MULTI-MODAL TRANSFORMERS FOR PREDICTING THE DENSITY OF STATES IN CRYSTALLINE STRUCTURES

Sudheesh Kumar Ethirajan, Saurabh Sivakumar & Kun-Lin Wu
Department of Chemical Engineering
University of California, Davis
Davis, CA 95616, USA
{sethirajan,sausiva,klwwu}@ucdavis.edu

1 INTRODUCTION

The density of states (DOS) provides essential information about the distribution of electron states across different energy levels in solid materials, impacting properties like conductivity, thermal behaviour, and optical characteristics. DOS effectively quantifies the number of available electronic states at each energy level. Accurately determining the DOS distribution, however, is challenging, especially in complex crystal structures or materials with strong electron-electron interactions. Typical methods to calculate DOS are time-consuming and often computationally intractable for large systems.

Machine learning (ML) offers an alternative approach that avoids the need for extensive experimentation and specialized domain expertise. For example, Chandrasekaran et al.[1] used neural networks to predict DOS by mapping the atomic environment around each grid point to the corresponding electron density and local density of states. These ML-based methods provide a more efficient and tractable solution for predicting DOS, potentially streamlining the material discovery and development process. Recently, Namkyeong et al.[2] introduced a multimodal transformer to integrate information from both the structure of the crystalline material and its energy levels, which in turn focused on predicting the DOS from the obtained representations by reflecting the nature of DOS.

In this project, we aim to integrate heterogeneous information from crystalline materials and energy data using a multimodal transformer for DOS prediction. We guide the model in learning the crystal structure-specific interactions between crystalline materials and energy by using the multimodal transformer,[2,3] prompt tuning,[4,5] and positional encoding.[6,7]. In addition, we compare the performance of `DOSTransformer_phonon` to three other model architectures using similar embeddings such as a `graphnetwork`, a `mlp` and `E3NN`.

2 DATASET

We use a dataset comprised of DOS phonons, a collection of electronic data for crystalline materials. We obtained the raw dataset from the GitHub repo[2] at github.com/HeewoongNoh/DOSTransformer and pre-processed it using information available on the materials project website[8] to include crystal information. This information is essential for the prompt tuning of the `DOSTransformer` as discussed further in the methods section. The dataset is then split into training, validation, and testing sets in an 80/10/10% ratio.

3 METHODS

Here, we evaluated the `DOSTransformer` against several baseline models, including Multi-Layer Perceptron,[9] Graph Neural Networks,[10,11,12,13] and E3NN- Euclidean networks,[14] which they encode the material representations of the atomic positions and bonds for the molecules in the dataset as graphs. We considered Mean squared error (MSE), mean absolute error (MAE), R^2 and the square root of MSE (RMSE) as performance metrics.

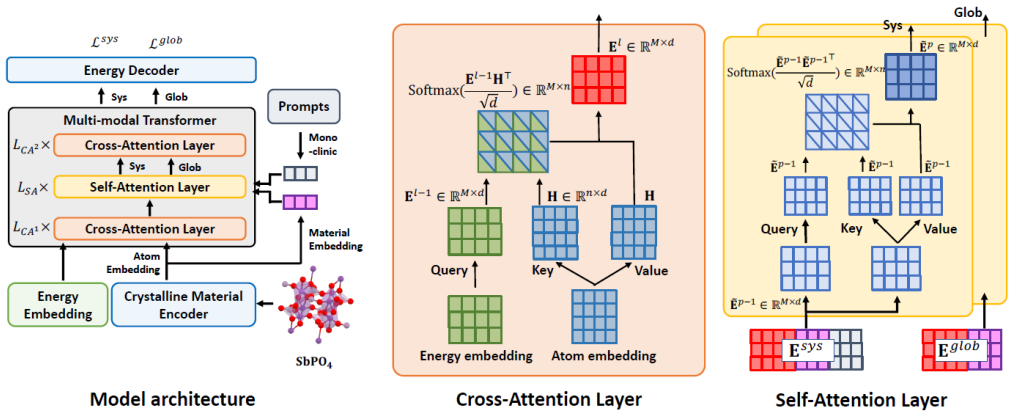


Figure 1: Overall DOSTransformer model architecture

3.1 DOSTransformer

`DOSTransformer` utilizes a multi-modal architecture to analyze both atoms and energy levels in crystals. It employs a multi-head attention mechanism to capture the electronic interactions within a certain radius (Cutoff Radius or R_{max}) (chemical bonds etc.) within the material structure.[15,16]. The code uses a graph network (GNN) to encode atom features and connectivity into embeddings. Energy levels are one-hot encoded and combined with atom embeddings. These combined embeddings enable capturing material-energy relationships. Multi-modal transformers with cross-attention layers integrate the information (signals) from atoms and energies, while global self-attention enhances dependencies between material and energy representations, similar to applications in computer vision and NLP.[17]

The multi-modal transformer being used here has an added self-attention layer after cross-attention to capture unique crystal structures. Seven learnable prompts (Cubic, Hexagonal, etc.) for crystal systems are included in this layer. Extra cross-attention refines material-energy connections considering crystal structure information. `DOSTransformer` leverages both prompt tuning and positional encoding for efficient learning. Prompt tuning, popular in NLP for fine-tuning large language models (LLMs), involves either designing task-specific prompts or training learnable prompts. Inspired by NLP success[4,5], `DOSTransformer` uses continuous prompt tuning to capture the unique properties of crystals, eliminating the need for pre-defined prompts. Positional encoding, traditionally based on a fixed function, is replaced with learnable vector embeddings for each energy value.[7] Similar to how words are positioned in a sentence, energy values guide the model's understanding of material properties. This approach overcomes limitations in the original Transformer architecture. The whole `DOSTransformer` architecture is shown in Figure 1.

