Characterizing pre-trained and task-adapted molecular representations

Celia Cintas
IBM Research
Kenya

Payel Das
IBM Research
USA

Jarret Ross
IBM Research
USA

Brian Belgodere
IBM Research
USA

Girmaw Abebe Tadesse
Microsoft AI for Good
Kenya

Vijil Chenthamarakshan
IBM Research
USA

Jannis Born
IBM Research
Switzerland

Skyler Speakman
IBM Research
Kenya

Abstract

Pre-trained deep learning models are emerging fast as a tool for enhancing scientific workflow and accelerating scientific discovery. Representation learning is a fundamental task to study the molecular structure–property relationship, which is then leveraged for predicting the molecular properties or designing new molecules with desired attributes. However, evaluating the emerging "zoo" of pre-trained models for various downstream tasks remains challenging. We propose an unsupervised method to characterize embeddings of pre-trained models through the lens of non-parametric group property-driven subset scanning (SS). We assess its detection capabilities with extensive experiments on diverse molecular benchmarks (ZINC-250K, MOSES, MoleculeNet) across predictive chemical language models (MoLFormer, ChemBERTa) and molecular graph generative models (GraphAF, GCPN). We further evaluate how representations evolve as a result of domain adaptation by finetuning or low-dimensional projection. Experiments reveal notable information condensation in the pre-trained embeddings upon task-specific fine-tuning as well as projection techniques. For example, among the top-120 most-common elements in the embedding (out of $\approx$ 700), only 11 property-driven elements are shared between the three tasks (BACE, BBBP, and HIV), while $\approx$ 70-80 of those are unique to each task. This work provides a post-hoc quality evaluation method for representation learning models and domain adaptation methods that is task and modality-agnostic.

1 Introduction

With the surge of pre-trained deep learning models [1; 2; 3], the question of which representation is best for deployment is gaining importance to ML researchers and practitioners [4; 5]. There exists a wide range of methods for this purpose, such as mutual information between representations and labels [6; 7], feature projection to new target domains [8], mechanistic intepretability [9; 10] and probing [11; 12; 13]. Most of the proposed frameworks for evaluation are heavily dependent on the type of model, tasks, and adaptation technique used. Pre-trained deep learning models are also emerging fast as a tool for enhancing scientific workflow and accelerating scientific discovery. For example, representation learning is a fundamental task to study the molecular structure–property relationship, which is then leveraged for predicting the molecular properties or designing new molecules with desired attributes. Given the complexity of molecular structure-function relationships, a plethora of deep learning-based models have emerged that take in text-based annotations, graphs, and 3D structure as input [14; 15; 16; 17; 18; 19; 20; 21; 22; 23]. Recently, self-supervised learning methods for molecular representation have been employed to address insufficient labeled molecules

Accepted at Unifying Representations in Neural Models Workshop at NeurIPS 2023

and learn a task-agnostic universal representation. The pre-trained molecular models are diverse in nature, vary in size and architecture, are trained using particular self- or un-supervised methods, or are domain-adapted via task-specific finetuning [24; 25; 26; 8]. While these models have shown improvement in performance for generative and predictive benchmarks, the semantics of the learned representations remain opaque. Examples of unsolved questions about learned representations include: Is there any disentanglement in terms of molecular structural and/or functional attributes? How does a pre-trained representation change upon task-specific fine-tuning or feature projection? How to provide a principled evaluation to ensure the trustworthiness of the chosen representation [27; 28; 4]. In this work, we characterize small organic molecular representations to (1) determine which pre-trained representation is more task-optimal (2) evaluate if and to what extent an adaptation method is needed for a pre-trained representation (and which approach among fine-tuning, projection, etc., will work best for a new task) and (3) to enable more fine-grained introspection in molecular generation (e.g., a user might want to generate molecules with a logical combination of multiple properties, e.g., *Scaffold* AND *Molecular weight* ≤ 270 OR ≥ 550 *Dalton* AND *LogP* ≥ 1:3).

2 Preliminaries

Representation learning models Chemical Language Models (CLMs) and Graph Generative Models (GGMs) are two of the most abundant approaches for molecule generation [25; 26; 29] and property prediction [24; 25; 30; 31]. In this work, we showcase the characterization of representation across two models of each category. For CLMs, we selected two encoders capable of producing task-agnostic and fine-tuned molecular embeddings. First, ChemBERTa [24], which is a transformer architecture for molecular property prediction, and second, MoLFormer [30], which uses a transformer-like architecture that is trained on an efficient linear attention mechanism. For GNNs, we consider Graph Convolutional Policy Network (GCPN) [32] and a Flow-based autoregressive model (GraphAF) [33]. GCPN employed a reinforcement learning strategy for molecular graph generation that optimized domain-specific characteristics through policy gradient [32]. GraphAF aimed to exploit the advantages offered by both autoregressive and flow-based approaches to provide enhanced flexibility, efficiency, and improved sampling process to encode domain knowledge [33].

Domain adaptation techniques We consider pre-trained embeddings to be fixed vectors from the encoder obtained during the training using self-supervised learning (e.g SMILES masking) [30]. We then examine two domain adaptation methods: fine-tuning and feature projection. In fine-tuning, some or all weights of a pre-trained model are adjusted to new data from the target domain. Instead, during feature projection, we learn a mapping from a pre-trained source domain to the target domain space, Pro^2 [8] is an interpolation of orthogonal features.

Baselines and evaluation techniques Cintas et al. [34] propose a simpler characterization of inner representations in neural networks for Out-Of-Distribution detection problems. This is a useful baseline because it returns the subset of elements associated with the abnormal pattern detection, in our case, a class belonging to a new task. This allows comparisons regarding the subset of elements found by our proposed evaluation. Tadesse et al. [28] reported characterizing the generation frequency of generative graph models, e.g., characteristics of molecules more or less frequently generated by a model compared to the training set or another model. However, their approach is limited to identifying these characteristics in the output space rather than the embedding subspace.

