Enhancing Molecule Generation Properties via 2-Stage Variational Autoencoders

Chenghui Zhou
Department of Machine Learning
Carnegie Mellon University
chenghuz@andrew.cmu.edu

Barnabás Póczos
Department of Machine Learning
Carnegie Mellon University
bapoczos@cs.cmu.edu

Abstract

Variational autoencoder (VAE) is a popular method for drug discovery and there had been a great deal of architectures and pipelines proposed to improve its performance. But the VAE model itself suffers from deficiencies such as poor manifold recovery when data lie on low-dimensional manifold embedded in higher dimensional ambient space and they manifest themselves in each applications differently. The consequences of it in drug discovery is somewhat under-explored. In this paper, we study how to improve the similarity of the data generated via VAE and the training dataset by improving manifold recovery via a 2-stage VAE where the second stage VAE is trained on the latent space of the first one. We experimentally evaluated our approach using the ChEMBL dataset as well as a polymer datasets. In both dataset, the 2-stage VAE method is able to improve the property statistics significantly from a pre-existing method.

1 Introduction

The use of generative models in the domain of drug discover has recently seen rapid progress. These methods can leverage large-scale molecule archives describing the structure of existing drugs to synthesize novel molecules with similar properties as potential candidates for future drugs [Duvenaud et al., 2015, Liu et al., 2018, Segler et al., 2018, You et al., 2018, Jin et al., 2018, 2020a, Polykovskiy et al., 2020, Jin et al., 2020b, Satorras et al., 2021]. There are two common ways of representing the structure of molecules SMILE strings [Weininger, 1988] and molecular graphs [Bonchev, 1991]. Graph neural networks can make effective use of the rich molecular graph representations by taking into account the atoms, edges and other structural information. SMILE strings convey less information about the molecular structure, but are more compatible with sequence models such as RNNs. Being able to generate valid molecules is the first step to machine learning drug discovery and various solutions have been proposed. For example, GNN methods [Liu et al., 2018, Jin et al., 2020a, Simonovsky and Komodakis, 2018] can constraint the output space based on the chemical rules and SMILE-based [Gómez-Bombarelli et al., 2018, Blaschke et al., 2018] approaches benefit from the abundant molecular data.

Despite the valid molecule outputs, the properties of the generated molecules, such as drug-likeness(QED) [Bickerton et al., 2012], Synthetic Accessibility Score (SA) [Ertl and Schuffenhauer, 2009] and molecular weight (MW) etc., are critical factors that decide whether they can be synthesized in a laboratory and be effective in real world applications. In order to learn to generate molecules that fulfill these properties, researchers curated molecule datasets for targeted purposes, such as ChEMBL [Mendez et al., 2019] and ZINC [Irwin and Shoichet, 2005]. The motivation is that by learning from a curated set of molecules, the generative models will learn to generate similar ones. Benchmark metrics [Polykovskiy et al., 2020] are created to measure how similar the generated molecules are to the target dataset but the results show that we still have more room to improve on this front. Learning to generate molecules that exhibit *similar* molecular properties to those in the target dataset is a prerequisite to achieve the desired properties in the generated molecules.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

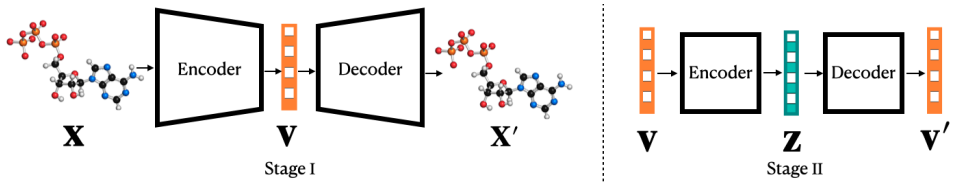


Figure 1: Overview of 2-stage VAE. In the first stage, the VAE trains with the molecule data x_i and obtain the latent variables v_i's from each of the training points. v_i's become the input of the second stage VAE. The second stage VAE's input dimension is equal to the output dimension. During sampling time, we sample z $\mathcal{N}(0, I)$ and obtain v through the second stage decoder. It is then used as the latent variable for the first stage VAE to be decoded into a molecule.

