BENCHMARKING TEXT REPRESENTATIONS FOR CRYSTAL STRUCTURE GENERATION WITH LARGE LANGUAGE MODELS

Shuyi Jia[1], **Aamod Varma**[1], **Pranav Manivannan**[1], **Dhruva Chayapathy**[2] **& Victor Fung**[1†]
[1] Computational Science and Engineering, Georgia Institute of Technology, Atlanta, GA, USA
[2] Alpharetta High School, Alpharetta, GA, USA
[†] Corresponding authors: victorfung@gatech.edu

ABSTRACT

The discovery of novel materials is essential for scientific and technological advancements but remains a significant challenge due to the vastness of the chemical space. Large language models (LLMs) have shown particular promise as generative models for materials discovery, where novel materials are generated in the form of textual representations of their crystal structures. In this work, we benchmark the performance of several textual representations with different levels of invariances and invertibility for crystal structure generation, covering Fractional, Z-matrix, distance matrix, and SLICES representations. We find that all representations can be effectively leveraged by LLMs for structure generation. However, we observe that the inclusion of translation and rotation invariances in more complex representations does not necessarily yield better generation performance, contrary to expectations. These findings suggest that established design principles for conventional structure representations do not apply for LLMs. This study establishes the first benchmark for textual representations in crystal structure generation using fine-tuned LLMs, offering a foundation for accelerating materials discovery with language models.

1 INTRODUCTION

Discovering novel materials with targeted functional properties is a longstanding challenge in materials science, largely due to the vast chemical space these materials can potentially span (Davies et al., 2016). Traditional materials discovery methods often rely on exhaustive laboratory or computational screening, which is both time-consuming and resource-intensive (Pyzer-Knapp et al., 2015; Liu et al., 2017). Recent advances in machine learning have led to the development of a diverse class of generative modeling techniques as an alternative approach towards materials discovery. These examples include diffusion models utilizing equivariant graph neural networks (GNNs) to directly generate crystal structures (Xie et al., 2021; Jiao et al., 2023; Zeni et al., 2023), as well as methods that utilize intermediate representations to encode existing structures and decode them into new materials (Hoffmann et al., 2019; Court et al., 2020; Long et al., 2021; Fung et al., 2022; Sinha et al., 2024). Additionally, rapid developments in large language models (LLMs) have led to their growing application in materials discovery (Miret & Krishnan, 2024; Jia et al., 2024; Ding et al., 2024), with recent studies exploring the generation of crystals using structure file formats such as CIF (Flam-Shepherd & Aspuru-Guzik, 2023; Antunes et al., 2023; Gruver et al., 2024; Mohanty et al., 2024; Kazeev et al., 2025). While the choice of CIF representation is intuitive as it is straightforward and contains all the necessary information to fully describe a 3D crystal structure, it fails to contain rotation, translation and permutation invariances which often cited as a key requirement in the representation of atomic structures (Musil et al., 2021).

Effectively representing materials in a textual form provides the opportunity to better leverage the remarkable advancements in LLMs for accelerating materials discovery. In a recent study, Alampara et al. (2024) introduced a suite of benchmarking tools and datasets to systematically evaluate the performance of language models in materials modeling, incorporating nine distinct text representations. However, their study primarily focuses on downstream tasks such as materials property

prediction, and most of the text representations used are not invertible, making them unsuitable for crystal structure generation.

In this study, we explore alternative textual representations of crystal structures that incorporate different invariances, aiming to enhance their generation through the fine-tuning of LLMs. Specifically, we benchmark three fundamentally distinct textual representations of crystal structures and compare their performance to the previously established CIF format introduced by Gruver et al. (2024). Our main findings show that LLM-based structure generation can work flexibly on very different types of text representations, and all obtain a competitive level of performance. However, more complex representations which incorporate translation and rotation invariances, and techniques for learning permutation invariances, were all found to be largely ineffective or detrimental to the performance.

Our code and dataset are available at `https://github.com/shuyijia/LLM4StructGen`.

2 METHODS

2.1 TEXTUAL REPRESENTATIONS OF CRYSTAL STRUCTURES

Fractional We adopt the reduced CIF representation proposed by Gruver et al. (2024), which we refer to as the Fractional representation. This representation includes the unit cell's lattice parameters, atom types, and the fractional coordinates of the atoms.

Z-matrix The Z-matrix describes each atom in a molecule using four attributes: atomic numbers, bond lengths, bond angles, and dihedral angles. As these attributes are internal coordinates, the representation is inherently invariant to rotation and translation. For materials, we extend this representation by including unit cell lattice parameters and adding three fictitious anchoring atoms at the corners of the unit cell.

Distance Matrix The distance matrix is an $N \times N$ matrix that encodes all pairwise distances between atoms in the unit cell. Since it is symmetric, only the lower triangular part is used. These distances, being internal coordinates, are invariant to rotation and translation. To fully represent the crystal structure, the lattice parameters of the unit cell are prepended to the distance matrix string.

