different conditions, and electrochemical properties. Transfer learning is also covered as a way to speed up the learning process with less data, by transferring knowledge from one domain to another, like from H_2 related properties to CH_4. Reinforcement learning can simulate the synthesis environment and optimize MOF properties. Genetic algorithms are also in use.

L. Zhang, H. Zhang, et al. (2025) focus on Amorphous Alloys and how ML is used for the prediction of amorphous alloy phases, glass-forming ability, and other material properties. They find that too much or too little data negatively affects prediction results, and 300 to 600 samples is a good middle ground. Additionally, predictor variables need to be carefully selected, as using multiple physically irrelevant features complicates the process.

Abdelhamid, H. Mohamed, and Kelouwani (2024) cover the use of ML to supervise additive manufacturing methods. First, they give a short explanation on how additive manufacturing works. Then they show how ML is used in this process. ML can be used to monitor the manufacturing process in real time, either using visual data or data from sensors. These models can then detect potential defects earlier, to reduce the costs of wasted materials. Additionally, ML techniques are also used to optimize process parameters.

3 Machine Learning in Materials Science: A Review

3.1 Review Setting

Papers were collected from Scopus (www.scopus.com) by filtering papers using "machine", "learning", and "material" (for materials science) in the keywords field, as well as those where the specific material (e.g. "superconductor", "perovskite") appears in the title, abstract, or keywords. However, some papers, including all the ones covered in section 3.2.5, were found without filtering for a specific material. The total number of papers available in Scopus for each search varies wildly, with superconductors having by far the fewest results, and polymers the most. For detailed numbers, refer to table 3.1.

Papers were then picked by hand by judging their novelty and uniqueness from their titles and abstracts. Many papers use well-known, supervised learning algorithms like RFs or ANNs to predict certain material properties. Of course, these are still relevant and were considered to be included in this work. But papers utilizing more unique or novel models were favored. This includes unsupervised learning models, papers covering inverse design, generative models, diffusion models, physics-informed models, and any other unique models, like the Mixture of Experts (MoE) model at the end of section 3.2.1 [101] and ChatMOF [32] in section 3.2.4.

The next section will cover the different papers and how the authors use ML for them.

Table 3.1: Number of papers using ML available in Scopus for each material as of July 15, 2025.

Material search term	Number of results
Superconductor	83
Perovskite	354
Polymer	752
Metal Organic Framework	251
None	13,326

Table 3.2: Different databases used by papers covered in this review in Section 3.2.

Database	example paper using it
SuperCon	Zhong, Y. Wang, et al. (2024)
Materials Project (MP)	Touati, Benghia, et al. (2024)
NOMAD	Seegmiller, Baird, et al. (2023)
Inorganic Crystal Structure Database (ICSD)	S. Kumar, Dutta, et al. (2023)
Open Quantum Materials Database (OQMD)	Chenebuah, Nganbe, and Tchagang (2024)
CoREMOF	Kang and Kim (2024)
QMOF	Kang and Kim (2024)
ARC-MOF	Yue, S. A. Mohamed, and Jiang (2024)
Conductive MOF (CM)	Lin, H. Zhang, et al. (2024)
Alexandria	Zeni, Pinsler, et al. (2025)
GEOM-QM9	Vieira Wyzykowski, Niazi, and Dickson (2025)
GEOM-Drugs	Vieira Wyzykowski, Niazi, and Dickson (2025)

Table 3.3: Some of the most common models and how often they approximately appear in the papers covered in Section 3.2.

Model	Number of appearances
RF	23
ANN	16
Gradient Boosting Decision Tree (GBDT)	9
Support Vector Machine (SVM)	9
XGBoost	6
DT	5
Light Gradient Boosting Machine (LGBM)	5
Categorical Boosting (CatBoost)	4

For an overview of different databases used, refer to table 3.2. Different models used will also be discussed. For an overview of some of the most used supervised regression and classification models, refer to table 3.3. RF and ANN are by far the most used models. However, GBDT, XGBoost, LGBM, and CatBoost can all be grouped under gradient boosting models, which would bring their total number of appearances to 24, which means that overall gradient boosting-based models are the most used, with RF being a close second.

3.2 Material Science View

This section will go into different material classes and how ML is used relating to them. Regressive models, made to predict certain properties of a given compound, have multiple uses. They can help explain physics-property relationships [101], guide researchers, or scan material databases for known compounds exhibiting unknown properties, like finding new superconductors in a database containing 2D materials [69].

3.2.1 Superconductors

Superconductors are a class of materials that show no electrical resistance and perfect diamagnetism below a certain temperature, called critical temperature T_c [10] or transition temperature [88, 21], measured in Kelvin. While searching for new superconductors, the goal is to find ones with a T_c as high as possible, ideally at room temperature. Since high T_c superconductors are not very well understood currently [88], ML is a promising approach to finding new ones.

Roter, Ninkovic, and Dordevic (2022) clustered superconductors using their chemical composition as the only feature of the clustering model, aiming to find possible structures in the data. The clustered data stems from the Supercon dataset and data collected from the literature. The outliers in the clustering result can be potential new families of superconductors. Additionally, they found that the clustering process is best done in multiple steps, meaning that after clustering the entire data, one can then apply the algorithm again to the clusters for further details.

Predicting T_c

Different models are in use, like ALIGNN and CatBoost. Besides the specific model, feature engineering and training data selection are the most important factors influencing prediction accuracy. In fact, it is possible to already achieve good prediction results

using only one feature [9]. Thus, research can focus on improved feature selection methods and novel types of ML models.

Accordingly Gashmard, Shakeripour, and Alaei (2024) first clean the SuperCon data set, using 13022 of its 33407 entries for training. Then, they create a Python package that generates 322 atomic descriptors for compounds. Of this large number of features, only the most important ones should be selected for training, for which purpose they developed another Python package that picks the 30 most relevant ones. This way, they could outperform past models using different feature sets and algorithms, reaching an R^2 of 0.952 and RMSE of $6.45K$.

On a similar note, S. G. Jung, G. Jung, and Cole (2024) introduce Gradient Boosted Feature Selection (GBFS), which leverages statistical analysis, using metrics like F1-score for classification or RMSE for regression models, to support feature selection. Additionally, they employ multicollinearity reduction, recursive feature engineering, and Bayesian hyperparameter optimization to improve the models. Using this approach, they train two GBDT models, one for the classification of superconductors (classifying them into $T_c \geq 10K$ and $T_c < 10K$) and one predicting T_c. Notably, they only use chemical descriptors and no crystalline descriptors, limiting predictive potential. The final classification model used 29 features, and the regression model used 34 features.

Models predicting T_c can also be used to find new superconductors via high-throughput search. By using an ALIGNN model, trained to specifically recognize hydride superconductors, and scanning through the ALEXANDRIA dataset with it, about 50 systems with $T_c > 20K$ could be found [10]. In another work, the authors scan the NOMAD dataset using a tool called DiSCoVeR, which stands for Descending from Stochastic Clustering Variance Regression, while simultaneously ensuring the scanned compounds are chemically valid. They find a lot of potential superconductors, with multiple of them showing transition temperatures upwards of $100K$ [66]. Pereti, Bernot, et al. (2023) create a model called DeepSet, making use of ANNs, to screen the minerals accepted by the International Mineralogical Association. Multiple potential superconductors are found, about 44% of which were already known as such. Additionally, three of the predicted, unknown superconductors are experimentally verified, two of which the model correctly predicted. In the search for new superconductors, J. Zhang, K. Zhang, et al. (2023) employ an integrated model consisting of a Light Gradient Boosting (LGB), Extra Tree, and GBDT trained on the Supercon dataset to predict transition temperatures. While mining the Materials Project database, the model identifies 20 compounds with a T_c above $50K$.

A slightly different approach was taken by Tran and Vu (2023), who instead predict λ and $\omega_l o g$, two properties related to the electron-phonon interactions, from atomic structures of compounds from the Materials Project database using a Gaussian Process Regression (GPR) model, from which T_c can be approximately calculated. T_c is

predicted for an environment at 0 pressure. This approach has the advantage that λ and $\omega_l og$ directly correlate to the atomic structure, making predictions more physics-inspired. Calculating T_c afterwards is trivial. This way, the authors find two potential new superconductors, and both materials have already been studied in a different context before.

Pogue, New, et al. (2023) train a model called RooSt, a GNN, in a closed loop, starting with a model trained on the SuperCon database. Then, this model scans the Materials Project and Open Quantum Materials databases for new candidate superconductors. A selection of potential superconductors is experimentally verified and then added to the training data of the model. Notably, both correct and wrong predictions are added to the training data. This way, the model is able to rediscover 5 known superconductors, as well as discover a previously unknown, new one.

A novel model is employed by Zhong, Y. Wang, et al. (2024), who train a MoE model integrating dopant recognition. The SuperCon database serves as a basis for training, with the training set consisting of 10,076 compounds after filtering. With it, ten elemental, physical, and doping experts are trained. Each of them focuses on a different descriptor for their predictions. The elemental experts consider the elemental composition of the superconductor, the elements, and their stoichiometric coefficients. The physical experts encompass a total of 116 features, including chemical, physical, and electrical properties. Finally, the doping experts are based on doping elements in the material. These models on their own already show decent performance, with their R^2 values reaching 0.932, 0.945, and 0.948 for the physical, elemental, and doping experts, respectively. Finally, a gating model is created, which dynamically determines the weight of each expert's prediction for the final result. The final prediction is the weighted sum of all the experts

$$\sum_{(i=1)}^{K} g_i(x) \cdot e_i(x) \tag{3.1}$$

with $g_i(x)$ and $e_i(x)$ being the weight and prediction of expert i respectively. This final model reaches an R^2 of 0.962, RMSE of 6.502K, and MAE of 3.257K, values surpassing other models. To highlight the importance of the doping factor, the authors additionally compare this models prediction for the superconductor $Bi_{1.4}Pb_{0.6}Sr_2Ca_{2\text{-}x}Ga_xCu_3O_7$, with the doping factor x varying between 0 and 0.8, to that of other models. The superconductor shows a sudden change in T_c for $x = 0.1$, which the MoE model predicts correctly, while the other models could not do so, as illustrated in Figure 3.1. Additionally, to find new superconductors, 1500 candidates are generated using the Supercon-Diffusion [102] model by the same authors discussed in section 3.2.1, 40 of which were new high-T_c superconductors with $T_c > 120K$.

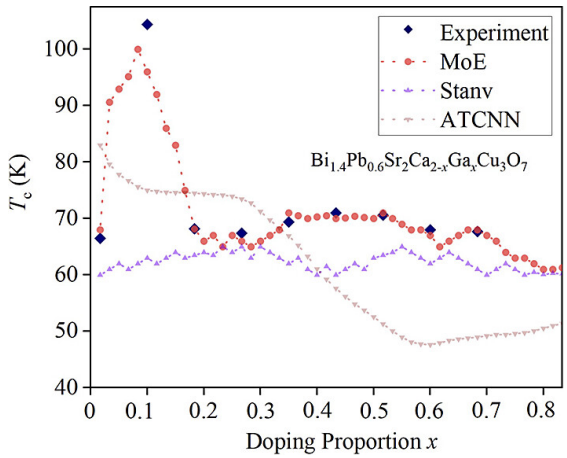

Figure 3.1: Comparison of different models' predictions of T_c for a given superconductor at different doping ratios. The experimental T_c-value is 104 K, the MoE model's prediction is 102 K. The next closest model predicts a T_c of 73 K. The image was taken from Zhong, Y. Wang, et al. (2024).

Generative Models

While regression models require a compound as input and thus have limited potential for finding new superconducting materials, generative models can create new materials exhibiting wanted properties directly.