In this paper, we introduce an easy-to-implement step to the VAE approach – training an additional VAE to generate the latent for the first-stage VAE– to improve the property metrics of the generated molecules by mitigating the manifold recovery problem. Experimentally, we first show how this approach can enhance the manifold recovery for synthetic data. We then evaluate our method in two domains using the ChEMBL dataset and the polymer datasets [St. John et al., 2019]. In both settings, the 2-Stage VAE approach is able to enhance the property statistics of the generated molecule set by bringing them closer in distribution to the training set.

2 Related Work

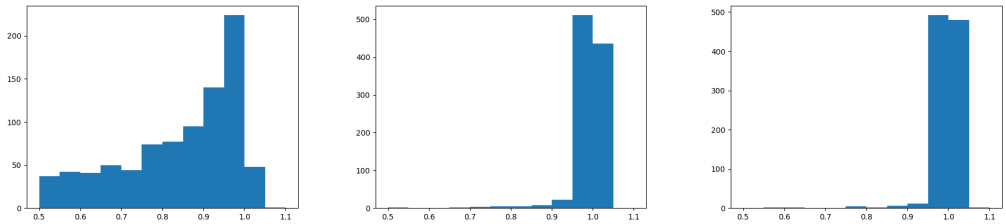
We structure our discussing based on the type of molecular representation underlying the individual methods. Most current approaches fall into one of the following families – namely, the SMILE strings approach, the molecular graph approach and the 3D point set approach. Many approaches have been proposed to generate molecules as SMILE strings [Segler et al., 2018, Gómez-Bombarelli et al., 2018]. Kusner et al. [2017], Dai et al. [2018] took advantage of the syntax of the SMILE strings to constrain the output of the VAE model in order to improve validity of the generated molecules. Other approaches to generate SMILE strings include generative adversarial model [Kadurin et al., 2017, Prykhodko et al., 2019, Guimaraes et al., 2017]. Molecular graphs carry more information about the molecular structures than the SMILE string format and GNN can effectively incorporate the additional information into the learning process [Duvenaud et al., 2015, Liu et al., 2018]. Jin et al. [2018] proposed to generate molecular graph in two steps – generate the tree-structured scaffolds first, and then combine with the substructures to form molecules. Jin et al. [2020a] improved upon this prior result and proposed to generate via substructures in a course-to-fine manner to adapt to bigger molecules, such as polymers. Satorras et al. [2021] introduced an equivariant graph neural network to apply on molecular graphs. 3D representations of molecules are gaining traction in the research communities as they provide additional spatial information of the molecules [Gebauer et al., 2019, 2022, Luo et al., 2021, Hoogeboom et al., 2022]. However, none of the methods use the VAE framework. Our paper is limited to improving the VAE approaches. Other generative approaches to drug discovery include generative adversarial model [Kadurin et al., 2017, Prykhodko et al., 2019, Guimaraes et al., 2017] and diffusion models [Hoogeboom et al., 2022, Xu et al., 2022].

3 Method

The VAE famework [Kingma and Welling, 2013] has enabled great success in the image generation domain and more recently VAE based approaches have become a popular approach for addressing the molecule generation problem. Many sophisticated architectures have been proposed to adapt the VAE approach to molecular data [Kusner et al., 2017, Dai et al., 2018, Jin et al., 2019, Satorras et al., 2021]. However, adapting the underlying neural architecture does not remedy VAE's learning deficiency in manifold recovery [Dai and Wipf, 2019, Koehler et al., 2021]. In the case of high-dimensional data that lies on low-dimensional manifold such as images and molecular representations, Koehler et al. [2021] found that the VAE is not guaranteed to recover the manifold where the nonlinear data lie. The 2-stage method can improve manifold recovery as demonstrated in a synthetic experiment (Figure 2) and further enhance the performance of a pre-existing model.