SLICES SLICES (Xiao et al., 2023) is an invariant string representation that encodes the composition and bonding of atoms within and across the unit cell, without relying on explicit atomic coordinates. It consists of atomic symbols followed by edge descriptions in the format $u\ v\ x\ y\ z$, where u and v are node indices, and $x\ y\ z$ denotes the unit cell location for bond connections.

Table 1 summarizes the invariances and invertibility of the representations. Notably, both the Fractional and Z-matrix representations can be decoded directly to structures. In contrast, the distance matrix and SLICES require specialized decoding strategies. For the distance matrix, we employ the gradient-based reconstruction algorithm proposed by (Fung et al., 2022) (see Appendix A). Fig. 1 provides an overview of the entire fine-tuning pipeline, accompanied by examples of the four textual representations for the material SiOs.

Table 1: Invariances and invertibility of the selected representations.

	Invariance			Invertibility
	Translation	Rotation	Permutation	
Fractional	×	×	×	direct
Z-matrix	√	√	×	direct
Distance	√	√	×	reconstruction
SLICES	√	√	×	reconstruction

2.2 EXPERIMENTAL SETUP

Datasets We conduct experiments using the MP-20 dataset curated by Xie et al. (2021). This dataset is divided into training, validation, and test splits containing 27,136, 9,046, and 9,047 stable inorganic structures, respectively, sourced from the Materials Project (Chen & Ong, 2022). Each structure in the dataset contains at most 20 atoms per unit cell.

Models We fine-tune two open-source LLMs, LLaMA-2 7B and LLaMA-3 8B, using the aforementioned textual representations with the `torchtune` package. To address limited GPU avail-

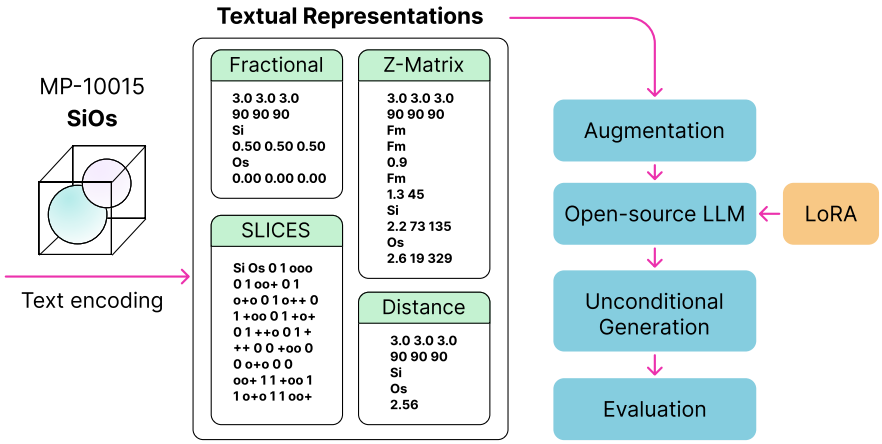


Figure 1: Overview of fine-tuning LLMs for crystal structure generation.

ability, we employ the LoRA fine-tuning technique (Hu et al., 2021) with a rank of 8 and an alpha of 16. Detailed hyperparameters and training configurations are provided in Appendix B.

Prompt Design We build upon the original unconditional generation prompt design introduced by Gruver et al. (2024) for the Fractional representation and adapt it to support the other three textual representations. We provide examples of the prompts in Appendix C.

Evaluation Metrics For structures generated by fine-tuned LLMs, we utilize the validity and diversity metrics proposed by Xie et al. (2021). Specifically, a structure is considered structurally valid if no pairwise distance is < 0.5 Å, and compositionally valid if the overall charge is neutral, as determined by `SMACT` (Davies et al., 2019). Coverage recall measures the percentage of ground truth materials correctly predicted, while coverage precision evaluates the percentage of predicted materials with high quality. Additionally, density (ρ, g/cm^3) and the number of unique elements (# elem.) are used to compute the Earth Mover's Distance (EMD) between the generated materials and the test materials. We collectively refer to this set of metrics as the CDVAE metrics. While these metrics are relatively basic, they are commonly adopted in materials generation literature due to their interpretability and low computational cost.

3 Results

We sample 10,000 valid textual representations from each fine-tuned LLM. The success rate is determined by whether a structure can be successfully decoded into an `ase.Atoms` object. All invalid strings are recorded to calculate the success rate as $(10000 - \#\text{ invalid})/10000$. The CDVAE metrics for the sampled structures are presented in Table 2. The baseline refers to the results reported by Gruver et al. (2024). For each representation, we explore permuting the order of atoms randomly within the unit cell as a form of data augmentation during fine-tuning. Notably, SLICES requires a relaxation step using the MACE interatomic potential (IAP) (Batatia et al., 2022).