3.2 Multi-layer Perceptron

The MLP baseline, a simple neural network, predicts the entire DOS directly from material representations. Two versions are compared: one with energy embeddings (like `DOSTransformer`) and one without. The model lacking energy information might miss subtle connections between material properties and their electronic structures. By outperforming MLP, `DOSTransformer` highlights its strength in capturing energy-dependent features crucial for accurate DOS prediction.

3.3 Graph Neural Network

GNNs, powerful for graph data, depict materials as atom nodes and bond edges. This captures spatial and atomic relationships. Like MLP, two GNN baselines are used for DOS prediction: one with energy embeddings (like `DOSTransformer`) and one without. The model lacking energy details might miss finer points in material properties' influence on electronic structures. By surpassing GNNs, `DOSTransformer` showcases its strength in capturing energy-specific interactions, leading to improved DOS prediction accuracy.

3.4 E(3)NN

E(3)NN, another advanced model, encodes crystals as graphs with atom nodes and edges based on distances (Cutoff Radius or R_{max}). This captures atomic interactions within a certain radius. Unlike `DOSTransformer`, E(3)NN doesn't consider energy levels during training. It uses convolutions to learn from these graphs, predicting DoS from final node features. Comparing `DOSTransformer` to E(3)NN showcases our model's advantage in handling energy-specific information, leading to superior prediction of complex material properties over state-of-the-art methods in predicting complex material properties.

4 RESULTS

4.1 MODEL IMPLEMENTATION

We've significantly enhanced the publicly available `DOSTransformer` code (github.com/HeewoongNoh/DOSTransformer) to facilitate broader research in material property prediction. Firstly, we streamlined the training process by incorporating both early stopping based on validation loss and an exponential learning rate scheduler. Secondly, the code's functionality now extends beyond DOSTransformer. We've implemented and written code for popular architectures like `E3NN`, `graph networks`, and `MLPs` (both with and without energy embeddings) to enable comparative analysis between these models and `DOSTransformer`. DOSTransformer now simplifies the analysis by enabling direct phonon DOS prediction and visualization. Additionally, we've incorporated checkpointing functionality, empowering researchers to pre-train and fine-tune models on a wider range of datasets.

These improvements empower researchers in several ways. They can directly compare the strengths and weaknesses of various architectures for material property prediction tasks. Additionally, the ability to study a broader range of models allows researchers to explore how hyperparameters impact performance across different architectures. This expanded scope might even lead to the discovery of new knowledge about the relationship between model design and prediction accuracy. These enhancements empower researchers to compare and optimize diverse models. Our work goes beyond replicating results, making DOSTransformer more user-friendly and paving the way for more comprehensive material property prediction research.

4.2 ARCHITECTURE PARAMETERS STUDY

We performed experiments by varying the hyperparameters of `Attention Drop`, `Transformer Layers`, `Maximum Radius`, `Number of processing layers` for the `DOSTransformer` architecture, and varied only the latter two parameters for the other architectures keeping all other parameters constant across the models. We trained these models for 500 epochs along with a learning rate decay with an initial learning rate of 0.0001 and early stopping criteria that are checked after 200 epochs and stops after there is no improvement on the RMSE on validation dataset over atleast 200 epochs. The detailed results from all the experiments along with the parameters and the best epoch identified with the early stopping criteria are shown in the appendix tables 3, 4. Note that some of the models show better performance in earlier epochs before 200 on the validation set and thus those are shown.

Shrinking the interaction radius (to 2 Å) hurts prediction accuracy while expanding it (to 8 Å) shows minimal improvement over the middle value (4 Å). This suggests that capturing interactions beyond a certain distance offers diminishing returns since most relevant atomic interactions occur within a specific range. Conversely, a radius that's too small misses crucial interactions, hindering the prediction from the model.

4.3 BEST MODEL PERFORMANCE AND ANALYSIS

The table 1 shows the performance of the best model for each architecture evaluated (best model defined by the RMSE on the valid dataset). The best parameters for the `DOSTransformer` are `Attention Drop` = 0.2, `Transformer Layers` = 4, `Maximum Radius` = 8, `Number of processing layers` = 4, for `E3NN` are `Maximum Radius` = 8,