3.1 Variational Autoencoder

The variational inference framework assumes that the data x is generated from a latent variable $z \sim p(z)$. The prior $p(z)$ is assumed to be a multivariate standard normal distribution in the application of a VAE. A VAE model seeks to maximize the likelihood of the data, denoted



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure 2: Multi-stage VAE on synthetic data. The x-axis represents the norm of the data point and the y-axis represents the number of data points that are of x distance away from the unit sphere. The figures across different stages of VAE training shows that the sphere surface is eventually recovered and improved starting from stage 2.

as $\log p_\theta(x) = \log \int p(z)p_\theta(x|z)dz$ where θ denotes the generative parameters. However, the marginalization is intractable in practice due to the inherent complexity of the generator, or the decoder, thus an approximation of the objective is needed. Let ϕ be the variational parameters, the VAE model consists of a tractable encoder $q_\phi(z|x)$ and a decoder $p_\theta(x|z)$. Together, they approximate a lower bound to the log likelihood of the data. Ideally, by optimizing this lower bound we aim to increase the likelihood. This approximation enables the efficient posterior inference of the latent variable z given the output x_i and for marginal inference of the output variable x. The objective function of VAE is:

$$\mathcal{L}(\theta, \phi; x) = -D_{KL}(q_\phi(z|x) \,||\, p(z)) + \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] \leq \log p_\theta(x) \tag{1}$$

For generation, latent variable z_i is sampled from the prior $p(z)$ which is a multivariate standard normal and the decoder transforms z_i into the output x_i.

3.2 2-Stage VAE

Despite its widespread use, VAE in its original form has many known flaws. Particularly, in the case where the data lies on low-dimensional manifold embedded in a high-dimensional ambient space. Dai and Wipf [2019] hypothesized that training a VAE with a fixed decoder variance could add additional noise to the output, while training with tunable decoder variance, the decoder variance has a tendency approach zero and the model will learn the correct manifold but not the correct density. In practice this can lead to low-quality output images in comparison to models such as GANs. The implications of the finding extends to molecular data as well. Subsequently, Dai and Wipf [2019] proposed a 2-stage VAE approach to enhance the manifold and density recovery of existing VAE approaches. The reasoning was that the first stage VAE with tunable decoder variance learns the low-dimensional manifold the data lies on as the decoder variance goes to zero and the probability mass collapses onto the correct low-dimensional manifold. The second stage VAE is constructed with its latent dimension equal to the output dimension and is theorized to recover the density when the ambient dimension is equal to the intrinsic dimension. Their algorithm visibly improved the appearance of the generated images but the claim of manifold or density recovery cannot be easily verified with image data.

Koehler et al. [2021] showed empirically and analytically that when the data is a nonlinear function of the latent variables, neither the manifold nor the density is guaranteed to be recovered by the first stage VAE. This could provide an explanation as to why the generated molecules from the VAE are dissimilar in properties to the training datset. Even though this conclusion rendered the reasoning behind a 2-stage VAE invalid, the algorithm itself is not without merits. As we will demonstrate in the following synthetic experiment, a 2-stage VAE can improve manifold recovery. This could significantly improve the properties of the generated molecules as we will demonstrate in the experiments.

Synthetic Experiment We show that a 2-stage VAE setup improves the recovery of the manifold. We demonstrate this in a synthetic experiment with data generated from a ground-truth manifold (see Figure 2). We generate data from a 2-dimensional unit sphere (the norms of all the generated data points are 1). The 3-dimensional vectors are then padded with 16 dimensions of zeros to embed the data in a higher ambient space. In this case, the intrinsic dimension of the data is 2 and the ambient

dimension is 17. We trained on this data in 3 stages – meaning, the latents from the previous stage are used for training in the next stage. For the second and the third stage VAE, the latent dimension is set to be the same as their input (Figure 1). The decoder variance is tunable for all stages and the decoder variance of the first stage approaches 0 upon convergence. During sampling, the output of the later stage VAE's becomes the latent of the previous stage VAE. The last-stage VAE's latents are sampled from standard normal distribution. We sample 1000 data points to visualize the results in the histograms. They show that the VAE in the first stage *does not* recover the manifold and many of the generated data points fall *inside* of the sphere, echoing the finding by Koehler et al. [2021]. In the second and third stage, we see that more data points fall *on* the sphere, indicating the recovery of the manifold.