From Table 2, we observe that the Fractional, Z-matrix, and distance matrix representations consistently achieve success rates in decoding, exceeding 99% for 10,000 sampled strings in most cases. In contrast, the fine-tuned models struggle with the SLICES representation, achieving a maximum sampling success rate of only 64.37%. Next, we observe that enabling or disabling random permutation of atoms during fine-tuning has minimal impact on the performance of the Fractional representation. However, for other representations, enabling permutation adversely affects performance. For example, in the Z-matrix and SLICES representations, disabling permutation leads to significant improvements in all metrics. Overall, the Fractional representation consistently outperforms other types, particularly the Z-matrix, which uses internal coordinates. Structures generated with the Fractional representation exhibit higher structural validity and better alignment with the test set distribution in terms of density and element count. While both textual representations scale linearly in terms of strings versus tokens (see Appendix D), the Z-matrix is arguably more complex, requir-

Table 2: Evaluation of textual representations for crystal structure generation using CDVAE metrics. Bold text highlights the best results, while underlined text indicates the second-best results. Post-generation relaxation via an interatomic potential is indicated by †.

	Model	Permutation	Success Rate	Validity ↑		Coverage ↑		Property ↓	
				Struct.	Comp.	Recall	Precision	ρ	# elem.
Baseline	LLaMA-2-7B	–	–	0.918	<u>0.879</u>	0.969	0.960	3.850	0.960
Fractional	LLaMA-2-7B	√	<u>0.9992</u>	0.946	0.789	<u>0.984</u>	0.983	0.862	0.488
Fractional	LLaMA-2-7B	×	<u>0.9992</u>	<u>0.954</u>	0.866	0.982	0.994	1.368	**0.126**
Fractional	LLaMA-3-8B	√	**0.9995**	0.936	0.823	<u>0.984</u>	0.981	**0.532**	0.511
Fractional	LLaMA-3-8B	×	0.9932	**0.955**	0.815	<u>0.984</u>	0.972	<u>0.613</u>	0.415
Z-matrix	LLaMA-2-7B	√	0.9962	0.793	0.791	0.773	0.994	0.864	0.691
Z-matrix	LLaMA-2-7B	×	0.9981	0.805	0.811	0.889	**0.997**	1.887	0.172
Z-matrix	LLaMA-3-8B	√	0.9975	0.740	0.807	0.795	0.988	1.249	0.987
Z-matrix	LLaMA-3-8B	×	0.9934	0.875	0.859	0.944	0.995	0.946	0.643
Distance	LLaMA-2-7B	√	0.9987	0.913	0.785	0.967	0.988	1.029	0.908
Distance	LLaMA-2-7B	×	0.9976	0.901	**0.889**	0.977	<u>0.996</u>	2.222	0.303
Distance	LLaMA-3-8B	√	0.9781	0.864	0.689	0.960	0.993	1.223	0.562
Distance	LLaMA-3-8B	×	0.9942	0.894	0.864	**0.989**	**0.997**	0.685	0.296
SLICES†	LLaMA-2-7B	√	0.6124	0.760	0.770	0.899	0.991	5.760	0.156
SLICES†	LLaMA-2-7B	×	0.4819	0.864	0.825	<u>0.984</u>	0.995	2.872	<u>0.147</u>
SLICES†	LLaMA-3-8B	√	0.6437	0.807	0.789	0.964	0.991	6.422	0.249
SLICES†	LLaMA-3-8B	×	0.4871	0.815	0.843	<u>0.984</u>	0.995	3.877	0.182

ing bond lengths, angles, and dihedrals, which vary significantly in scale. In contrast, the Fractional representation relies solely on fractional coordinates within the range of $[0, 1)$.

The distance matrix representation performs competitively with the Fractional representation, achieving the highest coverage recall and precision. This result suggests that textual representations, even when not directly invertible, can be used by LLMs for generating novel crystal structures, provided they are paired with a suitable decoding algorithm. In the future, this can potentially be extended to other non-invertible representations for generation by LLMs, including ones commonly used in machine-learned IAPs (Musil et al., 2021).

SLICES performs the worst among the four representations, even after relaxation with an IAP. This could be attributed to two potential factors. First, SLICES strings can become excessively long, as the number of edge labels is directly proportional to the number of neighbors within a 10.0 Å cutoff radius (Xiao et al., 2023). For example, structures from the training set have an average of 40.3 edge labels, corresponding to approximately 200 tokens for edges per structure. This is 2–3 times that of the other representations (see Appendix D for detailed calculations). Second, as a recently developed and specialized string representation, SLICES may be unfamiliar to LLMs trained on more common formats like SMILES. Another consideration is that specialized tokenization for SLICES might potentially be needed, rather than relying on the default LLaMA tokenizers.

