Transformer for Metal-Organic Frameworks Property Prediction

Zhonglin Cao
Department of Mechanical Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
zhonglic@andrew.cmu.edu

1 Introduction

Metal-organic frameworks (MOFs) are a type of porous crystalline materials which have been extensively researched during the past several decades [1, 2]. MOFs typically consist of several building blocks including metal based inorganic clusters (metal nodes) and organic linkers [3, 4, 5]. The assembly of those building blocks following certain topologies generates the 2-dimensional or 3-dimensional porous structures of MOFs. Because of the countless combinations of metal nodes, organic linkers and topologies [4, 6], there are enormous amount of MOFs which differentiate with each other in terms of mechanical properties and surface chemistry. Research interests have been induced by the porous structure and versatile nature of MOFs on their potential applications such as gas absorption/storage/separation [7, 5, 8]. With the growing global attention on reducing carbon emission to the environment, investigation on MOFs is gradually becoming popular as MOFs are efficient materials to capture greenhouse gases such as CO_2 in the atmosphere. However, given the enormous variety of possible MOF structures, selecting the ones with highest gas absorption rate can be an expensive and time-consuming task.

Computational methods that can accurately predict the mechanical and chemical properties of MOFs can significantly accelerate the screening of large quantity of MOFs and facilitate the research of them. Besides the traditional methods such as Monte Carlo simulation [8] or molecular dynamics (MD) simulation [9, 10], deep learning models have become increasingly popular recently. The advantage of deep learning model over the simulation methods is that deep learning models can instantly infer the properties of MOFs after training, while the simulation methods requires a computationally expensive rerun for every new MOF. In the last decade, multiple large scale MOF dataset, such as CoRE-MOF-2019 [11] and hypothetical MOFs [8], are released. Those dataset contains the structure of MOFs, their string representations called MOFid, and labels like CO_2 absorption rate. Those public MOF dataset made it possible for researcher to train deep learning models for the prediction of MOF properties. In this project, the goal is to build a transformer encoder model that takes a string representation of MOF structure as input to predict specific properties of MOFs. Experiments will be run to investigate whether or not pretraining will increase the performance of transformer model for this specific task, and how the number of layers and the number of hidden neurons will affect the accuracy. Moreover, t-SNE is used to visualize the latent representation of MOFs learned by the transformer model to examine if the model can distinguish between different MOFs.

2 Background and related works

In some previous works, many researchers attempt to use deep learning to predict the properties of MOFs. The work by Wang et al. [12] utilizes the crystal graph convolutional neural network (CGCNN) [13] to predict methane absorption of MOFs. CGCNN is a very popular model which has an architecture designed specifically for crystalline materials. It takes the optimized 3D coordinates and charges of atoms in the crystalline materials as input and is capable of extracting features that encodes rich chemical information through convolution operations on the crystal graph. However, one drawback for using CGCNN for MOFs property prediction is that obtaining the 3D coordinates and charges of all atoms in a MOF stucture can be computationally expensive. Also, some MOFs consist of many building blocks, rendering their crystal graphs too large for the memory capacity of personal computers.

Other researchers [14, 15] choose to use arbitrary geometrical features such as large cavity diameter (LCD) and pore limiting diameter (PLD) as input to a multilayer perceptrons (MLP) to predict MOFs properties. Although the training of MLP with a few layers can be fast, this method suffers from low accuracy due to the simplicity of network architecture and the naivety of features. Moreover, the selection of features requires extensive domain knowledge from the researchers and optimized 3D structures of MOFs, thus making this method even less generic. Given the drawbacks of the previous trial on using deep learning for MOFs property prediction, a novel method that can achieve high accuracy with simple and generic input of MOFs representations should be pursued.

Enlightened by the fact that all MOFs are combinations of metal nodes, organic linkers and topologies, Bucior et al.[16] invented a string representation of MOFs called MOFid. An ordinary MOFid consists of two important information of a MOF: the chemical information of building blocks, and the topology of the structure. In the example shown in Figure 1, the Cu-BTC MOF is deconstructed into inorganic building block (metal node), organic building block, topology and catenation. Catenation the pattern of the connections between same type of atoms, which is also considered as a topological information in this work. The building blocks are represented by an extensively used string representation of molecules called SMILES [17]. The topology and catenation are each represented by a code from the Reticular Chemistry Structure Resource (RCSR) database [18], which is broadly adopted for framework topologies. Other information in the MOFid are irrelevant to the structural and chemical information of the MOF and thus can be neglected. MOFid is a concise string representation of MOFs that preserves not only the chemical but also the majority of the structural information. The invention of MOFid enables the researchers to apply the state-of-the-art language models for the prediction of MOF properties.