Supercon-Diffusion [102] is a diffusion model for the purpose of generating high T_c superconductors with optimal doping ratios. Diffusion models work by gradually adding Gaussian noise to real data and then removing it again in the reverse diffusion process to create the generation result. A denoising model is trained to learn the denoising process during training by trying to re-create the input sample. After training during inference, a random Gaussian noise sample is denoised to create a generation result. See Section 3.3.5 for further details. 7315 superconductors are collected from the Supercon database, and sorted into low (20-40 K), medium (40-77 K), and high T_c (>77 K) categories, with each class containing around 2500 entries. To encode doping ratios of these materials, 3 matrices are created, with the first matrix representing the integer part of the stoichiometric coefficient, and the second and third matrices corresponding to the first and second decimal of the stoichiometric coefficient. Each matrix has the dimensions of 86×10, since 86 elements were present in the superconductors, and the respective coefficient is encoded using one-hot encoding. In total, each superconductor is encoded using a $3 \times 86 \times 10$ matrix. Additionally, at each time step of both the forward and reverse process, a conditioning for the class of

superconductor (low, medium, or high T_c) is also added in a one-hot encoded form. When generating a new superconductor, the same conditioning is also supplied. The superconductor is generated in the aforementioned matrix representation. With this model, the authors are able to create 200 potential new high T_c superconductors not documented yet. All of them are members of known families, created by doping existing superconductors.

Using diffusion, it is also possible to find entirely new families of superconductors, as shown by SuperDiff [88]. A 96×1 vector is used as a descriptor for the superconductors. Element n of this vector is the stoichiometric coefficient of the element with atomic number n in the encoded material. 96 was the highest atomic number present in the data, and thus the length of the vector. The elements in the vector allowed for non-integer values, so doping was also encoded this way. Four different models for different families of superconductors are trained from the SuperCon data: a cuprates model, a pnictides model, a model for other classes, and a model for all classes. To find new families of superconductors, the models are conditioned, meaning their reverse diffusion process was modified, after which they require a reference superconductor for the generation process. This reference influences the generation at each denoising step with the goal of the result sharing a similar chemical composition to the reference. The four trained models are copied and modified with the conditioning mechanism, resulting in 8 models total. From the unconditional models, the authors sample $500,000$ generations to compare the performance to the conditioned models. All the versions of SuperDiff are able to generate many unique and novel materials; however, the ability of the conditional models to do so depends strongly on the choice of conditioning reference. Most of the materials are predicted to be stable (have negative formation energies) as well. To find out if the models generate superconductors of new families, the results are clustered. New clusters of unknown families would indicate that new families have been found. While the unconditional versions are unable to generate new families, the conditional ones find multiple new families. Two examples are $Li_{1-x}Be_xGa_2Rh$ and $Na_{1-x}Al_{1-y}Mg_{x+y}Ge_{1-z}Ga_z$, generated by using $LiGa_2Rh$ and NaAlGe as conditioning respectively.

3.2.2 Perovskites and Solar Cells

Perovskites are an emerging material class, with one use field being photovoltaic cells [97]. They are classified by their ABX_3 crystal structure, where A, B, and X are organic cations, metal cations, and halide anions, respectively [68], in the case of a hybrid organic-inorganic perovskite. If the A-site cations are inorganic ions, then the perovskite is classified as an all-inorganic one [70]. They can also exist as a double perovskite, possessing an $ABB'X_3$ structure, following the same rules for what elements

can be in which position. [81]

Solar Cell related Properties

Different chemical or physical properties are of interest when considering a material for use in a solar cell. One of them is the bandgap, which influences the wavelength of light that can be absorbed [71]. Accordingly, ML has been used to quickly screen potential materials for favorable characteristics [97, 13]. ML models are commonly selected by training multiple different models, picking the best performing one when evaluated using R^2 and error metrics, like MSE, MAE, or RMSE.

For instance, Z. Zhang, S. Liu, et al. (2025) strategically investigate the prediction abilities of eleven models and four different structural descriptors when predicting bandgaps. Many-body tensor representation (MBTR) performs the best across the board for the descriptors. The majority of the ML models are pretty close together in performance, but the overall quality was heavily influenced by the choice of the descriptor. Two of the descriptors, sine matrix and Ewald sum matrix, are naturally suited for use in GNNs, and these models can outperform the Convolutional Neural Network (CNN)- and decision tree-based models used with these descriptors. However, the latter two model types can employ all of the descriptors, and among them, CustomCNN performs the best, surpassing the GNN models as well.

S. Kumar, Dutta, et al. (2023) also create a bandgap prediction model, trained from 14633 entries from the Inorganic Crystal Structure Database (ICSD), most of which are not perovskites. Additionally, a subset of this data is used to train a classification model, which can identify stable compounds. For both purposes, RF performs the best. Using these models, 246,820 prototype compounds, constructed from the periodic table, are screened for stability and bandgap.

Touati, Benghia, et al. (2024) collect 761 perovskites from the Material Project database. Then they train an XGBoost classification model to predict the crystal structure, and regression XGBoost and RF models to predict formation and bandgap energies. This way, stable perovskites for solar cells can be identified.

H. Wang, Ouyang, et al. (2024) utilize independence screening and specifying operator (SISSO) to find formulas to predict the bandgap of perovskites. They then search through databases to identify unknown materials with suitable bandgaps for solar cells, finding 14 new lead-free perovskites.

According to Y. Chen, H. Liu, et al. (2025), possessing a direct bandgap, as opposed to an indirect one, is another important factor influencing a perovskite's usability for solar cells. Thus, they create a ML model to predict the bandgap type of different double perovskites. The models used include gradient boosting (GB), RF, CatBoost, and LGBM. They are trained using data from the materials project database, and the final model the

authors select as the best performing one is LGBM, exhibiting a 90% accuracy. Given the general ABB'X composition of double perovskites, the authors can directly construct candidate materials from the periodic table and use their trained model to classify them into direct and indirect bandgap materials, finding 176 promising Br-based double perovskites.

A similar approach can be found in Ahmad, Ibrahim, et al. (2025), who first train an RF model to predict the highest occupied molecular orbit (HOMO) and lowest unoccupied molecular orbit (LUMO) energy levels of materials for solar cells. Then, they extract compounds similar to already known and used materials for solar cells and use the trained model to predict their HOMO and LUMO energy levels, showing that they are suitable for this application.

Leveraging the pattern-finding abilities of ML is also an opportunity to improve our physical understanding of material properties. One tool to achieve this is SHAP analysis, which extracts how much each feature influences the final prediction in a ML model. Subudhi, Sivapatham, et al. (2025) also train an RF model, which predicts the PCE of perovskites. However, they did not use the model for any predictions. Instead, they perform a SHAP analysis to extract the importance of each feature. These insights can then guide researchers in their efforts. Pindolia and Shinde (2024) also predict the PCE of perovskite solar cells, specifically $KSnI_3$-based ones, using a RF model. Abdellah and El-Shafei (2024) use autoQSPR models to predict solar cell properties, PCE among them. The model is trained on data extracted from the literature.

Besides the perovskite, other materials are needed for a solar cell. One of them is the hole transport material, the choice of which also influences the performance of the solar cell. Thus, Rashid, S. Lee, et al. (2024) employ a RF model to predict the hole mobility of hole transport materials for perovskite solar cells.

Other Perovskite Properties

Other properties of perovskites, not necessarily related to solar cells, are of interest as well. For instance, Alfares, Sha'aban, and Alhumoud (2025) train multiple models, RF, Support Vector Regression (SVR), ANN, and Ensemble Regression Trees (ERTs) among them, to predict the lattice constant of ABX_3 perovskites. The training data consists of 122 unique compounds from an earlier study. The GPR model performs the best, reaching an R^2 of 0.99 on the test set and 1 on the training set. Bi, M. Wang, et al. (2024) create a Deep Neural Network (DNN) and a MoE model utilizing three-dimensional convolutional neural networks (3DCNNs) that predict the adsorption energies of halogen-containing organic-inorganic hybrid perovskites. SHAP analysis on these models reveals which halogen atoms have the highest influence on adsorption energy. Other examples are a Light Gradient Boosting Machine (LightGBM) model

which predicts the phonon cutoff frequency of ABO_3 perovskites [22] and models predicting the spontaneous polarization of perovskites [72].

Jacobs, J. Liu, et al. (2024) use a RF model to predict the area-specific resistance (ASR), a property related to catalytic performance, of perovskites. They then screen a set of over 19 million constructed perovskites for promising new candidates.

Laref, Massuyeau, and Gautier (2024) train multiple ML models, like k-Nearest Neighbour, Logistic Regression, RF, DT, SVM, and GBDT, to predict whether materials could form a perovskite structure. They then analyse the importance of the features for the prediction. They find that the number of ammonium groups ($NH_3{}^+$) present majorly impacts the ability to form a perovskite structure.

Chenebuah, Nganbe, and Tchagang (2024) extract 36016 perovskites from the Open Quantum Materials Database (OQMD) and 6540 compounds from the Materials Project (MP) database. A mesh-grid descriptor for perovskites is designed. It is represented by a $32 \times 32 \times 3$ RGB image, made up of 3 individual 32×32 meshes. Each of them has a distinct purpose; The first one is the label mesh, encoding identification information, like the elements, number of atoms, and stoichiometry present. The second one is the property mesh, containing thermochemistry features. The third and last one is the X-Ray Diffraction (XRD) mesh for XRD patterns. The generation of perovskites using the model takes place over three phases. The first phase is a semi-supervised Variational Autoencoder (VAE) (SS-VAE). The VAE projects the perovskites onto a Gaussian latent space, while reducing the feature dimensionality. A Multi-Layer Perceptron (MLP) is used to find regions likely to contain highly stable materials of the correct crystal structures in this latent space. In this case, highly stable materials are defined as those with a formation energy $E_f \leq -1.5eV/atom$, which is a more demanding threshold, since generally compounds with $E_f < 0eV$ are considered stable. For the second phase, this latent space and the regions of interest are passed to an auxiliary generative adversarial network (A-GAN). This model generates latent space vectors of close similarity to the extracted vectors. Additionally, the auxiliary model predicts lattice features of the original vectors, ensuring validity. For the last phase, these compounds are decoded into perovskites, adopting the characteristic crystal structure. The results are screened for their charged neutrality, as neutrally charged compounds are more stable, and energetic stability. For this step, any materials with a negative formation energy pass screening, as opposed to the aforementioned $-1.5eV/atom$ threshold. Then, the perovskites undergo further screening via Bayesian Optimization (BO) and DFT, which simulate stress relaxation of the geometry. Converging materials are further examined. The final model is able to rediscover multiple materials from the OQMD and MP databases, and 72 perovskites are identified as new chemistries.

3.2.3 Polymers

Polymers are a very broad material class, used in many applications. Different properties are of interest depending on the use case and type of polymer. They are formed by multiple, smaller monomers reacting to build one large molecule.

Pure Polymers

One approach to improve the performance of ML models is to create physics-informed models. These models are tuned to take into account known physical characteristics. An example of such a model is SLIMNet, a physics-informed neural network, by Xu, X. Yu, et al. (2025), which leverages properties following scaling laws inherent to polymer materials. The properties of a single Gauss segment of the polymer also apply to the entire polymer chain. The model is split into two parts, one of which predicts the properties of the monomer, and the other predicts properties of the polymer using a physics-informed approach and the scaling trait mentioned previously. This way, the model is able to outperform RF on a relatively small dataset of 1070 molecules, with higher R^2 and lower error metrics. Notably, the RF model still performs decently on most of the data. However, it shows massive prediction errors for some of the molecules, while the physics-informed model gives much more accurate predictions for these cases.

Ran, An, and L. Zhang (2024) create another physics-informed neural network to predict the behaviour of phase-seperated homopolymer blends. Their model's loss function takes into account known physical relations related to phase separation behaviour. Additionally, they made use of symbolic regression to find analytical formulas for the relationships.

X. Fang, Murphy, et al. (2025) obtain block copolymer small-angle X-ray scattering (SAXS) data via automated chromatography. These polymers can come in six different morphologies. The authors then extract six physics-informed features from the X-ray data to train a RF model predicting the morphologies. The classification performance of this physics-informed model is compared to that of two other RF models, which use different feature sets. The physics-informed (PI) model proves to be way more accurate, only misclassifying a small fraction of the materials.

One use case of polymers is in membranes. The properties of interest for membranes are their permeability and selectivity. Zheng, S. Zhang, et al. (2024) utilize a RF model to find new membranes for carbon capturing. So in this specific case, N_2/CO_2 selectivity is of interest. The membranes for training are collected from the literature and screened for fragments used. Ten fragments of three categories (terminal reactive, terminal unreactive, and nonterminal) are identified. The membranes are encoded in a ten-bit

vector, where each bit represents the presence of the corresponding fragment. The resulting model is then used to screen candidate membranes for their selectivity and permeability.