Table 3 presents the CDVAE metrics for three selected variants of the Fractional, Z-matrix, and distance matrix representations. The generated structures from each variant were relaxed using the same IAP employed by SLICES, ensuring a fair comparison. As expected, optimizing the structures leads to an overall performance improvement, with validity metrics showing the most significant gains. This suggests that crystal structure generation using fine-tuned LLMs can be integrated with other tools or embedded into an end-to-end workflow to further enhance the quality of the generated structures. The formation energy distribution plots for both unrelaxed and relaxed structures are provided in Appendix E. Visualizations of selected materials are provided in Appendix F.

Next, we employ the Increase in Perplexity under Transformations (IPT) metric (Gruver et al., 2024) to evaluate the extent to which language models are invariant to continuous group transformations. Similar to the original definition, the IPT score for an input string representation s is

$$\mathrm{IPT}(s) = \mathbb{E}_{g \in G} \left[\mathrm{PPL}(t_g(s)) - \mathrm{PPL}(t_{g^*}(s)) \right], \tag{1}$$

Table 3: Evaluation of relaxed structures generated using the Fractional, Z-matrix, and distance matrix representations with the interatomic potential MACE. Bold text highlights the best results.

	Model	Permutation	Relaxation (MACE)	Validity ↑		Coverage ↑		Property ↓	
				Struct.	Comp.	Recall	Precision	ρ	# elem.
Fractional	LLaMA-3-8B	×	√	**0.989**	0.819	0.991	0.974	1.06	0.360
Z-matrix	LLaMA-3-8B	×	√	0.968	0.858	**0.998**	0.992	0.948	0.639
Distance	LLaMA-3-8B	×	√	0.968	**0.864**	0.991	**0.996**	**0.680**	**0.264**

where G denotes the transformation group with elements g and corresponding actions t_g, PPL is the sequence perplexity, and g^* is the element minimizing perplexity. In our case, G is the permutation group, where each g is a unique atomic sequence ordering, and t_g is its associated decoding scheme. Following the setup of Gruver et al. (2024), we sample 500 random structures from the test set, apply 20 random permutations to each, and compute the sample mean. Fig. 2 shows the IPT results for the LLaMA-3-8B models with permutation augmentation. Fractional and distance matrix representations yield relatively lower IPTs, indicating learning of permutation invariance by minimal perplexity changes. In contrast, Z-Matrix and SLICES show much higher IPTs, in line with their poorer performance shown in Table 2.

4 Conclusion

In this work, we demonstrate the relative performance of fine-tuned large language models (LLMs) for crystal structure generation using various distinct textual representations. While all representations were found to work well, simple designs, such as the Fractional representation, consistently achieve the best performance, even without incorporating any invariances. Overall, our study provides the first benchmark for textual representations in crystal structure generation using fine-tuned LLMs. Future work could explore extending these representations to encode additional physical properties or constraints, performing conditional generation, and employing more advanced metrics, such as running density functional theory, for further validation.

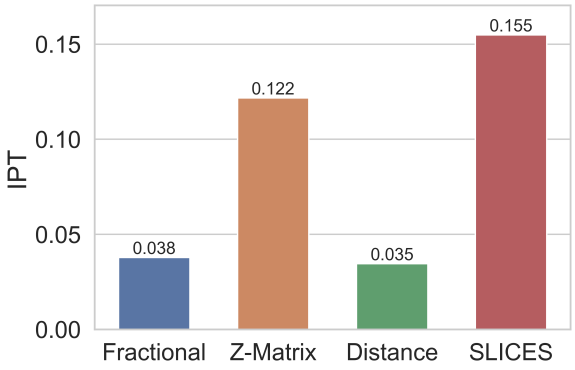


Figure 2: Increase in Perplexity under Transformations (IPT) for various representations using LLaMA-3-8B with permutation augmentation.

References

Nawaf Alampara, Santiago Miret, and Kevin Maik Jablonka. Mattext: Do language models need more than text & scale for materials modeling? *arXiv preprint arXiv:2406.17295*, 2024.

Luis M Antunes, Keith T Butler, and Ricardo Grau-Crespo. Crystal structure generation with autoregressive large language modeling. *arXiv preprint arXiv:2307.04340*, 2023.

Ilyes Batatia, David P Kovacs, Gregor Simm, Christoph Ortner, and Gábor Csányi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. *Advances in Neural Information Processing Systems*, 35:11423–11436, 2022.

Chi Chen and Shyue Ping Ong. A universal graph deep learning interatomic potential for the periodic table. *Nature Computational Science*, 2(11):718–728, 2022.

Callum J Court, Batuhan Yildirim, Apoorv Jain, and Jacqueline M Cole. 3-d inorganic crystal structure generation and property prediction via representation learning. *Journal of Chemical Information and Modeling*, 60(10):4518–4535, 2020.