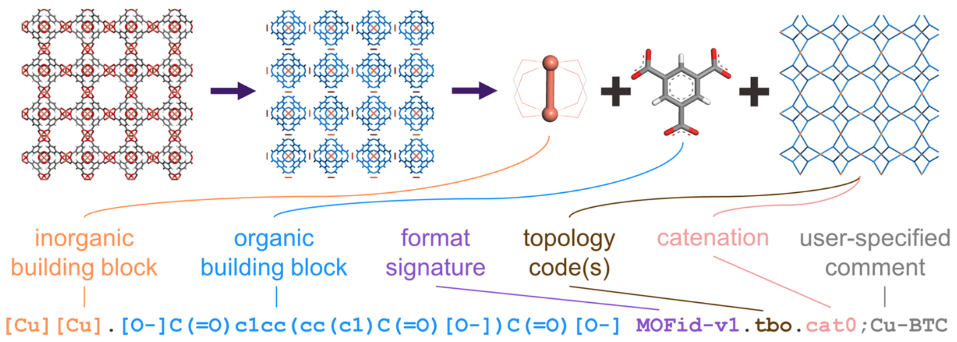

Figure 1: An example of representing a MOF structure with MOFid. This figure is adopted from Bucior et al.[16] Figure 2. In this example, Cu-BTC MOF is deconstructed into inorganic building block (metal node), organic building block, topology and catenation. The building blocks are represented by SMILES. Topology and catenation are each represented by a code.

Transformers as a type of self-attention based models, have rapidly become the top choice for the natural language processing tasks since proposed in 2017 by Vaswani et al[19]. The non-sequential nature of transformers helps to prevent it from suffering long-dependency issue as RNNs do. In addition, its multi-head self-attention mechanism and positional embedding helps it to catch the dependencies between words in a sequence. With advanced pre-training methods such as proposed in BERT [20] and RoBerta [21], transformer and its variants achieved state-of-the-art accuracy in many tasks such as machine translation, an d extend its influence even to image classification [22, 23], and molecular property prediction [24, 25]. The original transformer architecture [19] consist of a encoder and a decoder for the purpose of sequence generation. In this project, since the main focus is to perform regression task with MOFid as input to the model, only the encoder part of the Transformer model will be used.

3 Methods

3.1 Dataset and tokenizer

The dataset used in this project is the hypothetical MOFs (hMOF) created by Wilmer et al. [8] Only those MOFs with MOFid less than 512 characters are used for training and testing to prevent extremely long training time and impractically large GPU memory usage. Repeated MOFid and those without topology information are removed from the dataset. After the data cleaning, there are 102858 MOFs left in the hMOF dataset. The label that will be used for model training is the CO_2 absorption under 0.5 bar of pressure. The label ranges from 0 to 6.7 mol kg^{-1} for the whole dataset. The MOFs in the hMOF dataset has in total 30 different types of topology. The histograms of the label and the types of topology are shown in Figure 2. The probability (y-axis) of the histograms are shown in log scale to show the percentage of the MOFs with rare topologies and high CO_2 absorption. The mean, median and standard deviation of the labels in the dataset is 0.598, 0.332 and 0.682, respectively.