Phua, Terasoba, et al. (2024) use dimensionality reduction to reduce the data of Anion-Exchange Membrane Polymers into two dimensions using PCA and UMAP, visualizable using a coordinate system. Then, they use the k-means algorithm to cluster the results, identifying material classes. Researchers can use the resulting map to guide them in designing new Anion-Exchange membranes.

Nanjo, Arifin, et al. (2025) introduce SPACIER, using Bayesian Optimization to effectively search through polymers for those exhibiting wanted properties. In SPACIER, features of the repeating unit of the polymers are extracted into a 170-dimensional vector. The user gives an initial training set and search space, and determines how many new candidate polymers should be added at each optimization step. A Gaussian Process surrogate function is calculated to approximate the relationship between the feature vector and the target properties, as calculated by MD simulations. During each iteration, an acquisition function, which represents the probability of a compound being within the target region, is evaluated to find the most fitting candidates. The predetermined number of the best candidates undergo MD simulations to find their target properties, and are then added to the training set, and the surrogate function is updated to account for the new data. Any polymers added to the training set are removed from the candidate set. To illustrate the function of SPACIER, the authors use a set of 1077 polymers from another paper and calculate their specific heat capacity C_p and refractive index. Ten randomly selected polymers of this set form the initial training data. Ten candidates are added to the training set in every iteration. The authors define three target regions, and SPACIER finds all polymers in these regions within 20-30 iterations. This is compared to a Fix-GP approach, which functions the same but does not update the model, and randomly selects the candidate polymers at each step. The Fix-GP version is also able to find all the polymers within 30 cycles, but took more cycles than SPACIER, and the random approach, which selects polymers at each step completely randomly, finds less than half of the polymers in a given region within 30 cycles. Then, they explore polymers with both increasing refractive index and Abbe number. There is a tradeoff relationship between these two properties with an empirical upper limit, known as the Pareto boundary. As a candidate set, they use the 1077 polymers from before, plus an additional 101,487 compounds artificially generated from monomers. Within 20 iterations, SPACIER manages to find polymers gradually approaching and eventually surpassing the Pareto boundary. 64% of the found compounds surpassed the boundary at the end. The authors synthesize two of the found polymers, and their predicted properties are in good agreement with experimental values.

Jain, Armstrong, et al. (2024) use an GPR to recommend polymers for additive manufacturing (also known as 3D-printing). After an initial dataset and model training, they use Bayesian optimization to find new, fitting polymers to synthesize, analyze, and add to the training set before retraining. This process is iterated multiple times.

Hickey, Feinstein, et al. (2024) train a graph convolutional neural network (GCNN) to predict the glass transition temperature of polymers. The training data is scraped from the literature and consists of 7558 T_g values. This model is compared to a quantum chemistry approach, using 100 temperature values from the same source for the calculations. The GCNN shows an R^2 of 0.9 and RMSE of 38.08 °C. The QM model exhibits an R^2 of 0.86 and RMSE of 34.53 °C.

Mysona, Nealey, and Pablo (2024) predicts the lamellar period of homopolymer block polypeptides using an ANN. In a sample gathered from a simulation, all polymer chains follow the exact same structure and chemical species ordering. This is opposed to random block copolymers, where the species ordering is not consistent. Every polymer chain consists of two blocks, one consisting of 32 beads of the same species, the other of 32 beads of two different species. Data sets are constructed from simulations, and feed-forward neural networks are trained on them.

Fibre Reinforced Polymers

To further enhance the properties of polymers, they are commonly used for composites. Composites are materials made out of at least two different materials. The properties of the composite are then different from those of the individual elements [38]. This section focuses on FRPs composites.

Since these composites are commonly used in construction, predicting their fatigue and failure is important. Loh, H. T. Nguyen, and K. T. Nguyen (2024) use Deep Learning (DL) to predict failure properties of fibre-reinforced polymer laminates under fire exposure. Three models are trained: One predicts the time to failure, temperatures of the surfaces facing towards and away from the fire at failure, and the axial displacement at failure (DNN 1); the other two predict the temperature history of the unheated surface, but for materials with different flux (both referred to as DNN 2). The training data is obtained experimentally, by exposing the material to constant axial tension stress while applying constant radiant heat flux, so heat was transferred by electromagnetic radiation to one side. DNN 1 is trained on 38 data points, while DNN 2 low flux and DNN 2 high flux are trained on 57,154 and 88,227, respectively. The data stems from the same experiments, but DNN 1 only predicts the final outcome, while DNN 2 models the entire temperature history until failure, which is why it requires much more training data. DNN 1 shows good agreement with the experimental values in its predictions, with R^2-values ≥ 0.977 for all properties, albeit slightly overestimating the

axial displacement at failure, which the authors attribute to the same being true for the training data. The same holds true for DNN 2, with an R^2 of 0.944 on the test set for both the high- and low-flux versions.

Other works investigating FRP fatigue and damage include Osa-uwagboe, Udu, et al. (2024), who predict damage of FRPs under out-of-plane load. k-NN is the best-performing algorithm, but most models have very high R^2 scores. Another example is Mahapatra and Satapathy (2024), who investigate the erosion behavior of sponge iron slag-filled ramie-epoxy-based hybrid composites. Training data stems from past published work by the authors. The Gradient Boosting Machine (GBM) model performs the best over SVM, RF, and DT. Oh, D. Lee, and Park (2024) use changes in electrical resistance in carbon reinforced polymer pipes and unsupervised learning to detect damage in the pipes. Experimentally obtained data of impact damage is reduced to two dimensions using PCA and then clustered by damage type using k-means clustering. A. Kumar, Arora, and Nehdi (2024) predicts near surface mounted FRP rod-to-concrete bond strength and failure mode. Multiple models are tested, and XGBoost performs the best, outperforming past analytical models. The other models the authors test include linear regression, DT, GBDT, RF.

Albuquerque, Sarhadi, et al. (2024) use a CNN and GPR to reconstruct fatigue damage from thermal imaging. This technology can be used for monitoring. Both artificially constructed images using a multivariate Gaussian distribution and images taken from materials in experiments are used for training. Compared to a simple least squares regression, they perform slightly better, but the linear regression is already good, with R^2 values above 0.9.

Tian, L. Wang, and Xian (2024) use different ML models to predict the bond strength of fiber reinforced polymer bars to concrete. The authors deem CatBoost the most suitable for the task, and the models outperform existing mathematical equations from the literature.

Rabby, Das, et al. (2024) train multiple models to predict different properties related to the curing of fiber-reinforced polymer composites using dielectric features obtained from experiments. First, they create a SVM to categorize the curing degree. Next, they train a MLP to predict tensile strength. Lastly, another SVM classification model is trained, which can monitor the aging process and degradation of prepreg, the raw material used to create the composites.

Tunukovic, McKnight, et al. (2024) utilize ML for flaw detection of Carbon Fiber-reinforced polymer (CFRP) materials from ultrasonic testing (UT). UT works by sending an ultrasonic impulse into the material and analyzing the returning waves after they interact with the sample. An example of a UT image with and without a defect present can be seen in Figure 3.2. Multiple ultrasonic scans of 8 different CFRP samples are created for training. Notably, none of these contain any defects. The defect detection

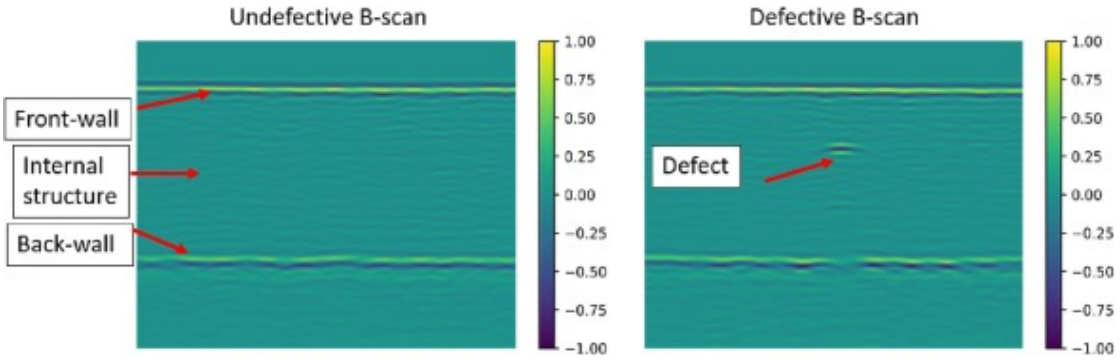

Figure 3.2: An example of a UT image, both with and without a defect. The image was taken from Tunukovic, McKnight, et al. (2024).

problem is transformed into an anomaly detection task instead, where anomalous results produced by the model indicate structural flaws. For this task, an unsupervised autoencoder is trained. The samples are transformed into a latent space, and the decoder learns to reconstruct the original input from the latent space representation. The difference between the input and output serves as the loss function; in this case, the MSE is used for the loss function. Furthermore, a DBSCAN clustering model is trained as a gating model. Defect/Anomaly detection works by analyzing the loss of the reconstructed image, where a higher error indicates defects. In certain structures with complex geometry and varying thickness, finding defects would be more difficult, as the small error around the defect gets drowned out in the MSE, which considers all pixels of the scan image. Such areas include the front- and back-walls, also marked in Figure 3.2. After clustering, areas to be removed before using the sample to train the autoencoder network can be identified. To test the model, scans from samples with artificial defects are utilized. The model trained on the gated dataset performs better and indicates defects more clearly. Visible reconstruction errors are identified in 36 out of 40 simple geometry samples, and 22 out of 24 are identified in the complex geometry samples.

3.2.4 Metal Organic Frameworks

MOFs are three-dimensional structures, consisting of metal corners connected by organic linkers [62].

Kang and Kim (2024) introduce ChatMOF, which finds or generates MOFs based on user input. These user inputs are natural language queries, interpreted by a LLM which generates scripts to fulfill requests. For instance, after the user asks how the density of a certain MOF compares to that of others, the model generates Python scripts that find

the density of the specific material and the average density of all compounds, compares these values, and then responds with the results. ChatMOF consists of three components: the agent, the toolkit, and the evaluator. The agent generates a strategy to obtain the information requested by the user, referring to the toolkit. The toolkit contains the tools necessary to obtain the information. They include table searcher, predictor, generator, and general utilities like calculators, visualizers, and internet searchers. If the required information is already present in the tabulated data, the table searcher finds it; otherwise, the predictor is employed, which utilizes the MOFTransformer model to predict it instead. For each request, the most appropriate, fine-tuned MOFTransformer model is selected. The generator can generate new structures fulfilling the required properties. To achieve this, a genetic algorithm is employed. The gene representations for this are textual, facilitating the usage of an LLM. The data for these processes stems from multiple databases. The CoREMOF database, containing various properties, and QMOF, used when electrical data, like formation energy, HOMO, or LUMO, is needed. Finally, the evaluator, another LLM, generates a final response from the results provided by the toolbox. The authors employ GPT-4, GPT-3.5-turbo, llama2-7B-chat, and llama-2-13B-chat as agent, toolkit, and evaluator.

For certain applications of MOFs, like catalysis, their stability in water (or other materials) is important. Without stability, they could break down and lose their functionality [73]. Accordingly, there are efforts to find stable MOFs using ML. Terrones, S.-P. Huang, et al. (2024) predict water stability of MOFs. First, a dataset of 1092 MOFs is derived from the WS14s and WS24s datasets. Each compound is assigned to one of four groups: unstable (U, cannot tolerate moisture), low kinetic stability (LK, can tolerate low humidity), high kinetic stability (HK, can tolerate high humidity), and thermodynamic stability (TS, stable in water for at least seven days, or at least one day in acidic, basic, or boiling conditions). Two model types are trained: 4-class models, assigning MOFs to one of the four aforementioned classes, and 2-class models, classifying compounds as either U+LK or HK+TS. RF is exclusively used for these models. Additionally, a genetic algorithm is employed to identify potential new (as in not yet synthesized) MOFs.

Z. Zhang, Pan, et al. (2024) also use RF classifiers to achieve the same goal. Additionally, they experimentally verify the prediction results for two compounds. The classifier correctly predicts both as stable and one as more stable than the other, by assigning a higher likelihood of being stable to the more stable one.

