NATIONAL RESEARCH UNIVERSITY
HIGHER SCHOOL OF ECONOMICS

as a manuscript

Daniil Polykovskiy

GENERATIVE MODELS FOR DRUG DISCOVERY

PhD Dissertation Summary
for the purpose of obtaining academic degree
Doctor of Philosophy in Computer Science

Moscow — 2021

The PhD Dissertation was prepared at National Research University Higher School of Economics.

Academic Supervisor: Dmitry P. Vetrov, Candidate of Sciences, National Research University Higher School of Economics.

1 Introduction

Topic of the thesis

Over the last several years, multiple teams adopted machine learning to discover new biological targets, propose molecular structures that can later become new drugs, predict and optimize their properties [1, 2, 3, 4]. Recent works demonstrated potent molecules generated using deep generative models: such molecules were tested *in vitro* and *in vivo* [5, 6, 7, 8]. In this work, we study distribution learning, conditional generation, and molecular property optimization problems and propose several novel approaches to solving these problems.

In distribution learning problem, we aim to produce novel molecular structures from the same distribution as the training set. Such an approach is useful for downstream tasks, including unsupervised pre-training and virtual screening—ranking molecules according to some quality function. In conditional generation problem, we generate molecules with specific properties. Such an approach narrows down the chemical space and biases the generative model towards desirable region. The aim of molecular property optimization is to discover molecules with the highest possible score. For example, such a score may be an activity predictor against a given target protein.

For each of the above mentioned problems, we propose novel machine learning models. In the first work [9], we demonstrate that node-level graph generative models fail on distribution learning problem unlike string-based models. We propose a new graph generative model with a hierarchical generation strategy and significantly outperform existing node-level graph generative models on distribution learning problem. We study conditional generation problem in the second work [5] and apply adversarial autoencoders to produce novel molecular structures with desirable properties. With a proposed model, we were able to generate a molecular structure that later showed selective micromolar *in vitro* activity against the selected target protein. In the third work, we analyze the molecular property optimization task using Bayesian optimization combined with variational autoencoders and propose to improve such a method with deterministic decoding [10].

Relevance

Computational approaches have been widely adopted to predict molecular properties [11] and to explore the chemical space with high throughput screening, combinatorial libraries, and evolutionary algorithms [12, 13, 14, 15]. Unlike traditional drug discovery with hand crafted molecules, generative models propose an automated approach where medicinal chemist's expertise is necessary only on final evaluation steps to confirm the quality of newly discovered structures. Such an approach is unbiased to human preferences and can take many explicit or implicit constraints into account. While a human expert can create molecular structures with certain binding points and shape, our approaches can also utilize highly accurate predictive models, conduct immediate novelty assessment and patent purity. Such a powerful tool can propose initial potent hypotheses within a matter of weeks and minimal human supervision [16].

We formulate drug discovery process as an optimization problem. Given an objective function $f(x)$ that scores a molecular structure x, we build a system that searches for the best possible structure. An example function, f may be an activity prediction model or a complex computational simulator. While building a relevant objective function is an interesting and challenging task involving domain expertise, for the purpose of this work we use standard toy functions to efficiently compare models in a unified environment. Real objective functions such as the ones used in our recent papers [16, 6] analyze generated structures' activity, novelty, synthetic accessibility, and other carefully curated terms.

The first problem when solving an optimization problem is how to represent a molecular structure. Two common ways to represent a structure are graphs and strings. Graph representation denotes atoms as nodes and bonds as edges. Alternatively, one can write down the molecule's atom symbols in depth first search traversal order with special tokens indicating cyclic bonds and branching. Such representation is called simplified molecular input-line entry system (SMILES) [17, 18]. There are other string representations that encode grammar rules using context-free grammar or reverse Polish notation to improve validity [19, 20]. String-based representations have an advantage that many previous works on natural language processing can be used out of the box. For example, a character-based neural language model can generate novel SMILES strings. It is also possible to incorporate grammar into the generative process [3]. Graph based models, on the contrary, are less studied and it is a rapidly developing topic. Substructure-based representations such as junction tree graphs are also used for multiple problems [2]. Every representation can be annotated with additional information such as 3D atom coordinates, molecular properties or fingerprints. Fingerprints are binary vectors capturing structural information. For example, Morgan [21] fingerprints iterate over atoms and encode their neighborhood into a fingerprint's index. Such descriptors can be used to predict molecular properties or to define similarity measure between molecules using Tanimoto coefficient—number of bits that are on for both molecules divided by the number of bits that are on for at least one molecule.