Daniel W Davies, Keith T Butler, Adam J Jackson, Andrew Morris, Jarvist M Frost, Jonathan M Skelton, and Aron Walsh. Computational screening of all stoichiometric inorganic materials. *Chem*, 1(4):617–627, 2016.

Daniel W Davies, Keith T Butler, Adam J Jackson, Jonathan M Skelton, Kazuki Morita, and Aron Walsh. Smact: Semiconducting materials by analogy and chemical theory. *Journal of Open Source Software*, 4(38):1361, 2019.

Qianggang Ding, Santiago Miret, and Bang Liu. Matexpert: Decomposing materials discovery by mimicking human experts. *arXiv preprint arXiv:2410.21317*, 2024.

Daniel Flam-Shepherd and Alán Aspuru-Guzik. Language models can generate molecules, materials, and protein binding sites directly in three dimensions as xyz, cif, and pdb files. *arXiv preprint arXiv:2305.05708*, 2023.

Victor Fung, Shuyi Jia, Jiaxin Zhang, Sirui Bi, Junqi Yin, and Panchapakesan Ganesh. Atomic structure generation from reconstructing structural fingerprints. *Machine Learning: Science and Technology*, 3(4):045018, 2022.

Nate Gruver, Anuroop Sriram, Andrea Madotto, Andrew Gordon Wilson, C Lawrence Zitnick, and Zachary Ulissi. Fine-tuned language models generate stable inorganic materials as text. *arXiv preprint arXiv:2402.04379*, 2024.

Jordan Hoffmann, Louis Maestrati, Yoshihide Sawada, Jian Tang, Jean Michel Sellier, and Yoshua Bengio. Data-driven approach to encoding and decoding 3-d crystal structures. *arXiv preprint arXiv:1909.00949*, 2019.

Rudolf Hoppe. Effective coordination numbers (econ) and mean fictive ionic radii (mefir). *Zeitschrift für Kristallographie-Crystalline Materials*, 150(1-4):23–52, 1979.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Shuyi Jia, Chao Zhang, and Victor Fung. Llmatdesign: Autonomous materials discovery with large language models. *arXiv preprint arXiv:2406.13163*, 2024.

Rui Jiao, Wenbing Huang, Peijia Lin, Jiaqi Han, Pin Chen, Yutong Lu, and Yang Liu. Crystal structure prediction by joint equivariant diffusion. *Advances in Neural Information Processing Systems*, 36:17464–17497, 2023.

Nikita Kazeev, Wei Nong, Ignat Romanov, Ruiming Zhu, Andrey Ustyuzhanin, Shuya Yamazaki, and Kedar Hippalgaonkar. Wyckoff transformer: Generation of symmetric crystals. *arXiv preprint arXiv:2503.02407*, 2025.

Yue Liu, Tianlu Zhao, Wangwei Ju, and Siqi Shi. Materials discovery and design using machine learning. *Journal of Materiomics*, 3(3):159–177, 2017.

Teng Long, Nuno M Fortunato, Ingo Opahle, Yixuan Zhang, Ilias Samathrakis, Chen Shen, Oliver Gutfleisch, and Hongbin Zhang. Constrained crystals deep convolutional generative adversarial network for the inverse design of crystal structures. *npj Computational Materials*, 7(1):66, 2021.

Santiago Miret and NM Krishnan. Are llms ready for real-world materials discovery? *arXiv preprint arXiv:2402.05200*, 2024.

Trupti Mohanty, Maitrey Mehta, Hasan M Sayeed, Vivek Srikumar, and Taylor D Sparks. Crystext: A generative ai approach for text-conditioned crystal structure generation using llm. 2024.

Felix Musil, Andrea Grisafi, Albert P Bartók, Christoph Ortner, Gábor Csányi, and Michele Ceriotti. Physics-inspired structural representations for molecules and materials. *Chemical Reviews*, 121 (16):9759–9815, 2021.

Mark Neumann, James Gin, Benjamin Rhodes, Steven Bennett, Zhiyi Li, Hitarth Choubisa, Arthur Hussey, and Jonathan Godwin. Orb: A fast, scalable neural network potential. *arXiv preprint arXiv:2410.22570*, 2024.

Shyue Ping Ong, William Davidson Richards, Anubhav Jain, Geoffroy Hautier, Michael Kocher, Shreyas Cholia, Dan Gunter, Vincent L Chevrier, Kristin A Persson, and Gerbrand Ceder. Python materials genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, 68:314–319, 2013.

Edward O Pyzer-Knapp, Changwon Suh, Rafael Gómez-Bombarelli, Jorge Aguilera-Iparraguirre, and Alán Aspuru-Guzik. What is high-throughput virtual screening? a perspective from organic materials discovery. *Annual Review of Materials Research*, 45(1):195–216, 2015.

Anshuman Sinha, Shuyi Jia, and Victor Fung. Representation-space diffusion models for generating periodic materials. *arXiv preprint arXiv:2408.07213*, 2024.