Yue, S. A. Mohamed, and Jiang (2024) also train a RF and Crystal Graph Convolutional Neural Network (CGCNN) to predict thermal expansion properties of MOFs. They are trained on compounds from the CoRE-MOF and ARC-MOF databases, and with data obtained from MD simulations. Notably, RF performs better. G. Zhao and Chung (2024) predict MOF partial atomic charges also using a CGCNN. These charges

are relevant for the adsorption properties of the materials.

Deng and Sarkisov (2024) use the CoRE MOF database and simulations to train an XGBoost algorithm to predict gas adsorption of ethane, propane, krypton, xenon, CO_2, and N_2. This work specifically focuses on highlighting the features used for the model. Bailey, Jackson, et al. (2023) train models to predict the H_2, CH_4, and CO_2 uptakes in MOFs. GBDT performs the best.

For usage in catalysis and gas separation, the diffusivity of a MOF is also important. Guo, X. Huang, et al. (2023) predict the diffusion selectivity of binary gas mixtures and diffusivity of the gases using RF, GBDTs, XGBoost, and LGBM. The 6013 MOFs the authors use for training stem from the 2019 CoRE-MOF database and the diffusivity of nine gases (Xe, CH_4, Kr, N_2, H_2S, O_2, CO_2, H_2, and He) from MD simulations. LGBM performs the best, with an average R^2 of 0.967. Additionally, the LGBM model performs well when extrapolating on C_2H_6, as confirmed by MD simulations. After conducting SHAP analysis on the same model, the authors find that pore limiting diameter is the most impactful feature determining molecular diffusivity, while the molecular polarizability primarily determines the diffusion selectivity of binary gas mixtures.

Sarikas, Fanourgakis, et al. (2024) employ a self-consistent approach for training a ML model to screen MOFs for hydrogen storage capabilities. This approach is iterative and works as follows: First, an initial training set is created by randomly selecting a set number of materials from the entire dataset. Then, their adsorption uptakes are calculated by grand canonical Monte Carlo simulations. With this training set the model is trained and predicts the gas capacity of the compounds in the entire data set. A set number (which can be different from the size of the initial training set) of the MOFs with the highest capacities is then added to the training set, and the whole process starts again with simulations. At some point, all the MOFs with the highest gas capacities, as predicted by the model, will already be in the training set, and it will no longer change after the last step. When this happens, the process is considered converged and ends. In this work, the size of the initial training set and the number of compounds added each iteration are set to 100. The goal is to get performance on par with regularly trained models while greatly reducing the size of the training set. To evaluate this approach, 98,695 MOFs are collected from 19 databases. The authors perform 100 runs of trying to find the 100 best performing MOFs in terms of volumetric and gravimetric capacity under pressure swings (PS) and temperature-pressure swings (TPS) from these 98,695 materials. The model is able to find all the compounds with the highest gravimetric capacities under both PS and TPS conditions in almost all runs, while its ability to do so was severely lacking in finding the highest volumetric capacities. The authors attribute this to high volumetric capacity MOFs spanning different regions of the feature space, and almost spanning the entire feature space at TPS conditions, as opposed to MOFs

with high gravimetric capacities appearing in only two regions of the feature space.

Lin, H. Zhang, et al. (2024) train ML models on 224 compounds from the Conductive MOF (CM) database. Multiple model types are trained both for classification and regression of electrical conductivity. For classification tasks, an ensemble model of CatBoost, RF, and ANN using a majority-vote system performs best, while ANN performs the best for regression.

MOFs also show potential for Photocatalytic water splitting (PWS), the process of splitting water into oxygen and hydrogen under sunlight. C. Wang, Wan, et al. (2024) utilize ML to find MOFs capable of efficient PWS. However, they only use a CGCNN to complete missing properties and then search for fitting MOFs using simulations.

3.2.5 Other Materials

Madani, Lacivita, et al. (2025) create CrysCo, a more general model to predict the properties of different materials. It is a hybrid model made of two separate networks; the first is a GNN taking in crystal structures, and the second receives compositional features and physical properties. It is trained as a whole at once using data from the Materials Project database. The graphs used to train the GNN are especially noteworthy. Three graphs are generated to capture specific features of the crystal structure, and any pair of atoms within 8Å is considered connected. In the first graph, G8 nodes represent atoms, and edges are bonds; the second graph L(G8) is a line graph of G8, where nodes are bonds and edges are bond angles, and lastly, L(G8d) is another line graph similar to L(G8), but the edges now represent dihedral angles. Using the final model, it is possible to predict properties of inorganic materials, like formation energy or bandgaps. Compared to other past models, this approach holds up very well and often comes out on top in terms of accuracy, using MAE as the metric.

R. Wu, Y. Zhang, et al. (2025) create both a black- and white-box ML model to predict fatigue properties of twinning-induced plasticity steels (TWIP steel). For the black-box approach, the experimentally obtained data is first clustered with the k-means clustering algorithms, after which the best regression model is determined and trained for each individual cluster. Five candidate models are tested for each cluster: Kernel Ridge Regression (KRR), Adaptive Boosting Regression (ABR), RF, SVR, and GPR. The white-box approach uses symbolic regression, starting from a big population of mathematical formulas and gradually replacing less fitting ones until a mathematical expression for property prediction is found.

J. Wang, Fan, and Cai (2025) use multiple ML algorithms to predict the fatigue performance of stainless steel and compare their performance. The models the authors test are LightGBM, SVM, RF, K-Nearest Neighbor (KNN), GB, XGBoost, ANN, and Genetic Algorithm-Artificial Neural Network (GA-ANN). GA-ANN differs from a

normal ANN in that it uses a genetic algorithm to fine-tune model parameters. Among these, GA-ANN performs the best, with ANN being in second place, with both being significantly better than the rest, using R^2, MAE, and MSE as metrics.

ML can also be used to classify unknown materials. López-Baldomero, Buzzelli, et al. (2025) use multiple different ML approaches (SVM, RF, Partial Least Squares Discriminant Analysis (PLS-DA), KNN, Linear Discriminant Analysis (LDA), DL) to classify the inks used in historical documents, which is of interest for preservation and restoration. The input data are photos taken by two cameras for visible and near-infrared (VNIR, 380-1080 nm) and short-wave infrared (SWIR, 888-1732 nm) light. The models are trained using a mixture of real historical documents, mock-ups, and a subset of the Microsoft COCO dataset. DL performed the best overall, with SVM being the best among the traditional models. The authors argue that the choice between the two should be made primarily on available resources, as DL requires more computational power.

Lai, Gomez, et al. (2025) predict the ultimate tensile strength (UTS) of cold-sintered composites using X-ray computed tomography (CT) images. Since the other tested models perform badly, they use a CNN where they replaced the last layer with a single-node output that predicts UTS. To further improve performance, the 5 models trained by the 5-fold validation are combined into an ensemble model, taking the average of all models as the final prediction. This ensemble model reaches an R^2 of up to 0.94 on unseen data, despite the training set only consisting of 31 samples. Furthermore, after utilizing a median meta-learner, the final test set R^2 can be raised to 0.95.

Hossain, Uddin, et al. (2024) create ANNs that predicts the compressive strength, tensile strength, and tensile strain of engineered cementitious composites. Samples collected from the literature serve as training data. Multiple ANN models are trained for each property, one with 2, 4, 6, 8, 10, 12, and 14 hidden layers each. The model with 10 hidden layers performs the best for the prediction of compressive and tensile strength, and the model with 12 hidden layers performs the best for tensile strain. Additionally, the authors analyze the impact of the features on the prediction results.

Mirzaei, Haghi, and Shokrieh (2024) use an ANN, whose hyperparameters are fine-tuned by the NSGA-II genetic optimization algorithm, to predict the fatigue life of laminated composites. The authors conducted both experiments and simulations to obtain the necessary training data. The model shows a high R^2, and strongly outperforms other, traditional models (DT, RF, SVR, Gradient Boosting). Additionally, the authors find that the magnitude of the applied force has the biggest impact on the fatigue life prediction by far.

W. Zhao, Pang, et al. (2025) investigate the residual stress and microhardness of metallic materials after laser shock peening (LSP) using an ANN. During LSP, a strong laser is used to create bubbles of plasma on the surface of a material. These bubbles

gradually heat up and expand as they absorb more energy from the laser until they explode, sending a shockwave through the sample. The goal of this procedure is to improve the fatigue, wear, and corrosion resistance of the material. The model the authors train is physics-inspired, as the input features, which are related to the peak pressure of the shockwave, the propagation of the shockwave, and the deformations of the material under the shockwave, are directly, physically related to the predicted properties. Training data is obtained from the literature. The authors compare the results to past ANNs with the same goal, but whose features were more traditional and not physics-inspired. The physics-based ANN reaches an R^2 of 0.99935 and MAE of 0.00644 for microhardness, while the traditional model has an R^2 of 0.87661 and MAE of 0.10805. For predicting residual stress, the results are closer, but the physics-inspired ANN still outperforms the traditional one. Additionally, when compared to the results of an empirical formula, the author's model also produces better results.

Kameni, Palessonga, et al. (2024) use an ANN with one input and output, and two hidden layers with ten neurons per layer to predict the conductivity from the electromagnetic shielding of three different composites. A sample is taken for each material, and training data is gathered experimentally. The three tested composites are an aluminum foil and plexiglass one, an aluminum grid in a dielectric matrix, and a resin filled with carbon fibers.

Mallah, Güleryüz, et al. (2025) use ML to find new organic semiconductors. They train multiple models using 1142 datapoints from a literature review and websites containing chemical data. The best performing models are Ridge Regression and RF.

There also exist generative models for more general usage. MatterGen[90] is a diffusion model that generates inorganic crystal compounds. To train the base model, the authors use 607,683 compounds from the Materials Project and the Alexandria datasets. The crystalline materials are defined by their chemical elements, coordinates, and periodic lattice. For the reverse diffusion process, a score network is trained on a large number of stable materials. To ensure generated materials fall within the given constraints, an adapter module is injected into the score network at each layer, which fine-tunes the model using labeled data. This adapter-based approach has the advantage of requiring many fewer training samples while still being effective. MatterGen is able to both generate possible crystal structures from elements and materials exhibiting given magnetic, electronic, and mechanical properties. The model is able to rediscover already known materials not seen in the training data. During a test run, the authors generate 1000 structures, all of which are unique, and the ratio of unique structures only drops to 52% after generating 10 million, 61% of which are new, illustrating the model's ability to generate varied structures. One of the generated compounds, $TaCr_2O_6$, is synthesized. The generation for this material uses a bulk modulus target of 200 GPa, and DFT predicts a value of 222 GPa for this property. Additionally, of the total 8,192

generated compounds when targeting 4 different bulk modulus values (50 GPa, 100 GPa, 150 GPa, and 200 GPa), 101 match already known ICSD compounds. The 95 property values successfully calculated by DFT for these materials align well with the target, resulting in an MAE of 23 GPa and an RMSE of 32 GPa.

Vieira Wyzykowski, Niazi, and Dickson (2025) introduce AGDIFF, a diffusion model trained on the GEOM-QM9 and GEOM-Drugs datasets. The goal of the model is to predict 3D locations of atoms in a molecule from its 2D graph representation. The graph representation of the molecules contains further properties besides the general structure, like bond types and the presence of charges, by labeling edges and nodes with this information. The model utilizes two encoder streams, a local and a global one. While the global encoder takes the entire molecular structure into account, or the entire molecular graph in this case, the local encoder focuses on a smaller set of edges. Local edges are determined by their distance being within a set cutoff distance of 10 Å. The global encoder handles both local and global edges, while the local encoder only gets local edges passed to it. Furthermore, the encoder is improved from past works by adding an attention mechanism and a 10 Å cutoff distance past which atomic interactions are ignored, among others. To test the model, its performance on alanine dipeptide is evaluated. Multiple different configurations for the molecule are generated, and the distribution of the results is in good agreement with past studies of the compound. Additionally, its performance on the test sets from the GEOM-QM9 and GEOM-Drugs datasets is evaluated. For this COV-R, the proportion of reference conformations generated by the model, and MAT-R, the average deviation of each generation from the nearest reference, are used as metrics. AGDIFF reaches a mean COV-R of 93.08% and a mean MAT-R of 0.1965 Å on the GEOM-QM9 dataset, and a mean COV-R of 91.31% mean MAT-R of 0.8237 Å on the GEOM-Drugs dataset.