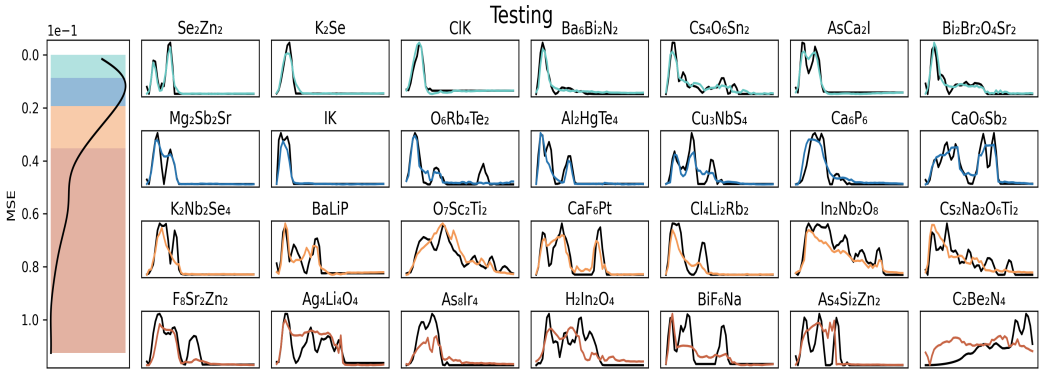


Figure 2: Predictions of DOS on the testing data with the DOSTransformer

`Number of processing layers` = 4, for `Graph Networks` are `Maximum Radius` = 2, `Number of processing layers` = 3 with energy embeddings and `Maximum Radius` = 8, `Number of processing layers` = 4 without and for `MLP` with energy embeddings are `Maximum Radius` = 4, `Number of processing layers` = 4 and `Maximum Radius` = 4, `Number of processing layers` = 4 without. The loss curves for the best model architectures are shown in Figure 3 in the Appendix.

Model	**RMSE**	**MSE**	**MAE**
Energy ✓			
MLP	0.1596	0.0266	0.0955
Graph Network	0.1635	0.0277	0.1102
Energy ×			
MLP	0.1718	0.0311	0.1098
Graph Network	0.1635	0.0277	0.1102
E3NN	0.1817	0.0341	0.1062
DOSTransformer	0.1423	0.0242	0.0911

Table 1: Best model performances

	Training	**Inference**
Model		
Energy ✓		
MLP	0.96	1773.43
Graph Network	1.34	1685.02
Energy ×		
MLP	0.72	2128.63
Graph Network	2.31	1040.60
E3NN	6.79	107.21
DOSTransformer	5.69	174.06

Table 2: Training time per epoch (s) and inference throughput (s^{-1}) for the best models

4.4 Phonon DOS predictions

The figure 2 shows the predictions of Phonon DOS for various crystal structures from the test dataset compared with the actual DOS. It is obvious that there is a good agreement for most of the structures and the MSE is small. However, it's performance on materials with transition metals is quite poor. The predictions could possibly be improved by further hyperparameter optimizations or training on a larger dataset. The training and inference times for the best models are shown in table 2.

5 Conclusion & Future directions

`DOSTransformer` tackles DOS prediction across energies in crystals. It uses attention layers to link material and energy information. Crystal system prompts further enhanced performance for phonon DOS, even in unseen scenarios. However, it cannot be applied to materials with transition metals. This is because current models, including `DOSTransformer`, treat all materials equally, potentially leading to mixed signals. Future work could explore specialized models for each material type, potentially using expert models tailored to specific categories, and including other types of encoding such as auto-encoders. This would ensure robust predictions for all materials, boosting the model's impact and advancing our grasp of material properties.

6 Contributions

- Sudheesh - Ideation, code implementation, running experiments and results visualization.
- Saurabh - Ideation, code implementation, writing report/ PPT and results visualization.
- Kun-Lin - Ideation, bug fixes, result analysis and writing report/ PPT.

Overall, all the authors recognize all contributions as equal.

References

[1] Anand Chandrasekaran et al. "Solving the electronic structure problem with machine learning". In: *npj Computational Materials* 5.1 (2019), p. 22.

[2] Namkyeong Lee et al. "Density of States Prediction of Crystalline Materials via Prompt-guided Multi-Modal Transformer". In: *Advances in Neural Information Processing Systems* 36 (2024).

[3] Peng Xu, Xiatian Zhu, and David A Clifton. "Multimodal learning with transformers: A survey". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).

[4] Brian Lester, Rami Al-Rfou, and Noah Constant. "The power of scale for parameter-efficient prompt tuning". In: *arXiv preprint arXiv:2104.08691* (2021).

[5] Xiao Liu et al. "P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks". In: *arXiv preprint arXiv:2110.07602* (2021).

[6] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[7] Jeremy Howard and Sebastian Ruder. "Universal language model fine-tuning for text classification". In: *arXiv preprint arXiv:1801.06146* (2018).

[8] Anubhav Jain et al. "Commentary: The Materials Project: A materials genome approach to accelerating materials innovation". In: *APL Materials* 1.1 (July 2013), p. 011002. ISSN: 2166-532X. DOI: 10.1063/1.4812323. eprint: https://pubs.aip.org/aip/apm/article-pdf/doi/10.1063/1.4812323/13163869/011002_1_online.pdf. URL: https://doi.org/10.1063/1.4812323.

[9] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[10] Joan Bruna et al. "Spectral networks and locally connected networks on graphs". In: *arXiv preprint arXiv:1312.6203* (2013).

