MolSiam: Simple Siamese Self-supervised Representation Learning for Small Molecules

Anonymous Author(s)
Affiliation
Address
email

Abstract

We investigate a self-supervised learning technique from the Simple Siamese (SimSiam) Representation Learning framework on 2D molecule graphs. SimSiam does not require negative samples during training, making it 1) more computationally efficient and 2) less vulnerable to faulty negatives compared with contrastive learning. Leveraging unlabeled molecular data, we demonstrate that our approach, MolSiam, effectively captures the underlying features of molecules and shows that those with similar properties tend to cluster in UMAP analysis. By fine-tuning pre-trained MolSiam models, we observe performance improvements across four downstream therapeutic property prediction tasks without training with negative pairs.

1 Introduction

Machine learning (ML) is a rapidly growing field that has significantly contributed to molecular design for drug discovery [1, 2], which is traditionally a complex and time-consuming process. Studies have shown that supervised machine learning algorithms can predict drug efficacy, toxicity, and side effects [3], providing a promising approach to reduce the number of failed drug candidates and lower the cost of development. Deep learning, particularly graph neural networks (GNN), has played a significant role in designing and characterizing small molecule therapeutics [4].

Despite the success of supervised learning in molecular property prediction, obtaining labeled experimental data can be costly and time-consuming. Given the scarcity of labeled data (typically $10^2 - 10^4$ examples per task), supervised learning methods usually face a significant obstacle to learning a generalized representation of the vast chemical landscape [5, 6].

Numerous approaches have been suggested to learn effective molecular representations. In [7, 8], the authors show pre-training on the graph is beneficial for downstream molecular property prediction. [9] reviews the variational autoencoder (VAE) as a tool for representation learning on SMILES strings. [10, 11] utilize SMILES strings with BERT-like [12] pretraining to learn molecular representations. [13] introduce ChemGPT for joint representation learning and generation for molecules using SELFIES. [14] propose a novel geometry-enhanced molecular representation learning method (GEM).

In recent years, self-supervised learning (SSL) with pairwise augmentation has also shown promising results on computer vision tasks [15, 16, 17, 18], as well as for pre-training with applications to graph-structured data [19, 20]. Among them, contrastive learning (CLR) has been explored [19] for learning molecular representations and pre-training for downstream tasks, while other SSL frameworks like Bootstrap Your Own Latent (BYOL) [16] [17] [18] for molecules drug discovery. [21, 22, 23] use contrastive learning on protein sequences and 3D structures. [24] used Barlow twins [25] and SimSiam for material property prediction. In this study, we investigate a self-supervised learning technique from the Simple Siamese (SimSiam) Representation Learning framework on molecular

Submitted to the NeurIPS 2023 Workshop on New Frontiers of AI for Drug Discovery and Development (AI4D3 2023). Do not distribute.

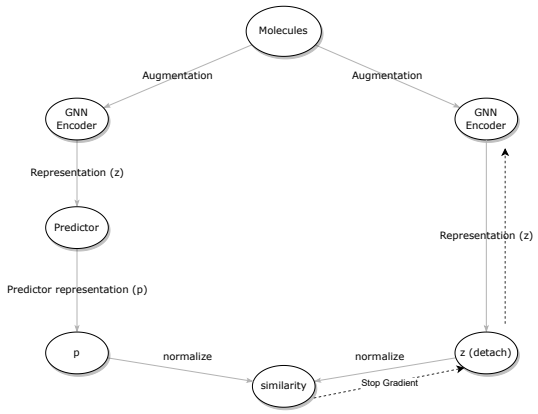

Figure 1: The MolSiam pipeline: we utilize Siamese GNN encoders for pre-training. One of the representations z from the encoder is further passed into the predictor MLP to get the predictor representation p. The loss function is the similarity between p and z while z is detached, which stops the gradient.

2D graphs. We demonstrate that molecular ML models pre-trained with SimSiam, herein called MolSiam, can improve downstream performance on a number of molecular property prediction tasks in drug discovery.

2 Related Work

2.1 Contrastive Learning and MolCLR

SSL is a widely-adopted approach for model pre-training [15, 17, 20, 18, 16]. A recent SSL approach, Molecular Contrastive Learning of Representations (MolCLR), was demonstrated to be effective for improving the performance of 2D GNNs in QM property prediction [19]. Common augmentation tasks in graph processing involve techniques such as subgraph masking and randomly removing nodes or edges. Graphs augmented from a shared source are considered positive pairs, whereas those generated from distinct sources are regarded as negative pairs. The objective of pre-training is to maximize the similarity between positive pairs and minimize the similarity between negative pairs in the embedding space, known as the normalized temperature-scaled cross entropy loss (*NT-Xent loss*) [15].

