

DiffractGPT: Atomic Structure Determination from X-ray Diffraction Patterns Using a Generative Pretrained Transformer

Kamal Choudhary*



Cite This: *J. Phys. Chem. Lett.* 2025, 16, 2110–2119



Read Online

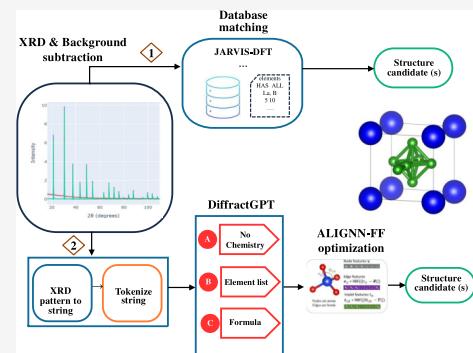
ACCESS |

Metrics & More

Article Recommendations

Supporting Information

ABSTRACT: Crystal structure determination from powder diffraction patterns is a complex challenge in materials science, often requiring extensive expertise and computational resources. This study introduces DiffractGPT, a generative pretrained transformer model designed to predict atomic structures directly from X-ray diffraction (XRD) patterns. By capturing the intricate relationships between diffraction patterns and crystal structures, DiffractGPT enables fast and accurate inverse design. Trained on thousands of atomic structures and their simulated XRD patterns from the JARVIS-DFT data set, we evaluate the model across three scenarios: (1) without chemical information, (2) with a list of elements, and (3) with an explicit chemical formula. The results demonstrate that incorporating chemical information significantly enhances prediction accuracy. Additionally, the training process is straightforward and fast, bridging gaps between computational, data science, and experimental communities. This work represents a significant advancement in automating crystal structure determination, offering a robust tool for data-driven materials discovery and design.



Since the discovery of X-rays in 1895, they have been widely used in medical imaging, crystallography, and astronomy.¹ Numerous experimental techniques in materials science rely on X-rays, including X-ray diffraction (XRD), X-ray fluorescence (XRF), X-ray photoelectron spectroscopy (XPS), small-angle X-ray scattering (SAXS), X-ray tomography (XRT), X-ray reflectometry (XRR), grazing incidence X-ray diffraction (GIXRD), and resonant inelastic X-ray scattering (RIXS).^{2,3} Among these, XRD plays a crucial role in determining atomic structures and uncovering the mechanisms underlying mechanical strength, electronic properties, optical behavior, and chemical reactivity.^{4,5} However, crystal structure determination currently involves extensive trial and error as well as expert knowledge. The main challenge lies in the reduction of chemical and three-dimensional structural information into one-dimensional diffraction patterns, which causes the loss of phase information and complicates structure determination.

Additionally, the presence of peaks in the diffraction data of newly discovered compounds, complex materials, or multiphase systems further exacerbates this challenge. Over the past few decades, Rietveld refinement, simulated annealing, and evolutionary algorithms have been developed to address this problem by iteratively fitting data to potential candidate structures.^{2,3} Several widely used software tools, such as FullProf,⁶ the General Structure Analysis System (GSAS),⁷ GenX,⁸ TOTal Pattern Analysis Solutions (TOPAS),⁹ and Materials Analysis Using Diffraction (MAUD),¹⁰ are available for this purpose. While these methods have been successful, they often require significant domain expertise, computational resources, and

manual intervention, particularly when dealing with ambiguous or incomplete data.

In recent years, machine learning has emerged as a powerful tool in materials science, offering the potential to accelerate materials discovery and characterization.^{11–13} In particular, high-throughput materials design and process modeling, which are key driving forces behind the Materials Genome Initiative and the Creating Helpful Incentives to Produce Semiconductors (CHIPS) Act,¹⁴ require a bridge between experiments and multiscale modeling components, where large language models (LLMs) could play a significant role. Moreover, two recent Nobel Prizes in Physics and Chemistry in 2024 for neural networks and AlphaFold clearly demonstrate the wide applicability of AI/ML in scientific research.

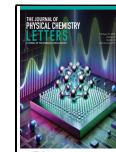
The AI/ML techniques have been successfully used for both forward (structure to property) and inverse (property to structure) tasks in materials design.¹¹ Generating crystal structures from XRD can be considered a generative AI-based inverse design task. Recent advancements in machine learning related to X-ray diffraction¹⁵ include works by Park et al.,¹⁶ NeuralXRD,¹⁷ XRD_is_All_You_Need,¹⁸ Crystallography Companion Agent (XCA),¹⁹ ARiXD-ML,²⁰ Zaloga et al.,²¹

Received: October 30, 2024

Revised: January 20, 2025

Accepted: January 21, 2025

Published: February 20, 2025



XTEC,²² Li et al.,²³ Maffettone et al.,²⁴ Oviedo et al.²⁵ and several others.^{26–28} These works demonstrate the application of ML models for a wide range of tasks, including crystal lattice and space group classification, peak detection, and structure generation. In particular, the application of deep generative models such as Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs) has demonstrated the ability to generate complex atomic structures based on insights.

The potential of GPT in natural language processing (NLP), such as ChatGPT, has spurred interest in their applications beyond textual data, particularly in domains such as chemistry and materials science. The success of AtomGPT (Atomistic Generative Pretrained Transformer),²⁹ which demonstrated the capability to generate atomic structures and predict material properties using transformer-based architectures, highlights the power of transformer models in handling materials data. AtomGPT establishes the relationship between atomic configurations as text and material properties, allowing it to tackle both forward and inverse design problems.

The GPT is a type of LLM originally developed for natural language processing and has demonstrated remarkable success in generating coherent and contextually relevant text.^{30–32} Models such as ChatGPT³³ have been used for code generation, debugging, literature reviews, and numerous other tasks. However, if we attempt to perform forward/inverse materials design tasks, the outcomes can be quite poor.^{34–36} Nevertheless, inspired by its simplicity of use and the massive success of ChatGPT, an alternate model, AtomGPT, was introduced, tailored for forward and inverse materials design.

While AtomGPT enables scalar material properties to be generated from atomic structures, its application for generating atomic structures from experimental properties, such as XRD, has not yet been explored. Based on these developments, we introduce DiffractGPT (DGPT), a specialized generative model designed to directly predict crystal structures from powder X-ray diffraction (PXRD) patterns. DiffractGPT leverages the powerful architecture of AtomGPT, adapting it to the unique challenges of PXRD-based crystal structure determination. By training on large data sets such as JARVIS-DFT (JDFT), which comprises simulated PXRD patterns alongside their corresponding atomic structures, DiffractGPT learns to map complex diffraction data to accurate crystal structures. This approach enables the direct prediction of atomic arrangements from diffraction data, significantly reducing the need for iterative fitting and manual intervention. We further evaluate various application scenarios for DiffractGPT, such as XRD with no known chemical constituents, with guessed elements, and with explicit chemical formulas for structure design tasks. We also provide a web framework and tools to match the XRD patterns with existing data, as well as to generate new structures using the generative models. Most importantly, although we apply the models to XRD data, they can also be useful for other experiments, such as neutron and electron diffraction and other spectroscopic experiments.