Quesada-Molina, Mofatteh, et al. (2025) first generate Voronoi patterns with different numbers of seeds (or centers) and edge spacings. The patterns had two distinct phases, which were utilized as a base to construct two-phase composites. In this case, approximately 6000 images are generated, and one phase is set as thermoplastic polyurethane (TPU), and the other as polylactic acid (PLA), resulting in about 12000 materials. The image is mirrored to generate 3 additional ones, and the representative unit cell (RUC) is extracted from the result. This RUC is additionally discretized, resulting in 22500 CPS4 finite elements (FE), on which FE simulations are conducted to find elastic properties, specifically Young's modulus E_1, and the volume fraction of the stiff phase v_f^{PLA}. To confirm the simulation results, two different samples are manufactured and undergo stress testing, resulting in a below 5% error margin between the two methods of obtaining data. Thus, the simulation results are considered the ground truth moving forward. These images are used to train a CNN, which predicts the aforementioned elastic properties. Furthermore, a Deep Convolutional

Generative Adversarial Network (DCGAN) is trained to generate synthetic images of Voronoi patterns. A CNN generates the synthetic 128×128 pixel images from a latent space vector, and another CNN classifies the images into fake and real ones. The feedback from the classification model is used to adjust the generative model during training, so that it can generate highly realistic images. Furthermore, an optimization algorithm for the inverse design of materials following the Voronoi-inspired structure is implemented. For this, the generative network of the DCGAN and the CNN predicting E_1 and v_f^{PLA} are utilized. The algorithm works as follows: First, latent space vectors are randomly sampled until one that causes the generative model to generate images, whose properties, as predicted by the regressive CNN, are within a given maximum error of the requested values. This error is a weighted sum of squares

$$L_{SSE} = \lambda_1(\hat{E}_1 - E_1)^2 + \lambda_2(\hat{v}_f^{PLA} - v_f^{PLA})^2, \tag{3.2}$$

where $\hat{E}_1$ and $\hat{v}_f^{PLA}$ are the predicted properties, E_1 and v_f^{PLA} are their wanted values, and λ_1 and λ_2 are weighting factors given by the user. After a suitable vector is found, the optimization phase of the algorithm starts. For each iteration of this phase, each individual element of the latent space vector is perturbed by a small, predetermined amount, and the gradient of the loss function L_{SSE} with respect to the perturbed element is approximated. Then, the same element of the vector is updated by a given step size according to the gradient. Note that the updated components are not taken into account for perturbations and loss function calculations until the next iteration. The number of iterations this process undergoes is also predetermined by the user.

Li, Miklaucic, and J. Hu (2025) address the problem of out of distribution (ood) property prediction with limited labeled data available. This is of interest when few materials of a new family have been found, and now more members of this family are sought after. To test their model, they have to first create an ood dataset. 84,190 compounds from the MP database are selected, with 145 composition-based properties serving as features X, and either formation energy or bandgap being the predicted property Y. Three different cases of ood data are identified: Covariate shift, Prior shift, and Relation shift. A Covariate shift occurs when the distribution of the input data X is different between training and inference. They obtain ood data of this type by collecting datapoints on the periphery of clusters after reducing the input data to two dimensions using dimensionality reduction. Prior shifts are similar to covariate shifts, but refer to a change of the distribution of the target data Y. Extreme (outlier) values of Y are collected for prior shift data, specifically the top or bottom 10%. Piezoelectric materials are collected as relation shift samples, which are characterized by a shift in the joint distribution of X and Y. All the aforementioned ood samples are removed from the training data. To tackle the problem of accurately predicting these ood cases, the authors introduce Crystal Adversarial Learning (CAL). Periodically during training,

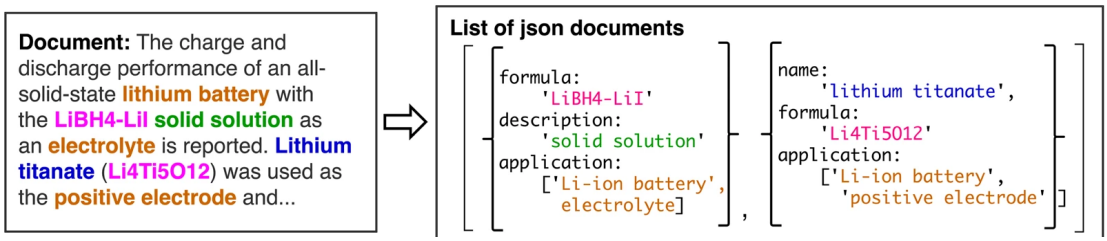

Figure 3.3: A sample text section and what information the model should extract from it. The image is an adapted version from Figure 1 [15], which is published under Creative Commons Attribution 4.0 International License (https://creativecommons.org/licenses/by/4.0/).

the samples with the highest loss are selected and exclusively focused on until new samples with the highest loss are reselected. These samples, called partial samples, are used together with selected test domain samples to generate adversarial samples. Specifically, covariates are classified into stable and unstable ones, and the unstable ones are perturbed using gradients. Using this method, the CAL model is able to achieve much lower error values on the ood sets compared to regular models.

3.2.6 Other Uses of Machine Learning

Another way to obtain training data for ML is by automatically extracting information from past research papers. For this purpose, natural language processing (NLP) algorithms and LLMs are promising approaches. For example, Dagdelen, Dunn, et al. (2024) fine-tune the GPT-3 and Llama-2 LLMs for entity recognition and relation extraction (NERRE) from scientific papers. Then, text passages, like the abstract of a paper, can be given to the model, which then outputs entities and relations in a defined structure. This structure can be natural language sentences or JSON objects with set keys. An example of what a text passage and the information extracted from it could look like can be found in Figure 3.3. To fine-tune the models, 400-650 text passages are manually annotated in the way that the model should later output information in. Specifically, fine-tunes for doping information, MOFs, and general materials are created. Since the same information can potentially be conveyed in multiple ways, automatic performance evaluation methods requiring the output to exactly match the input do not paint an accurate picture, but rather a lower bound of performance. So, additionally, the accuracy of the model is also manually evaluated by domain experts judging if correct information is extracted. Overall, both models show good performance, with the GPT-3-based one being better overall. One major advantage to fine-tuning existing models is that it can be done by a scientist without any domain knowledge in LLMs or

NLP. However, these models are still prone to something called hallucinations, which is the generation of information that is not in the text. The authors suggest that this could be avoided by requiring extracted information to be present word-for-word in the source text, but this would come at the cost of automated data normalization. The same information can be presented in multiple ways in the literature (e.g., N-doped, nitrogen doped, doped with nitrogen, etc.), and any is factually correct. By properly fine-tuning, the model can implicitly learn to transform these different forms into the same form for consistency, even if this form does not appear in a given passage, reducing the need for post-processing. The authors differentiate this automatic normalization from hallucinations, and this ability could be lost by requiring the word-for-word presence of extracted information in the text.

As mentioned before, an important part in the ML workflow is selecting features for the model. To help with this, Sharma Priyadarshini, Thota, and Hernandez (2025) introduce ReLMM, a reinforcement learning based model that selects an optimal, minimal feature set to predict a given property. A set of potential features and a property to predict are predefined, as well as a model for which the optimal feature set is to be found. In this work, the predictive model is always a DT, but others can be used instead. Each training episode starts with a randomly selected subset of features. Then, each agent removes or adds a feature based on its respective policy, which is a neural network trained over multiple episodes. The number of agents is equal to the total number of potential features. These features are then used by the model, which should be optimized, to predict a property for a validation data set. The loss $\mathcal{L}_{t,A}$ at step t is then computed using the MSE function found in equation 2.2, and the reward for each agent A is

$$R_{t,A}(s_t) = \frac{1}{\mathcal{L}_{t,A}} \tag{3.3}$$

at state s. This reward is used to calculate the Q-values

$$Q_{t,A}(s_t, a_{t,A}) = R_{t,A}(s_t) + \gamma \max_{a_{t+1,A}} Q_{t+1,A}(s_{t+1}, a_{t+1,A}), \tag{3.4}$$

where $a_{t,A}$ is the action a being taken by agent A at time step t, and γ is a hyperparameter set to 0.95 for ReLMM. The neural network serving as policy is trained to predict the second term of this sum over multiple episodes. The next action $a_{t+1,A}$ taken by each agent is the one that results in the highest Q-value. To evaluate this approach, it is tested on a real dataset to find features for the prediction of bandgaps in metal halide perovskites. The features selected by ReLMM result in lower MSE compared to those selected by XGBoost and LASSO, two other methods of feature selection, while also not selecting multiple features that are strongly correlated with each other.

3.3 Machine Learning Methods

3.3.1 Decision Trees and Random Forest

DTs resemble a tree in their structure and are made out of nodes. A DT starts at a root node, and each node has two or more children. Inference starts at the root node, and at each node, the algorithm proceeds to one of the child nodes, based on one or more input variables, as defined by the current node. This repeats until the final node, called the leaf node, is reached, where the final decision is made. During training, the tree is constructed recursively, and child nodes are created in a way that maximizes the information gain [94]. This model is not used very often, but forms the basis for a much more popular model, RF.

RFs are ensemble learners and work by constructing multiple DTs. However, at each node of the individual trees, a random set of variables is excluded as candidates for a decision [74], and the variable chosen is the best performing one [94]. The final result of an inference on x is

$$\hat{y}(x) = \frac{1}{N} \sum_{i=1}^{N} T_i(x), \tag{3.5}$$

the average result of all individual trees, where N is the number of trees and $T_i(x)$ is the prediction of tree i [33].

3.3.2 Boosting Algorithms

Different types of boosting ML algorithms exist. At their core, they are ensemble learners, which construct multiple base learners one by one in an iterative manner. During this process, the loss of past learners is taken into account to construct the next one. The entire training data is used during each iteration to train the base learner g_i, and additionally, all base learners are assigned weights α, which get optimized over the iterations as well. The final boosting model G_M with M base learners uses the formula

$$G_M(x) = \sum_{i=0}^{M} \alpha_i g_i(x) \tag{3.6}$$

to calculate the prediction for input x [94]. DTs are a common choice as base learners, used by GBDT and XGBoost for example [74].

3.3.3 Artificial Neural Networks

Generally, ANNs consist of an input layer, an output layer, and a variable number of hidden layers. Each layer is made up of a set number of so-called neurons, and its

neurons are connected to the neurons of the layer before and after it. The input layer neurons accept the input, and the output layer neurons give the prediction results. The number of neurons in the former is equal to the dimensions of the features used for the model, and the number of neurons in the latter depends on the task of the model. In the context of materials science, regressive ANNs generally predict only one property, in which case the output layer consists of one neuron, while it conists of a number of neurons equal to the number of possible classes in the case of classification models [19]. The number of hidden layers and the number of neurons in each hidden layer vary between models.

The value of each neuron after the input layer can be calculated by

$$x_i = f(\sum_j w_{ij} x_j + b), \tag{3.7}$$

where w_{ij} refers to the weight of the connection from node j to node i, and f is a specific activation function, and b is a bias value. In other words, the value of each neuron is equal to the result of the weighted sum of all neurons of the previous layer after being passed to a set activation function. This forward propagation process repeats and cascades through the network until the final layer is reached and the final prediction result is given [99]. Multiple choices exist as activation functions; tanh

$$f(x) = tanh(x), \tag{3.8}$$

ReLU

$$f(x) = max(0, x), \tag{3.9}$$

and Sigmoid

$$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}} \tag{3.10}$$

are some of the most widely used ones [19].

The weights of the connections between the neurons and biases are set during training. For this, a loss function is defined, which the algorithm tries to minimize by modifying the weights and biases during training [99].

3.3.4 Convolutional Neural Networks

CNNs are based on ANNs and can accept images directly as input. They utilize convolutional layers and pooling layers. Convolutional layers use a kernel function to extract information from the input. Pooling layers reduce the dimensionality of the input using different methods, like max pooling. How often each layer occurs and their arrangement depends on the specific architecture of the model [84].