The contrastive objective may erroneously treat identical or similar augmented graphs from different examples in a dataset as negative pairs. This is particularly relevant in small-molecule design, where new designs may only be slight variations on a common scaffold. This raises the concern that embeddings of very similar graphs may be separated, which goes against the spirit of contrastive learning. For example, [26] found faulty negatives could hurt the performance of downstream tasks in the MolCLR setting and hence incorporate cheminformatics similarities between molecule pairs.

SimCLR, SwAV[17], BYOL [16], and SimSiam are all self-supervised learning algorithms. SimCLR uses contrastive learning techniques to maximize the agreement between augmented views of the same sample and minimize the similarity between negative pairs. BYOL removes the necessity of negative pairs but requires large batch sizes (e.g. 4096) to have a significant effect.

2.2 SimSiam (positive-only non-CLR)

SimSiam provides an elegant way to perform self-supervised learning with only positive pairs and a smaller batch size (e.g., 256), making it an adequate framework for pre-training for most of the use cases without access to substantial computational resources. SimSiam is an easily implemented non-CLR mechanism widely used and studied in computer vision [18]. In both the CLR and SimSiam methods, samples generated from the same data inputs are considered positive pairs, and the model is trained to increase the cosine similarity between their embeddings. However, in SimSiam, no

negative pairs are introduced. A major question of non-CLR methods like SimSiam is how to avoid collapse in representation space without negative samples.

To prevent the representations from collapsing into identical vectors while minimizing the loss, the authors in [18] claim that the use of a `stopgrad` operation with a Projector MLP is crucial (see Section 4). Recent works have demonstrated the potential of using SimSiam on molecular graphs and crystal structures [27, 24, 28]. [29] demonstrate that incorporating SimSiam networks on augmented views of 3D molecular structures increases manifold smoothness during supervised learning. However, the protocol requires 3D point cloud structures, which are not easy to obtain for large unlabeled molecular datasets for representation learning. Therefore, in this work, we study the Simple Siamese (SimSiam) Representation Learning framework on molecular 2D graphs.

3 Data

3.1 Pre-training Dataset

For MolSiam pre-training, we utilized approximately 10 million unique SMILES of unlabeled molecules obtained from PubChem [30, 31]. The molecule graphs were constructed using RDKit [32]. Each node in the molecule graph represents an atom, while each edge represents a chemical bond. The pre-training dataset was randomly divided into a 95:5 ratio for training and validation sets.

3.2 Downstream Dataset

To validate the effectiveness of MolSiam, a handful of datasets were selected from MoleculeNet [33] and the Therapeutic Data Commons (TDC) [34] for evaluation. Below is a brief overview of each dataset, and we encourage the reader to visit the MoleculeNet[1] and TDC[2] websites and original references for more information. All the downstream tasks are binary classification, the loss function for fine-tuning the GNN encoders is binary cross entropy (BCE), and the evaluation metric is `roc_auc_score`.

3.2.1 Pgp

The **Pgp** dataset consists of 1,212 molecules with affinity labels for binding to P-glycoprotein receptors [35].

3.2.2 BACE

The **BACE** dataset provides quantitative IC_{50} and qualitative (binary label) binding results for a set of inhibitors of human beta-secretase 1 (BACE-1). All data are experimental values reported in scientific literature over the past decade, some with detailed crystal structures available. A collection of 1522 compounds is provided, along with the regression labels of IC50.

3.2.3 HIV

The AIDS Antiviral Screen dataset (**HIV**) is a dataset of screens over tens of thousands of compounds for evidence of anti-HIV activity [36]. The available screen results are chemical graph-structured data of these various compounds with experimentally measured abilities to inhibit HIV replication.

3.2.4 Bioavailability

The **Bioavailability** dataset contains 640 drugs in SMILES representation. The dataset records the rate and extent to which the active ingredient or active moiety is absorbed from a drug product and becomes available at the site of action. The task is to predict bioavailability given a drug representation.