The Joint Automated Repository for Various Integrated Simulations (JARVIS) - density functional theory (DFT)^{37,38} database used in this work contains nearly 80,000 bulk 3D materials and 1,100 2D materials. The JARVIS-DFT project originated about six years ago and has amassed millions of material properties, along with carefully converged atomic structures using tight convergence parameters and various exchange-correlation functionals. JARVIS-DFT encompasses a

wide range of material classes, including metallic, semiconducting, insulating, superconducting, high-strength, topological, solar, thermoelectric, piezoelectric, dielectric, two-dimensional, magnetic, porous, defect, and various other types of bulk materials.

In this paper, we describe the architecture and training methodology of DiffractGPT and evaluate its performance on the PXRD data set. DiffractGPT uses transformer architecture based on the Mistral AI model³⁹ but can be easily adapted to other LLMs as well. We demonstrate that DiffractGPT not only matches the accuracy of traditional methods but also significantly reduces the computational time and expertise required for crystal structure determination. AtomGPT and DiffractGPT are analogous to AlphaFold (mentioned above) in their approach to solving complex structure–property relationships using machine learning. They adapt generative predictive frameworks to tackle fundamental challenges in materials science, mirroring what AlphaFold⁴⁰ has achieved for biology. The results show the promise of using generative machine learning models for automating the crystal structure determination process, opening up new avenues for materials discovery and design. The code used in this study will be made available on the AtomGPT GitHub page: <https://github.com/usnistgov/atomgpt>.

The data set used for this work is taken from the JARVIS-DFT database, which includes nearly 80,000 atomic structures and several material properties derived from density functional theory and powder X-ray diffraction patterns.^{37,38,41} From an atomic structure and a given X-ray wavelength (here Cu K α), the corresponding PXRD patterns can be easily calculated. The PXRD pattern was computed from the atomic structure by first calculating the reciprocal lattice vectors and interplanar spacings d_{hkl} for each set of Miller indices (hkl). Bragg's law, $n\lambda = 2d_{hkl} \sin \theta$, was used to convert these d -spacings into scattering angles 2θ . The structure factor $F(hkl)$ for each reflection was then calculated as the sum of atomic scattering contributions from all atoms in the unit cell, taking into account their positions and associated phase shifts. The atomic scattering factor $f(\theta)$, which varies with the scattering angle, was used to model the electron density distribution around each atom accurately. The diffraction intensity for each reflection was obtained using the relation $I(hkl) \propto |F(hkl)|^2$. A Gaussian broadening function was also applied to account for experimental resolution effects. The final XRD pattern was generated by summing the corrected intensities over all relevant reflections. All calculations were performed using custom scripts in the JARVIS-tools package to simulate the diffraction patterns for comparison with experimental data.

Such XRD predictions were carried out for all the data in the JARVIS-DFT (JDFT) data set. The XRD data set was split into a 90:10 ratio for training and testing the DiffractGPT models. This requires fine-tuning LLM models such as Mistral AI,³⁹ which are based on transformer architecture. Each transformer block contains two main components: a multihead self-attention mechanism and a position-wise feed-forward network. The input to the model is a sequence of tokens, which are first converted into embeddings and then passed through the transformer blocks. The scaled dot-product attention used in a transformer model can be written as

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

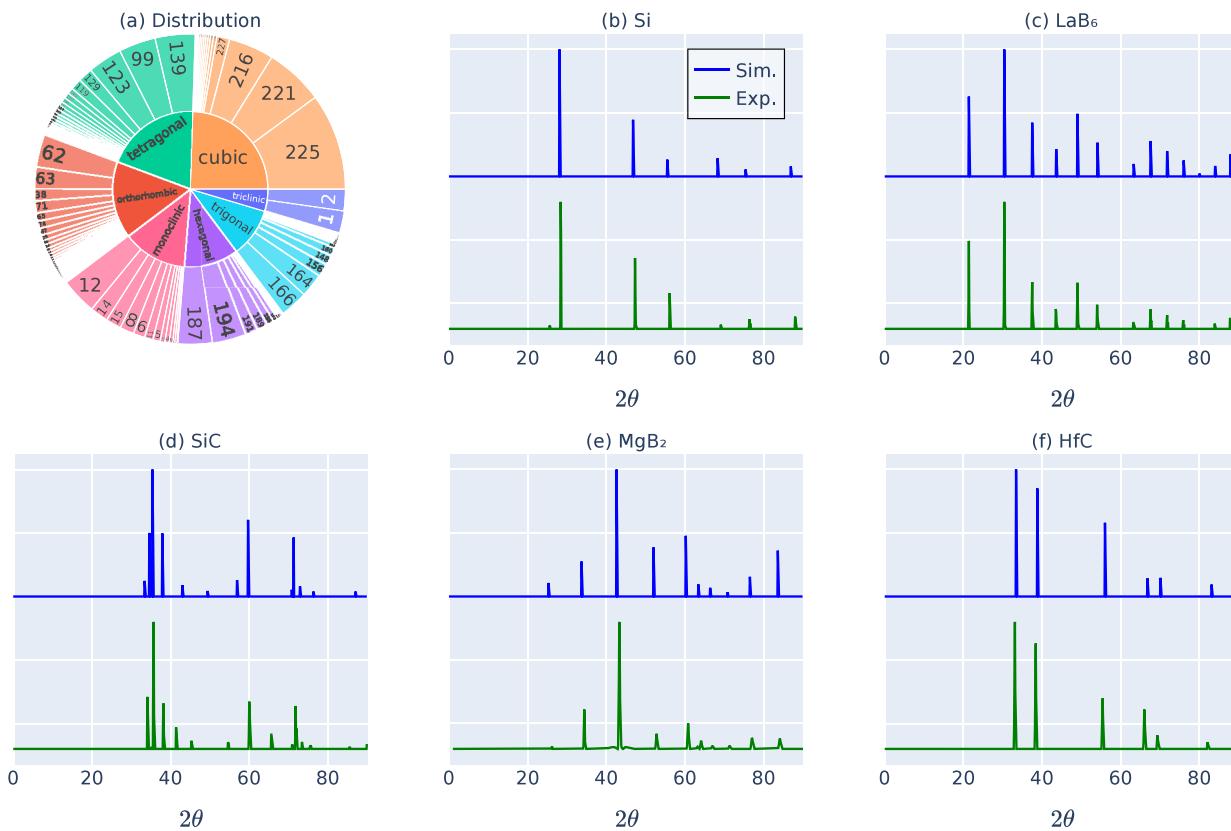


Figure 1. Crystal lattice and spacegroup data-distribution in the JARVIS-DFT (JDFT) database and comparison of a few simulated XRD-patterns with experimental measurements. (a) Crystal lattice and spacegroup distribution in the JDFT atomic structure database. (b) Simulated and experimental PXRD for silicon. The experimental data was taken from RRUFF database with ID R050145 while the simulated data from JDFT ID JVASP-1002. (c) Simulated and experimental PXRD for lanthanum boride. The experimental data was obtained as a part of this work while the simulated data from JDFT with ID of 15014. (d) Simulated and experimental PXRD for silicon carbide (Moissanite). The experimental data was taken from RRUFF database with ID R061083 while the simulated data from JDFT ID JVASP-107. (e) Simulated and experimental PXRD for magnesium boride. The experimental data was obtained as a part of this work while the simulated data from JDFT ID JVASP-1151. (f) Simulated and experimental PXRD for hafnium carbide. The experimental data was obtained as a part of this work while the simulated data from JDFT ID JVASP-17957.

where Q , K , and V represent the query, key, and value matrices, respectively. Here, d_k is the dimensionality of the key vectors. The multihead attention is obtained by concatenating multiple such attention heads. The multihead self-attention mechanism allows the model to focus on different parts of the input sequence when computing the output for a particular token.