3 Scanning Over Molecular Representations

Subset scanning has been used to detect anomalous samples in various computer vision and audio tasks [35; 36; 34; 37]. Previously, it was mainly aimed as a detection method of abnormal samples; We believe that extending this type of methodology to characterization and evaluation of the quality of molecular representations has the potential to provide a metric to contrast different domain adaptation methods as well as quality assessment of generative processes. Consider a set of samples from the embedding vectors $X = \{X_1 \cdots X_M\}$ and elements $O = \{O_1 \cdots O_J\}$ generated e.g., by $CLM_{Encoder}$. Where $CLM_{Encoder}$ is a Chemical Language Model Encoder capable of producing task-agnostic and fine-tuned molecular embeddings [30; 24]. Let $X_S \subseteq X$ and $O_S \subseteq O$, we then define the subsets S under consideration to be $S = X_S \times O_S$. The goal is to find the most property-driven subset:$S = \arg\max_S F(S)$, where the function $F(S)$ defines the property-driven

score of a subset of samples from the elements of a given component from $CLM_{Encoder}$ or a Graph Generative Model (GGM). Group subset scanning computes an empirical p-value for each element, as a measurement of how divergent the activation (I^P) or embedding (e) value of a potentially novel sample is at a given element. Group property-driven subset scanning algorithms can be seen in the Appendix A. Group property-driven subset scanning uses non-parametric scan statistics (NPSS) that have been used in other pattern detection methods [38; 39; 34; 36; 35]. There are three steps to using the non-parametric scan statistics on the model's activations and embeddings. **(1) Expectation**: Forming a distribution of "expected" activations at each element (H_0). We generate this distribution by letting the generative process create canonical samples that are known to be from the training data, in the case of GGMs, and the most common samples for CLMs scenarios, sometimes referred to as "background" samples, and record the activations. **(2) Scoring**: Scoring a group of samples in a test set that may contain candidates with a given property or not, this means that the samples can belong to our alternative hypothesis (H_1), which is our case, could be a particular class of a new task or comparing different adaptation techniques. We record the activations induced by the group of test samples and compare them to the baseline activations created in the first step. This comparison results in a p-value for each sample in the test set at each element. **(3) Quantify**: We measure the degree of anomalousness of the resulting p-values by finding X_S and O_S that maximize the NPSS, which estimates how much an observed distribution of p-values deviates from the uniform distribution. More details of NPSS scoring functions can be found in Appendix A.1.

4 Experimental Setup

We have done extensive experiments to validate the generalizability and characterization capabilities of the proposed framework. To this end, we evaluated two groups of models: GGMs and CLMs. Models in each group were evaluated across different domain adaptation techniques and downstream tasks. See more details in Appendix B. We assess the quality of molecular embeddings from different CLMs task-agnostic and fine-tuned representations obtained from MoLFormer [30][1] and ChemBERTa [24][2]. For both CLMs, we used publicly available pre-trained and fine-tuned models. For Pro^2 [8] projection method, we find an optimal number of features for the projection, we did a grid search of 10 to 200 elements for each task. We also employed two autoregressive GGMs to evaluate further the capability of our framework in providing fine-grained control in the generation process. Specifically, we used GCPN [32] and GraphAF [33] as the graph generator. We followed training procedures and pre-trained models detailed in the GT4SD platform [40] for ZINC-250K [41] and MOSES [42]. We chose to scan over a summarization layer (I^P), which concatenates node and edge representation. We evaluate the learned representation on two different tasks. First, the detection of invalid graphs in the learned representation space, and the second task is to identify candidates with a given set of property MPEGO rules [28]; examples can be seen in Fig. 1.

Baseline and Scanning Setup We compare our proposed approach with an existing baseline [34]. Our experiments use elements extracted from task-agnostic and fine-tuned embeddings ($|e| = 768$) generated by MoLFormer [30] and ChemBERTa [24], as well as summarization layers ($|I^P| = 256$) from GCPN [32] and GraphAF [33]. We extract H_0 from a forward pass of the known data through the $CLM_{Encoder}$ or a GGM and record the activations at each element. For the downstream tasks such as HIV, BACE, and BBBP, H_0 contains the most common class, and H_1 will only have the remaining class. During testing, we set 100 randomized runs. In each run, we create test sets with samples from both H_0 and H_1 to assert the detection capabilities. More details are in Appendix B.2.1.

5 Results & Conclusion

Representation in CLMs Embeddings Initially, we compared task-agnostic embeddings from both ChemBERTa [24] and MoLFormer [30], see Table 1. Since MoLFormer embeddings provided the best performance in both downstream tasks, we continue domain adaptation experiments only with this model. We observe the detection power improves when scanning the fine-tuned (FT) embeddings compared to the task-agnostic (TA) version for each of the three binary classification tasks. While the detection power increases in FT embeddings, the cardinality of elements needed to detect a given class is significantly smaller when scanning the FT representation ($\approx$ 130 elements) compared to

[1] https://github.com/IBM/molformer Last accessed 15th May 2023.
[2] https://huggingface.co/seyonec/ChemBERTa-zinc-base-v1 Last accessed 15th May 2023.

Table 1: Averaged Detection Power (AUC) across five runs for tasks under the scenario large ($n > 1000$) and small sample sizes in the target task ($n = 100$). For projection [8], we have the optimal feature vector with 10 features for BBBP and 100 features for BACE. (*) Sample size for BACE task and for BBBP (**).