Application on Molecule Generation How improving manifold recovery of the generative model would benefit molecule generations does not have a straightforward answer. Evaluation metrics on molecule generation are multifaceted. Validity alone provides only a shallow examination on the quality of the generated molecules and other property metrics need to be considered to evaluate the generated molecules for real world applications. We provide empirical studies in the experiment section to study the suitability of a 2-stage VAE approach in the molecule generation domain by providing evaluation results on sample quality, structural as well as property statistics [Polykovskiy et al., 2020]. We found that the 2-stage VAE helps to generate molecules that are more similar in properties to the ones in the training dataset. We present the precise steps to train a 2-stage VAE [Dai and Wipf, 2019]:

1. Train a VAE on the molecular dataset $\{x_i | i = 1, 2, \dots n\}$ with architectures of your choice. Upon convergence, save the latent vectors $v_i \sim q_\phi(v|x_i)$, for all the molecules in the dataset. The tunable decoder covariance requirements are satisfied with decoders that follow multinomial distribution such as in molecule generations, the decoder variance approaching zero is equivalent to the probability mass concentrating on one choice;
2. With v_i's as input, train the second stage VAE with tunable decoder variance. We denote the latent of the second stage VAE as z. In this paper, we use feed-forward architectures for both the decoder $p_{\theta'}(v|z)$ and the encoder $q_{\phi'}(z|v)$. The dimension of v is equal to the dimension z for maximum power.
3. During the generation process, sample the latent of the last stage VAE by $z \sim \mathcal{N}(0, I)$. Obtain the output from the second stage decoder $v_i \sim p_{\theta'}(v|z)$ as the latent for the first stage VAE. And get the molecule sample x from the first stage decoder via $x \sim p_\theta(x|v)$.

One way to interpret this method is that while the first-stage VAE learns a mapping between the latent representations and the molecular data, the second-stage VAE learns to generate latent variables from the distribution of the latent representations of the dataset.

4 Experiments

In this section, we explore in detail how the 2-stage VAE improves the generated molecules. We adopt two model architectures – hierarchical GNN and character-level RNN – to compare the outcomes of a 2-stage and 3-stage VAE on different model architectures. We adopt a GAN-based model as an additional comparison. We conducted experiments on two molecule datasets – ChEMBL [Mendez et al., 2019] and polymers [St. John et al., 2019] for a comprehensive study of the method.

We introduce the two datasets used in the experiments – ChEMBL dataset Mendez et al. [2019] and polymers dataset St. John et al. [2019]. Details on our benchmark metrics are in Appendix B

- **ChEMBL Dataset**[Mendez et al., 2019] consists of 1,799,433 bioactive molecules with drug-like properties. It is split into training, testing, validation, test scaffold and validation scaffold dataset containing 1,463,775, 81,321, 8,321, 86,508 and 86,507 molecules respectively.
- **The Polymer Dataset**[St. John et al., 2019] contains 86,353 polymers and it's divided into training, test and validation set that contains 76,353, 5000 and 5000 molecules each. There is no scaffold split for the polymers dataset. Polymers generally have heavier weight than the molecules in the ChEMBL dataset and the dataset size is smaller.