There are several approaches to optimize $f(x)$—using reinforcement learning, genetic algorithms, or Bayesian optimization. It is possible to optimize molecular structures directly using genetic algorithms or Bayesian optimization. In the latter case, graph kernels or similar should be used to train a surrogate function [22]. Gómez-Bombarelli et al. [1] train a variational autoencoder on molecular structures and optimize the objective function using the variational autoencoder's latent codes. The authors use Bayesian optimization approach, but other optimization techniques have later been used to optimize the objective function in the domain of latent codes.

Besides molecular property optimization, machine learning is used for distribution learning. Given a set of molecular structures sampled from an unknown distribution, distribution learning models learn the underlying distribution and produce new samples. In [23], we proposed a dataset and a diverse set of metrics to compare generated sets from different perspectives: uniqueness, validity, diversity, similarity to nearest neighbor, and many others. We implemented multiple baseline models and compared them on the basis of these metrics. Distribution learning models are useful for building virtual screening libraries. Such models capture implicit rules from the training set and produce new datasets that can be enumerated, stored and used for quick scoring and search. For example, instead of optimizing a new function $f(x)$, one can use a virtual library for virtual screening to retrieve high scoring compounds. Such an approach saves time and can quickly discover high scoring structures.

Over the last few years, we implemented several novel models and integrated them into an automated drug discovery platform called Chemistry42. Chemistry42 supports both ligand-based and structure-based drug design, producing high scoring structures within a week. In my thesis, I describe some of the models developed during this time and illustrate applications on standard datasets.

The goal of this work is to develop new molecular generative models for conditional generation, molecular property optimization, and distribution learning.

2 Key results and conclusions

Contributions. The main contributions of this work are three generative models and their applications to drug discovery problem.

1. We analyzed node-level graph generative models and proposed a hierarchical generation procedure and a fragment-oriented atom ordering. We obtained state-of-the-art results across node-level graph generative models for molecular property optimization and distribution learning tasks.

2. The Entangled Conditional Adversarial Autoencoder extends the supervised adversarial autoencoder and successfully handles multiple binary and continuous conditions. We show that the proposed model can generate molecular structures for conditions outside the original training range and generate structures with micromolar activity.

3. For molecular property optimization, we studied Bayesian optimization on the latent codes of variational autoencoders and proposed deterministic decoding to avoid issues with standard stochastic decoding. We proposed the training approach based on relaxed training objective and proved convergence to the original optimization problem. We also proposed bounded support proposals to ensure that there exists a set of encoder-decoder parameters providing lossless encoding-decoding.

Theoretical and practical significance. The proposed models pave the way for further advancements in deep learning for drug discovery. These models can accelerate discovery of new drugs and significantly reduce costs of initial hit finding, which is especially crucial during the time of a global pandemic. For conditional modeling, we proposed a novel algorithm that was able to produce selective molecules with micromolar activity against the selected protein. We also analyzed molecular property optimization problem and proposed a new training approach for variational autoencoders with deterministic decoding. Finally, we improved the quality of distribution learning models for node-level graph generative models using hierarchical generation—we obtained 3.5-fold improvement in the main distribution learning metric (Fréchet ChemNet Distance).

Key aspects/ideas to be defended.

1. A hierarchical graph generative model for molecular generation and its application to distribution learning and molecular property optimization problems
2. An entangled conditional adversarial autoencoder model for conditional molecular generation
3. A method for training variational autoencoders with deterministic decoders and application of this method for molecular property optimization

Personal contribution. In the second and third papers, the method was proposed and implemented by the author, all experiments were conducted by the author, the text has been written by an author; other authors supervised the research and helped with domain expertise. In the first paper, the author designed the experiments, supervised the research, and wrote the text.