There are thousands of LLMs, especially transformer models, that are publicly available. In particular, we use the Mistral AI 7 billion parameter model,³⁹ which employs Low-Rank Adaptation (LoRA) for parameter-efficient fine-tuning (PEFT)⁴² adopted from the UnisloAI package.⁴³ Mistral is a powerful model with 7.3 billion parameters and has been shown to outperform the Large Language Model Meta AI (LLaMA) 2 13B,⁴⁴ LLaMA 1 34B,⁴⁵ and ChatGPT³³ on several publicly available benchmarks. The Mistral 7B model combines efficiency and performance within a 7 billion parameter architecture. It introduces several key innovations, including Grouped-Query Attention for reduced computational complexity, Sliding Window Attention for processing longer sequences, and Rotary Positional Embeddings (RoPE) for improved position encoding. The model features 32 layers, a hidden size of 4096, and 32 attention heads. It employs prenormalization, Swish-Gated Linear Unit (SwiGLU) activation in feed-forward layers, and various training optimizations. This model was also successfully used in the previous AtomGPT work.²⁹

Now, fine-tuning requires transforming the instructions into a specialized protocol such as Alpaca.⁴⁶ The Alpaca instructions consist of Python dictionaries with keys for instruction, input, and output texts. The instruction key was set to “Below is a description of a material.” The XRD patterns were interpolated on a grid of 180 points, with intervals of 0.5° 2θ , using three floating-point precision, and then converted to a string with a newline character as separators. A fixed pattern length allows for uniform token lengths for LLMs, irrespective of different simulation and experimental settings for PXRD data. Note that with decreasing intervals (here 0.5), the number of tokens increases, and hence, the training and inference time will be higher. The input key used was of three types: (1) with no chemical information, (2) with elemental lists only, and (3) with an explicit chemical formula. For the input with no chemical information, the input key was simply “The XRD is ... Generate atomic structure description with lattice lengths, angles, coordinates, and atom types.” Similarly, for the second and third cases, the inputs were “The chemical elements are ... The XRD is ... Generate atomic structure description with lattice lengths, angles, coordinates, and atom types.” and “The chemical formula is ... The XRD is ... Generate atomic structure description with lattice lengths, angles, coordinates, and atom types,” respectively. Finally, the output key was a string of lattice lengths, angles, and chemical coordinates along with three fractional coordinates in XYZ format. Two decimal precision

was used for lattice parameters and three decimal precision for coordinates.

As directly fine-tuning such an LLM can be computationally expensive, the PEFT method was used within the Hugging Face ecosystem. Additionally, Transformer Reinforcement Learning (TRL) and RoPE⁴⁷ were employed to patch the Mistral model with fast LoRA⁴² weights for reduced memory training. After obtaining the PEFT model, corresponding tokenizer, and Alpaca data set, supervised fine-tuning tasks were carried out with a batch size of 5, using the AdamW 8-bit optimizer and a cross-entropy loss function for 5 epochs. This loss function measures the difference between the predicted probability distribution over the vocabulary and the true distribution (i.e., the one-hot encoded target words). After the model is trained, it is evaluated on the test set with respect to reconstruction/test performance. To further clarify, after training the model on the training set, while keeping the instruction and input keys in the test set, the trained model is employed to generate outputs. After parsing the outputs to create corresponding crystal structures, the StructureMatcher algorithm⁴⁸ is used to find the best match between two structures, considering all invariances of materials. The root-mean-square error (RMS) is averaged over all matched materials. Because the interatomic distances can vary significantly for different materials, the RMS is normalized following the work in ref..⁴⁹ Note that this is just one of the metrics for generative models for atomic structures, and there can be numerous other types of metrics.

In addition to developing GPT models, convolutional neural networks (CNN) and gradient boosting regression tree (GBR) models were developed to predict lattice lengths given XRD patterns, with the same train-test split as for GPT models. For the GBR and CNN models, the XRD signals are used as inputs and the three lattice constants as outputs. For GBR, we used 1000 estimators, a learning rate of 0.01, and a maximum depth of 3 with a mean absolute error loss function. The CNN model used in this study, referred to as CNNRegressor, is designed to perform regression tasks by extracting features from one-dimensional input data. The architecture begins with two 1D convolutional layers: the first layer has 16 filters and the second layer has 32 filters, both with a kernel size of 3 and padding of 1 to preserve the input size. Each convolutional layer is followed by a Rectified Linear Unit (ReLU) activation function to introduce nonlinearity. MaxPooling layers with a kernel size of 2 and stride of 2 are applied to downsample the feature maps, reducing dimensionality and computational load. After these operations, the output is flattened to a shape of 32×45 , which feeds into a fully connected layer with 64 neurons. The final output layer contains 3 neurons, corresponding to the three target values predicted by the model. This architecture allows the network to efficiently learn relevant features from the input data for accurate regression. The CNN model was trained for 50 epochs with a batch size of 32.

Finally, XRD measurements were also performed for this work to validate the simulated XRD patterns. The crystal structures were characterized using spatially resolved powder X-ray diffraction with a Bruker D8 Discover. We explored Bragg angles ranging from $10^\circ 2\theta$ to $90^\circ 2\theta$ using Cu K α radiation (wavelength 1.54184 Å) at 50 kV, with a step size of 0.02° and a scan rate of 6° per minute.

In Figure 1, we show the crystal lattice and space group data distribution in the JDFT database and a comparison of several simulated XRD patterns with experimental measurements. In Figure 1a, we observe that most of the crystals are cubic, while

the least number belongs to the triclinic lattice out of the seven crystal systems. Similarly, out of 230 space groups, 225, which belong to the cubic lattice system, is prevalent. Such analysis provides a basic understanding of the predictive limits of the models. For instance, if the model is trained with a sufficiently large cubic data set but not with a triclinic data set, it might generalize well for cubic systems but not for triclinic ones.

There are various proprietary databases that contain PXRD and atomic structure information. However, in this work, we choose to use the publicly available JARVIS-DFT data set for proof of concept. Note that although a simulated PXRD database is used here, it can be easily extended to include experimental data in the future. Analyzing the accuracy of the simulated PXRD compared to experimental results is important. In Figure 1b–f, we present a few such comparisons. The experimental data was either obtained from RRUFF database or as part of the experimental component of this work.

The simulated and experimental PXRD for silicon, which is undoubtedly the most important material, especially for the semiconductor industry, is shown in Figure 1b. The experimental data was taken from the RRUFF database with ID R050145, while the simulated data is from JDFT with ID JVASP-1002. All the simulation and experimental data were rescaled between 0 and 1 based on the maximum height available in that pattern for uniform comparison. We can observe close agreement between the simulated (Sim.) and experimental (Exp.) patterns, suggesting high fidelity of the simulated data. We note that the relative peak heights may not be exactly identical for all the peaks, which can be attributed to the collection of crystal planes encountered during PXRD experiments.