3.3.5 Diffusion Models

Diffusion models are made out of two parts: the forward process and the backwards (denoising) process. The forward process starts at time step 0 with a datapoint Z_0 and gradually adds noise to it over multiple steps, until it is transformed into pure Gaussian noise. This is represented by a Markov chain, where each step is given by

$$q(Z_t|Z_{t-1}) = \mathcal{N}(Z_t; \sqrt{1-\beta_t}Z_{t-1}, \beta_t I), \tag{3.11}$$

where Z_t is the sample at step t, β_t is a sequence of values controlling the amount of noise added each step, $\mathcal{N}$ refers to the multivariate Gaussian distribution, and I is the identity matrix [102]. During training, noise is added to a sample, and the model then learns to remove this noise in the reverse process to reproduce the original data [88]. This reverse process is defined as

$$p_\theta(Z_{t-1}|Z_t) = \mathcal{N}(Z_{t-1}; \mu_\theta(Z_t, t), \Sigma_\theta(Z_t, t)), \tag{3.12}$$

with θ being the denoising model's parameters, and μ_θ and Σ_θ being its fitted mean and variance functions respectively [102, 88]. After training during the generation process, random noise is sampled as Z_t, and the model trained for the reverse process is used to remove this noise [102].

4 Conclusion

This review gave an introduction to ML and materials science, and how ML algorithms are utilized for materials science. Multiple different model types, like RF and ANNs, and how they work were covered. Different material groups and how ML is used for researching them were covered, with superconductors, perovskites, polymers, FRPs, and MOFs being covered in more detail. RF and gradient boosted models are commonly employed for various regression and classification tasks. Generative models are most commonly built on diffusion models, but are rather rare compared to regression and classification models. Generative Adversarial Networks (GANs) are another type of generative models. Besides uses directly relating to the materials, ML is also used for other, important things, like model feature selection and data collection.

Overall, ML is a promising tool for the field of materials science. Models can predict different properties and help design and find new materials for any application. Additionally, ML can also be employed for monitoring defects and wear. The only limiting factor is the availability of training data, as one first needs to gather enough data on the properties one wants to predict. This data is not always available and might need to be gathered experimentally or through simulations first. One of the most common applications of a trained model is high-throughput search. The model is used to scan a large number of compounds to find those that fulfill predetermined requirements. For instance, to find new superconductors, commonly, models predicting a material's transition temperature are trained, and then are used to scan databases for the compounds with the highest T_c, and filter out any non-superconductors as a side effect. In a similar vein, classification models are also used for high-throughput search, for instance by classifying MOFs by their water stability. Some of the most powerful tools, albeit difficult to implement, are generative models. They can directly generate a compound within given constraints, removing the need to search through vast amounts of materials entirely. Specifically, diffusion models are used most often to create generative models. For general regression applications, many different model types are employed, and commonly multiple of them are trained and evaluated to find the one performing the best at the task. Specifically for crystal structures (like MOFs), graph-based neural networks are worth mentioning, as modeling the crystal structure as a graph lends itself very well for these types of models. Across all types of regressive tasks, RF and various types of gradient boosting models are commonly

used and generally show good results, if not the best among the tested models. ANNs are also worth mentioning here, as they often show good performance; However, they have the disadvantage of requiring more computational resources as well as generally more training data. Lastly, these models can improve our physical understanding of the materials. Tools like SHAP analysis reveal the impact each feature had on a prediction, increasing interpretability and deepening our understanding of the material. There even exist techniques like symbolic regression, which can directly produce a formula to calculate a property.

The prediction accuracy of the models could be further improved in the future in several ways. Firstly, more data should be gathered and made publicly accessible. While certain material classes have their own dedicated databases, others lack them. This makes gathering the data needed for training much harder. Additionally, other models for the automatic collection of data from the literature can help facilitate this process. Furthermore, it should be ensured that the data available in databases is of high quality and complete. As it stands right now, researchers usually need to clean the data first by removing invalid materials and dealing with missing property values. The ML models themselves also can be further improved. Especially physics-informed models show potential and should be further developed. By directly taking physical and chemical laws into account, models can achieve higher accuracy with less training data. Furthermore, feature selection is an important factor to keep in mind. Domain knowledge helps in this area as well, in addition to statistical analysis, revealing the impact of each feature.

Bibliography

[1] Z. Abdelhamid, H. Mohamed, and S. Kelouwani. "The use of machine learning in process–structure–property modeling for material extrusion additive manufacturing: a state-of-the-art review." In: *Journal of the Brazilian Society of Mechanical Sciences and Engineering* 46.2 (2024). DOI: 10.1007/s40430-023-04637-5.

[2] I. M. Abdellah and A. El-Shafei. "A machine learning approach for in silico prediction of the photovoltaic properties of perovskite solar cells based on dopant-free hole-transport materials." In: *New Journal of Chemistry* 48.44 (2024), pp. 18666–18682. DOI: 10.1039/d4nj03777d.

[3] N. Ahmad, M. A. Ibrahim, S. R. Sayed, S. S. Ahmad Shah, M. H. Tahir, and Y. Zou. "Data-mining and machine learning based search for optimal materials for perovskite and organic solar cells." In: *Solar Energy* 287 (2025). DOI: 10.1016/j.solener.2024.113223.

[4] R. Q. Albuquerque, A. Sarhadi, M. Demleitner, H. Ruckdäschel, and M. A. Eder. "Fatigue damage reconstruction in glass/epoxy composites via thermal analysis and machine learning: A theoretical study." In: *Composite Structures* 331 (2024). DOI: 10.1016/j.compstruct.2023.117855.

[5] A. Alfares, Y. A. Sha'aban, and A. Alhumoud. "Machine learning -driven predictions of lattice constants in ABX3 Perovskite Materials." In: *Engineering Applications of Artificial Intelligence* 141 (2025). DOI: 10.1016/j.engappai.2024.109747.

[6] X. Bai and X. Zhang. "Artificial Intelligence-Powered Materials Science." In: *Nano-Micro Letters* 17.1 (2025). DOI: 10.1007/s40820-024-01634-8.

[7] T. Bailey, A. Jackson, R.-A. Berbece, K. Wu, N. Hondow, and E. Martin. "Gradient Boosted Machine Learning Model to Predict H2, CH4, and CO2 Uptake in Metal-Organic Frameworks Using Experimental Data." In: *Journal of Chemical Information and Modeling* 63.15 (2023), pp. 4545–4551. DOI: 10.1021/acs.jcim.3c00135.

[8] H. Bi, M. Wang, L. Liu, J. Yan, R. Zeng, Z. Xu, and J. Wang. "The influence of perovskite crystal structure on its stability." In: *Journal of Materials Chemistry A* 12.21 (2024), pp. 12744–12751. DOI: 10.1039/d3ta07457a.

[9] Á. D. Carral, M. Roitegui, and M. Fyta. "Interpretably learning the critical temperature of superconductors: Electron concentration and feature dimensionality reduction." In: *APL Materials* 12.4 (2024). DOI: 10.1063/5.0189714.

[10] T. F. T. Cerqueira, Y.-W. Fang, I. Errea, A. Sanna, and M. A. L. Marques. "Searching Materials Space for Hydride Superconductors at Ambient Pressure." In: *Advanced Functional Materials* 34.40 (2024). DOI: 10.1002/adfm.202404043.

[11] T. F. T. Cerqueira, A. Sanna, and M. A. L. Marques. "Sampling the Materials Space for Conventional Superconducting Compounds." In: *Advanced Materials* 36.1 (2024). DOI: 10.1002/adma.202307085.

[12] M. Chen, Z. Yin, Z. Shan, X. Zheng, L. Liu, Z. Dai, J. Zhang, S. (Liu, and Z. Xu. "Application of machine learning in perovskite materials and devices: A review." In: *Journal of Energy Chemistry* 94 (2024), pp. 254–272. DOI: 10.1016/j.jechem.2024.02.035.

[13] Y. Chen, H. Liu, X. Fang, Y. Li, J. Chen, L. Peng, X. Liu, and J. Lin. "A machine learning workflow for large-scale discovery of direct bandgap double perovskites." In: *Solar Energy Materials and Solar Cells* 282 (2025). DOI: 10.1016/j.solmat.2025.113402.

[14] E. T. Chenebuah, M. Nganbe, and A. B. Tchagang. "A deep generative modeling architecture for designing lattice-constrained perovskite materials." In: *npj Computational Materials* 10.1 (2024). DOI: 10.1038/s41524-024-01381-9.

[15] J. Dagdelen, A. Dunn, S. Lee, N. Walker, A. S. Rosen, G. Ceder, K. A. Persson, and A. Jain. "Structured information extraction from scientific text with large language models." In: *Nature Communications* 15.1 (2024). DOI: 10.1038/s41467-024-45563-x.

[16] Z. Deng and L. Sarkisov. "Engineering Machine Learning Features to Predict Adsorption of Carbon Dioxide and Nitrogen in Metal-Organic Frameworks." In: *Journal of Physical Chemistry C* 128.24 (2024), pp. 10202–10215. DOI: 10.1021/acs.jpcc.4c01692.

[17] V. L. Deringer, M. A. Caro, and G. Csányi. "Machine Learning Interatomic Potentials as Emerging Tools for Materials Science." In: *Advanced Materials* 31.46 (2019). All Open Access, Green Open Access. DOI: 10.1002/adma.201902765.

[18] R. Du, R. Xin, H. Wang, W. Zhu, R. Li, and W. Liu. "Machine learning: An accelerator for the exploration and application of advanced metal-organic frameworks." In: *Chemical Engineering Journal* 490 (2024). DOI: 10.1016/j.cej.2024.151828.

[19] D. Fanelli, L. Bindi, L. Chicchi, C. Pereti, R. Sessoli, and S. Tommasini. "A short introduction to neural networks and their application to Earth and Materials Science." In: *Rendiconti Lincei* 35.4 (2024), pp. 881–892. DOI: 10.1007/s12210-024-01271-8.

[20] X. Fang, E. A. Murphy, P. A. Kohl, Y. Li, C. J. Hawker, C. M. Bates, and M. Gu. "Universal Phase Identification of Block Copolymers From Physics-Informed Machine Learning." In: *Journal of Polymer Science* 63.6 (2025), pp. 1433–1440. DOI: 10.1002/pol.20241063.

[21] H. Gashmard, H. Shakeripour, and M. Alaei. "Predicting superconducting transition temperature through advanced machine learning and innovative feature engineering." In: *Scientific Reports* 14.1 (2024). DOI: 10.1038/s41598-024-54440-y.

[22] C. Gong, J. Liu, S. Dai, H. Hao, and H. Liu. "Machine learning assisted prediction of the phonon cutoff frequency of ABO3 perovskite materials." In: *Computational Materials Science* 239 (2024). DOI: 10.1016/j.commatsci.2024.112943.

[23] S. Guo, X. Huang, Y. Situ, Q. Huang, K. Guan, J. Huang, W. Wang, X. Bai, Z. Liu, Y. Wu, and Z. Qiao. "Interpretable Machine-Learning and Big Data Mining to Predict Gas Diffusivity in Metal-Organic Frameworks." In: *Advanced Science* 10.21 (2023). DOI: 10.1002/advs.202301461.

[24] K. Hickey, J. Feinstein, G. Sivaraman, M. MacDonell, E. Yan, C. Matherson, S. Coia, J. Xu, and K. Picel. "Applying machine learning and quantum chemistry to predict the glass transition temperatures of polymers." In: *Computational Materials Science* 238 (2024). DOI: 10.1016/j.commatsci.2024.112933.

[25] S. Hossain, M. N. Uddin, K. Yan, M. M. Hossain, M. S. H. Golder, and M. A. Hoque. "Prediction of the mechanical performance of polyethylene fiber-based engineered cementitious composite (PE-ECC)." In: *Low-Carbon Materials and Green Construction* 2.1 (2024). DOI: 10.1007/s44242-024-00040-y.

[26] Y. Hu, Q. Wang, and H. Ma. "Machine-learning-assisted searching for thermally conductive polymers: A mini review." In: *Journal of Applied Physics* 135.12 (2024). DOI: 10.1063/5.0201613.

[27] R. Jacobs, J. Liu, H. Abernathy, and D. Morgan. "Machine Learning Design of Perovskite Catalytic Properties." In: *Advanced Energy Materials* 14.12 (2024). DOI: 10.1002/aenm.202303684.