Publications and probation of the work

First-tier publications

1. **Daniil Polykovskiy**, *Alexander Zhebrak, Dmitry Vetrov, Yan Ivanenkov, Vladimir Aladinskiy, Polina Mamoshina, Marine Bozdaganyan, Alexander Aliper, Alex Zhavoronkov, and Artur Kadurin.* Entangled Conditional Adversarial Autoencoder for de Novo Drug Discovery. Molecular pharmaceutics, 15(10):4398–4405, 2018. Q1 journal, indexed by SCOPUS.
2. **Daniil Polykovskiy** *and Dmitry Vetrov.* Deterministic Decoding for Discrete Data in Variational Autoencoders. In Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, volume 108 of Proceedings of Machine Learning Research, pages 3046–3056. Core A conference.
3. *Maxim Kuznetsov and* **Daniil Polykovskiy**. MolGrow: A graph normalizing flow for hierarchical molecular generation. Association for the Advancement of Artificial Intelligence Conference 2021. Core A* conference.

Reports at conferences

1. Neural information processing systems, Dec 2, 2018. Expo Tutorial. Topic: "Generative models for drug discovery".

2. Neural information processing systems, Dec 2, 2018. Expo Workshop. Topic: "Machine Learning for Drug discovery and Biomarker development".

3. Undoing Aging, March 30, 2019. Topic: "Deep Generative Approach for Transcriptome Analysis of Human Aging"

4. International conference on machine learning, June 9, 2019. Expo Tutorial. Topic: "Generative models for drug discovery".

Volume and structure of the work. The thesis contains an introduction, contents of publications and a conclusion. The full volume of the thesis is 67 pages.

3 Content of the work

3.1 MolGrow: A Graph Normalizing Flow for Hierarchical Molecular Generation

Recent works [24] demonstrated that graph representation is useful for molecular property optimization, since graph is a more natural representation of a molecule. However, prior works did not study distribution learning and molecular property optimization problems simultaneously. Although pretrained as generative models, previous node-level graph generative models perform significantly worse than simple string-based generative models. In this section, we propose a new graph-based normalizing flow generative model for molecular generation to narrow the performance gap between these domains.

Previous works on graph generation produced graphs either sequentially [25, 26], or simultaneously in one-shot manner [27, 28]. We propose a new generation approach—starting with a single node graph, we iteratively split each node into two and repeat this procedure until we obtain a graph of a given size. We formulated a set of invertible transformations for node splitting and merging, noise injection and separation. We also noticed that standard breadth-first search ordering is prone to producing unwanted macrocycles in the generated structures; hence, we proposed a new fragment-oriented atom ordering. In such ordering, we first split a molecular structure into a set of meaningful fragments and then align these fragments with node merging and splitting.

We represent a graph with node attribute matrix $V \in \mathbb{R}^{N \times d_v}$ and edge attribute tensor $E \in \mathbb{R}^{N \times N \times d_e}$, where d_v and d_e are feature dimensions. For the input data, V_i defines atom type and charge, $E_{i,j}$ defines edge type. Since molecular graphs are non-oriented,

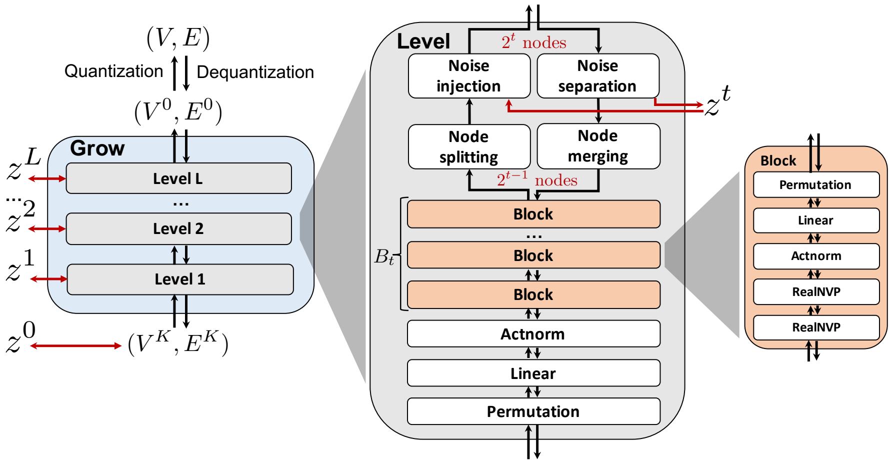

Figure 1: MolGrow architecture. **Left:** Full architecture combines multiple *levels* to generate latent codes $z^L, \dots, z^0$ from a graph (V, E) and vice versa. **Middle:** Each *level* separates noise, merges node pairs, applies multiple *blocks* and linear transformations; **Right:** Each *block* applies three channel-wise transformations and two RealNVP layers.