CLM	Adaptation	Downstream Tasks BBBP	Downstream Tasks BACE	Embedding Dimensions $\|e\|_{BBBP}$	Embedding Dimensions $\|e\|_{BACE}$	n
ChemBERTa [24]	Pre-trained	0.56 ± 0.03	0.58 ± 0.04	768	768	-
MoLFormer [30]	Pre-trained	$\mathbf{0.89 \pm 0.00}$	$\mathbf{0.66 \pm 0.03}$	768	768	-
MoLFormer [30]	Finetuned	0.91 ± 0.05	0.99 ± 0.00	768	768	100
MoLFormer [8]	Projected	0.98 ± 0.01	0.90 ± 0.02	10	100	100
MoLFormer [30]	Finetuned	$\mathbf{1.0 \pm 0.00}$	$\mathbf{0.99 \pm 0.01}$	768	768	1065 , 1400

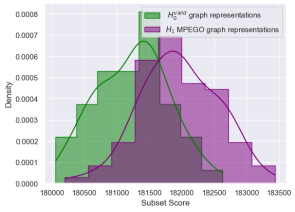
	GCPN [43]	GraphAF [33]
Score Distributions		
MPEGO Ruleset	Scaffold=0 $\cap$ Weight $\leq$ 272.4 $\cap$ LogP $\geq$ 1.3	Ring $\leq$ 3 $\cap$ SAS > 3.1 $\cap$ QED $\leq$ 0.6
Detection Power	0.819	0.969

Figure 1: Score Distributions (SD) for group of elements in I^P that contain representations that have the properties described in MPEGO Ruleset [28].

the TA (≈ 240 elements), which can be a step forward for detecting the subset of elements that are more likely to improve the quality of the representation for a given task via FT (Appendix Figs. 3, 4 and 5). When we look at the most common top-120 elements across all runs and compare them between the tasks, only 11 property-driven elements are shared in the three tasks, while $\approx 70 - 80$ of those are unique to each task. Further, we can see that HIV and BACE, share almost double the elements (27); both of these tasks involve enzyme inhibition, compared to the 14 and 13 nodes share with BBBP, which is associated with a fundamentally different mechanism. When we averaged the precision across 100 test runs for task-agnostic and fine-tuned embeddings for all tasks, we see ≈ 0.23 improvement in the average precision (P). Lastly, we compared in a reduced data scenario when a small amount of target data for domain adaptation is available ($n = 100$ in this experiment); we compared the performance of projection methods and fine-tuning (See Table 1). We can observe that for the BBBP task, the projected embedding improves the Detection Power while requiring less computation and generating a reduced feature space than using the complete finetuned embedding. Nonetheless, we observe that for the BACE task, reduced finetuning is still a better option than projection, hence the need for methods to select the best domain adaptation technique for a given task.

Representation in GGMs The score distributions for the MPEGO rulesets can be seen in Fig. 1. In the case of the MPEGO ruleset, both H_0 and H_1 contain valid representations. Particularly, H_1, contains valid representations with the given ruleset, which makes a more difficult detection problem. Furthermore, we evaluated the impact of the H_0 definition in Appendix Fig. 2. Furthermore, we can observe that the trained I^P from GraphAF, shows a clear discrimination between activations that will generate a given ruleset compared to the rest of the possible properties combination. This confirms, from activation data, a potential bias of the model to be over-generating samples with those properties, as supported by the findings of [28] in the output space.

This study aimed to quantify the relative goodness of pre-trained representations in terms of task-specific information consolidation. Our framework works across models with different architecture, inner representation types, and input features. We propose a non-parametric group property-driven subset scanning to analyze representation learning models and domain adaptation techniques. Currently, the study faces limitations given that all the experiments are done with small molecules and binary classification. Future research will explore the disentanglement of the feature vectors found in fine-tuned and projected embeddings.

References

[1] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[2] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.

[3] Zixuan Ke, Yijia Shao, Haowei Lin, Tatsuya Konishi, Gyuhak Kim, and Bing Liu. Continual learning of language models. *arXiv preprint arXiv:2302.03241*, 2023.

[4] William F Whitney, Min Jae Song, David Brandfonbrener, Jaan Altosaar, and Kyunghyun Cho. Evaluating representations by the complexity of learning low-loss predictors. *arXiv preprint arXiv:2009.07368*, 2020.

[5] Yichu Zhou and Vivek Srikumar. A closer look at how fine-tuning changes bert. *arXiv preprint arXiv:2106.14282*, 2021.

[6] Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. Information-theoretic probing for linguistic structure. *arXiv preprint arXiv:2004.03061*, 2020.

[7] Yann Dubois, Douwe Kiela, David J Schwab, and Ramakrishna Vedantam. Learning optimal representations with the decodable information bottleneck. *Advances in Neural Information Processing Systems*, 33:18674–18690, 2020.

[8] Annie S Chen, Yoonho Lee, Amrith Setlur, Sergey Levine, and Chelsea Finn. Project and probe: Sample-efficient domain adaptation by interpolating orthogonal features. *arXiv preprint arXiv:2302.05441*, 2023.

[9] Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *arXiv preprint arXiv:2211.00593*, 2022.

[10] Siqi Miao, Mia Liu, and Pan Li. Interpretable and generalizable graph learning via stochastic attention mechanism. In *International Conference on Machine Learning*, pages 15524–15543. PMLR, 2022.

[11] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *5th International Conference on Learning Representations (ICLR 2017)*, 2017.

[12] John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. *arXiv preprint arXiv:1909.03368*, 2019.

[13] Yichu Zhou and Vivek Srikumar. Directprobe: Studying representations without classifiers. *arXiv preprint arXiv:2104.05904*, 2021.

[14] Vijil Chenthamarakshan, Payel Das, Samuel Hoffman, Hendrik Strobelt, Inkit Padhi, Kar Wai Lim, Benjamin Hoover, Matteo Manica, Jannis Born, Teodoro Laino, et al. Cogmol: target-specific and selective drug design for covid-19 using deep generative models. *Advances in Neural Information Processing Systems*, 33:4320–4332, 2020.

[15] Samuel C Hoffman, Vijil Chenthamarakshan, Kahini Wadhawan, Pin-Yu Chen, and Payel Das. Optimizing molecules using efficient queries from property evaluations. *Nature Machine Intelligence*, 4(1):21–31, 2022.

[16] Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

[17] Ryan-Rhys Griffiths and José Miguel Hernández-Lobato. Constrained bayesian optimization for automatic chemical design using variational autoencoders. Chemical science, 11(2):577–586, 2020.

[18] Jannis Born, Tien Huynh, Astrid Stroobants, Wendy D. Cornell, and Matteo Manica. Active site sequence representations of human kinases outperform full sequence representations for affinity prediction and inhibitor generation: 3d effects in a 1d model. Journal of Chemical Information and Modeling, 62(2):240–257, 2022. doi: 10.1021/acs.jcim.1c00889. URL https://doi.org/10.1021/acs.jcim.1c00889. PMID: 34905358.