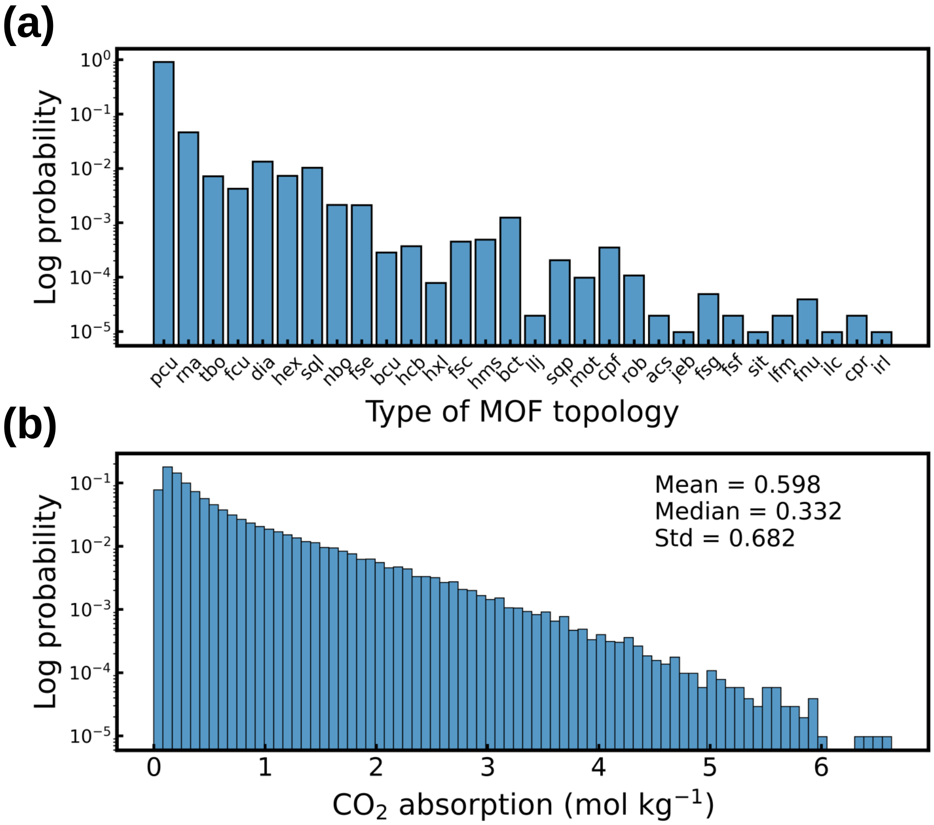

Figure 2: Histogram of the **(a)** topologies and **(b)** label (CO_2 absorption in mol kg^{-1}) of all MOFs in the hypothetical MOF dataset.

The original MOFid is slightly modified to keep only the important information (SMILES of the building blocks, topology and catenation codes). The SMILES of building blocks and topological codes are seperated by a “&&" symbol to help the model to distinguish the two section of MOFid. The modified MOFid needs to be tokenized by a tokenizer before fed into the transformer model. In this project, a MOFid tokenizer is implemented based on the SMILES tokenizer[25] in the DeepChem module[26]. The SMILES of building blocks are tokenized following the manner of [25], while the topological code are tokenized separately. Similar to the BERT model[20], a [CLS] token is added to the beginning of each MOFid, and [PAD] tokens are added to the end of the MOFid if its length is less than 512. The total number of tokens is 4012. The modification of the original MOFid and the tokenization are shown in Figure 3.

3.2 Model and pretraining

The architecture of the Transformer encoder used in this project is the same as originally proposed [19]. The input and output dimension is 512, corresponding to the maximum length of tokenized MOFid. There is a positional encoding layer before the transformer encoder layers. There are N_{layers} layers in the transformer encoder, each having 8 attention head and the hidden layer dimension is N_{hidden}. To investigate the effect of N_{layers} and N_{hidden} to the performance of model, I will test models with N_{layers} = 4 or 6, and N_{hideen} = 512 or 2048. The dropout ratio is set to 0.1.