We use hierearchical GNN (HGNN) [Jin et al., 2019] and vanilla RNN (RNN) [Polykovskiy et al., 2020] as the first stage VAE and a GAN-based model, latent GAN [Prykhodko et al., 2019], as baseline:

	Sample Quality				Structural Statistics			Property Statistics			
Stage #	Valid ↑	Unique ↑	Novelty ↑	FCD ↓	SNN ↑	Frag ↑	Scaf ↑	LogP ↓	SA ↓	QED ↓	MW ↓
HGNN#1	**1.0**	**1.0**	0.99	5.1	**0.42**	0.97	**0.46**	$0.92_{0.016}$	$0.070_{4.3e-3}$	$0.024_{9.5e-4}$	$68.8_{0.83}$
HGNN#2	**1.0**	**1.0**	0.99	**1.1**	0.41	**1.0**	0.43	$0.095_{0.019}$	$\mathbf{0.069}_{5.8e-3}$	$\mathbf{0.0067}_{1.0e-3}$	$5.0_{0.72}$
HGNN#3	**1.0**	**1.0**	**1.0**	1.2	0.41	**1.0**	**0.46**	$\mathbf{0.059}_{4.5e-3}$	$0.069_{6.3e-3}$	$0.016_{1.6e-3}$	$7.7_{0.42}$
RNN#1	0.86	**1.0**	**1.0**	1.84	0.38	**1.0**	0.38	$0.088_{7.8e-3}$	$0.25_{7.8e-3}$	$0.0088_{1.6e-3}$	$3.2_{0.55}$
RNN#2	0.87	**1.0**	**1.0**	1.86	0.38	**1.0**	0.36	$0.099_{5.5e-3}$	$0.27_{7.7e-3}$	$0.0099_{1.5e-3}$	$\mathbf{2.8}_{0.29}$
LatentGan	0.77	0.98	0.99	17.3	0.34	0.68	0.21	$0.69_{0.019}$	$0.63_{7.3e-3}$	$0.047_{2.0e-3}$	$27.2_{0.88}$

Table 1: Properties of the generated molecules trained on the ChEMBL dataset.

	Sample Quality				Structural Statistics			Property Statistics			
Stage #	Valid ↑	Unique ↑	Novelty ↑	FCD ↓	SNN ↑	Frag ↑	Scaf ↑	LogP ↓	SA ↓	QED ↓	MW ↓
HGNN#1	**1.0**	**1.0**	0.57	0.62	0.67	0.98	0.37	$1.3_{0.030}$	$0.089_{3.0e-3}$	$0.020_{1.2e-3}$	$72.2_{1.42}$
HGNN#2	**1.0**	**1.0**	0.51	**0.27**	**0.69**	**0.99**	0.37	$\mathbf{0.10}_{0.033}$	$0.031_{3.3e-3}$	$0.0041_{9.5e-4}$	$\mathbf{7.7}_{1.1}$
HGNN#3	**1.0**	**1.0**	0.52	0.29	**0.69**	**0.99**	0.38	$0.24_{0.017}$	$\mathbf{0.024}_{4.1e-3}$	$\mathbf{0.0024}_{2.9e-4}$	$9.4_{2.3}$
RNN#1	0.53	0.99	0.13	1.6	0.69	0.87	0.50	$2.6_{0.011}$	$0.31_{9.2e-3}$	$0.047_{9.4e-4}$	$178.4_{1.2}$
RNN#2	0.53	**1.0**	0.13	1.5	0.69	0.87	**0.51**	$2.5_{0.011}$	$0.31_{3.2e-3}$	$0.044_{6.9e-4}$	$176.1_{0.68}$
LatentGan	0.94	**1.0**	**0.82**	0.51	0.66	**0.99**	0.33	$0.26_{0.031}$	$0.041_{3.3e-3}$	$0.0029_{4.37e-4}$	$11.8_{1.7}$

Table 2: Properties of the generated molecules trained on the polymers dataset.

We sampled 10,000 molecules from each model to generate the results in Table 1 and Table 2. We included sample quality, structural and property statistics. The numbers in the tables are averaged over 6 sets of samples generated with 6 different random seeds from the model. We included the standard deviations for the property statistics but eliminated the rest as those are below 0.01.