[19] Tomohiro Nakamura, Shinsaku Sakaue, Kaito Fujii, Yu Harabuchi, Satoshi Maeda, and Satoru Iwata. Selecting molecules with diverse structures and properties by maximizing submodular functions of descriptors learned with graph neural networks. Scientific reports, 12(1):1124, 2022.

[20] Fang Wu, Nicolas Courty, Shuting Jin, and Stan Z Li. Improving molecular representation learning with metric learning-enhanced optimal transport. Patterns, 4(4), 2023.

[21] Benjamin Sanchez-Lengeling, Carlos Outeiral, Gabriel L Guimaraes, and Alan Aspuru-Guzik. Optimizing distributions over molecular space. an objective-reinforced generative adversarial network for inverse-design chemistry (organic). 2017.

[22] Alex Zhavoronkov, Yan A Ivanenkov, Alex Aliper, Mark S Veselov, Vladimir A Aladinskiy, Anastasiya V Aladinskaya, Victor A Terentiev, Daniil A Polykovskiy, Maksim D Kuznetsov, Arip Asadulaev, et al. Deep learning enables rapid identification of potent ddr1 kinase inhibitors. Nature biotechnology, 37(9):1038–1040, 2019.

[23] Nathaniel H Park, Matteo Manica, Jannis Born, James L Hedrick, Tim Erdmann, Dmitry Yu Zubarev, Nil Adell-Mill, and Pedro L Arrechea. Artificial intelligence driven design of catalysts and materials for ring opening polymerization using a domain-specific language. Nature Communications, 14(1):3686, 2023.

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

[25] Jannis Born and Matteo Manica. Regression transformer enables concurrent sequence regression and generation for molecular language modelling. Nature Machine Intelligence, pages 1–13, 2023.

[26] Dimitrios Christofidellis, Giorgio Giannone, Jannis Born, Ole Winther, Teodoro Laino, and Matteo Manica. Unifying molecular and textual representations via multi-task language modelling. arXiv preprint arXiv:2301.12586, 2023.

[27] Connor W Coley, Natalie S Eyke, and Klavs F Jensen. Autonomous discovery in the chemical sciences part ii: outlook. Angewandte Chemie International Edition, 59(52):23414–23436, 2020.

[28] Girmaw Abebe Tadesse, Jannis Born, Celia Cintas, William Ogallo, Dmitry Zubarev, Matteo Manica, and Komminist Weldemariam. Domain-agnostic and multi-level evaluation of generative models. arXiv preprint arXiv:2301.08750, 2023.

[29] Andrei Cristian Nica, Moksh Jain, Emmanuel Bengio, Cheng-Hao Liu, Maksym Korablyov, Michael M Bronstein, and Yoshua Bengio. Evaluating generalization in graphnets for molecule design. In ICLR2022 Machine Learning for Drug Discovery, 2022.

[30] Jerret Ross, Brian Belgodere, Vijil Chenthamarakshan, Inkit Padhi, Youssef Mroueh, and Payel Das. Molformer: Large scale chemical language representations capture molecular structure and properties. 2022.

[31] Yu Rong, Yatao Bian, Tingyang Xu, Weiyang Xie, Ying Wei, Wenbing Huang, and Junzhou Huang. Self-supervised graph transformer on large-scale molecular data. Advances in Neural Information Processing Systems, 33:12559–12571, 2020.

[32] Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in neural information processing systems*, 31, 2018.

[33] Chence Shi, Minkai Xu, Zhaocheng Zhu, Weinan Zhang, Ming Zhang, and Jian Tang. Graphaf: a flow-based autoregressive model for molecular graph generation. *arXiv preprint arXiv:2001.09382*, 2020.

[34] Celia Cintas, Skyler Speakman, Victor Akinwande, William Ogallo, Komminist Weldemariam, Srihari Sridharan, and Edward McFowland. Detecting adversarial attacks via subset scanning of autoencoder activations and reconstruction error. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 876–882, 2021.

[35] Victor Akinwande, Celia Cintas, Skyler Speakman, and Srihari Sridharan. Identifying audio adversarial examples via anomalous pattern detection. In *Workshop on Adversarial Learning Methods for Machine Learning and Data Mining, KDD 2020*, 2020.

[36] Celia Cintas, Payel Das, Brian Quanz, Girmaw Abebe Tadesse, Skyler Speakman, and Pin-Yu Chen. Towards creativity characterization of generative models via group-based subset scanning. IJCAI, 2022.

[37] Hannah Kim, Girmaw Abebe Tadesse, Celia Cintas, Skyler Speakman, and Kush Varshney. Out-of-distribution detection in dermatology using input perturbation and subset scanning. In *IEEE 19th International Symposium on Biomedical Imaging*, pages 1–4, 2022. doi: 10.1109/ISBI52829.2022.9761412.

[38] Edward McFowland III, Skyler D Speakman, and Daniel B Neill. Fast generalized subset scan for anomalous pattern detection. *The Journal of Machine Learning Research*, 14(1):1533–1561, Jun 2013.

[39] Feng Chen and Daniel B Neill. Non-parametric scan statistics for event detection and forecasting in heterogeneous social media graphs. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1166–1175, 2014.

[40] Matteo Manica, Jannis Born, Joris Cadow, Dimitrios Christofidellis, Ashish Dave, Dean Clarke, Yves Gaetan Nana Teukam, Giorgio Giannone, Samuel C Hoffman, Matthew Buchan, et al. Accelerating material design with the generative toolkit for scientific discovery. *npj Computational Materials*, 9(1):69, 2023.

[41] John J Irwin, Teague Sterling, Michael M Mysinger, Erin S Bolstad, and Ryan G Coleman. Zinc: a free tool to discover chemistry for biology. *Journal of chemical information and modeling*, 52 (7):1757–1768, 2012.