Hang Xiao, Rong Li, Xiaoyang Shi, Yan Chen, Liangliang Zhu, Xi Chen, and Lei Wang. An invertible, invariant crystal representation for inverse design of solid-state materials using generative deep learning. *Nature Communications*, 14(1):7027, 2023.

Tian Xie, Xiang Fu, Octavian-Eugen Ganea, Regina Barzilay, and Tommi Jaakkola. Crystal diffusion variational autoencoder for periodic material generation. *arXiv preprint arXiv:2110.06197*, 2021.

Claudio Zeni, Robert Pinsler, Daniel Zügner, Andrew Fowler, Matthew Horton, Xiang Fu, Sasha Shysheya, Jonathan Crabbé, Lixin Sun, Jake Smith, et al. Mattergen: a generative model for inorganic materials design. *arXiv preprint arXiv:2312.03687*, 2023.

A GRADIENT-BASED DECODING ALGORITHM FOR DISTANCE MATRIX

We note the distance matrix is not directly invertible, meaning there is no explicit mapping from the string representation back to a 3D structure. To reconstruct structures from the generated distance matrices, we utilize the gradient-based algorithm proposed by Fung et al. (2022).

Specifically, we denote the sampled string from a fine-tuned LLM as S, from which we extract the atomic numbers $\boldsymbol{A} \in \mathbb{R}^N$, lattice parameters $\boldsymbol{L} \in \mathbb{R}^6$, and the distance matrix $\boldsymbol{D} \in \mathbb{R}^{N \times N}$. To obtain the exact 3D coordinates, we use Algo. 1.

Algorithm 1 Gradient-based Reconstruction Algorithm

Input: $\boldsymbol{A}$, $\boldsymbol{L}$, $\boldsymbol{D}$
$\mathcal{M}$:= number of initializations
$\mathcal{N}$:= number of basin hops
T := number of iterations over positions

for $i = 1$ to $\mathcal{M}$ **do**
  Randomly initialize a set of Cartesian positions $\tilde{\boldsymbol{X}}_0$
  for $j = 1$ to $\mathcal{N}$ **do**
    for $t = 1$ to T **do**
      Calculate the distance matrix $\tilde{\boldsymbol{D}}_{t-1}$ from $\tilde{\boldsymbol{X}}_{t-1}$
      $\tilde{\boldsymbol{X}}_t \leftarrow \tilde{\boldsymbol{X}}_{t-1} - \eta \nabla_{\tilde{\boldsymbol{X}}_{t-1}} \mathcal{L}\left(\boldsymbol{D}, \tilde{\boldsymbol{D}}_{t-1}\right)$
    end for
    $\tilde{\boldsymbol{X}}_T \leftarrow \tilde{\boldsymbol{X}}_T + \varepsilon$, where $\varepsilon \sim N(0, \boldsymbol{I})$.
  end for
end for

Essentially, we initialize a set of random positions and iteratively optimize them using the loss function $\mathcal{L}\left(\boldsymbol{D}, \tilde{\boldsymbol{D}}_{t-1}\right)$. In other words, the goal is to find a set of positions that minimizes the difference between its distance matrix $\tilde{\boldsymbol{D}}$ and the target distance matrix $\boldsymbol{D}$ derived from the sampled string S.

B HYPERPARAMETERS AND TRAINING SETUP

In Table 4, we include the hyperparameter settings for fine-tuning LLaMA-2 7B and LLaMA-3 8B with LoRA. The fine-tuning and sampling are completed on A100 40GB GPUs.

Table 4: LLaMA-2 7B & LLaMA-3 8B fine-tuning with LoRA hyperparameters

Parameter	Value or function
Batch size	4
Data type	BF16
No. of epochs	10
Gradient accumulation steps	64
Learning rate	0.0003
Optimizer	AdamW
Optimizer weight decay	0.01
Learning rate scheduler	cosine schedule with warmup
No. of warm up steps	100
LoRA rank	8
LoRA alpha	16

C PROMPT TEMPLATES

Fractional

Below is a description of a bulk material. Generate a description of the lengths and angles of the lattice vectors and then the element type and coordinates for each atom within the lattice:

`<Fractional string>`

Z-matrix

Below is a description of a bulk material where each atom is described by its element type and three attributes: 1. distance to the previous atom, 2. angle to the previous two atoms, 3. dihedral angle to the previous three atoms. The first three Fm atoms are dummies that help define the rest of the material. Generate a description of the lengths and angles of the lattice vectors and the three dummy Fm atoms, followed by the element type and the three attributes for each atom within the lattice:

`<Z-matrix string>`

Distance Matrix

Below is a description of a bulk material where each atom is described by its element type and distances to the preceding atoms. Generate a description of the lengths and angles of the lattice vectors, followed by the element type and distances for each atom within the lattice, ensuring that each atom solely references distances to preceding atoms, resembling the lower triangular portion of a distance matrix:

`<Distance matrix string>`

SLICES

Below is a description of a bulk material. Generate a SLICES string, which is a text-based representation of a crystal material:

`<SLICES string>`

D TOKENIZATION OF REPRESENTATIONS

In the following calculation, we disregard the prompt heading and constant overheads, such as the lattice parameters of the unit cell.