Similarly, the simulated and experimental PXRD for lanthanum boride, considered an important reference material for XRD, is shown in Figure 1c. The experimental data was obtained as part of this work, while the simulated data is from JDFT with ID JVASP-15014. Here, we observe excellent agreement in peak positions and peak height values, especially up to $60^\circ 2\theta$ values, after which peak heights begin to differ. The simulated and experimental PXRD for silicon carbide (Moissanite) is shown in Figure 1d. The experimental data was taken from the RRUFF database with ID R061083, while the simulated data is from JDFT with ID JVASP-107. Here, we see more peaks in the simulation around $30^\circ 2\theta$, which can also be attributed to the reasons mentioned above regarding crystal planes encountered during experiments. PXRD should measure an aggregate of all present crystal planes that diffract X-ray that fulfill the Braggs criterion. However, in experiments, it is possible to miss some of the plane orientations in the powder sample. Finally, the simulated and experimental PXRDs for magnesium boride and hafnium carbide are shown in Figure 1e,f. In the case of magnesium boride, we are missing a peak around the $20^\circ 2\theta$ value, as well as peaks after $60^\circ 2\theta$. We observe excellent agreement in the hafnium carbide case, especially up to $60^\circ 2\theta$ values, after which the experimental data shows fewer peaks than the simulated data. After generating such PXRD patterns for all the materials in JDFT, we perform LLM training following the details mentioned above, and the resultant models can be used for fast prediction of crystal structures.

As the first evaluation of the model's performance, the lattice constants in the x, y, and z crystallographic directions are compared for crystals in the test set and those generated using the DGPT models. This test set was never exposed to the model during training. The lattice constants from XRD can also be

predicted using other ML techniques such as gradient boosting regression tree (GBR), convolutional neural networks (CNN), and various DiffractGPT (DGPT) models, as shown in Table 1.

Table 1. Performance Measurement in Terms of Mean Absolute Error (MAE) for Predicting Lattice Constants (\AA) Using Gradient Boosting Regression (GBR), Convolutional Neural Network (CNN), and Varieties of DiffractGPT (DGPT) Models^a

Prop/MAE	GBR	CNN	DGPT-no formula	DGPT-element list	DGPT-formula
a	1.03	0.28	0.25	0.18	0.17
b	0.99	0.27	0.26	0.20	0.18
c	1.27	0.28	0.38	0.28	0.27
RMS-d	-	-	0.23	0.21	0.07

^aWe also compare root mean square distance in predicted vs target structures using DGPT models.

The mean absolute errors (MAE) for predicting a, b, and c lattice constants on the test set for GBR are 1.03 \AA , 0.99 \AA , and 1.27 \AA . Similarly, for CNN models, MAEs of 0.28 \AA , 0.27 \AA , and 0.28 \AA are observed, which is a significant improvement compared to GBR. Now, the performance of three types of DiffractGPT models—those with chemical information, with element lists, and with explicit formulas—shows the minimum error for the model with explicit formulas, which is intuitively correct. Specifically, the lowest error in lattice constant predictions was observed for the a-lattice parameter at 0.17 \AA . This value is close to the CNN model predictions. Li et al. performed a similar task for predicting lattice constants and found a mean absolute error (MAE) of 0.48 \AA ⁵⁰ and an R^2 of 0.80. Although the data sets for these two works are different, a

MAE of 0.17 \AA suggests promising results. As larger databases are used for DiffractGPT in the future, the MAE may further decrease. Note that DiffractGPT provides not only lattice constants but also full atomic structure information, such as chemical elements and coordinates. Hence, as a second evaluation, we compare the root-mean-square distance (RMS-d) between the predicted and target materials in the test set and find that the lowest error is observed for the DGPT model with explicit formulas. The RMS-d of 0.07 \AA is comparable to the AtomGPT value of 0.08 \AA for the superconductor design task.²⁹

To illustrate further, we show the predicted lattice constants and volumes for the DiffractGPT chemical formula + XRD pattern model in Figure 2. The color of the dots in the plot represents different crystal lattice types. The cubic, tetragonal, orthorhombic, hexagonal, trigonal, monoclinic, and triclinic systems are represented by blue, green, red, cyan, magenta, purple, and black colors, respectively. The values that lie on the $x = y$ line represent perfect agreement, while points away from it represent outliers. We barely observe outliers from symmetric lattice systems such as cubic materials. Most of the outliers are from the red and purple dots, representing orthorhombic and monoclinic systems. We find the maximum R^2 score of 0.85 (for lattice constant b) and the minimum R^2 of 0.78 for lattice constant a.

Now, we present an overview of the usability of the DiffractGPT framework in Figure 3. DiffractGPT can be used to predict the complete crystal structure given a PXRD pattern. A user provides a PXRD pattern as input. These patterns contain background noise, which can be automatically detected and subtracted using scripts available in JARVIS-Tools. As a first option, the spectrum can be matched with structures from atomic structure databases, such as those in JDFT or similar databases, based on simulated XRD patterns using cosine

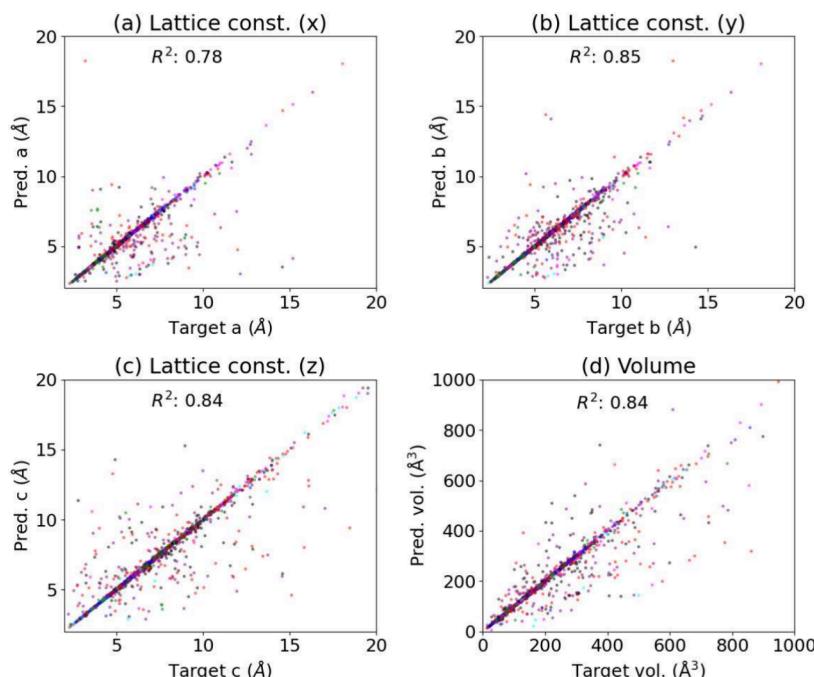


Figure 2. Performance of DiffractGPT chemical formula+XRD pattern to atomic structure model for lattice constants in (a) x-crystallographic direction, (b) y-crystallographic direction, (c) z-crystallographic direction, (d) volume. The color of the dots in the plot represents different crystal lattice types. The cubic, tetragonal, orthorhombic, hexagonal, trigonal, monoclinic, and triclinic systems are represented by blue, green, red, cyan, magenta, purple, and black colors, respectively. The values that lie on the $x = y$ line represent perfect agreement, while points away from it represent outliers.