we preserve the symmetry constraint on all intermediate layers: $E_{i,j,k} = E_{j,i,k}$. The final graph has $N = 2^L$ nodes, where L is a number of node-splitting layers in the model.

We use node merging and splitting operations to control the graph size. These operations are inverse of each other, and both operate by rearranging node and edge features. Consider a graph (V^k, E^k) with N_k nodes. Node merging operation joins nodes $2i$ and $2i+1$ into a single node by concatenating their features and features of the edge between them. We concatenate edge features connecting the merged nodes:

$$\underbrace{V_i^{k+1}}_{2d_v+d_e} = \operatorname{cat}\Big(\underbrace{V_{2i}^k}_{d_v}, \underbrace{V_{2i+1}^k}_{d_v}, \underbrace{E_{2i,2i+1}^k}_{d_e}\Big), \tag{1}$$

$$\underbrace{E_{i,j}^{k+1}}_{4d_e} = \operatorname{cat}\Big(\underbrace{E_{2i,2j}^k}_{d_e}, \underbrace{E_{2i,2j+1}^k}_{d_e}, \underbrace{E_{2i+1,2j}^k}_{d_e}, \underbrace{E_{2i+1,2j+1}^k}_{d_e}\Big). \tag{2}$$

Node splitting is the inverse of node merging layer: it slices features into original components. MolGrow produces a latent vector for each *level*. We derive the latent codes by separating half of the node and edge features before node merging and impose Gaussian prior on these latent codes. During generation, we sample the latent code from the prior and concatenate it with node and edge features. As we show in the experiments, latent codes on different levels affect the generated structure differently. Latent codes from smaller intermediate graphs (top level) influence global structure, while bottom *level* features define local structure. We illustrate the model in Figure 1.

Table 1: Molecular property optimization: penalized octanol-water partition coefficient (penalized logP) and quantitative estimation of drug-likeness (QED). Results for baseline models from [29, 27].

Method	Penalized logP			QED		
	1st	2nd	3rd	1st	2nd	3rd
ZINC250k	4.52	4.30	4.23	**0.948**	**0.948**	**0.948**
Graph-based models						
GCPN [24]	7.98	7.85	7.80	**0.948**	0.947	0.946
MolecularRNN [26]	8.63	6.08	4.73	0.844	0.796	0.736
GraphNVP [27]	-	-	-	0.833	0.723	0.706
GraphAF [29]	12.23	11.29	11.05	**0.948**	**0.948**	**0.948**
MoFlow [28]	-	-	-	**0.948**	**0.948**	**0.948**
Proposed model						
MolGrow (GE)	**14.01 ± 0.364**	**13.95 ± 0.424**	**13.92 ± 0.422**	**0.9484 ± 0.0**	**0.9484 ± 0.0**	**0.9484 ± 0.0**
MolGrow (GE, Top only)	11.66 ± 0.31	11.65 ± 0.319	11.63 ± 0.306	**0.9484 ± 0.0**	**0.9484 ± 0.0**	**0.9484 ± 0.0**
MolGrow (GE, Bottom only)	10.29 ± 3.32	10.29 ± 3.33	10.28 ± 3.32	**0.9484 ± 0.0**	**0.9484 ± 0.0**	**0.9484 ± 0.0**
MolGrow (predictor-guided optimization)	5.2 ± 0.347	4.94 ± 0.262	4.84 ± 0.22	**0.9484 ± 0.0**	0.9483 ± 0.0	0.9483 ± 0.0
MolGrow (REINFORCE)	4.81 ± 0.285	4.47 ± 0.145	4.39 ± 0.126	0.9468 ± 0.001	0.9459 ± 0.001	0.9455 ± 0.001
SMILES and fragment-based models						
DD-VAE [10]	5.86	5.77	5.64	-	-	-
Grammar VAE [3]	2.94	2.88	2.80	-	-	-
SD-VAE [30]	4.04	3.50	2.96	-	-	-
JT-VAE [2]	5.30	4.93	4.49	**0.948**	0.947	0.947