[28] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. A. Persson. "Commentary: The materials project: A materials genome approach to accelerating materials innovation." In: *APL Materials* 1.1 (2013). DOI: 10.1063/1.4812323.

[29] A. Jain, C. D. Armstrong, V. R. Joseph, R. Ramprasad, and H. J. Qi. "Machine-Guided Discovery of Acrylate Photopolymer Compositions." In: *ACS Applied Materials and Interfaces* 16.14 (2024), pp. 17992–18000. DOI: 10.1021/acsami.4c00759.

[30] S. G. Jung, G. Jung, and J. M. Cole. "Machine-Learning Predictions of Critical Temperatures from Chemical Compositions of Superconductors." In: *Journal of Chemical Information and Modeling* 64.19 (2024), pp. 7349–7375. DOI: 10.1021/acs.jcim.4c01137.

[31] A. Kameni, D. Palessonga, Z. Semmoumy, and M. Bensetti. "Effective Electromagnetic Properties of Composite Material Computed From Neural Network Approach." In: *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields* 37.5 (2024). DOI: 10.1002/jnm.3303.

[32] Y. Kang and J. Kim. "ChatMOF: an artificial intelligence system for predicting and generating metal-organic frameworks using large language models." In: *Nature Communications* 15.1 (2024). DOI: 10.1038/s41467-024-48998-4.

[33] A. Kumar, H. C. Arora, and M. L. Nehdi. "XGBoost algorithm based estimation of near surface mounted FRP rod-to-concrete bond strength and failure mode." In: *Journal of Building Engineering* 90 (2024). DOI: 10.1016/j.jobe.2024.109418.

[34] S. Kumar, S. Dutta, R. Jaafreh, N. Singh, A. Sharan, K. Hamad, and D. H. Yoon. "Accelerated discovery of perovskite materials guided by machine learning techniques." In: *Materials Letters* 353 (2023). DOI: 10.1016/j.matlet.2023.135311.

[35] P.-H. Lai, E. D. Gomez, B. D. Vogt, and W. F. Reinhart. "Predicting the Strength of Composites with Computer Vision Using Small Experimental Datasets." In: *ACS Materials Letters* 7.4 (2025), pp. 1503–1511. DOI: 10.1021/acsmaterialslett.4c02424.

[36] R. Laref, F. Massuyeau, and R. Gautier. "Role of Hydrogen Bonding on the Design of New Hybrid Perovskites Unraveled by Machine Learning." In: *Small* 20.5 (2024). DOI: 10.1002/smll.202306481.

[37] Q. Li, N. Miklaucic, and J. Hu. "Out-of-Distribution Material Property Prediction Using Adversarial Learning." In: *Journal of Physical Chemistry C* 129.13 (2025), pp. 6372–6385. DOI: 10.1021/acs.jpcc.4c07481.

[38] Y. Liang, X. Wei, Y. Peng, X. Wang, and X. Niu. "A review on recent applications of machine learning in mechanical properties of composites." In: *Polymer Composites* 46.3 (2025), pp. 1939–1960. DOI: 10.1002/pc.29082.

[39] J. Lin, H. Zhang, M. Asadi, K. Zhao, L. Yang, Y. Fan, J. Zhu, Q. Liu, L. Sun, W. J. Xie, C. Duan, F. Mo, and J.-H. Dou. "Machine Learning-Driven Discovery and Structure-Activity Relationship Analysis of Conductive Metal-Organic Frameworks." In: *Chemistry of Materials* 36.11 (2024), pp. 5436–5445. DOI: 10.1021/acs.chemmater.4c00229.

[40] Y. Liu, Y. Dong, and H. Wu. "Comprehensive overview of machine learning applications in MOFs: from modeling processes to latest applications and design classifications." In: *Journal of Materials Chemistry A* 13.4 (2024), pp. 2403–2440. DOI: 10.1039/d4ta06740a.

[41] T. W. Loh, H. T. Nguyen, and K. T. Nguyen. "Prediction of temperature and structural properties of fibre-reinforced polymer laminates under simulated fire exposure using artificial neural networks." In: *Composites Part B: Engineering* 287 (2024). DOI: 10.1016/j.compositesb.2024.111858.

[42] A. B. López-Baldomero, M. Buzzelli, F. Moronta-Montero, M. Á. Martínez-Domingo, and E. M. Valero. "Ink classification in historical documents using hyperspectral imaging and machine learning methods." In: *Spectrochimica Acta - Part A: Molecular and Biomolecular Spectroscopy* 335 (2025). DOI: 10.1016/j.saa.2025.125916.

[43] M. Madani, V. Lacivita, Y. Shin, and A. Tarakanova. "Accelerating materials property prediction via a hybrid Transformer Graph framework that leverages four body interactions." In: *npj Computational Materials* 11.1 (2025). DOI: 10.1038/s41524-024-01472-7.

[44] S. K. Mahapatra and A. Satapathy. "Development of machine learning models for the prediction of erosion wear of hybrid composites." In: *Polymer Composites* 45.9 (2024), pp. 7950–7966. DOI: 10.1002/pc.28315.

[45] S. H. Mallah, C. Güleryüz, S. H. Sumrra, A. U. Hassan, H. Güleryüz, A. Mohyuddin, H. A. Kyhoiesh, S. Noreen, and A. Y. Elnaggar. "Benzothiophene semiconductor polymer design by machine learning with low exciton binding energy: A vast chemical space generation for new structures." In: *Materials Science in Semiconductor Processing* 190 (2025). DOI: 10.1016/j.mssp.2025.109331.

[46] A. Mirzaei, P. Haghi, and M. Shokrieh. "Prediction of fatigue life of laminated composites by integrating artificial neural network model and non-dominated sorting genetic algorithm." In: *International Journal of Fatigue* 188 (2024). DOI: 10.1016/j.ijfatigue.2024.108528.

[47] J. A. Mysona, P. F. Nealey, and J. J. de Pablo. "Machine Learning Models and Dimensionality Reduction for Prediction of Polymer Properties." In: *Macromolecules* 57.5 (2024), pp. 1988–1997. DOI: 10.1021/acs.macromol.3c02401.

[48] S. Nanjo, Arifin, H. Maeda, Y. Hayashi, K. Hatakeyama-Sato, R. Himeno, T. Hayakawa, and R. Yoshida. "SPACIER: on-demand polymer design with fully automated all-atom classical molecular dynamics integrated into machine learning pipelines." In: *npj Computational Materials* 11.1 (2025). DOI: 10.1038/s41524-024-01492-3.

[49] S. Y. Oh, D. Lee, and Y.-B. Park. "Impact damage characterization approach for CFRP pipes via self-sensing." In: *International Journal of Mechanical Sciences* 281 (2024). DOI: 10.1016/j.ijmecsci.2024.109511.

[50] N. Osa-uwagboe, A. G. Udu, M. K. Ghalati, V. V. Silberschmidt, A. Aremu, H. Dong, and E. Demirci. "A machine learning-enabled prediction of damage properties for fiber-reinforced polymer composites under out-of-plane loading." In: *Engineering Structures* 308 (2024). DOI: 10.1016/j.engstruct.2024.117970.

[51] S. M. Pai, K. A. Shah, S. Sunder, R. Q. Albuquerque, C. Brütting, and H. Ruckdäschel. "Machine learning applied to the design and optimization of polymeric materials: A review." In: *Next Materials* 7 (2025). DOI: 10.1016/j.nxmate.2024.100449.

[52] C. Pereti, K. Bernot, T. Guizouarn, F. Laufek, A. Vymazalová, L. Bindi, R. Sessoli, and D. Fanelli. "From individual elements to macroscopic materials: in search of new superconductors via machine learning." In: *npj Computational Materials* 9.1 (2023). DOI: 10.1038/s41524-023-01023-6.

[53] Y. K. Phua, N. Terasoba, M. Tanaka, T. Fujigaya, and K. Kato. "Unsupervised Machine Learning-Derived Anion-Exchange Membrane Polymers Map: A Guideline for Polymers Exploration and Design." In: *ChemElectroChem* 11.14 (2024). DOI: 10.1002/celc.202400252.

[54] G. Pindolia and S. M. Shinde. "Prediction of Efficiency for KSnI3 Perovskite Solar Cells Using Supervised Machine Learning Algorithms." In: *Journal of Electronic Materials* 53.6 (2024), pp. 3268–3275. DOI: 10.1007/s11664-024-10988-z.

[55] E. A. Pogue, A. New, K. McElroy, N. Q. Le, M. J. Pekala, I. McCue, E. Gienger, J. Domenico, E. Hedrick, T. M. McQueen, B. Wilfong, C. D. Piatko, C. R. Ratto, A. Lennon, C. Chung, T. Montalbano, G. Bassen, and C. D. Stiles. "Closed-loop superconducting materials discovery." In: *npj Computational Materials* 9.1 (2023). DOI: 10.1038/s41524-023-01131-3.

[56] J. P. Quesada-Molina, H. Mofatteh, A. Akbarzadeh, and S. Mariani. "Navigating beyond the training set: A deep learning framework for inverse design of architected composite materials." In: *Engineering Applications of Artificial Intelligence* 150 (2025). DOI: 10.1016/j.engappai.2025.110473.

[57] M. M. Rabby, P. P. Das, M. Rahman, V. Vadlamudi, and R. Raihan. "Fast and accurate prediction of cure quality and mechanical performance in fiber-reinforced polymer composite using dielectric variables and machine learning." In: *Polymer Composites* 45.2 (2024), pp. 1810–1825. DOI: 10.1002/pc.27891.

[58] Y. Ran, J. An, and L. Zhang. "Inference of Constitutive Relation of Phase-Separated Polymers by Integrating Physics-Informed Neural Networks and Symbolic Regression." In: *Macromolecular Chemistry and Physics* 225.20 (2024). DOI: 10.1002/macp.202400184.

[59] M. A. M. Rashid, S. Lee, K. H. Kim, J. Kim, and K. Jeong. "Machine Learning Approach for Predicting the Hole Mobility of the Perovskite Solar Cells." In: *Advanced Theory and Simulations* 7.6 (2024). DOI: 10.1002/adts.202300978.

[60] B. Roter, N. Ninkovic, and S. Dordevic. "Clustering superconductors using unsupervised machine learning." In: *Physica C: Superconductivity and its Applications* 598 (2022). DOI: 10.1016/j.physc.2022.1354078.

[61] A. E. Santo, W. Khor, and F. Ciampa. "Statistical and Machine Learning-Based Imaging with Long Pulse Thermography for the Detection of Non-standardised Defects in CFRP Composites." In: *Journal of Nondestructive Evaluation* 44.1 (2025). DOI: 10.1007/s10921-024-01138-w.

[62] A. P. Sarikas, G. S. Fanourgakis, K. Gkagkas, and G. E. Froudakis. "Comparison of machine learning approaches for the identification of top-performing materials for hydrogen storage." In: *Sustainable Chemistry for the Environment* 5 (2024). DOI: 10.1016/j.scenv.2023.100056.

[63] M. Scheidgen, L. Himanen, A. N. Ladines, D. Sikter, M. Nakhaee, Á. Fekete, T. Chang, A. Golparvar, J. A. Márquez, S. Brockhauser, S. Brückner, L. M. Ghiringhelli, F. Dietrich, D. Lehmberg, T. Denell, A. Albino, H. Näsström, S. Shabih, F. Dobener, M. Kühbach, R. Mozumder, J. F. Rudzinski, N. Daelman, J. M. Pizarro, M. Kuban, C. Salazar, P. Ondračka, H.-J. Bungartz, and C. Draxl. "NOMAD: A distributed web-based platform for managing materials science research data." In: *Journal of Open Source Software* 8.90 (2023), p. 5388. DOI: 10.21105/joss.05388.

[64] J. Schmidt, M. R. G. Marques, S. Botti, and M. A. L. Marques. "Recent advances and applications of machine learning in solid-state materials science." In: *npj Computational Materials* 5.1 (2019). DOI: 10.1038/s41524-019-0221-0.

[65] T. Schuett, P. Endres, T. Standau, S. Zechel, R. Q. Albuquerque, C. Brütting, H. Ruckdäschel, and U. S. Schubert. "Application of Digital Methods in Polymer Science and Engineering." In: *Advanced Functional Materials* 34.8 (2024). All Open Access, Hybrid Gold Open Access. DOI: 10.1002/adfm.202309844.