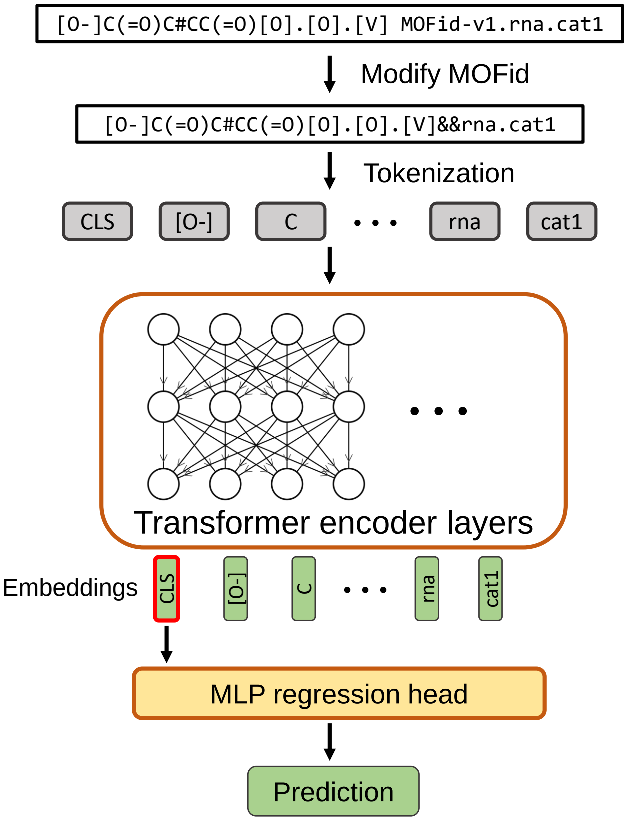
[O-]C(=O)C#CC(=O)[O].[O].[V] MOFid-v1.rna.cat1
Modify MOFid
[O-]C(=O)C#CC(=O)[O].[O].[V]&&rna.cat1
Tokenization
CLS
[O-]
C
• • •
rna
cat1
• • •
Transformer encoder layers
Embeddings
CLS
[O-]
C
• • •
rna
cat1
MLP regression head
Prediction

Figure 3: Pipeline of the model. Each MOFid is firstly modified and then tokenized before fed into the transformer model. In the regression task, the embedding of the [CLS] token is used by the regression head as input to predict the CO_2 absorption of the MOF. During pretraining, however, the embedding of the masked or substituted token will be used to classify what is the original token.

Pretraining of the transformer model using masked language modeling (MLM) [20, 21, 27] is adopted. Besides the fact that MLM is shown to be very effective in NLP tasks, the physical reason behind using MLM in this project is that the relative position of tokens indicates the chemical structure of the MOF building blocks. For example, in Figure 3, the “[O-]" followed by a “C" token represents a carbon connected to an negatively charged oxygen atom in the building block. MLM pretraining can force the transformer model to learn the chemical structure of the building blocks of the MOFs. During the MLM pretraining, 15% of the tokens are randomly selected. 90% of the selected tokens are masked (substituted by a [MASK] token), and 10% of the selected tokens are replaced by another random token. The model will then predict what should be the original tokens given the context. The model is then optimized using softmax cross-entropy loss. Each model are pretrained for 100 epochs using the whole dataset using Adam optimizer with learning rate as 3×10^{-4} based on the suggestion of [20].

After pretraining, the model is then finetuned in a supervised learning fashion using the labeled dataset. 70% of the dataset are used as the training set and the rest 30% are used as the test set. The encoding of the [CLS] token is used as the representation of the whole MOFid to be fed into a MLP regression head for the property prediction. This is a common practice in many works [24, 25, 22] because the global attention in Transformer model makes the embedding of [CLS] a robust representation of the whole sequence. The regression head has 3 hidden layers with 256, 128, and 64 neurons in each layer. The model is finetuned for another 100 epochs using AdamW optimizer with learning rate and and weight decay both as 1×10^{-4}. The architecture of the model in this project is shown in Figure 3.

4 Results and discussion

4.1 Effect of model size on accuracy

The size of the transformer model can significantly affect the model's performance [20]. Larger models generally performs better than smaller models. However, the vanilla transformer model is not memory efficient due to the nature of self-attention mechanism. Therefore, finding a model size that balance the trade-off between accuracy and memory usage is important. In this project, I trained the transformer model with 3 different sizes using the pretraining + finetuning method. The small model consists of 4 transformer layers, each with 512 hidden neurons. The medium model consists of 6 transformer layers, each with 512 hidden neurons (same as the small model). The large model, on the other hand, has 2048 hidden neurons in each layer, but has the same amount of layers compared with the medium model. By comparing prediction errors of the three models, we can evaluate how the number of layers and the number of hidden neurons can affect the accuracy of the models. Based on Table 1, the large model achieves the lowest mean absolute error on both the training and test set. By comparing the performance of the small and medium model, we can conclude that increasing the depth (N_{layers}) of the model can only marginally benefit the performance of the model. However, increasing the width of the model (N_{hidden}) from 512 to 2048 significantly reduced the training MAE and slightly reduced the test MAE. For this specific task, adding more hidden neurons in each layer is more effective than adding more layers to the model. Since there is no previous work using a language model for the prediction of MOF properties, the accuracy comparison is conducted only between different sizes of the transformer model. One limitation that I discovered from the comparison between models is that increasing the size of transformer model does not have significantly positive effect on the test MAE.