In the experiments, we compare our model with state of the art graph and string based generative models. Table 1 we show that MolGrow outperforms current best graph and string-based generators for two of the most commonly used objective functions—penalized octanol-water partition coefficient (penalized logP) and quantitative estimation of drug-likeness (QED). We also significantly outperformed the best node-level graph model on distribution learning task (Table 2) in terms of Fréchet ChemNet Distance (FCD).

3.2 Entangled Conditional Adversarial Autoencoder for de Novo Drug Discovery

In this section, we switch to conditional generation problem, where a model has to produce novel molecular structures with given properties. Earlier papers on molecular generation [32, 33] produced molecular structures' fingerprints and retrieved the most similar molecules from a large database of molecular structures based on similarity search. Such an approach requires little data, and discovered structures are readily available for pur-

Table 2: Distribution learning metrics on MOSES dataset.

Method	FCD/Test (↓)	Frag/Test (↑)	Unique@10k (↑)	Novelty (↑)
Graph-based models				
MolecularRNN [26]	23.13	0.56	98.6%	99.9%
GraphVAE [31]	49.39	0.0	5%	**100%**
GraphNVP [27]	29.95	0.62	99.7 %	99.9%
GraphAF (BFS) [29]	21.84	0.651	97%	99.9%
MoFlow [28]	28.05	0.685	100%	99.99%
Proposed model				
MolGrow (fragment-oriented)	**6.284 ± 0.986**	0.929 ± 0.025	99.28 ± 0.62%	99.26 ± 0.12%
MolGrow (BFS)	9.962 ± 0.795	0.932 ± 0.01	**100 ± 0.0%**	99.37 ± 0.08%
MolGrow (BFS on fragments)	16.15 ± 1.026	0.868 ± 0.018	**100 ± 0.0%**	**100 ± 0.0%**
MolGrow (random permutation)	40.17 ± 4.709	0.051 ± 0.034	58.96 ± 38.11%	**100 ± 0.0%**
MolGrow (GAT instead of CAGE)	6.523 ± 0.302	**0.941 ± 0.013**	99.36 ± 0.3%	99.32 ± 0.05%
MolGrow (No positional embedding)	6.771 ± 0.555	0.937 ± 0.006	99.49 ± 0.19%	99.41 ± 0.06%
SMILES and fragment-based models				
CharRNN (from MOSES benchmark)	**0.073**	**0.9998**	99.73%	84.19%
VAE (from MOSES benchmark)	0.099	0.9994	99.84%	69.49%
JTN-VAE (from MOSES benchmark)	0.422	0.9962	**100%**	**91.53%**

chasing. In contrast, training models directly on SMILES strings requires more data even to produce semantically valid strings. We decided to combine fingerprints generator with a conditional generative model to produce novel SMILES strings. A conditional generative model learns a distribution $p(x \mid y)$ over the molecular structures x with given properties y.

In our paper [5], we proposed an adversarial autoencoder-based conditional model. Adversarial Autoencoders (AAE) [34] are generative models that model the data distribution $p_{\mathrm{d}}(x)$ by training a regularized autoencoder. The regularizer forces a distribution of the latent code $q(z) = \int Q_E(z \mid x) p_{\mathrm{d}}(x) dx$ to match a tractable prior $p(z)$. In this paper, we will only consider deterministic autoencoders: the encoding distribution $Q_E(z \mid x)$ and decoding distribution $P_G(x \mid z)$ are parameterized by neural networks E and G respectively: $z = E(x)$ and $x = G(z)$.