On both datasets, the HGNN#2 improves upon the first stage by many folds on property statistics. The most notable improvement from the ChEMBL dataset is the QED (from 0.024 to 0.0067), MW (from 68.8 to 5.0) and LogP (0.92 to 0.059). On the polymer dataset, the second stage VAE improves significantly across all metrics – from 72.2 to 7.7 on MW, 0.020 to 0.0024 on QED, 0.089 to 0.031 on SA and 1.3 to 0.1 on LogP. A lower value on these statistics for the molecules generated through two stages signals that they are much more similar to the test set on these properties. Structural statistics generally did not change a lot throughout the 2-stage and 3-stage training. The performance on these metrics of the later stages models may be bottle-necked by the first-stage graph decoder. We also repeat the second-stage VAE to perform the third-stage. However, there is no consistent improvement from the second stage across the board as seen in Table 1 and Table 2. This is also in line with our synthetic experiment demonstrated in Figure 2, where the second and the third stage of VAE training made no substantial improvement in manifold recovery compared to the improvements from the first stage to the second stage. Overall, the second stage VAE to the HGNN model outperforms both stages of RNN VAE and the latentGAN on majority of the evaluation metrics of both datasets.

The second stage to the RNN model does not provide significantly improvement on either datasets. The RNN VAE performs particularly poorly on the the polymer dataset as only half of the molecules the model generates are valid. This may be because that the RNN architecture is sensitive to the amount of data used for training. The polymer dataset is smaller than the ChEMBL dataset while polymers generally contain more atoms. This could negative impact RNN's performance. The second stage VAE slightly improves upon the first stage on the polymer datasets. On ChEMBL dataset, the second stage VAE does not have consistent improvement on any metrics. Our hypothesis for the poor performance of the 2-stage VAE with this RNN model as the first stage model is that the variance of the first stage decoder did not fulfill the condition of approaching 0 upon convergence. We will investigate the underlying reasoning behind this behavior for our future work.

5 Discussion

Manifold recovery is a challenge for VAE methods that train on data that lie on a low-dimensional manifold embedded in a higher-dimensional ambient space. In this paper, we presented a 2-stage VAE method that improves manifold recovery as demonstrated in a synthetic experiment. In experiments with molecular data such as ChEMBL and polymers, the method significantly improve the property statistics of a pre-existing VAE method and brings the generated molecules closer to the training dataset in property distributions. The nature of our approach makes it applicable to a wide range of other VAE based molecule generation methods. In future work we want to extend the method further to successfully adapt to more types of VAE architectures.

References

Guy W Bemis and Mark A Murcko. The properties of known drugs. 1. molecular frameworks. *Journal of medicinal chemistry*, 39(15):2887–2893, 1996.

G Richard Bickerton, Gaia V Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L Hopkins. Quantifying the chemical beauty of drugs. *Nature chemistry*, 4(2):90–98, 2012.

Thomas Blaschke, Marcus Olivecrona, Ola Engkvist, Jürgen Bajorath, and Hongming Chen. Application of generative autoencoder in de novo molecular design. *Molecular informatics*, 37(1-2): 1700123, 2018.

Danail Bonchev. *Chemical graph theory: introduction and fundamentals*, volume 1. CRC Press, 1991.

Bin Dai and David Wipf. Diagnosing and enhancing vae models. *arXiv preprint arXiv:1903.05789*, 2019.

Hanjun Dai, Yingtao Tian, Bo Dai, Steven Skiena, and Le Song. Syntax-directed variational autoencoder for structured data. *arXiv preprint arXiv:1802.08786*, 2018.

Jörg Degen, Christof Wegscheid-Gerlach, Andrea Zaliani, and Matthias Rarey. On the art of compiling and using'drug-like'chemical fragment spaces. *ChemMedChem: Chemistry Enabling Drug Discovery*, 3(10):1503–1507, 2008.

David Duvenaud, Dougal Maclaurin, Jorge Aguilera-Iparraguirre, Rafael Gómez-Bombarelli, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. *arXiv preprint arXiv:1509.09292*, 2015.