	$\mathbf{N_{hidden}}$	$\mathbf{N_{layers}}$	**Train MAE**	**Test MAE**
Small	512	4	0.164	0.253
Medium	512	6	0.161	0.251
Large	2048	6	0.107	0.247

Table 1: Influence of the number of hidden neurons and the number of layers on the prediction error of the transformer model.

4.2 Effect of self-supervised pretraining

Pretraining the transformer model using self-supervised methods such as masked language modeling (MLM) has been proven to benefit the performance of the model in downstream tasks. To quantify the positive effect of pretraining on the CO_2 absorption prediction accuracy of transformer model, I compared the training and test MAE of the large model ($N_{hidden} = 2048$ and $N_{layers} = 6$) with or without pretraining. The pretrained model is finetuned 100 epochs after pretraining using MLM, and the non-pretrained model is simply trained in supervised manner for 100 epochs. The training and testing MAE curves (Figure 4a) are from the finetuning stage of the pretrained model. It is obvious that the pretrained model can achieve a much lower training MAE compared with the non-pretrained model (Figure 4a). The pretrained model also achieved a lower test MAE compared with the non-pretrained model, although the difference is not large. One thing that is noteworthy is that the test MAE of the pretrained model stopped decreasing after approximately 20 epochs of training, indicating a much faster convergence rate compared with the non-pretrained model. 100 epochs of training on pretrained model might caused slight overfitting on the test set.

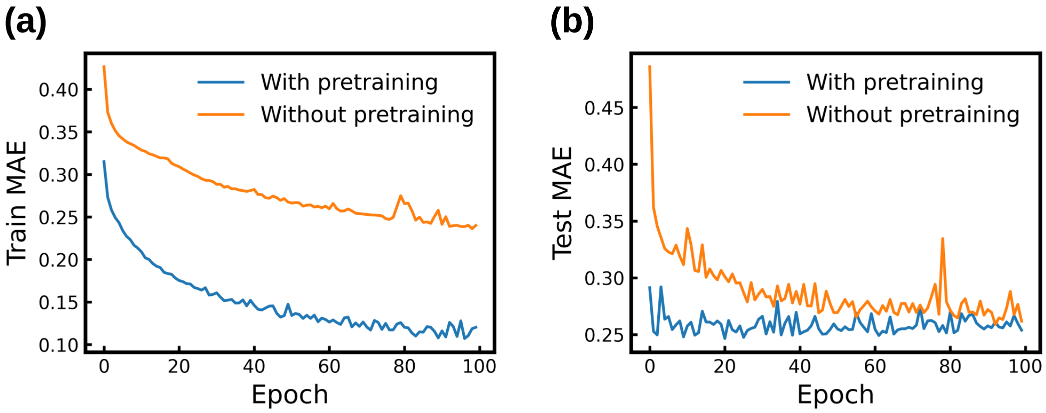

Figure 4: Comparison of the **(a)** training and **(b)** test MAE between the pretrained and non-pretrained large transformer model.

Figure 5 shows the comparison between the predicted value and ground truth on the whole dataset for both the pretrained and non-pretrained model. The black dashed line represents that perfect alignment between prediction and ground truth. Each data point is a MOF structure and is colored based on its topology. Only the MOFs with the top 10 most common topologies are plotted in this figure because the top 10 most common topologies include >99.7% of the whole dataset. The pretrained model achieved a lower MAE of 0.161 compared with 0.230 of the non-pretrained model. The pearson coefficient (R^2) of pretrained model is 0.798, much higher than 0.659 of the non-pretrained model. Pretrained model demonstrate much better accuracy on predicting MOFs with relatively high CO_2 absorption. The scatter points in Figure 5a demonstrate more obvious linear relationship between the prediction and ground truth compared with Figure 5b, indicating that pretrained model has a much better fitting of the data. Combining the train/test MAE curve during training (Figure 4a) and the scatter plot comparison between prediction and ground truth (Figure 4b), we can conclude that the pretraining did have a positive effect on the prediction accuracy of the model, but it does not provide significant benefit on the test accuracy of the model.