[11] Gabriele Corso et al. "Graph neural networks". In: *Nature Reviews Methods Primers* 4.1 (2024), p. 17.

[12] Zonghan Wu et al. "A comprehensive survey on graph neural networks". In: *IEEE transactions on neural networks and learning systems* 32.1 (2020), pp. 4–24.

[13] Jie Zhou et al. "Graph neural networks: A review of methods and applications". In: *AI open* 1 (2020), pp. 57–81.

[14] Mario Geiger and Tess Smidt. "e3nn: Euclidean neural networks". In: *arXiv preprint arXiv:2207.09453* (2022).

[15] Mozhdeh Gheini, Xiang Ren, and Jonathan May. "Cross-attention is all you need: Adapting pretrained transformers for machine translation". In: *arXiv preprint arXiv:2104.08771* (2021).

[16] Hezheng Lin et al. "Cat: Cross attention in vision transformer". In: *2022 IEEE international conference on multimedia and expo (ICME)*. IEEE. 2022, pp. 1–6.

[17] Md Shamim Hussain, Mohammed J Zaki, and Dharmashankar Subramanian. "Global self-attention as a replacement for graph convolution". In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2022, pp. 655–665.

7 APPENDIX

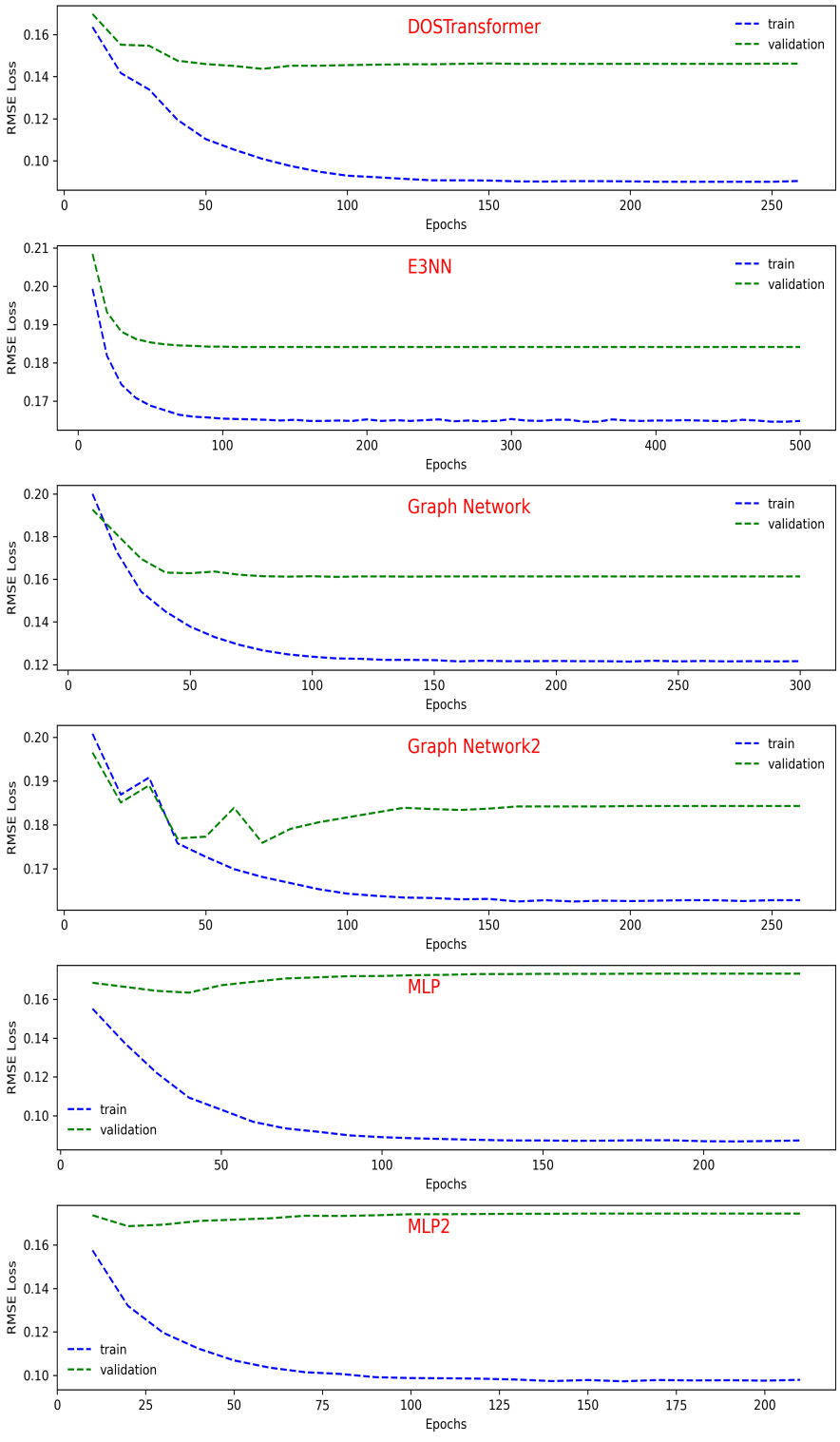
7.1 MODEL METRICS

Model	Cutoff	Layers	Best Epoch	Best RMSE	Best MSE	Best MAE	Best R^2
Energy ✓							
MLP	**4.0**	**4**	**40**	**0.160**	**0.027**	**0.096**	**0.677**
MLP	2.0	3	60	0.160	0.027	0.100	0.673
MLP	4.0	3	60	0.160	0.027	0.100	0.673