Peter Ertl and Ansgar Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of cheminformatics*, 1(1):1–11, 2009.

Niklas Gebauer, Michael Gastegger, and Kristof Schütt. Symmetry-adapted generation of 3d point sets for the targeted discovery of molecules. *Advances in neural information processing systems*, 32, 2019.

Niklas WA Gebauer, Michael Gastegger, Stefaan SP Hessmann, Klaus-Robert Müller, and Kristof T Schütt. Inverse design of 3d molecular structures with conditional generative neural networks. *Nature communications*, 13(1):1–11, 2022.

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.

Gabriel Lima Guimaraes, Benjamin Sanchez-Lengeling, Carlos Outeiral, Pedro Luis Cunha Farias, and Alán Aspuru-Guzik. Objective-reinforced generative adversarial networks (organ) for sequence generation models. *arXiv preprint arXiv:1705.10843*, 2017.

Emiel Hoogeboom, Vıctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *International Conference on Machine Learning*, pages 8867–8887. PMLR, 2022.

John J Irwin and Brian K Shoichet. Zinc- a free database of commercially available compounds for virtual screening. *Journal of chemical information and modeling*, 45(1):177–182, 2005.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*, pages 2323–2332. PMLR, 2018.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Hierarchical graph-to-graph translation for molecules. *arXiv preprint arXiv:1907.11223*, 2019.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Hierarchical generation of molecular graphs using structural motifs. In *International Conference on Machine Learning*, pages 4839–4848. PMLR, 2020a.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Multi-objective molecule generation using interpretable substructures. In *International Conference on Machine Learning*, pages 4849–4859. PMLR, 2020b.

Artur Kadurin, Sergey Nikolenko, Kuzma Khrabrov, Alex Aliper, and Alex Zhavoronkov. drugan: an advanced generative adversarial autoencoder model for de novo generation of new molecules with desired molecular properties in silico. *Molecular pharmaceutics*, 14(9):3098–3104, 2017.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Frederic Koehler, Viraj Mehta, Andrej Risteski, and Chenghui Zhou. Variational autoencoders in the presence of low-dimensional data: landscape and implicit bias. *arXiv preprint arXiv:2112.06868*, 2021.

Matt J Kusner, Brooks Paige, and José Miguel Hernández-Lobato. Grammar variational autoencoder. In *International conference on machine learning*, pages 1945–1954. PMLR, 2017.

Qi Liu, Miltiadis Allamanis, Marc Brockschmidt, and Alexander L Gaunt. Constrained graph variational autoencoders for molecule design. *arXiv preprint arXiv:1805.09076*, 2018.

Shitong Luo, Jiaqi Guan, Jianzhu Ma, and Jian Peng. A 3d generative model for structure-based drug design. *Advances in Neural Information Processing Systems*, 34:6229–6239, 2021.

David Mendez, Anna Gaulton, A Patrícia Bento, Jon Chambers, Marleen De Veij, Eloy Félix, María Paula Magariños, Juan F Mosquera, Prudence Mutowo, Michał Nowotka, et al. Chembl: towards direct deposition of bioassay data. *Nucleic acids research*, 47(D1):D930–D940, 2019.

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.

Kristina Preuer, Philipp Renz, Thomas Unterthiner, Sepp Hochreiter, and Gunter Klambauer. Fréchet chemnet distance: a metric for generative models for molecules in drug discovery. *Journal of chemical information and modeling*, 58(9):1736–1741, 2018.

Oleksii Prykhodko, Simon Viet Johansson, Panagiotis-Christos Kotsias, Josep Arús-Pous, Esben Jannik Bjerrum, Ola Engkvist, and Hongming Chen. A de novo molecular generation method using latent vector based generative adversarial network. *Journal of Cheminformatics*, 11(1):1–13, 2019.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

Victor Garcia Satorras, Emiel Hoogeboom, Fabian B Fuchs, Ingmar Posner, and Max Welling. E (n) equivariant normalizing flows. *arXiv preprint arXiv:2105.09016*, 2021.