Regularization of the latent space is implemented by an adversarial training procedure [35] with the Discriminator model $D(z)$. The Discriminator is trained to discriminate between samples from the latent distribution $q(z)$ and the prior $p(z)$. The Encoder E is trained to modify the latent code so the discriminator could not distinguish the latent distribution from the prior. This results in a minimax game $\min_E \max_D \mathcal{L}_{\text{adv}}$, where

$$\mathcal{L}_{\text{adv}} = \mathbb{E}_{x\sim p_{\text{d}}} \log D\left(E(x)\right) + \mathbb{E}_{z\sim p(z)} \log\left(1 - D(z)\right) \tag{3}$$

The adversarial training with the reconstruction penalty constitutes the following optimization task:

$$\min_{E,G} \max_{D} \mathbb{E}_{x\sim p_{\text{d}}} \log D\left(E(x)\right) + \mathbb{E}_{z\sim p(z)} \log\left(1 - D(z)\right) - \mathbb{E}_{x\sim p_{\text{d}}} \log p\left(x \mid G\left(E\left(x\right)\right)\right). \tag{4}$$

The proposed model—Entangled Adversarial Autoencoder—introduces a conditional prior model $p_\theta(z \mid y) \sim \mathcal{N}(\mu_\theta(y), \Sigma_\theta(y))$ and uses a conditional discriminator to train the model. We utilize a reparameterization trick $\overline{z} = g_\theta(z, y) = \Sigma_\theta^{-1/2}(y)\left(z - \mu_\theta(y)\right)$ to simplify the training objective:

$$\begin{aligned}\min_{E,G,\theta} \max_{D} \mathbb{E}_{(x,y)\sim p_{\text{d}}} \log D\left(g_\theta(E(x), y), y\right) + \mathbb{E}_{y\sim p(y)}\mathbb{E}_{\overline{z}\sim p(\overline{z})} \log\left(1 - D(\overline{z}, y)\right)\\ - \mathbb{E}_{(x,y)\sim p_{\text{d}}} \log p\left(x \mid G\left(E\left(x\right), y\right)\right).\end{aligned} \tag{5}$$

We also proposed an additional regularizer that improved training. In the optimization problem above, discriminator's objective can be interpreted as enforcing independence of a reparameterized latent code from the condition. We proposed a technique called predictive disentanglement that uses a separate predictive model to infer y from z and adjusts latent codes to fool the predictor. The additive regularizer takes a form of

$$\min_{E} \max_{q} \lambda \mathbb{E}_{(x,y)\sim p_{\text{d}}} \log q(y \mid g_\theta(E(x), y)). \tag{6}$$

In the experiments, we studied different generation conditions and compared different AAE modifications, including a model with a fixed prior network—$p_\theta(z \mid y) = \mathcal{N}(0, I)$. In Table 3, we compared the proposed models on fingerprint-conditioned generation. In this table, 'No' corresponds to Supervised Adversarial Autoencoder, 'Predictive' corresponds to employing only predictive disentanglement and not supplying the condition to the discriminator. 'Joint' corresponds to supplying condition to the discriminator. 'Combined' corresponds to predictive disentanglement and supplying condition to the discriminator.

In Table 4, we conditioned the model on three continuous properties—octanol-water partition coefficient, synthetic accessibility score, and binding energy towards MCL1 protein. With such a model we were able to generate novel molecules with better binding energy than the best molecule from the training set. We also conducted a similar experiment by training a conditional model on inhibition concentration 50 (IC50) for JAK3 protein and discovered a molecular structure that showed micromolar *in vitro* activity.

Table 3: Performance of models trained with different disentanglement techniques using fingerprint vectors as the condition. Notice the large gap between the model with no disentanglement (corresponding to [34]) and other models. First four models utilize an unconditional model of a prior distribution.

Disentanglement	Tanimoto, %	Hamming	Exact, %	Remaining MI
No	80.0	10.49	4.4	2.75
Predictive	86.2	7.13	11.4	0.64
Joint	88.7	5.78	17.4	1.56
Combined	91.8	4.18	27.8	0.32
Entangled, no Predictive	93.5	3.31	40.9	2.51
Entangled	**93.6**	**3.28**	**41.3**	1.30

Table 4: Performance of semi-supervised models on partially labeled binding energy dataset in terms of Pearson correlation r between the requested value and the generated one.