[66] C. C. Seegmiller, S. G. Baird, H. M. Sayeed, and T. D. Sparks. "Discovering chemically novel, high-temperature superconductors." In: *Computational Materials Science* 228 (2023). DOI: 10.1016/j.commatsci.2023.112358.

[67] M. Sharma Priyadarshini, N. K. Thota, and R. Hernandez. "ReLMM: Reinforcement Learning Optimizes Feature Selection in Modeling Materials." In: *Journal of Chemical Information and Modeling* 65.1 (2025), pp. 153–161. DOI: 10.1021/acs.jcim.4c01934.

[68] N. Shrivastav, A. Abu-Jrai, P. Kanjariya, H. Hassan, A. Verma, J. Madan, and R. Pandey. "Advanced Computational Techniques for Optimizing Manganese-Based Perovskite Solar Cells: From SCAPS-1D Simulations to Machine Learning Predictions." In: *Journal of Electronic Materials* 54.2 (2025), pp. 1209–1217. DOI: 10.1007/s11664-024-11638-0.

[69] T. H. B. da Silva, T. Cavignac, T. F. T. Cerqueira, H.-C. Wang, and M. A. L. Marques. "Machine-learning accelerated prediction of two-dimensional conventional superconductors." In: *Materials Horizons* (2025). DOI: 10.1039/d4mh01753f.

[70] S. Subba, P. Rai, and S. Chatterjee. "Machine Learning Approaches in Advancing Perovskite Solar Cells Research." In: *Advanced Theory and Simulations* 8.3 (2025). DOI: 10.1002/adts.202400652.

[71] P. Subudhi, S. Sivapatham, R. Narasimhan A, B. Kumar, and D. Punetha. "Enhancing photovoltaic performance in tin-based perovskite solar cells: A unified approach utilizing numerical simulation and machine learning techniques." In: *Journal of Power Sources* 639 (2025). DOI: 10.1016/j.jpowsour.2025.236639.

[72] Y. Sun, X. Wang, C. Hou, and J. Ni. "Interpretable Machine Learning to Discover Perovskites with High Spontaneous Polarization." In: *Journal of Physical Chemistry C* 127.49 (2023), pp. 23897–23905. DOI: 10.1021/acs.jpcc.3c05742.

[73] G. G. Terrones, S.-P. Huang, M. P. Rivera, S. Yue, A. Hernandez, and H. J. Kulik. "Metal-Organic Framework Stability in Water and Harsh Environments from Data-Driven Models Trained on the Diverse WS24 Data Set." In: *Journal of the American Chemical Society* 146.29 (2024), pp. 20333–20348. DOI: 10.1021/jacs.4c05879.

[74] L. Tian, L. Wang, and G. Xian. "Machine learning prediction of interfacial bond strength of FRP bars with different surface characteristics to concrete." In: *Case Studies in Construction Materials* 21 (2024). DOI: 10.1016/j.cscm.2024.e03984.

[75] S. Touati, A. Benghia, Z. Hebboul, I. K. Lefkaier, M. B. Kanoun, and S. Goumri-Said. "Predictive machine learning approaches for perovskites properties using their chemical formula: towards the discovery of stable solar cells materials." In: *Neural Computing and Applications* 36.26 (2024), pp. 16319–16329. DOI: 10.1007/s00521-024-09992-5.

[76] H. Tran, H.-C. Dam, C. Kuenneth, V. Ngoc Tuoc, and H. Kino. "Superconductor Discovery in the Emerging Paradigm of Materials Informatics." In: *Chemistry of Materials* 36.22 (2024), pp. 10939–10966. DOI: 10.1021/acs.chemmater.4c01757.

[77] H. Tran and T. N. Vu. "Machine-learning approach for discovery of conventional superconductors." In: *Physical Review Materials* 7.5 (2023). DOI: 10.1103/PhysRevMaterials.7.054805.

[78] V. Tunukovic, S. McKnight, R. Pyle, Z. Wang, E. Mohseni, S. Gareth Pierce, R. K. W. Vithanage, G. Dobie, C. N. MacLeod, S. Cochran, and T. O'Hare. "Unsupervised machine learning for flaw detection in automated ultrasonic testing of carbon fibre reinforced plastic composites." In: *Ultrasonics* 140 (2024). DOI: 10.1016/j.ultras.2024.107313.

[79] A. B. Vieira Wyzykowski, F. F. Niazi, and A. Dickson. "AGDIFF: Attention-Enhanced Diffusion for Molecular Geometry Prediction." In: *Journal of Chemical Information and Modeling* 65.4 (2025), pp. 1798–1811. DOI: 10.1021/acs.jcim.4c01896.

[80] C. Wang, Y. Wan, S. Yang, Y. Xie, S. Chu, Y. Chen, and X. Yan. "Revealing the Untapped Potential of Photocatalytic Overall Water Splitting in Metal Organic Frameworks." In: *Advanced Functional Materials* 34.13 (2024). DOI: 10.1002/adfm.202313596.

[81] H. Wang, R. Ouyang, W. Chen, and A. Pasquarello. "High-Quality Data Enabling Universality of Band Gap Descriptor and Discovery of Photovoltaic Perovskites." In: *Journal of the American Chemical Society* 146.26 (2024), pp. 17636–17645. DOI: 10.1021/jacs.4c03507.

[82] J. Wang, D. Fan, and C. Cai. "Application and feasibility analysis of knowledge-based machine learning in predicting fatigue performance of stainless steel." In: *Case Studies in Construction Materials* 22 (2025). DOI: 10.1016/j.cscm.2024.e04090.

[83] J. Wang, Q. Lou, Z. Xu, Y. Jin, G. Luo, and H. Zhou. "Accelerating ionic liquid research in perovskite solar cells through machine learning:Opportunities and challenges." In: *Materials Today Electronics* 12 (2025). DOI: 10.1016/j.mtelec.2025.100143.

[84] J. Wei, X. Chu, X.-Y. Sun, K. Xu, H.-X. Deng, J. Chen, Z. Wei, and M. Lei. "Machine learning in materials science." In: *InfoMat* 1.3 (2019). All Open Access, Gold Open Access, pp. 338–358. DOI: 10.1002/inf2.12028.

[85] R. Wu, Y. Zhang, Z. Peng, D. Song, and H. Li. "Predicting multiple fatigue properties of twinning-induced plasticity steels by black-box and white-box machine learning." In: *Mechanics of Materials* 205 (2025). DOI: 10.1016/j.mechmat.2025.105307.

[86] H. Xu, X. Yu, J. Liu, and X. Gao. "Scaling law-informed machine learning for predicting thermal and electrical properties of polymers: A physics-based approach." In: *Computational Materials Science* 253 (2025). DOI: 10.1016/j.commatsci.2025.113887.

[87] B. Yu, L. Zhang, X. Ye, J. Wu, H. Ying, W. Zhu, Z. Yu, and X. Wu. "State-of-the-art review on various applications of machine learning techniques in materials science and engineering." In: *Chemical Engineering Science* 306 (2025), p. 121147. ISSN: 0009-2509. DOI: https://doi.org/10.1016/j.ces.2024.121147.

[88] S. Yuan and S. Dordevic. "Diffusion models for conditional generation of hypothetical new families of superconductors." In: *Scientific Reports* 14.1 (2024). DOI: 10.1038/s41598-024-61040-3.

[89] Y. Yue, S. A. Mohamed, and J. Jiang. "Classifying and Predicting the Thermal Expansion Properties of Metal-Organic Frameworks: A Data-Driven Approach." In: *Journal of Chemical Information and Modeling* 64.13 (2024), pp. 4966–4979. DOI: 10.1021/acs.jcim.4c00057.

[90] C. Zeni, R. Pinsler, D. Zügner, A. Fowler, M. Horton, X. Fu, Z. Wang, A. Shysheya, J. Crabbé, S. Ueda, R. Sordillo, L. Sun, J. Smith, B. Nguyen, H. Schulz, S. Lewis, C.-W. Huang, Z. Lu, Y. Zhou, H. Yang, H. Hao, J. Li, C. Yang, W. Li, R. Tomioka, and T. Xie. "A generative model for inorganic materials design." In: *Nature* 639.8055 (2025), pp. 624–632. DOI: 10.1038/s41586-025-08628-5.

[91] J. Zhang, K. Zhang, S. Xu, Y. Li, C. Zhong, M. Zhao, H.-J. Qiu, M. Qin, X.-D. Xiang, K. Hu, and X. Lin. "An integrated machine learning model for accurate and robust prediction of superconducting critical temperature." In: *Journal of Energy Chemistry* 78 (2023), pp. 232–239. DOI: 10.1016/j.jechem.2022.11.047.

[92] J. Zhang, J. Wu, V. M. Le Corre, J. A. Hauch, Y. Zhao, and C. J. Brabec. "Advancing perovskite photovoltaic technology through machine learning-driven automation." In: *InfoMat* (2025). DOI: 10.1002/inf2.70005.

[93] L. Zhang, H. Zhang, B. Ji, L. Liu, X. Liu, and D. Chen. "Application of Machine Learning in Amorphous Alloys." In: *Materials* 18.8 (2025). DOI: 10.3390/ma18081771.

[94] T. Zhang. *An introduction to materials informatics: The elements of machine learning*. Springer, 2025, pp. 1–479. DOI: 10.1007/978-981-99-7992-9.

[95] Z. Zhang, S. Liu, Q. Xiong, and Y. Liu. "Strategic Integration of Machine Learning in the Design of Excellent Hybrid Perovskite Solar Cells." In: *Journal of Physical Chemistry Letters* 16.3 (2025), pp. 738–746. DOI: 10.1021/acs.jpclett.4c03580.

[96] Z. Zhang, F. Pan, S. A. Mohamed, C. Ji, K. Zhang, J. Jiang, and Z. Jiang. "Accelerating Discovery of Water Stable Metal-Organic Frameworks by Machine Learning." In: *Small* 20.42 (2024). DOI: 10.1002/smll.202405087.

[97] Z. Zhang, S. Wang, C. Chen, M. Sun, Z. Wang, Y. Cai, Y. Tuo, Y. Du, Z. Han, X. Yun, X. Guan, S. Shi, J. Xie, G. Liu, and P. Lu. "Design of photovoltaic materials assisted by machine learning and the mechanical tunability under micro-strain." In: *Journal of Materials Science and Technology* 227 (2025), pp. 108–121. DOI: 10.1016/j.jmst.2024.11.055.

[98] G. Zhao and Y. G. Chung. "PACMAN: A Robust Partial Atomic Charge Predicter for Nanoporous Materials Based on Crystal Graph Convolution Networks." In: *Journal of Chemical Theory and Computation* 20.12 (2024), pp. 5368–5380. DOI: 10.1021/acs.jctc.4c00434.

[99] W. Zhao, Z. Pang, C. Wang, W. He, X. Liang, J. Song, Z. Cao, S. Hu, M. Lang, and S. Luo. "Hybrid ANN-physical model for predicting residual stress and microhardness of metallic materials after laser shock peening." In: *Optics and Laser Technology* 181 (2025). DOI: 10.1016/j.optlastec.2024.111750.

[100] G. Zheng, S. Zhang, L. Meng, S. Zhang, and X. Wang. "Machine Learning-Guided Design and Synthesis of Eco-Friendly Poly(ethylene oxide) Membranes for High-Efficacy CO2/N2 Separation." In: *Advanced Functional Materials* 34.51 (2024). DOI: 10.1002/adfm.202410075.

[101] C. Zhong, Y. Wang, Y. Long, J. Liu, K. Hu, J. Zhang, J. Chen, and X. Lin. "Enhancing Superconductor Critical Temperature Prediction: A Novel Machine Learning Approach Integrating Dopant Recognition." In: *ACS Applied Materials and Interfaces* 16.44 (2024), pp. 60472–60481. DOI: 10.1021/acsami.4c11997.

[102] C. Zhong, J. Zhang, Y. Wang, Y. Long, P. Zhu, J. Liu, K. Hu, J. Chen, and X. Lin. "High-performance diffusion model for inverse design of high Tc superconductors with effective doping and accurate stoichiometry." In: *InfoMat* 6.5 (2024). DOI: 10.1002/inf2.12519.