Fractional For the Fractional representation, each atom is described by its chemical symbol and fractional coordinates rounded to two decimal places. For example:

$$\underbrace{\texttt{Si}}_{1}\quad\underbrace{\texttt{0.00}}_{3}\quad\underbrace{\texttt{0.00}}_{3}\quad\underbrace{\texttt{0.00}}_{3}$$

Using the LLaMA-3 tokenizer, this corresponds to $1 + 3 \times 3 = 10$ tokens per atom. Therefore, for a unit cell containing N atoms, the total number of tokens is $10N$.

Z-matrix For the Z-matrix representation, each atom is described by its chemical symbol, bond length, bond angle and dihedral angle. The bond length is round to the nearest decimal, while the angles are rounded to the nearest integer. For example:

$$\underbrace{\texttt{Si}}_{1}\quad\underbrace{\texttt{1.6}}_{3}\quad\underbrace{\texttt{45}}_{1}\quad\underbrace{\texttt{120}}_{1}$$

Using the LLaMA-3 tokenizer, this corresponds to $1 + 3 + 1 + 1 = 6$ tokens per atom. Therefore, for a unit cell containing N atoms, the total number of tokens is $6N$.

Distance Matrix For the distance matrix representation, each atom is described by its chemical symbol, followed by its distances to the preceding atoms. Here, only the lower triangular part of the symmetric distance matrix is used. For instance, the i-th atom includes $i-1$ distances to the previous $i-1$ atoms. For a structure with N atoms, the total number of distances forms an arithmetic sequence, summing up to $N(N-1)/2$. Since each number with one decimal place requires 3 tokens, the total number of tokens is therefore

$$\underbrace{N}_{\text{chemical symbols}} + \frac{3N(N-1)}{2} - \overbrace{3N}^{\text{start counting from 2nd atom}} = \frac{3N(N-1)}{2} - 2N$$

SLICES For SLICES, the number of edges is determined by the number of neighbors within a 10 Å cutoff radius, as calculated using `pymatgen.analysis.local_env.EconNN` (Ong et al., 2013) based on Hoppe's algorithm (Hoppe, 1979). Thus, the number of neighbors depends on both the cutoff radius and the number of unit cells included in the calculation. Since SLICES considers only first-order neighboring cells, the theoretical upper bound for the number of edges, assuming an infinite cutoff radius, is

$$\left[\overbrace{(3\times 3\times 3)}^{\text{first-order cells}} \underbrace{(N\times N)}_{\text{for each atom}} - \overbrace{N}^{\text{remove self-loops}}\right] \div \underbrace{2}_{\text{remove duplicate edges}} = \frac{27N(N-1)}{2}$$

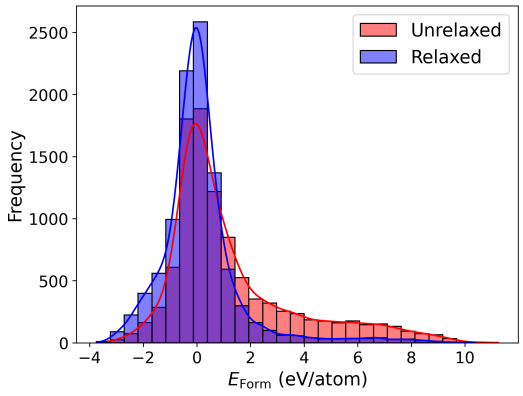
Since each edge description in SLICES can take up to 5 tokens, the total number of tokens is $N + \frac{135N(N-1)}{2}$.

In Table 5, we show the estimated number of tokens per structure for each representation using the training dataset. For SLICES, the token count of 6609.2 is calculated based on the average number of atoms per structure, while the bracketed value of 211.9 represents the token count derived from the average number of edges per structure.

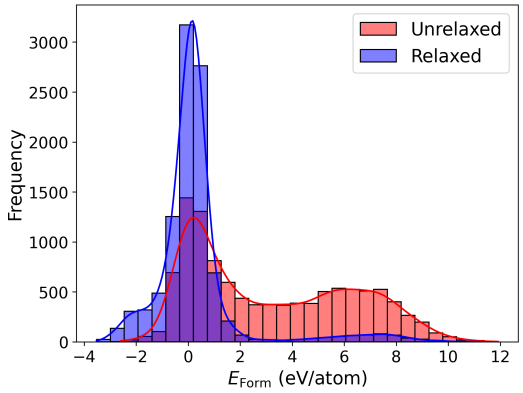
Table 5: Estimated token count per structure for each textual representation based on the training dataset.