Disentanglement	logP, r	SA, r	E, r
No	0.311 ± 0.01	0.0522 ± 0.009	0.02 ± 0.04
Predictive	0.687 ± 0.006	0.0893 ± 0.008	0.063 ± 0.05
Joint	0.595 ± 0.007	0.0838 ± 0.008	0.109 ± 0.04
Combined	0.677 ± 0.007	0.0896 ± 0.007	0.116 ± 0.04
Entangled	$\mathbf{0.804 \pm 0.005}$	$\mathbf{0.593 \pm 0.007}$	$\mathbf{0.406 \pm 0.04}$

3.3 Deterministic Decoding for Discrete Data in Variational Autoencoders

While distribution learning and conditional modelling are useful for quickly exploring the chemical space, the ultimate goal of drug discovery is to find one or several "perfect" molecules. Molecular property optimization problem is commonly used for such applications as hit finding and hit-to-lead optimization, where a computational approach can reliably estimate a quality of a given molecular structure, and the model's goal to discover a molecule with the maximum quality. In this section, we revise a common approach to solving this task with variational autoencoders (VAE) and Bayesian optimization [1]. We discuss issues issues with stochastic decoding in VAEs and propose deterministic decoding as a solution [10].

Molecular property optimization task, some times referred to as goal directed learning, is one of the important tasks in computational chemistry. Given some objective function $f(x)$ that measures the compound's quality, the goal is to find a compound x_* that has the highest quality: $x_* \in \arg\max_{x \in \mathcal{X}} f(x)$. Depending on the task, $f(x)$ can guide optimization towards structures with desirable physiochemical properties or biological activity [36]. It is also possible to restrict the search space $\mathcal{X}$ to compounds similar to the reference one. In this case, the optimization problem is referred to as a constrained molecular optimization. An example of such a problem is hit optimization when a promising molecular structure is optimized towards higher activity and better properties. Computation of $f(x)$ is commonly considered expensive and time consuming since for practical problems $f(x)$ either requires complex simulations or synthesis and *in vitro* testing.

Using recent advances in representation learning with variational autoencoders (VAE) [37], Gómez-Bombarelli et al. [1] proposed to adapt VAEs for molecular property optimization task. Their approach was to first train a VAE on a large collection of molecular structures and then optimize molecular properties with respect to the latent codes of VAE. More formally, given a labeled training set, they computed latent codes of all training examples and trained a regression model on the corresponding latent codes. Next, they trained a sparse Gaussian process regression model [38] and found the latent code corresponding to the structure with the highest expected improvement—a commonly used Bayesian optimization approach. They then added the newly labeled example to the training set and repeated this procedure for a few iterations.

The paper mentioned above uses recurrent neural network encoder and decoder trained on a string representation of molecular structure—a simplified molecular input line entry system (SMILES). Such complex decoders in VAE tend to ignore the latent codes since they can model the generative distribution on their own. However, for the Bayesian optimization to succeed, its latent codes must carry useful information about the corresponding objects—the easier it is to predict target properties from the latent codes, the easier it is to search for the optimal structures. To avoid the decoder ignoring the latent codes, we consider deterministic decoding, where each latent code corresponds to a single molecular structure. A simple way to turn a stochastic decoder into a deterministic one is to replace all sampling operations with greedily selecting the most probable token at each iteration. However, such sampling scheme is biased and can decrease diversity of the generated structures. We propose to optimize the evidence lower bound of a variational autoencoder with deterministic decoding directly.

Consider an evidence lower bound of a variational autoencoder with sequential decoder:

$$\mathcal{L}(\theta, \phi) = \mathbb{E}_{x \sim p(x)} \left[\mathbb{E}_{z \sim q_\phi(z|x)} \sum_{i=1}^{|x|} \log \pi^\theta_{x,i,x_i}(z) - \mathcal{KL}\left(q_\phi(z \mid x) \parallel p(z)\right) \right], \tag{7}$$

where $p(x)$ is the data distribution, q_ϕ is an encoding distribution, and $\pi^\theta_{x,i,s}(z)$ is the decoding distribution $p_\theta(x_i = s \mid z, x_1, \dots, x_{i-1})$. In deterministic decoders, decoded sequence $\widetilde{x}_\theta(z)$ is

$$\widetilde{x}_i = \arg\max_s p_\theta(s \mid z, x_1, \dots, x_{i-1}) = \arg\max_s \pi^\theta_{x,i,s}(z) \tag{8}$$