[1] https://moleculenet.org/

[2] https://tdcommons.ai/single_pred_tasks/overview/

4 Approach and Methods

There exist several methods for augmentation on graph data [37, 38]. Following `MolCLR`[19], we adopt a mixture of three augmentation strategies: 1) node removal, 2) bond (edge) removal, and 3) subgraph removal. We augment each input with the above transformations to create pairs, which are passed to the GNN encoder for representation learning (*vide infra*).

4.1 GIN encoder

We used Graph Isomorphism Network (GIN) as our encoder network [39, 19]. GINs aim to solve the graph isomorphism problem, which is the task of determining whether two graphs are structurally equivalent. To do this, GINs define a neural network architecture that maps nodes of a graph to a fixed-length vector representation, and the GIN is trained such that isomorphic graphs are mapped to the same representation. This allows the GIN to be used for tasks such as graph classification and graph similarity computation. Unlike other GNN models, which typically use graph convolutional layers to propagate information, GINs use multi-layer perceptrons (MLPs) to update the node representations. The MLPs in a GIN are designed to be permutation-invariant, meaning that they produce the same output regardless of the order of the input elements.

Our GIN has five hidden layers, each of which are followed by batch normalization (BN) and `ReLU` activation. Our hidden and embedding layers are of dimension 512 and 300, respectively, and mean pooling is applied at the output for the GIN encoder.

4.2 Predictor

Our predictor is a two-layer MLP with bottleneck structure that was shown to be crucial to prevent representation collapse [18]. In this work, we kept the format of the predictor the same as in SimSiam. The prediction MLP (h) has BN and ReLu applied to its hidden layers and not to the output layer. The dimension of h's input and output (z and p) is 512, and h's hidden layer's dimension is 256, making h a bottleneck structure.

4.3 Loss function and stop gradient

For pre-training `MolSiam`, the loss for optimization combines symmetrized loss setting on representation z and p. The term z is the direct output of the GNN encoder, and p is the output of h. We applied two augmentations on the same molecule graph to obtain (z_1, z_2) and (p_1, p_2). The loss function is described as follows:

$$\mathcal{L} = \frac{1}{2} D(p_1, z_2) + \frac{1}{2} D(p_2, z_1)\,, \tag{1}$$

$$\mathcal{D}(p, z) = -\left[\frac{p}{\|p\|_2} \cdot \frac{z}{\|z\|_2}\right]. \tag{2}$$

We adopt the `stopgrad` operation on z. We use `Adam` as our pre-training optimizer. The batch size = 512 and we train for 100 epochs with initial learning rate of 0.005 and weight decay of 10^{-5}.

5 Results and Discussion

5.1 Representation Learning

To understand the molecular representations of MolSiam, learned through self-supervised learning, we first obtain the 512-dimensional molecular embeddings for the molecules of interest from the GIN-encoder. Then we use the UMAP algorithm [40] to reduce the dimensionality of the embeddings to a lower-dimensional space. Figure 3 shows the UMAP on the QM8 dataset[33], which has quantum energy labels. We notice that lower energy compounds (second-order approximate coupled-cluster, CC2, indicated by colorbar) tend to cluster together.

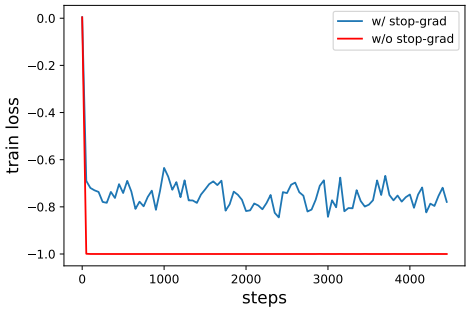

Figure 2: Training loss of the MolSiam during the pre-training stage. Without stop-gradient as the training starts, the loss function asymptomatically approaches -1 as soon as the training starts.

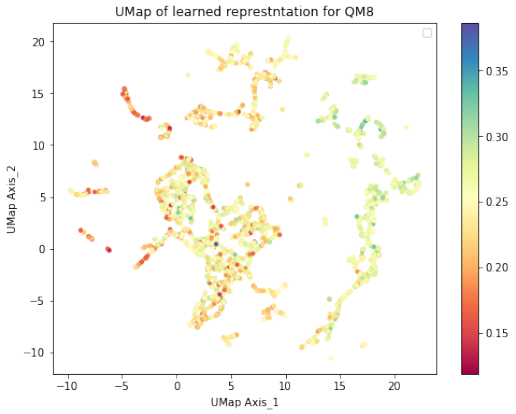

Figure 3: Visualization of molecular representations learned by MolSiam via umap with QM8 dataset.