[42] Daniil Polykovskiy, Alexander Zhebrak, Benjamin Sanchez-Lengeling, Sergey Golovanov, Oktai Tatanov, Stanislav Belyaev, Rauf Kurbanov, Aleksey Artamonov, Vladimir Aladinskiy, Mark Veselov, et al. Molecular sets (moses): a benchmarking platform for molecular generation models. *Frontiers in pharmacology*, 11:1931, 2020.

[43] Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in neural information processing systems*, 31, 2018.

[44] Daniel B Neill. Fast subset scan for spatial pattern detection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(2):337–360, 2012.

[45] Skyler Speakman, Sriram Somanchi, Edward McFowland III, and Daniel B Neill. Penalized fast subset scanning. *Journal of Computational and Graphical Statistics*, 25(2):382–404, 2016.

[46] David Donoho and Jiashun Jin. Higher criticism for detecting sparse heterogeneous mixtures. 2004.

A Property-driven Subset Scanning Algorithm

Group property-driven subset scanning uses an iterative ascent procedure that alternates between two steps: a step identifying the most property-driven subset of molecule samples for a fixed subset of elements, or a step that identifies the converse. In each step, there are $2^M - 1$ possible subsets of samples that need to be reduced to only M while guaranteeing that the highest-scoring subset will be identified. This optimization function, OptimizeCandidates (Algorithm 2) is achieved by the Linear-time Subset Scanning property (LTSS). In OptimizeCandidates, each element is sorted by its priority (SortByProportion), which is its proportion of p-values less than a significance threshold. After elements are sorted, we proceed to score them; we provide more details regarding the scoring in Section A.1. The LTSS property states that the highest-scoring subset will consist of the top-k elements [44, 45]. This drastic reduction in the search space is the key feature that enables subset scanning to scale to large networks, embeddings, and sets of samples.

Algorithm 1: Group property-driven single step over M molecule candidates (or I^P representations) and J elements.

input : $(M \quad J)$ p-values
output : score, X_S, O_S

```
score = -1;
X_S = RandomM(M);
O_S = RandomJ(J);
while score is increasing and not max iterations do
    (M  J') = (M  J)|O_S;
    score, X_S = OptimizeCandidates((M  J'));
    (M'  J) = (M  J)|X_S;
    score, O_S = OptimizeCandidates((J  M'));
return score, X_S, O_S
```

A.1 NPSS score function

The general form of the NPSS score function is

$$F(S) = \max_{\alpha} F_{\alpha}(S) = \max_{\alpha} \alpha(\alpha, N_{\alpha}(S), N(S)) \tag{1}$$

where $N(S)$ is the number of empirical p-values contained in subset S and $N_{\alpha}(S)$ is the number of p-values less than (significance level) $\alpha \in (0, 1)$ contained in subset S. It has been shown that for a subset S consisting of $N(S)$ empirical p-values, it holds $E_H[N_{\alpha}(S)] = \alpha N(S)$ [38]. Group-based subset scanning attempts to find the subset S that shows the most evidence of an observed significantly higher than an expected significance, $N_{\alpha}(S) > \alpha N(S)$, for a significance level α. This work uses the Higher-Criticism test [46] as our scan statistic. This can be interpreted as the test statistic of a Wald test for the amount of significant p-values given that N_{α} is binomially distributed with parameters N and α.

$$p' \frac{|N_{\alpha} - N\alpha|}{\sqrt{N\alpha(1-\alpha)}} \tag{2}$$

Because Higher-Criticism normalizes by the standard deviation of N_{α}, it tends to be more sensitive to small subsets with very extreme p-value ranges as this would produce large values in the numerator and smaller ones in the denominator.

B Experimental Setup Details

All group property-driven subset scanning experiments presented in this work were performed in a desktop machine (2.9 GHz Quad-Core Intel Core i7, 16 GB 2133 MHz LPDDR3). All models were off-the-shelf trained models. Table 2 shows the details of our experiments, including a description of the alternative hypotheses $H_1(\alpha)$ and sample sizes used to build the hypotheses.

Algorithm 2: OptimizeCandidates. Optimizes over fixed subsets of E (M molecule candidates and C (J elements. It returns the max_score and max_subset, the subset that maximizes the score over all possible subsets with different thresholds.

```
input : p-values from all molecule candidates (E) and relevant (C) elements
output: max_score, max_subset
max_score ← −1;
arg_max_subset ← ∅;
for t in LinearSpace(0, 1) do
    sorted_priority ← SortByProportion(E, t) /* Sort E by p-value ratio across C < t */;
    score, subset ← NPSS(sorted_priority, t);
    if score > max_score then
        max_score ← score;
        max_subset ← subset;
return max_score, max_subset
```

B.1 Metrics

We employ the area under the receiver operating characteristic curve (AUC) and precision (Prec) as our performance metrics in both generation and representation analyses. In group property-driven scanning results, AUC can be thought of as detection power, which is the method's ability to distinguish between test sets that contain some proportion of molecule candidates from H_1 and test sets containing only samples from H_0. Prec reflects detection performance, which is the method's ability to label which candidates in the test set belong to H_1.

Table 2: Description of the base models used to extract pre-trained, fine-tuned, and projected embeddings from CLMs and summarization layers from GGMs, type of the alternative hypothesis defined in each experiment (H_1), the number (#) of samples used to build the null hypothesis (H_0), and to test H_1 in the different scenarios.