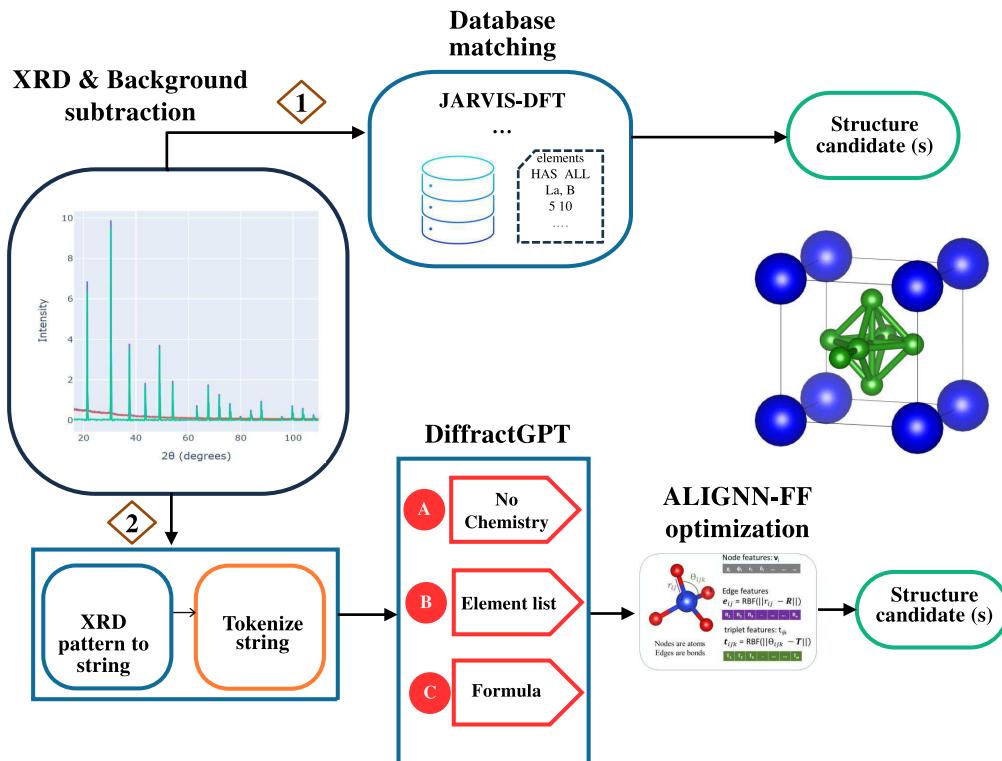


Figure 3. Schematic overview of crystal structure determination from XRD patterns using the DiffractGPT workflow. It begins with the user providing an XRD pattern as input. Utilizing the scripts available in JARVIS-Tools, background subtraction is automatically performed. First, the spectrum can be matched with structures from atomic structure databases, such as those in JDFT or similar databases, based on simulated XRD patterns using cosine similarity or other metrics. Alternatively, there are multiple scenarios where the user might (1) not know the constituent elements at all, (2) have some idea about the involved elements, or (3) explicitly know the chemical formula. Based on the provided information, the XRD pattern can be converted to strings followed by tokenization, after which one or more pretrained DiffractGPT models can be applied to generate potential crystal structures. Subsequently, further optimization can be performed using a unified GNN force field, such as ALIGNN-FF, to generate additional structure candidates. A tentative application for this workflow is available at the Web site <https://jarvis.nist.gov/jxrd>.

similarity or other metrics. A web application for this option is available at the JARVIS-XRD Web site (<https://jarvis.nist.gov/jxrd>). This process can predict the top candidates for the input XRD pattern. However, if the XRD patterns are complex or if the material does not exist in the current databases, the second option can be employed as follows. There are multiple scenarios: the user might (1) not know the constituent chemical elements at all, (2) have some idea about the involved elements, or (3) explicitly know the chemical formula. We have independent DiffractGPT models for all these scenarios. Based on the provided information, we can convert the XRD pattern to strings followed by tokenization, after which one or more pretrained DiffractGPT models can be applied to generate potential crystal structures. Note that transformer architectures allow for fast sampling, which can also be used to generate multiple options for the crystal structure if necessary.

As an optional subsequent step, further optimization of the generated structures can be performed using a unified graph neural network (GNN) force field (FF), such as the atomistic line graph neural network (ALIGNN)-FF,⁵¹ to generate additional structure candidates. It was developed for fast crystal structure optimization and to handle chemically and structurally diverse crystalline systems, with the entirety of the JARVIS-DFT data set used for training. This data set contains 4 million energy-force entries for 89 elements of the periodic table, of which 307,113 entries were utilized for training.⁵¹ ALIGNN-FF is seamlessly integrated into the DiffractGPT framework.

In Figure 4, we evaluate the performance of the DiffractGPT (DGPT)-formula model with and without ALIGNN-FF (AFF) optimization for a few selected materials. In these examples, the input chemical formula and X-ray diffraction (XRD) pattern are fed into the DGPT model to generate an initial atomic structure. The theoretical XRD pattern of the generated structure is shown, along with the mean absolute error (MAE) when compared to the original input XRD pattern. To further demonstrate the impact of optimization, we apply the ALIGNN-FF (AFF) force field to relax the DGPT-generated structure, and the resulting XRD pattern for the optimized structure is shown along with its corresponding MAE. We observe some of the limitations of the model. For example, in Figure 4a, there are 6 peaks while the DGPT model generates model with 7 peaks for Silicon as shown in Figure 4b. After applying the ALIGNN-FF optimization, we observe that the number of peaks is corrected to 6, as expected as shown in Figure 4c. A similar trend is observed for La_2O_3 , where the input XRD pattern has 13 peaks (Figure 4f), but the DGPT model initially predicts 14 peaks (Figure 4g). This discrepancy is also corrected with ALIGNN-FF optimization. On the other hand, for the HfC case shown in Figure 4j, the predicted XRD pattern consistently matches the correct number of peaks, suggesting that ALIGNN-FF optimization may not be necessary in this case. We further quantify these observations with mean absolute error (MAE) values, comparing the target and predicted XRD patterns. The structure with the lower MAE can be considered the better candidate structure for the XRD pattern. Moreover, while for