MLP	8.0	4	40	0.160	0.027	0.096	0.677
MLP	8.0	3	60	0.160	0.027	0.100	0.673
MLP	2.0	4	40	0.160	0.027	0.096	0.677
Graph Network	4.0	4	130	0.165	0.029	0.111	0.653
Graph Network	**2.0**	**3**	**110**	**0.164**	**0.028**	**0.110**	**0.662**
Graph Network	4.0	3	130	0.170	0.030	0.114	0.637
Graph Network	8.0	4	60	0.168	0.029	0.111	0.648
Graph Network	8.0	3	130	0.166	0.029	0.112	0.649
Graph Network	2.0	4	70	0.165	0.028	0.110	0.656
Energy ×							
MLP2	**4.0**	**4**	**30**	**0.172**	**0.031**	**0.110**	**0.623**
MLP2	2.0	3	30	0.180	0.034	0.117	0.589
MLP2	4.0	3	30	0.180	0.034	0.117	0.589
MLP2	8.0	4	30	0.172	0.031	0.110	0.623
MLP2	8.0	3	30	0.180	0.034	0.117	0.589
MLP2	2.0	4	30	0.172	0.031	0.110	0.623
Graph Network2	4.0	4	50	0.189	0.037	0.121	0.551
Graph Network2	2.0	3	50	0.186	0.035	0.118	0.566
Graph Network2	4.0	3	80	0.190	0.037	0.121	0.545
Graph Network2	**8.0**	**4**	**70**	**0.185**	**0.035**	**0.118**	**0.571**
Graph Network2	8.0	3	50	0.192	0.038	0.120	0.536
Graph Network2	2.0	4	60	0.196	0.040	0.124	0.512
E3NN	4.0	4	500	0.186	0.036	0.109	0.559
E3NN	2.0	3	500	0.211	0.046	0.123	0.438
E3NN	4.0	3	500	0.194	0.039	0.114	0.524
E3NN	**8.0**	**4**	**500**	**0.182**	**0.034**	**0.106**	**0.582**
E3NN	8.0	3	500	0.189	0.037	0.111	0.548
E3NN	2.0	4	500	0.215	0.047	0.126	0.423

Table 3: Model Results for the four baseline models (Best models are in bold)