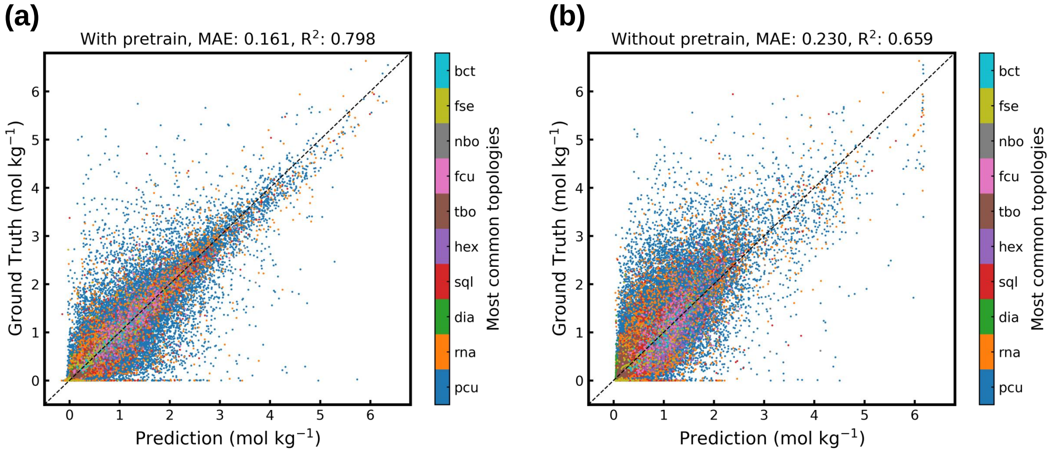

Figure 5: Comparison of the prediction and ground truth CO_2 absorption between the pretrained and non-pretrained model. Each data point is a MOF and is colored based on the topology of the MOF structure. Only the MOFs which have the top 10 most common topologies are shown.

4.3 Visualization of the latent representation of MOFs

Another interesting application of the transformer model is that the embedding learned by it can be used as fingerprint or latent representation of MOFs. Here I extract the embedding of the

"[CLS]" token learned by the pretrained and finetuned large transformer model and use it as a latent representation of each MOF. The representations of all MOFs are then projected into a 2D space using t-distributed stochastic neighbour embedding (t-SNE). t-SNE is designed to preserve local similarity, indicating that similar representations are mapped close to each other (clustered) in 2D space. Assuming that the latent representation of a MOF contains enough information of the MOF structure, it should be clustered with other MOFs that have similar properties (e.g. CO_2 absorption, topology, etc.). The visualization of the representations after t-SNE projection is shown in Figure 6. Each data point is a MOF and it is colored based on either its CO_2 absorption (Figure 6a) or its topology (Figure 6b). Again only the MOFs which have the top 10 most common topologies are shown. We can see that the MOFs with higher CO_2 absorption are clustered at the tail-like region at the bottom of Figure 6a, especially at the tip of the tail. This trend shows that the representations learned by the transformer model can accurately capture the important information related to the CO_2 absorption property of MOFs. Moreover, Figure 6b shows that MOFs of the same topology tend to be clustered together. For example, MOFs with dia topology (green), tbo topology (brown) and hex topology (purple) form three groups at the southeast corner of the panel. MOFs with the rna topology (orange) form multiple smaller clusters across the panel. The clear segregation of MOFs with different topologies indicates that the latent representation, even though it is the embedding of the "[CLS]" token, encodes the topology information of the MOF structure. This is can be attributed to the multi-head global attention mechanism in the transformer model.