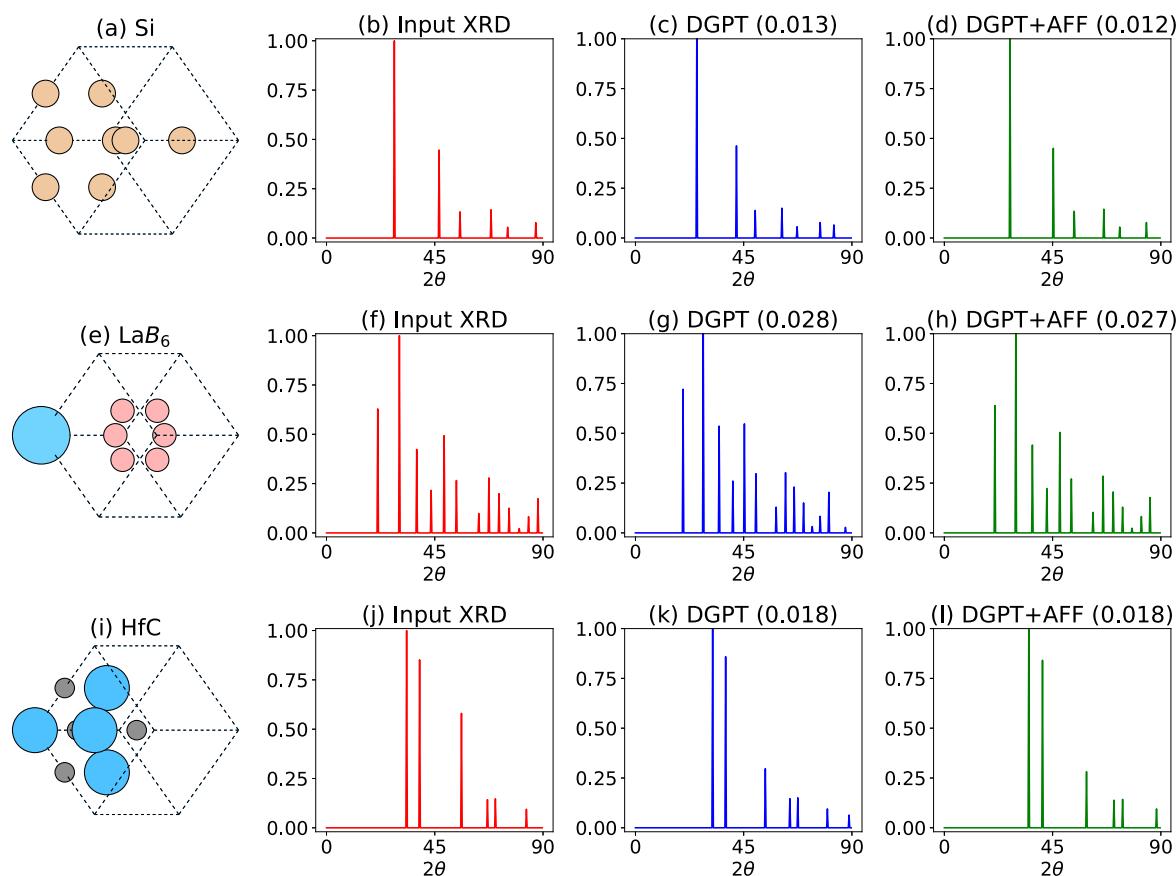


Figure 4. Evaluating the performance of the DiffractGPT (DGPT)-formula model with and without ALIGNNN-FF (AFF) optimization for a few example materials. The input chemical formula and XRD pattern are fed into the DGPT model to generate the atomic structure. The theoretical XRD pattern of the generated structure is shown as DGPT, along with the mean absolute error (MAE) of the XRD pattern in comparison with the input XRD. The DGPT structure is further optimized with AFF, and the XRD of the optimized structure, along with its MAE, is shown. (a) Silicon atomic structure, (b) input XRD pattern for Si, (c) XRD pattern of the DGPT-generated structure given the chemical formula and XRD, (d) XRD pattern for the AFF-optimized DGPT structure. Similar results for LaB₆ (e–h) and HfC (i–l) are shown.

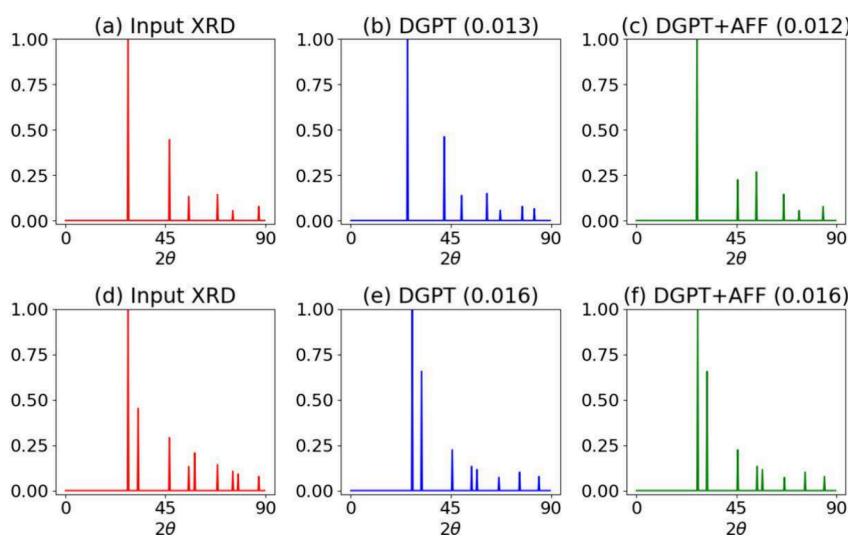


Figure 5. XRD patterns for both perfect and defective two-atom silicon (JVASP-1002) structure, with and without the displacement of an atom from its equilibrium position, are shown. The x-coordinate of the first atom is translated by 0 (panels a–c) and 0.2 (panels d–f), with the 0 translation representing the perfect crystal. After generating the crystals, we predict their simulated patterns. We then use these patterns, along with the chemical formula Si, to generate the DGPT-based atomic structure and its corresponding diffraction pattern. Furthermore, the DGPT-generated structure is optimized using ALIGNNN-FF, and the corresponding XRD patterns are also presented.

above analysis simulated XRD patterns were used as inputs, the same for experimental patterns is shown in Figure S1. The

experimental XRD pattern is scaled between 0 and 1 and peaks less than 0.04 as a threshold value are removed to align with the

training based simulated data. Interestingly, we observe excellent agreement for Si and LaB₆ case, but for HfC case we observe noticeable difference.

While the above analysis provides insights into the performance of the model in different scenarios, obtaining deeper physical insights into why these discrepancies occur is a more complex task. Due to the nature of deep learning models with billions of parameters, they tend to be less explainable, making it difficult to extract detailed physical explanations. However, we plan to explore such investigations in future work to better understand these behaviors.

Furthermore, there could be different types of real world diffraction patterns including defects. An example of silicon structure with and without defects (translated atom) is shown in Figure 5. After constructing a perfect silicon structure with two atoms in the primitive cell, The x-coordinate of the first atom is translated by 0 (panels a–c) and 0.2 (panels d–f), with the 0 translation representing the perfect crystal. After generating the crystals, we predict their simulated patterns. We then use these patterns, along with the chemical formula Si, to generate the DGPT-based atomic structure and its corresponding diffraction pattern. Furthermore, the DGPT-generated structure is optimized using ALIGNN-FF, and the corresponding XRD patterns are also presented. We observe that for the defective structure, the peaks show reasonable agreement for peaks before 45° 2θ, but after that, it begins to differ compared to input XRD pattern. This can be attributed to the fact that the current work has primarily focused on perfect materials with no defective structures explicitly included during training. However, it could be extended to defective materials in the future. Detecting defects, such as vacancies, dislocations or other imperfections, in materials through X-ray diffraction (XRD) is a challenging task. While XRD is commonly used to study crystalline materials, the presence of defects introduces complexities in the diffraction patterns. Previous studies, such as those utilizing convolutional neural networks^{52–54} and Long Short-Term Memory (LSTM) networks⁵⁵ for identifying vacancies, strain in semiconductors, have made progress in this area. Our model, trained on diffraction patterns from ideal structures, can be extended to defective systems by incorporating additional training data from materials with known defects. With such data, the model should be able to generalize and capture the diffraction features associated with defects and dislocations.