Marwin HS Segler, Thierry Kogej, Christian Tyrchan, and Mark P Waller. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS central science*, 4(1): 120–131, 2018.

Martin Simonovsky and Nikos Komodakis. Graphvae: Towards generation of small graphs using variational autoencoders. In *International conference on artificial neural networks*, pages 412–422. Springer, 2018.

Peter C St. John, Caleb Phillips, Travis W Kemper, A Nolan Wilson, Yanfei Guan, Michael F Crowley, Mark R Nimlos, and Ross E Larsen. Message-passing neural networks for high-throughput polymer screening. *The Journal of chemical physics*, 150(23):234111, 2019.

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.

Scott A Wildman and Gordon M Crippen. Prediction of physicochemical parameters by atomic contributions. *Journal of chemical information and computer sciences*, 39(5):868–873, 1999.

Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. Geodiff: A geometric diffusion model for molecular conformation generation. *arXiv preprint arXiv:2203.02923*, 2022.

Jiaxuan You, Bowen Liu, Rex Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. *arXiv preprint arXiv:1806.02473*, 2018.

A Information on the Models Used for Experiments

We chose one graph neural network based VAE and one RNN based VAE for diversity. We chose latentGAN as a non-VAE method for baseline.

Hierarchical GNN[Jin et al., 2019]: The proposed method first extracts chemically valid motifs, or substructures, from the molecular graph such that the union of these motifs covers the entire molecule. The model consists of a fine-to-coarse encoder that encodes from atoms to motifs and a coarse-to-fine decoder that selects motifs to create the molecule while deciding the attachment point between the motif and the emerging molecule. We used the configuration from the orginal model. The latent size of the VAE is 32 and we used 0.1 as the KL coefficient.

Vanilla RNN[Polykovskiy et al., 2020]: The inputs to the model are SMILE strings and the vocabulary consists of the low-level symbols in the SMILE strings. The encoder is a 1-layer GRU and the decoder is a 3-layer GRU. The latent size of the VAE is 128. We used most of the original configuration except the KL coefficient.

Latent GAN [Prykhodko et al., 2019] is also a 2-stage method. The first stage is heteroencoder that takes SMILE strings as input while the second stage is a Wasserstein GAN with gradient penalty (WGAN-GP) that trains on the latents of the first stage VAE. The heteroencoder consists of an encoder and a decoder like an autoencoder and is trained with categorical cross-entropy loss. Afterwards, the GAN is trained to generate latent vectors for the decoder from the heteroencoder. We used the original parameters for training.

B Benchmark Metrics

Property Statistics includes LogP (The Octanol-Water Partition Coefficient), SA (Synthetic Accessibility Score), QED (Quantitative Estimation of Drug-Likeness) and MW (Molecular Weight). These metrics determines the practicality of the generated molecules, for example, LogP measures the solubility of the molecules in water or an organic solvent [Wildman and Crippen, 1999], SA estimates how easily the molecules can be synthesized based on molecule structures [Ertl and Schuffenhauer, 2009], QED estimates how likely it can be a viable candidate of drugs [Bickerton et al., 2012]. The values listed in the table for each metric is the Wasserstein distance between the distribution of the property statistics in the test set and the generate molecule set.

Structural statistics includes SNN (Similarity to Nearest Neighbor), Frag (Fragment Similarity), and Scaf (Scaffold Similarity). These statistics calculate two molecular datasets' structural similarity based on their extended-connectivity fingerprints [Rogers and Hahn, 2010], BRICS fragments [Degen et al., 2008] and Bemis–Murcko scaffolds [Bemis and Murcko, 1996].

The sample quality metrics are a lot more intuitive. Valid calculates the percentage of valid molecule outputs. Unique calculates the percentage of unique molecules in the first k molecules where $k = 1000$ for the ChEMBL dataset and $k = 500$ for the polymer dataset. Novelty calculates the percentage of molecules generated that are not present in the training set. FCD is the Fréchet ChemNet Distance [Preuer et al., 2018].