Model	Cutoff	Layers	Transformer	Attn. Drop	Best Epoch	Best RMSE	Best MSE	Best MAE	Best R2
DOSTransformer	8.0	4	3	0.0	70	0.144	0.025	0.091	0.695
DOSTransformer	8.0	4	4	0.0	80	0.148	0.026	0.093	0.678
DOSTransformer	4.0	3	2	0.1	60	0.149	0.027	0.097	0.674
DOSTransformer	4.0	3	3	0.1	70	0.145	0.025	0.093	0.692
DOSTransformer	2.0	4	2	0.2	60	0.146	0.025	0.094	0.690
DOSTransformer	4.0	3	4	0.1	60	0.146	0.025	0.093	0.691
DOSTransformer	2.0	4	3	0.2	60	0.145	0.025	0.093	0.692
DOSTransformer	2.0	4	2	0.0	50	0.147	0.026	0.094	0.683
DOSTransformer	8.0	3	2	0.0	60	0.148	0.026	0.096	0.680
DOSTransformer	2.0	4	4	0.2	90	0.146	0.025	0.092	0.688
DOSTransformer	8.0	3	3	0.0	60	0.146	0.026	0.093	0.682
DOSTransformer	8.0	3	4	0.0	70	0.144	0.025	0.092	0.698
DOSTransformer	2.0	3	2	0.2	60	0.143	0.025	0.093	0.700
DOSTransformer	2.0	3	3	0.2	100	0.143	0.025	0.091	0.698
DOSTransformer	2.0	3	2	0.0	70	0.148	0.026	0.096	0.678
DOSTransformer	4.0	4	2	0.0	60	0.149	0.027	0.096	0.672
DOSTransformer	2.0	3	4	0.2	60	0.146	0.025	0.091	0.691
DOSTransformer	4.0	4	4	0.2	90	0.145	0.025	0.092	0.692
DOSTransformer	4.0	4	3	0.0	60	0.148	0.027	0.095	0.668
DOSTransformer	4.0	4	4	0.0	50	0.149	0.026	0.096	0.678
DOSTransformer	8.0	4	2	0.2	60	0.144	0.025	0.094	0.699
DOSTransformer	8.0	4	3	0.2	60	0.142	0.024	0.091	0.706
DOSTransformer	8.0	4	2	0.0	50	0.144	0.025	0.094	0.697
DOSTransformer	4.0	3	2	0.0	70	0.151	0.028	0.097	0.658
DOSTransformer	**8.0**	**4**	**4**	**0.2**	**70**	**0.142**	**0.024**	**0.091**	**0.704**
DOSTransformer	4.0	3	4	0.2	70	0.146	0.025	0.094	0.690
DOSTransformer	4.0	3	3	0.0	70	0.148	0.027	0.096	0.673
DOSTransformer	4.0	3	4	0.0	60	0.149	0.026	0.094	0.680
DOSTransformer	2.0	4	3	0.1	70	0.146	0.025	0.092	0.690
DOSTransformer	2.0	4	4	0.1	120	0.145	0.025	0.091	0.691
DOSTransformer	8.0	3	2	0.2	60	0.144	0.025	0.093	0.697
DOSTransformer	8.0	3	3	0.2	50	0.143	0.025	0.091	0.697
DOSTransformer	8.0	3	4	0.2	80	0.144	0.025	0.092	0.697
DOSTransformer	2.0	3	3	0.1	70	0.146	0.025	0.094	0.689
DOSTransformer	2.0	3	4	0.1	60	0.147	0.026	0.092	0.682
DOSTransformer	4.0	4	2	0.2	80	0.146	0.026	0.094	0.686
DOSTransformer	4.0	4	3	0.2	60	0.144	0.025	0.092	0.694
DOSTransformer	8.0	4	2	0.1	70	0.144	0.025	0.092	0.701
DOSTransformer	8.0	4	3	0.1	80	0.143	0.025	0.091	0.701
DOSTransformer	8.0	4	4	0.1	90	0.144	0.025	0.091	0.694
DOSTransformer	4.0	3	2	0.2	60	0.146	0.025	0.095	0.690
DOSTransformer	4.0	3	3	0.2	70	0.144	0.025	0.094	0.695
DOSTransformer	2.0	4	3	0.0	60	0.147	0.026	0.092	0.685
DOSTransformer	2.0	4	4	0.0	50	0.148	0.026	0.095	0.685
DOSTransformer	2.0	4	2	0.1	90	0.146	0.025	0.092	0.688
DOSTransformer	8.0	3	2	0.1	110	0.147	0.026	0.096	0.686
DOSTransformer	8.0	3	3	0.1	50	0.143	0.025	0.092	0.701
DOSTransformer	8.0	3	4	0.1	60	0.145	0.025	0.092	0.697
DOSTransformer	2.0	3	3	0.0	60	0.145	0.025	0.093	0.690
DOSTransformer	2.0	3	4	0.0	60	0.149	0.027	0.095	0.673
DOSTransformer	2.0	3	2	0.1	60	0.144	0.025	0.094	0.698
DOSTransformer	4.0	4	2	0.1	80	0.147	0.026	0.094	0.680
DOSTransformer	4.0	4	3	0.1	70	0.146	0.026	0.094	0.682
DOSTransformer	4.0	4	4	0.1	90	0.145	0.025	0.093	0.689

Table 4: Model Results for the DOSTransformer


DOSTransformer
train
validation
RMSE Loss
Epochs
E3NN
train
validation
RMSE Loss
Epochs
Graph Network
train
validation
RMSE Loss
Epochs
Graph Network2
train
validation
RMSE Loss
Epochs
MLP
train
validation
RMSE Loss
Epochs
MLP2
train
validation
RMSE Loss
Epochs