In conclusion, this study introduces an efficient approach for determining crystal structures from powder X-ray diffraction patterns. It goes beyond existing generative AI applications focused on scalar properties by facilitating structure generation and demonstrating the potential of using spectral data, such as XRD. The DiffractGPT model is capable of predicting material properties with high accuracy, particularly when the chemical elements of the materials are known. Notably, DiffractGPT outperforms conventional machine learning models, such as gradient boosting and convolution neural network, in predicting lattice constants while also providing the option to generate complete crystal structures. Additionally, the training process for DiffractGPT is straightforward, fast and relatively easy to learn, thereby bridging the gap between the computational, data science, and experimental communities. As a complementary tool, we offer a framework that matches experimental XRD patterns with existing databases, incorporating automated background subtraction. This work represents a significant advancement in the automation of crystal structure determination and provides a robust tool for data-driven materials

design, paving the way for enhanced research and development in materials science.

■ ASSOCIATED CONTENT

SI Supporting Information

The Supporting Information is available free of charge at <https://pubs.acs.org/doi/10.1021/acs.jpcllett.4c03137>.

Additional examples of evaluating the performance of the DiffractGPT-formula model with experimental XRD patterns as inputs ([PDF](#))

■ AUTHOR INFORMATION

Corresponding Author

Kamal Choudhary – Material Measurement Laboratory, National Institute of Standards and Technology, Gaithersburg, Maryland 20899, United States;  orcid.org/0000-0001-9737-8074; Email: kamal.choudhary@nist.gov

Complete contact information is available at: <https://pubs.acs.org/10.1021/acs.jpcllett.4c03137>

Notes

The author declares no competing financial interest.

■ ACKNOWLEDGMENTS

K.C. thanks the National Institute of Standards and Technology (NIST) for computational resources. K.C. thanks Maureen E. Williams, Adam J. Biacchi and Adam A. Creuziger at NIST for helpful discussion. This work was performed with funding from the CHIPS Metrology Program, part of CHIPS for America, National Institute of Standards and Technology, U.S. Department of Commerce. Certain commercial equipment, instruments, software, or materials are identified in this paper in order to specify the experimental procedure adequately. Such identifications are not intended to imply recommendation or endorsement by NIST, nor it is intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

■ REFERENCES

- (1) Als-Nielsen, J.; McMorrow, D. *Elements of modern X-ray physics*; John Wiley & Sons: West Sussex, UK, 2011.
- (2) Giacovazzo, C. *Fundamentals of crystallography*; Oxford University Press: Oxford, UK, 2002.
- (3) Wyon, C. X-ray metrology for advanced microelectronics. *European Physical Journal-Applied Physics* **2010**, *49*, 20101.
- (4) Holder, C. F.; Schaak, R. E. Tutorial on powder X-ray diffraction for characterizing nanoscale materials. *ACS Nano* **2019**, *13*, 7359–7365.
- (5) Brown, J. G. *X-rays and Their Applications*; Springer: New York, 2012.
- (6) Rodriguez-Carvajal, J.; Roisnel, T. *FullProf. 98 and WinPLOTR: new windows 95/NT applications for diffraction*; Commission for Powder Diffraction, International Union of Crystallography, Newsletter, 1998, 20, May–August.
- (7) Larson, A. C.; Von Dreele, R. B. *General Structure Analysis System (GSAS)*; Los Alamos National Laboratory Report LAUR: 1985; Vol. 86.
- (8) Glavic, A.; Björck, M. GenX 3: the latest generation of an established tool. *Journal of applied crystallography* **2022**, *55*, 1063–1071.
- (9) Coelho, A. A. TOPAS and TOPAS-Academic: an optimization program integrating computer algebra and crystallographic objects written in C++. *J. Appl. Crystallogr.* **2018**, *51*, 210–218.