	Formula	Training Set		Token Count
		Avg. # atoms	Avg. # edges	
Fractional	$10N$	10.4	–	104
Z-matrix	$6N$	10.4	–	62.4
Distance	$3N(N-1)/2-2N$	10.4	–	125.8
SLICES	$N+135N(N-1)/2$	10.4	40.3	6609.2 (211.9)

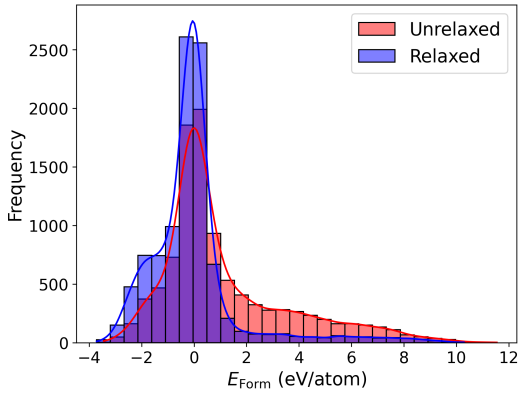
E FORMATION ENERGY DISTRIBUTION



(a) Fractional



(b) Z-matrix

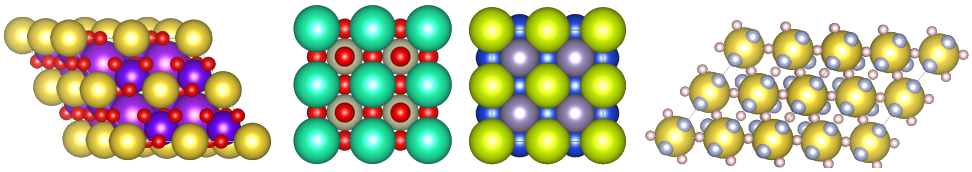


(c) Distance matrix

Figure 3: Distribution of formation energies per atom (eV/atom) for structures generated by LLaMA-3 8B with the Fractional, Z-matrix and distance matrix representations, before (unrelaxed) and after (relaxed) structural optimization. Structural relaxation was performed using MACE (Batatia et al., 2022), and formation energies were calculated using `Orb-v2` (Neumann et al., 2024).

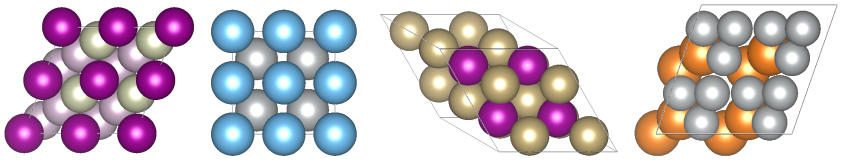
F VISUALIZATION OF SELECTED MATERIALS

In this section, we present periodic materials generated using the four textual representations fine-tuned on LLaMA-3 8B without permutation. For SLICES, the displayed structures are relaxed using MACE, while the other three representations are shown without relaxation. All structures are shown in $2 \times 2 \times 2$ supercells.



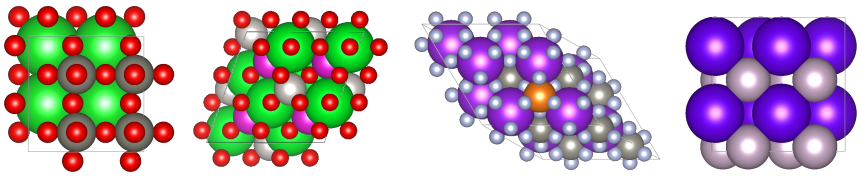
(a) $KNaTb_2O_6$ (b) $LuRhO_3$ (c) $Ce_2Cu_4Sn_2$ (d) NaF_6H_5

Figure 4: Materials generated using the Fractional representation.



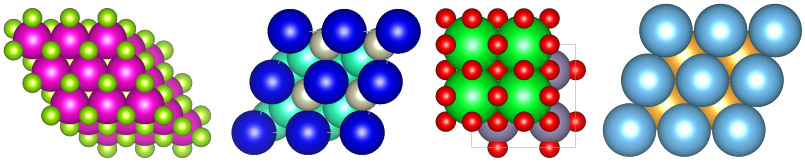
(a) $MnRhTc_2$ (b) $AgTi$ (c) $MnOs_2$ (d) Mg_2Ni_4

Figure 5: Materials generated using the Z-matrix representation.



(a) $BaZnO_3$ (b) $CdPtSr_2O_6$ (c) $MgWK_2F_6$ (d) Tb_2Tc_2

Figure 6: Materials generated using the distance matrix representation.



(a) EuSe (b) Lu_2PmRh (c) $SrSnO_3$ (d) TiAu

Figure 7: Materials generated using the SLICES representation.