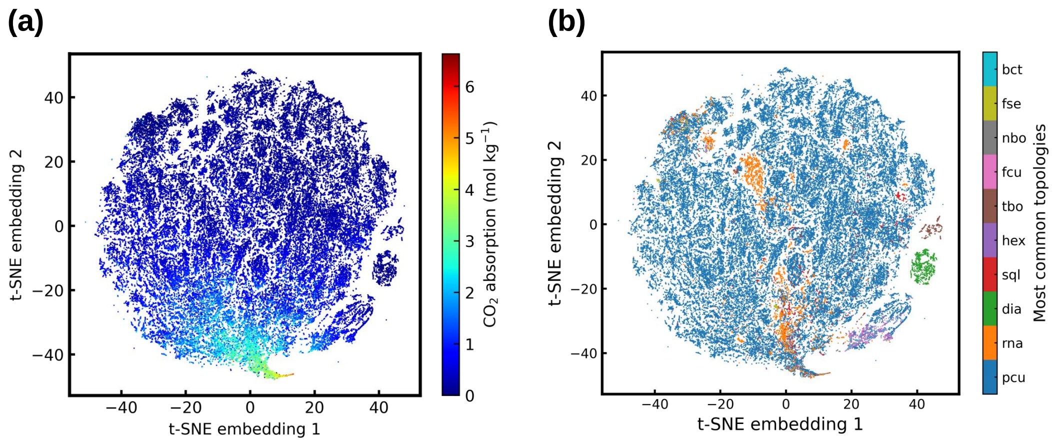

Figure 6: Visualization of the latent representation of MOFs projected to 2D space using t-SNE. Each data point is colored based on either **(a)** the CO_2 absorption or **(b)** topology of the MOF.

5 Conclusion

In this project, I used a transformer encoder model to predict the CO_2 absorption of MOFs. Based on MOFid [16] which is invented to encode the chemical and structural information of MOFs in the format of text string, I customized a tokenizer which convert the MOFid into the input of the transformer model. By comparing the prediction accuracy of three different transformer models, I showed that the large model ($N_{hidden} = 2048$ and $N_{layers} = 6$) has the lowest MAE for both the training and test set. Also, increasing the model depth (N_{layers}) is not as effective as increasing the model width (N_{hidden}) in improving the prediction accuracy of the model. Also, I evaluated the effect of masked language modeling pretraining on the performance of the model. I showed that pretraining can help the model to achieve lower MAE and converge faster in finetuning. One limitation with the transformer model on this specific task is that it cannot reach very low test MAE despite the outstanding accuracy on training set. One reason can be that the transformer is very data hungry, and the dataset used in this project is not large enough to train a model that generalized well on all MOF structures. At last, t-SNE is used to project the latent representation of the MOFs ("[CLS]" token embedding) to a 2D space. The visualization shows that the latent representations of MOFs successfully encode information of MOF properies such as CO_2 absorption and topology. In conclusion, despite the relatively high test MAE, transformer encoder can be used as an effective model for MOF property prediction.

References

[1] Stuart L James. Metal-organic frameworks. *Chemical Society Reviews*, 32(5):276–288, 2003.

[2] Hong-Cai Zhou, Jeffrey R Long, and Omar M Yaghi. Introduction to metal–organic frameworks. *Chemical reviews*, 112(2):673–674, 2012.

[3] Conor H Sharp, Brandon C Bukowski, Hongyu Li, Eric M Johnson, Stefan Ilic, Amanda J Morris, Dilip Gersappe, Randall Q Snurr, and John R Morris. Nanoconfinement and mass transport in metal–organic frameworks. *Chemical Society Reviews*, 2021.

[4] Seyed Mohamad Moosavi, Aditya Nandy, Kevin Maik Jablonka, Daniele Ongari, Jon Paul Janet, Peter G Boyd, Yongjin Lee, Berend Smit, and Heather J Kulik. Understanding the diversity of the metal-organic framework ecosystem. *Nature communications*, 11(1):1–10, 2020.

[5] Peter G Boyd, Arunraj Chidambaram, Enrique García-Díez, Christopher P Ireland, Thomas D Daff, Richard Bounds, Andrzej Gładysiak, Pascal Schouwink, Seyed Mohamad Moosavi, M Mercedes Maroto-Valer, et al. Data-driven design of metal–organic frameworks for wet flue gas co 2 capture. *Nature*, 576(7786):253–256, 2019.

[6] Paolo Falcaro, Anita J Hill, Kate M Nairn, Jacek Jasieniak, James I Mardel, Timothy J Bastow, Sheridan C Mayo, Michele Gimona, Daniel Gomez, Harold J Whitfield, et al. A new method to position and functionalize metal-organic framework crystals. *Nature communications*, 2(1):1–8, 2011.

[7] Alauddin Ahmed, Saona Seth, Justin Purewal, Antek G Wong-Foy, Mike Veenstra, Adam J Matzger, and Donald J Siegel. Exceptional hydrogen storage achieved by screening nearly half a million metal-organic frameworks. *Nature communications*, 10(1):1–9, 2019.