5.2 Downstream task

As shown in Table 1, we found that pre-training with MolSiam improved performance in three of the four downstream tasks, with the strongest improvement observed for **Pgp**. We also study the difference in training loss function 1 similar to the ablation study done in [18] to understand the impact of training with/without `stopgrad`. As shown in figure 2, without the stop-gradient, the loss collapses immediately.

6 Conclusion and Outlook

We demonstrate `MolSiam` as an efficient and simple way of pre-training representation learners for downstream molecular property prediction. For future work, there are many directions worth exploring, including 1) augmentation strategy, 2) encoder architecture, 3) effect of the bottleneck in prediction MLP, 4) effect of batch size, 5) effect of batch normalization on MLP heads, and 6) effect

Table 1: Downstream task RoC-AUC comparison of pre-trained MolSiam vs supervised-only models

Target	MolSiam	Supervised-only
BACE	**0.8523** $\pm$ **0.013**	0.8427 $\pm$ 0.019
HIV	0.7699 $\pm$ 0.039	**0.7731** $\pm$ **0.037**
Pgp	**0.7354** $\pm$ **0.011**	0.694 $\pm$ 0.026
Bioavailability	**0.6669** $\pm$ **0.025**	0.6595 $\pm$ 0.022

of (a)symmetrized loss hyper parameter. In addition, we plan to compare our approach with different baselines and more downstream tasks in future work.

In conclusion, we present MolSiam as a valuable approach to learning on molecular graphs. It benefits from vast amounts of unannotated chemical data, giving downstream models the potential to generalize to new chemical spaces and making it an attractive option for many applications in chemistry and drug discovery. With the increasing availability of large molecular datasets, self-supervised learning methods, including MolSiam, are likely to play a crucial role in advancing molecular representation learning and property prediction.

References

[1] Suresh Dara, Swetha Dhamercherla, Surender Singh Jadav, CH Madhu Babu, and Mohamed Jawed Ahsan. Machine learning in drug discovery: a review. *Artificial Intelligence Review*, 55(3):1947–1999, 2022.

[2] Antonio Lavecchia. Machine-learning approaches in drug discovery: methods and applications. *Drug discovery today*, 20(3):318–331, 2015.

[3] David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. *Advances in neural information processing systems*, 28, 2015.

[4] Jessica Vamathevan, Dominic Clark, Paul Czodrowski, Ian Dunham, Edgardo Ferran, George Lee, Bin Li, Anant Madabhushi, Parantu Shah, Michaela Spitzer, et al. Applications of machine learning in drug discovery and development. *Nature reviews Drug discovery*, 18(6):463–477, 2019.

[5] Elman Mansimov, Omar Mahmood, Seokho Kang, and Kyunghyun Cho. Molecular geometry prediction using a deep generative graph neural network. *Scientific reports*, 9(1):20381, 2019.

[6] Ying Zhang and Chen Ling. A strategy to apply machine learning to small datasets in materials science. *Npj Computational Materials*, 4(1):25, 2018.

[7] Jun Xia, Yanqiao Zhu, Yuanqi Du, and Stan Z Li. A survey of pretraining on graphs: Taxonomy, methods, and applications. *arXiv preprint arXiv:2202.07893*, 2022.

[8] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for pre-training graph neural networks. *arXiv preprint arXiv:1905.12265*, 2019.

[9] Zaccary Alperstein, Artem Cherkasov, and Jason Tyler Rolfe. All smiles variational autoencoder. *arXiv preprint arXiv:1905.13343*, 2019.

[10] Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. Chemberta: Large-scale self-supervised pretraining for molecular property prediction. *arXiv preprint arXiv:2010.09885*, 2020.

[11] Sheng Wang, Yuzhi Guo, Yuhong Wang, Hongmao Sun, and Junzhou Huang. Smiles-bert: large scale unsupervised pre-training for molecular property prediction. In *Proceedings of the 10th ACM international conference on bioinformatics, computational biology and health informatics*, pages 429–436, 2019.

[12] 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.

[13] Nathan Frey, Ryan Soklaski, Simon Axelrod, Siddharth Samsi, Rafael Gomez-Bombarelli, Connor Coley, and Vijay Gadepally. Neural scaling of deep chemical models. 2022.