Base Model	Adaptation	# for H_0	# for H_1	H_1	Dataset
MoLFormer [30]	Pre-trained	822	691	Class 1	BACE
	Pre-trained	1567	483	Class 0	BBBP
	Pre-trained	3983	130	Class 1	HIV
	Finetuned	822	691	Class 1	BACE
	Finetuned	1567	483	Class 0	BBBP
	Finetuned	3983	130	Class 1	HIV
	Projected	767	646	Class 1	BACE
	Projected	1497	453	Class 0	BBBP
ChemBERTa [24]	Pre-trained	822	691	Class 1	BACE
	Pre-trained	1567	483	Class 0	BBBP
	Projected	767	646	Class 1	BACE
	Projected	1497	453	Class 0	BBBP
GraphAF [33]	Pre-trained	10000	1168	Valid	ZINC250K
	Pre-trained	10000	1653	Valid	MOSES
	Pre-trained	4000	408	MPEGO	ZINC250k
GCPN [43]	Pre-trained	8000	567	Valid	ZINC250K
	Pre-trained	4000	227	MPEGO	ZINC250K
	Pre-trained	8000	490	Valid	MOSES

B.2 Representation Analysis in Graph Generative Models

We employed two autoregressive Graph Generative Models (GGMs) to further evaluate the capability of our framework in providing fine-grained control in the generation process. Specifically, we used two GGMs: Graph Convolutional Policy Network (GCPN) [32] and a Flow-based Autoregressive (GraphAF) [33] as the graph generator. GCPN employed a reinforcement learning strategy for molecular graph generation that optimized domain-specific characteristics through policy gradient [32]

GraphAF aimed to exploit the advantages offered by both autoregressive and flow-based approaches in order to provide enhanced flexibility, efficiency, and improved sampling process to encode domain knowledge [33]. In our experimental setup, we followed similar implementation, training procedures, and pre-trained models detailed in the GT4SD platform [40].

To this end, both GCPN and GraphAF were trained on the publicly available ZINC-250K [41][3] dataset, which contains $249,455$ small molecules. Additionally, we used a refined version of ZINC[4] molecules, as proposed by the benchmark platform MOSES [42], which undergoes filtering by certain parameters, such as molecular weight ranges and the number of rotatable bonds, among others. For both GCPN and GraphAF models, we chose to scan over a summarization layer (l^P_y), which concatenates node and edge representation. In this case, we evaluate the learned representation on two different tasks. First, the detection of invalid graphs in the learned representation space (l^P_y), and the second task, is to identify candidates with a given set of rules generated by MPEGO [28]. An example of Rulesets generated by MPEGO for each graph generative model and dataset can be seen in Fig. 1. These sets of properties correspond to molecules being generated with higher or lower frequencies.

B.2.1 Baseline and Scanning Setup in GGMs

In the validity test, H_0 is designed to contain only invalid graph representations; these are generated at a higher rate in both generative graph generative models. In the MPEGO ruleset case, H_0 will contain valid representations that do not contain properties found in the ruleset. The H_1 distributions are built from a forward pass with only samples belonging to a given class, valid representations (for the validity use-case), and representations that generate molecules with a given set of properties for the MPEGO use-case.

B.2.2 Change of Expectation H_0 for MPEGO rulesets

We can observe that the detection of power reduces compared to the validity task; this is partially because the expectation for the validity task is clearly divided from our alternative hypothesis, i.e., H_0 corresponds to invalid graph representations, while H_1 contains only valid graph representations. Furthermore, we consider two H_0 for the experiments for MPEGO rulesets [28] (See Fig. 2). The first one will be to have a mix of valid and invalid representations of molecules generated by each model, which we observe more discrimination as MPEGO rules will never have invalid graph representations. And the second, more realistic H_0 will only contain valid graph representations as appears in the final experiments.

C Run-time benchmark

Scalability is often an issue in most existing evaluation frameworks, particularly in the molecular space, when the group of samples is large. After all, there are exponentially many subsets with respect to the group size. To this end, we utilize linear-time subset scanning property that helps to scan across samples in linear time via its ranking function. In Table 3, we can see the execution time for subset scanning under a summarization layer (l^P_y) from both GraphAF and GCPN as well embeddings from MoLFormer and ChemBERTa. We performed 100 runs with the High Criticism score function for multiple molecule candidates for the evaluation. The tests were performed in a desktop machine (2.9 GHz Quad-Core Intel Core i7, 16 GB 2133 MHz LPDDR3) under Linux 4.15.0-139-generic operating system.

D Extended Results

D.1 Group property-driven subset scanning over fine-tuned and task-agnostic MoLFormer

In Fig. 3, we observe that for the BBBP task, the detection power for pre-trained embeddings is 0.89 and 1.0 for fine-tuned representation with all target datasets (train test splits following [30]). Similarly, for HIV task, pre-trained embeddings are 0.98 and 1.0 for fine-tuned representation. This

[3] https://www.kaggle.com/datasets/basu369victor/zinc250k Last Accessed 15th May 2023.
[4] https://zinc.docking.org/

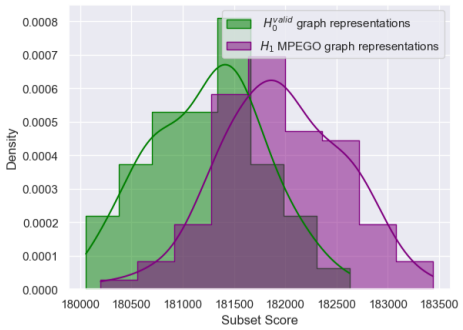
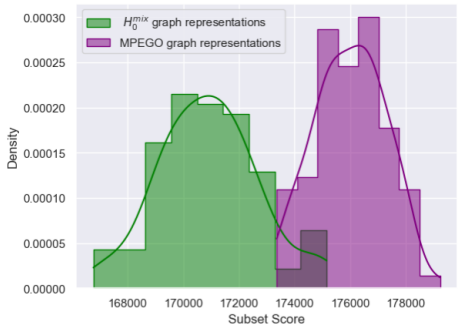
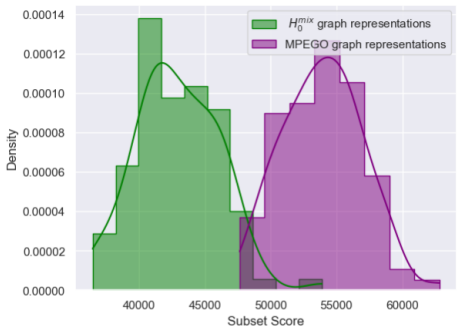
Model	Detection Power (↑)	Scores Distributions in I^P	H_0
GCPN [43]	0.819		H_0^{valid}
GCPN [43]	0.984		H_0^{mix}
GraphAF [33]	0.969		H_0^{valid}
GraphAF [33]	0.991		H_0^{mix}

Figure 2: Scores distributions for different rulesets obtained from MPEGO [28].

can imply that fine-tuned representations have higher discriminative power for the given task, see Table 1 for more comparisons. Further, the cardinality of identified elements needed to detect a given sample type in the embeddings reduces to half. See Fig. 4 for more details regarding node identification.