- (10) Wenk, H.-R.; Lutterotti, L.; Vogel, S. Rietveld texture analysis from TOF neutron diffraction data. *Powder Diffraction* **2010**, *25*, 283–296.
- (11) Choudhary, K.; DeCost, B.; Chen, C.; Jain, A.; Tavazza, F.; Cohn, R.; Park, C. W.; Choudhary, A.; Agrawal, A.; Billinge, S. J.; et al. Recent advances and applications of deep learning methods in materials science. *npj Computational Materials* **2022**, *8*, 59.
- (12) Vasudevan, R. K.; Choudhary, K.; Mehta, A.; Smith, R.; Kusne, G.; Tavazza, F.; Vlcek, L.; Ziatdinov, M.; Kalinin, S. V.; Hattrick-Simpers, J. Materials science in the artificial intelligence age: high-throughput library generation, machine learning, and a pathway from correlations to the underpinning physics. *MRS Commun.* **2019**, *9*, 821–838.
- (13) Schmidt, J.; Marques, M. R.; Botti, S.; Marques, M. A. Recent advances and applications of machine learning in solid-state materials science. *npj Computational Materials* **2019**, *5*, 83.
- (14) CHIPS.Gov — nist.gov. <https://www.nist.gov/chips> [accessed October 10, 2024].
- (15) Surdu, V.-A.; György, R. X-ray diffraction data analysis by machine learning methods—a review. *Applied Sciences* **2023**, *13*, 9992.
- (16) Park, W. B.; Chung, J.; Jung, J.; Sohn, K.; Singh, S. P.; Pyo, M.; Shin, N.; Sohn, K.-S. Classification of crystal structure using a convolutional neural network. *IUCrJ* **2017**, *4*, 486–494.
- (17) Zhdanov, M.; Zhdanov, A. Machine learning-assisted close-set X-ray diffraction phase identification of transition metals. *arXiv preprint arXiv:2303.15410*, **2023** [accessed October 10, 2024].
- (18) Lee, B. D.; Lee, J.-W.; Park, W. B.; Park, J.; Cho, M.-Y.; Pal Singh, S.; Pyo, M.; Sohn, K.-S. Powder X-ray diffraction pattern is all you need for machine-learning-based symmetry identification and property prediction. *Advanced Intelligent Systems* **2022**, *4*, 2200042.
- (19) Banko, L.; Maffettone, P. M.; Naujoks, D.; Olds, D.; Ludwig, A. Deep learning for visualization and novelty detection in large X-ray diffraction datasets. *Npj Computational Materials* **2021**, *7*, 104.
- (20) Yanxon, H.; Weng, J.; Parraga, H.; Xu, W.; Ruett, U.; Schwarz, N. Artifact identification in X-ray diffraction data using machine learning methods. *Journal of Synchrotron Radiation* **2023**, *30*, 137–146.
- (21) Zaloga, A. N.; Stanovov, V. V.; Bezrukova, O. E.; Dubinin, P. S.; Yakimov, I. S. Crystal symmetry classification from powder X-ray diffraction patterns using a convolutional neural network. *Materials Today Communications* **2020**, *25*, 101662.
- (22) Venderley, J.; Mallayya, K.; Matty, M.; Krogstad, M.; Ruff, J.; Pleiss, G.; Kishore, V.; Mandrus, D.; Phelan, D.; Poudel, L.; et al. Harnessing interpretable and unsupervised machine learning to address big data from modern X-ray diffraction. *Proceedings of the National Academy of Sciences* **2022**, *119*, No. e2109665119.
- (23) Lee, J.-W.; Park, W. B.; Lee, J. H.; Singh, S. P.; Sohn, K.-S. A deep-learning technique for phase identification in multiphase inorganic compounds using synthetic XRD powder patterns. *Nat. Commun.* **2020**, *11*, 86.
- (24) Maffettone, P. M.; Banko, L.; Cui, P.; Lysogorskiy, Y.; Little, M. A.; Olds, D.; Ludwig, A.; Cooper, A. I. Crystallography companion agent for high-throughput materials discovery. *Nature Computational Science* **2021**, *1*, 290–297.
- (25) Oviedo, F.; Ren, Z.; Sun, S.; Settens, C.; Liu, Z.; Hartono, N. T. P.; Ramasamy, S.; DeCost, B. L.; Tian, S. I.; Romano, G.; et al. Fast and interpretable classification of small X-ray diffraction datasets using data augmentation and deep neural networks. *npj Computational Materials* **2019**, *5*, 60.
- (26) Chen, L.; Wang, B.; Zhang, W.; Zheng, S.; Chen, Z.; Zhang, M.; Dong, C.; Pan, F.; Li, S. Crystal Structure Assignment for Unknown Compounds from X-ray Diffraction Patterns with Deep Learning. *J. Am. Chem. Soc.* **2024**, *146*, 8098.
- (27) Xin, C.; Yin, Y.; Song, B.; Fan, Z.; Song, Y.; Pan, F. Machine learning-accelerated discovery of novel 2D ferromagnetic materials with strong magnetization. *Chip* **2023**, *2*, 100071.
- (28) Xin, C.; Song, B.; Jin, G.; Song, Y.; Pan, F. Advancements in High-Throughput Screening and Machine Learning Design for 2D Ferromagnetism: A Comprehensive Review. *Advanced Theory and Simulations* **2023**, *6*, 2300475.
- (29) Choudhary, K. AtomGPT: Atomistic Generative Pretrained Transformer for Forward and Inverse Materials Design. *J. Phys. Chem. Lett.* **2024**, *15*, 6909–6917.
- (30) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*; 2017; Vol. 30.
- (31) Tunstall, L.; Von Werra, L.; Wolf, T. *Natural language processing with transformers*; O'Reilly Media, Inc.: 2022.
- (32) Rothman, D. *Transformers for Natural Language Processing: Build innovative deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and more*; Packt Publishing Ltd: 2021.
- (33) Wu, T.; He, S.; Liu, J.; Sun, S.; Liu, K.; Han, Q.-L.; Tang, Y. A brief overview of ChatGPT: The history, status quo and potential future development. *IEEE/CAA Journal of Automatica Sinica* **2023**, *10*, 1122–1136.
- (34) Pimentel, A.; Wagener, A.; da Silveira, E. F.; Picciani, P.; Salles, B.; Follmer, C.; Oliveira Jr, O. N. Challenging ChatGPT with Chemistry-Related Subjects. *Chemrxiv preprint* **2023**, DOI: [10.26434/chemrxiv-2023-xl6w3](https://doi.org/10.26434/chemrxiv-2023-xl6w3). Accessed October 10, 2024.
- (35) Jablonka, K. M.; Schwaller, P.; Ortega-Guerrero, A.; Smit, B. Leveraging large language models for predictive chemistry. *Nature Machine Intelligence* **2024**, *6*, 161.
- (36) Polak, M. P.; Morgan, D. Extracting accurate materials data from research papers with conversational language models and prompt engineering. *Nat. Commun.* **2024**, *15*, 1569.
- (37) Wines, D.; Gurunathan, R.; Garrity, K. F.; DeCost, B.; Biacchi, A. J.; Tavazza, F.; Choudhary, K. Recent progress in the JARVIS infrastructure for next-generation data-driven materials design. *Applied Physics Reviews* **2023**, *10*, 041302.
- (38) Choudhary, K.; Garrity, K. F.; Reid, A. C.; DeCost, B.; Biacchi, A. J.; Hight Walker, A. R.; Trautt, Z.; Hattrick-Simpers, J.; Kusne, A. G.; Centrone, A.; et al. The joint automated repository for various integrated simulations (JARVIS) for data-driven materials design. *npj computational materials* **2020**, *6*, 173.
- (39) Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; Casas, D. d. l.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. Mistral 7B. *arXiv preprint arXiv:2310.06825*, **2023** [Accessed October 10, 2024].
- (40) Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Zidek, A.; Potapenko, A.; et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583–589.
- (41) Choudhary, K.; Zhang, Q.; Reid, A. C.; Chowdhury, S.; Van Nguyen, N.; Trautt, Z.; Newrock, M. W.; Congo, F. Y.; Tavazza, F. Computational screening of high-performance optoelectronic materials using OptB88vdW and TB-mBJ formalisms. *Scientific data* **2018**, *5*, 1–12.
- (42) Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, **2021** [Accessed October 10, 2024].
- (43) Unslot AI Unslot - GitHub Repository. <https://github.com/unslotai/unslot>. Accessed: October 10, 2024.
- (44) Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, **2023** [Accessed October 10, 2024].
- (45) Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, **2023** [Accessed October 10, 2024].
- (46) Taori, R.; Gulrajani, I.; Zhang, T.; Dubois, Y.; Li, X.; Guestin, C.; Liang, P.; Hashimoto, T. B. Stanford alpaca: an instruction-following llama model (2023). URL: https://github.com/tatsu-lab/stanford_alpaca [Accessed October 10, 2024].
- (47) Su, J.; Ahmed, M.; Lu, Y.; Pan, S.; Bo, W.; Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing* **2024**, *568*, 127063.

- (48) Ong, S. P.; Richards, W. D.; Jain, A.; Hautier, G.; Kocher, M.; Cholia, S.; Gunter, D.; Chevrier, V. L.; Persson, K. A.; Ceder, G. Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis. *Comput. Mater. Sci.* **2013**, *68*, 314–319.
- (49) Xie, T.; Fu, X.; Ganea, O.-E.; Barzilay, R.; Jaakkola, T. Crystal diffusion variational autoencoder for periodic material generation. *arXiv preprint arXiv:2110.06197*, 2021 [Accessed October 10, 2024].
- (50) Li, Y.; Yang, W.; Dong, R.; Hu, J. Mlatticeabc: generic lattice constant prediction of crystal materials using machine learning. *ACS omega* **2021**, *6*, 11585–11594.
- (51) Choudhary, K.; DeCost, B.; Major, L.; Butler, K.; Thiyagalingam, J.; Tavazza, F. Unified graph neural network force-field for the periodic table: solid state applications. *Digital Discovery* **2023**, *2*, 346–355.
- (52) Lim, B.; Bellec, E.; Dupraz, M.; Leake, S.; Resta, A.; Coati, A.; Sprung, M.; Almog, E.; Rabkin, E.; Schulli, T.; et al. A convolutional neural network for defect classification in Bragg coherent X-ray diffraction. *npj Computational Materials* **2021**, *7*, 115.
- (53) Boulle, A.; Debelle, A. Convolutional neural network analysis of x-ray diffraction data: strain profile retrieval in ion beam modified materials. *Machine Learning: Science and Technology* **2023**, *4*, 015002.
- (54) Judge, W.; Chan, H.; Sankaranarayanan, S.; Harder, R. J.; Cabana, J.; Cherukara, M. J. Defect identification in simulated Bragg coherent diffraction imaging by automated AI. *MRS Bull.* **2023**, *48*, 124–133.
- (55) Motamedi, M.; Shidpour, R.; Ezoji, M. LSTM-based framework for predicting point defect percentage in semiconductor materials using simulated XRD patterns. *Sci. Rep.* **2024**, *14*, 24353.