7.2 Model DOS Predictions

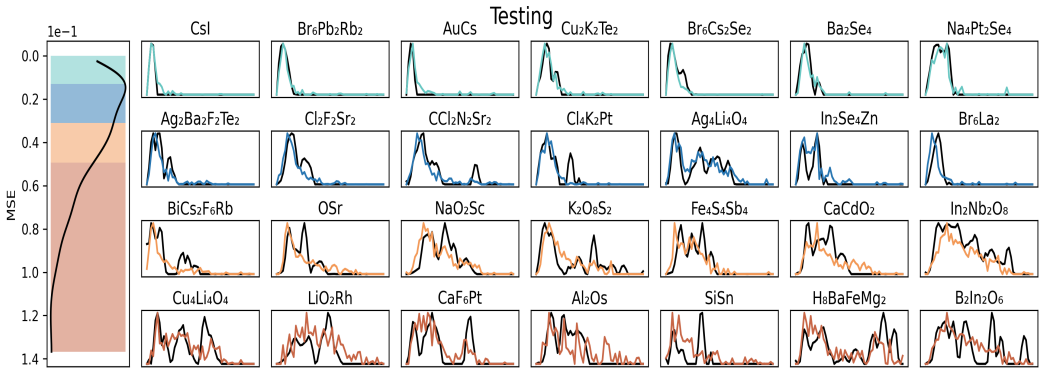


Figure 4: Predictions of DOS on the testing data with the E3NN

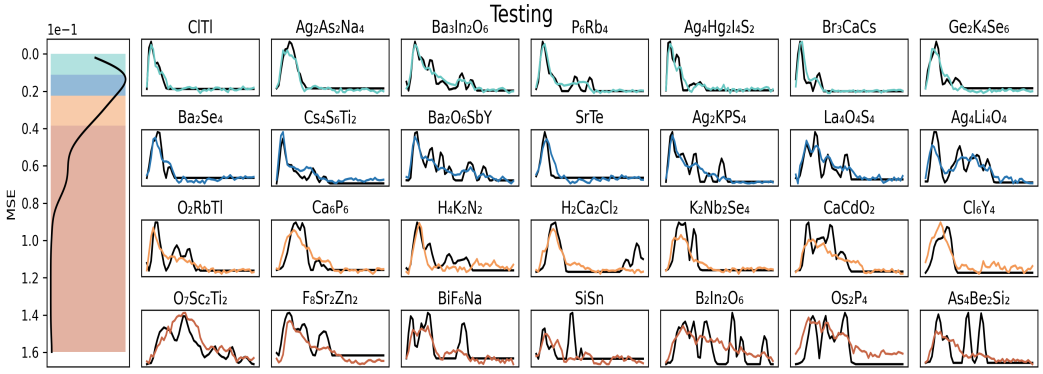


Figure 5: Predictions of DOS on the testing data with the GNN with energy embedding

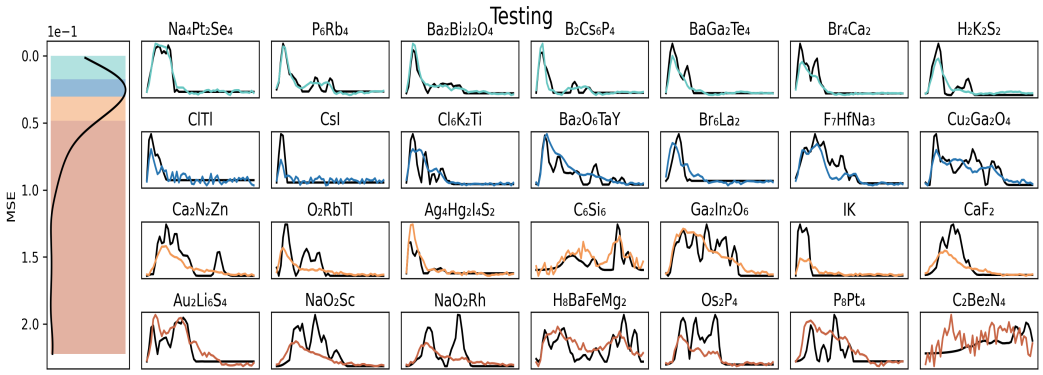


Figure 6: Predictions of DOS on the testing data with the GNN without energy embedding

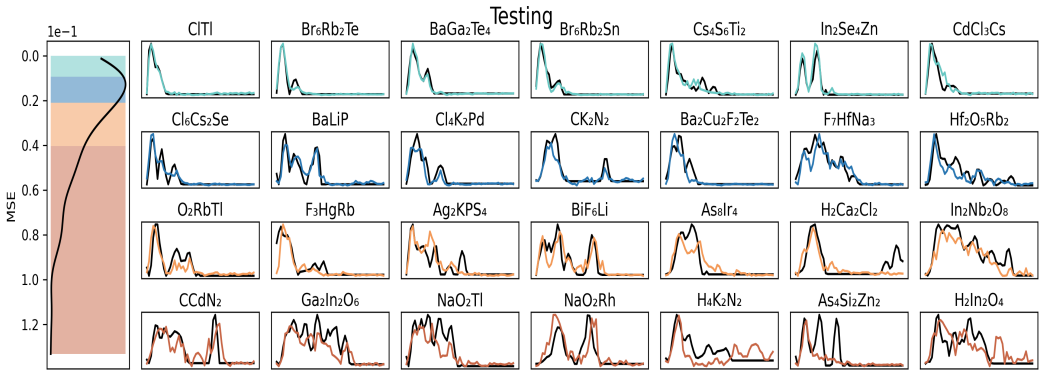


Figure 7: Predictions of DOS on the testing data with the MLP with energy embedding

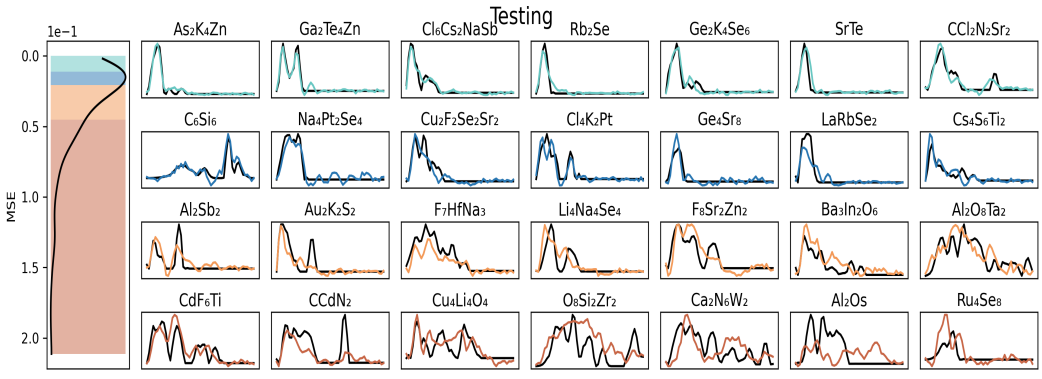


Figure 8: Predictions of DOS on the testing data with the MLP without energy embedding