Table 3: Run-time benchmark. Scan time involves p-value calculation and scanning process for both evaluation samples. Total time measures the complete pipeline from activation extraction to output metrics recording and visualization.

Model	task	scan time (secs)	Total time (secs)
ChemBERTa [24]	BACE	$10{:}75 \pm 0{:}06$	$18{:}41 \pm 0{:}89$
MoLFormer [30]	BACE	$14{:}03 \pm 0{:}49$	$20{:}75 \pm 1{:}11$
GCPN [43]	Invalid	$7{:}69 \pm 0{:}23$	$14{:}3 \pm 0{:}21$
GraphAF [33]	Invalid	$7{:}68 \pm 0{:}03$	$6{:}99 \pm 0{:}40$

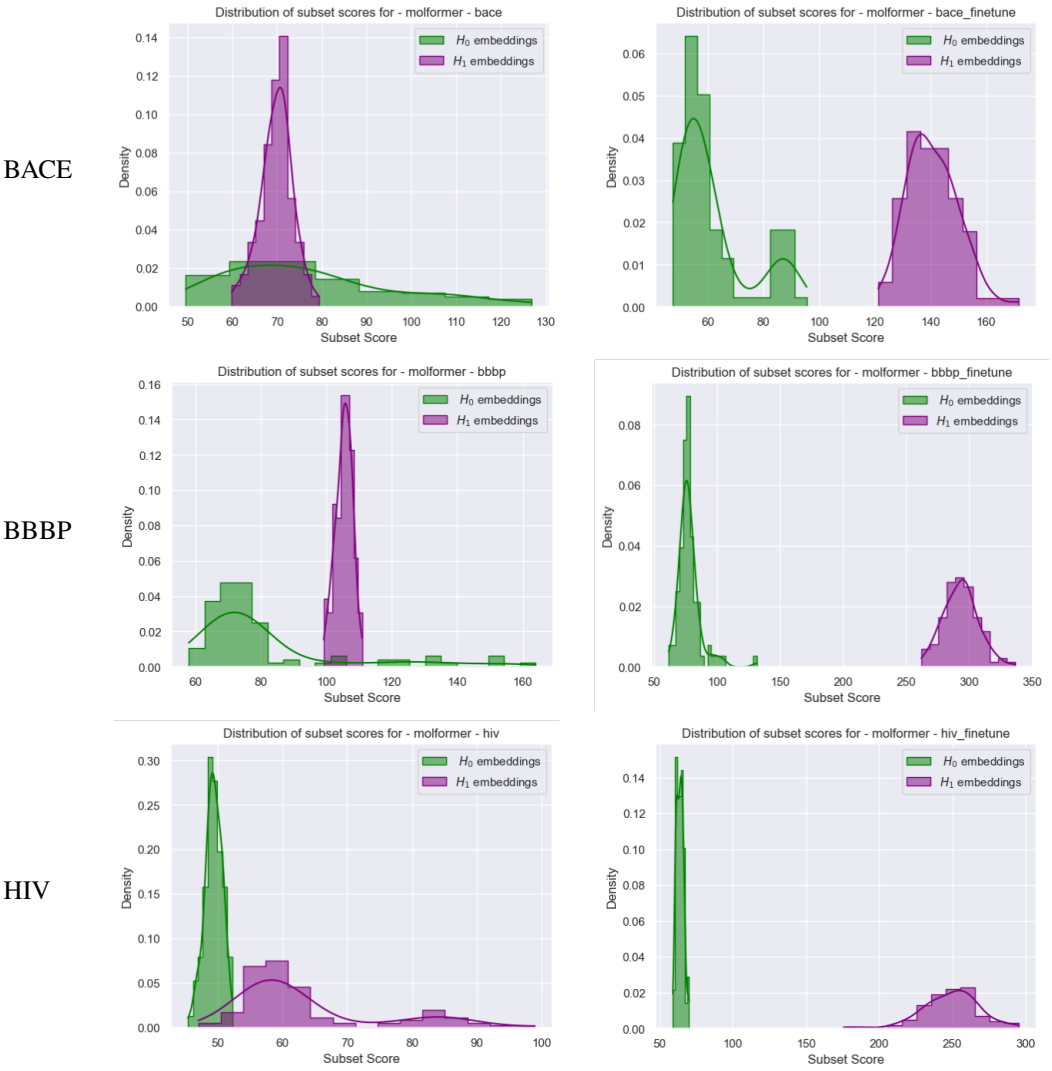


Figure 3: Score distributions for group property-driven subset scanning over fine-tuned and task-agnostic MoLFormer embeddings [30].

D.2 Identified candidates

A random example of an extracted group of molecule candidates that were identified by the same group of elements in the task-agnostic embedding to belong to class 1 under BBBP task with a precision of $P = 0{:}83$ can be seen in Fig. 6.

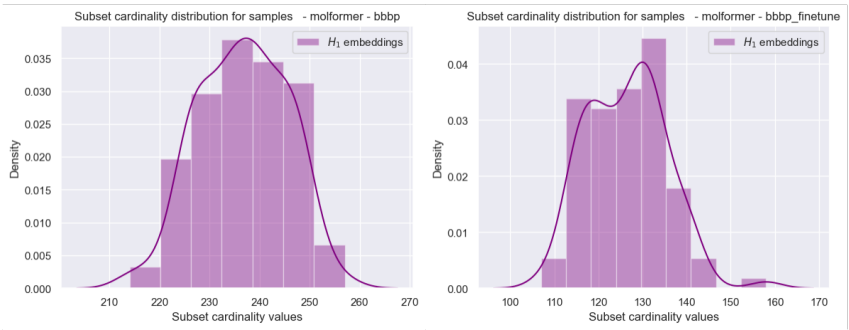


Figure 4: Cardinality distribution of detected elements for BBBP with task-agnostic and fine-tune embeddings.