[14] Xiaomin Fang, Lihang Liu, Jieqiong Lei, Donglong He, Shanzhuo Zhang, Jingbo Zhou, Fan Wang, Hua Wu, and Haifeng Wang. Geometry-enhanced molecular representation learning for property prediction. *Nature Machine Intelligence*, 4(2):127–134, 2022.

[15] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[16] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[17] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

[18] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15750–15758, 2021.

[19] Yuyang Wang, Jianren Wang, Zhonglin Cao, and Amir Barati Farimani. Molecular contrastive learning of representations via graph neural networks. *Nature Machine Intelligence*, 4(3):279–287, 2022.

[20] Yinan Zhang and Wenyu Chen. Incorporating siamese network structure into graph neural network. In *Journal of Physics: Conference Series*, volume 2171, page 012023. IOP Publishing, 2022.

[21] Pedro Hermosilla and Timo Ropinski. Contrastive representation learning for 3d protein structures. *arXiv preprint arXiv:2205.15675*, 2022.

[22] Michael Heinzinger, Maria Littmann, Ian Sillitoe, Nicola Bordin, Christine Orengo, and Burkhard Rost. Contrastive learning on protein embeddings enlightens midnight zone. *NAR genomics and bioinformatics*, 4(2):lqac043, 2022.

[23] Zuobai Zhang, Minghao Xu, Arian Jamasb, Vijil Chenthamarakshan, Aurelie Lozano, Payel Das, and Jian Tang. Protein representation learning by geometric structure pretraining. *arXiv preprint arXiv:2203.06125*, 2022.

[24] Rishikesh Magar, Yuyang Wang, and Amir Barati Farimani. Crystal twins: self-supervised learning for crystalline material property prediction. *npj Computational Materials*, 8(1):231, 2022.

[25] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320. PMLR, 2021.

[26] Yuyang Wang, Rishikesh Magar, Chen Liang, and Amir Barati Farimani. Improving molecular contrastive learning via faulty negative mitigation and decomposed fragment contrast. *Journal of Chemical Information and Modeling*, 62(11):2713–2725, 2022.

[27] Jinhua Zhu, Yingce Xia, Lijun Wu, Shufang Xie, Tao Qin, Wengang Zhou, Houqiang Li, and Tie-Yan Liu. Unified 2d and 3d pre-training of molecular representations. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2626–2636, 2022.

[28] Teddy Koker, Keegan Quigley, Will Spaeth, Nathan C Frey, and Lin Li. Graph contrastive learning for materials. *arXiv preprint arXiv:2211.13408*, 2022.

[29] Michael Maser, Ji Won Park, Joshua Yao-Yu Lin, Jae Hyeon Lee, Nathan C Frey, and Andrew Martin Watkins. Supsiam: Non-contrastive auxiliary loss for learning from molecular conformers. In *ICLR 2023-Machine Learning for Drug Discovery workshop*, 2023.

[30] Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, et al. Pubchem in 2021: new data content and improved web interfaces. *Nucleic acids research*, 49(D1):D1388–D1395, 2021.

[31] Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, et al. Pubchem 2023 update. *Nucleic Acids Research*, 51(D1):D1373–D1380, 2023.

[32] Greg Landrum. RDKit: Open-source cheminformatics., 3 2022.

[33] Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018.

[34] Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor W Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. *arXiv preprint arXiv:2102.09548*, 2021.

[35] Fabio Broccatelli, Emanuele Carosati, Annalisa Neri, Maria Frosini, Laura Goracci, Tudor I Oprea, and Gabriele Cruciani. A novel approach for predicting p-glycoprotein (abcb1) inhibition using molecular interaction fields. *Journal of medicinal chemistry*, 54(6):1740–1751, 2011.

[36] Kaspar Riesen, Horst Bunke, et al. Iam graph database repository for graph based pattern recognition and machine learning. In *SSPR/SPR*, volume 5342, pages 287–297, 2008.

[37] Omar Mahmood, Elman Mansimov, Richard Bonneau, and Kyunghyun Cho. Masked graph modeling for molecule generation. *Nature communications*, 12(1):3156, 2021.

[38] Kaize Ding, Zhe Xu, Hanghang Tong, and Huan Liu. Data augmentation for deep graph learning: A survey. *ACM SIGKDD Explorations Newsletter*, 24(2):61–77, 2022.

[39] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

[40] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.