7.3 DATASET INFO

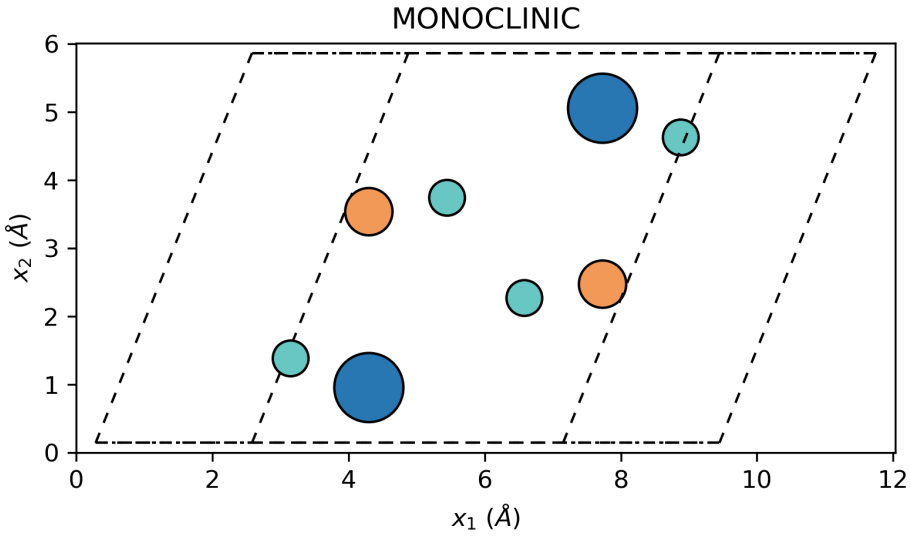


Figure 9: Example of a crystal structure

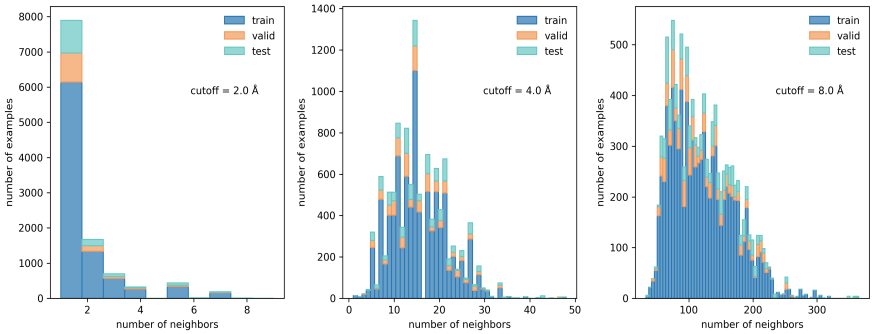


Figure 10: Neighbor Information based on cutoff

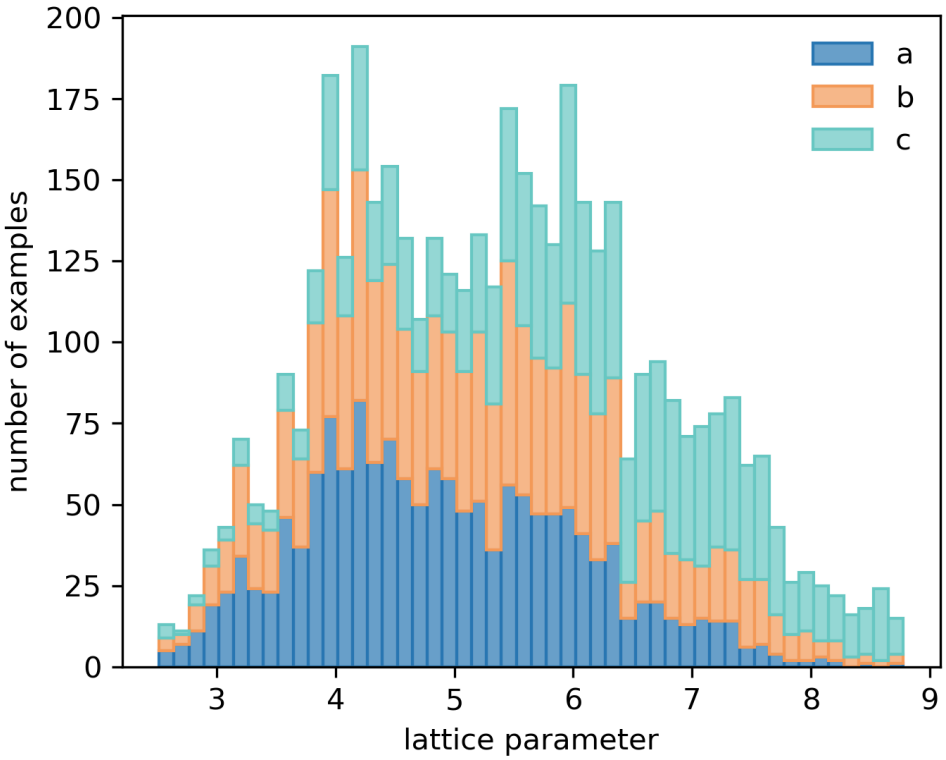


Figure 11: Lattice parameter (quantities specifying the of the periodicity of the atomic arrangement) statistics