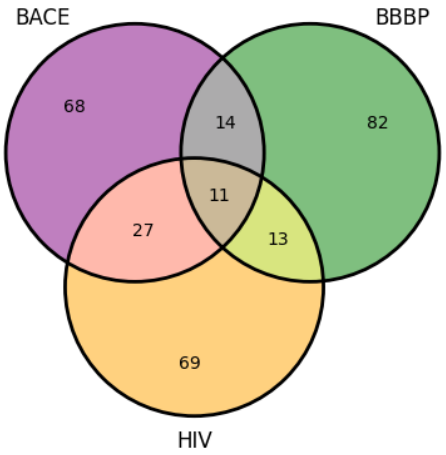


Figure 5: When we look at the most common top−120 elements across all runs and compare them between the tasks, only 11 property-driven elements are shared in the three tasks, while between 70 to 80 of those are unique to each task. We can also observe that HIV and BACE, share almost double the elements (27); both of these tasks involve enzyme inhibition, compared to the 13 and 14 nodes share with BBBP, respectively, which is associated with a fundamentally different mechanism.

Table 4: Detection power (AUC) across two different Generative Models (GGMs) GraphAF [33] and GCPN [32] trained in two datasets, ZINC250k and MOSES. The first H_1 corresponds to finding the graph inner representation of valid molecules, and the second H_1 corresponds to a group of molecules with a set of chemistry properties found in [28]. NA: a ruleset is not available for that dataset.

		Invalid Molecules		MPEGO Rulesets	
GGM	Dataset	SS (Ours)	Cintas et al.[34]	SS (Ours)	Cintas et al.[34]
GraphAF [33]	ZINC250K	**0.984**	0.639	**0.969**	0.596
	MOSES	**0.998**	0.768	NA	NA
GCPN [43]	ZINC250K	**0.878**	0.713	**0.819**	0.369
	MOSES	**0.996**	0.644	NA	NA

D.3 Group property-driven subset scanning over GGMs

Table 4 shows the detection power of our approach for two different tasks across generative models and datasets. For all cases, we observe that group-based scanning yields higher detection power than the baseline [34]. We hypothesize this is thanks to the unique ability to identify anomalous elements across a group of candidate molecules to detect different tasks.

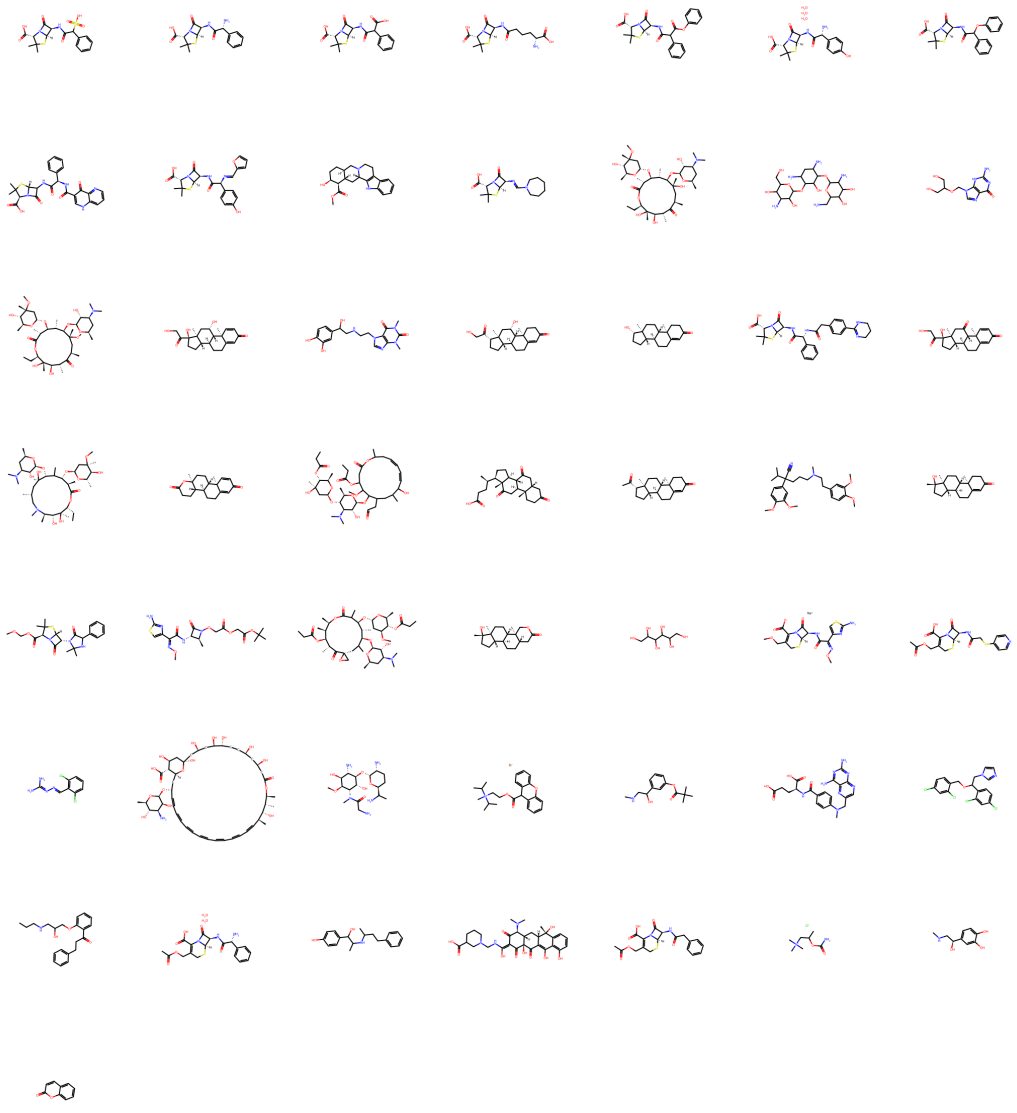


Figure 6: Subset of molecules identified by our proposed approach in the task-agnostic embedding space as candidates to belong to class 1 under BBBP task with a precision of 0:839 and a recall of 0:766.