[8] Christopher E Wilmer, Michael Leaf, Chang Yeon Lee, Omar K Farha, Brad G Hauser, Joseph T Hupp, and Randall Q Snurr. Large-scale screening of hypothetical metal–organic frameworks. *Nature chemistry*, 4(2):83–89, 2012.

[9] Aydin Ozcan, Rocio Semino, Guillaume Maurin, and A Ozgur Yazaydin. Modeling of gas transport through polymer/mof interfaces: a microsecond-scale concentration gradient-driven molecular dynamics study. *Chemistry of Materials*, 32(3):1288–1296, 2020.

[10] Zhonglin Cao, Vincent Liu, and Amir Barati Farimani. Water desalination with two-dimensional metal–organic framework membranes. *Nano letters*, 19(12):8638–8643, 2019.

[11] Yongchul G Chung, Emmanuel Haldoupis, Benjamin J Bucior, Maciej Haranczyk, Seulchan Lee, Hongda Zhang, Konstantinos D Vogiatzis, Marija Milisavljevic, Sanliang Ling, Jeffrey S Camp, et al. Advances, updates, and analytics for the computation-ready, experimental metal–organic framework database: Core mof 2019. *Journal of Chemical & Engineering Data*, 64(12):5985–5998, 2019.

[12] Ruihan Wang, Yeshuang Zhong, Leming Bi, Mingli Yang, and Dingguo Xu. Accelerating discovery of metal–organic frameworks for methane adsorption with hierarchical screening and deep learning. *ACS Applied Materials & Interfaces*, 12(47):52797–52807, 2020.

[13] Tian Xie and Jeffrey C Grossman. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Physical review letters*, 120(14):145301, 2018.

[14] Jake Burner, Ludwig Schwiedrzik, Mykhaylo Krykunov, Jun Luo, Peter G Boyd, and Tom K Woo. High-performing deep learning regression models for predicting low-pressure co2 adsorption properties of metal–organic frameworks. *The Journal of Physical Chemistry C*, 124(51):27996–28005, 2020.

[15] Peyman Z Moghadam, Sven MJ Rogge, Aurelia Li, Chun-Man Chow, Jelle Wieme, Noushin Moharrami, Marta Aragones-Anglada, Gareth Conduit, Diego A Gomez-Gualdron, Veronique Van Speybroeck, et al. Structure-mechanical stability relations of metal-organic frameworks via machine learning. *Matter*, 1(1):219–234, 2019.

[16] Benjamin J Bucior, Andrew S Rosen, Maciej Haranczyk, Zhenpeng Yao, Michael E Ziebel, Omar K Farha, Joseph T Hupp, J Ilja Siepmann, Alán Aspuru-Guzik, and Randall Q Snurr. Identification schemes for metal–organic frameworks to enable rapid search and cheminformatics analysis. *Crystal Growth & Design*, 19(11):6682–6697, 2019.

[17] David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

[18] Michael O'Keeffe, Maxim A Peskov, Stuart J Ramsden, and Omar M Yaghi. The reticular chemistry structure resource (rcsr) database of, and symbols for, crystal nets. *Accounts of chemical research*, 41(12):1782–1789, 2008.

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[21] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[22] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[23] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *International Conference on Machine Learning*, pages 4055–4064. PMLR, 2018.

[24] Philippe Schwaller, Teodoro Laino, Théophile Gaudin, Peter Bolgar, Christopher A Hunter, Costas Bekas, and Alpha A Lee. Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. *ACS central science*, 5(9):1572–1583, 2019.

[25] Philippe Schwaller, Daniel Probst, Alain C Vaucher, Vishnu H Nair, David Kreutter, Teodoro Laino, and Jean-Louis Reymond. Mapping the space of chemical reactions using attention-based neural networks. *Nature Machine Intelligence*, 3(2):144–152, 2021.

[26] Bharath Ramsundar, Peter Eastman, Patrick Walters, Vijay Pande, Karl Leswing, and Zhenqin Wu. *Deep Learning for the Life Sciences*. O'Reilly Media, 2019. `https://www.amazon.com/Deep-Learning-Life-Sciences-Microscopy/dp/1492039837`.

[27] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mass: Masked sequence to sequence pre-training for language generation. *arXiv preprint arXiv:1905.02450*, 2019.