A reconstruction probability and an evidence lower bound for deterministic decoding are

$$p\left(x \mid \widetilde{x}_\theta\left(z\right)\right) = \begin{cases} 1, & \widetilde{x}_\theta(z) = x \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

$$\mathcal{L}_*(\theta, \phi) = \mathbb{E}_{x \sim p(x)} \left[\mathbb{E}_{z \sim q_\phi(z|x)} \log p\left(x \mid \widetilde{x}_\theta(z)\right) - \mathcal{KL}\left(q_\phi(z \mid x) \parallel p(z)\right) \right] \tag{10}$$

We propose to optimize $\mathcal{L}_*$ by approximating $\arg\max$ with a smooth function and annealing the temperature.

$$\mathbb{I}\left[i = \arg\max_j r_j \right] = \prod_{j \neq i} \mathbb{I}\left[r_i > r_j\right] \approx \prod_{j \neq i} \sigma_\tau(r_i - r_j), \tag{11}$$

$$\sigma_\tau(x) = \frac{1}{1 + \exp\left(-x/\tau\right)\left(\frac{1}{\tau} - 1\right)} \xrightarrow[\tau \to 0]{} \mathbb{I}\left[x > 0\right] \tag{12}$$

$$\mathcal{L}_\tau(\theta, \phi) = \mathbb{E}_{x \sim p(x)} \left[\mathbb{E}_{z \sim q_\phi(z|x)} \sum_{i=1}^{|x|} \sum_{s \neq x_i} \log \sigma_\tau\left(\pi^\theta_{x,i,x_i}(z) - \pi^\theta_{x,i,s}(z)\right) - \mathcal{KL}\left(q_\phi(z \mid x) \parallel p(z)\right) \right] \tag{13}$$

We faced two challenges: how to ensure that for some parameters (θ, ϕ) objective function is finite ($\mathcal{L}_* > -\infty$) and how does optimization of $\mathcal{L}_\tau$ relates to optimization of $\mathcal{L}_*$.

We show that it is impossible to obtain finite $\mathcal{L}_*$ if proposal distribution $q(x \mid z)$ has full support: if encoder maps every object to every latent code, the decoder should decode every object from every latent code. Hence, a deterministic decoder will always have a non-zero reconstruction error rate and produce infinitely small ELBO. To overcome this issue, we proposed bounded support proposals parameterized with shifted and scaled factorized kernels:

$$q_\phi(z \mid x) = \prod_{i=1}^{d} \frac{1}{\sigma_i^\phi(x)} K\left(\frac{z_i - \mu_i^\phi(x)}{\sigma_i^\phi(x)}\right). \tag{14}$$

We derived closed-form Kullback-Leibler divergence for a handful of kernels for a standard Gaussian and a uniform priors. Bounded support proposals and sufficiently flexible encoder and decoder ensure that for some (θ, ϕ) $\mathcal{L}_*$ is finite.

To connect $\mathcal{L}_\tau$ to $\mathcal{L}_*$, we proved the following theorem.

Theorem 1. *Let $\Omega = \{(\theta, \phi) \mid \mathcal{L}_*(\theta, \phi) > -\infty\}$. Let $\Delta(\widetilde{x}_\theta, \phi)$ be a sequence-wise reconstruction error for the given encoder-decoder pair, and $\Delta(\phi)$ be a sequence-wise reconstruction error rate for an optimal decoder (given by maximum a-posteriori probability across all possible decoding sequences). Assume that $\Omega \neq \emptyset$, length of sequences is bounded ($\exists L : |x| \leq L, \forall x \in \chi$), and Θ and Φ are compact sets of possible parameter values. Assume that $q_\phi(z \mid x)$ is equicontinuous in total variation for any ϕ and x:*

$$\begin{aligned} &\forall \epsilon > 0, \exists \delta = \delta(\epsilon, x, \phi) > 0 : \\ &\|\phi - \phi'\| < \delta \Rightarrow \int |q_\phi(z \mid x) - q_{\phi'}(z \mid x)|\, dz < \epsilon. \end{aligned} \tag{15}$$

Let τ_n, ϕ_n, θ_n be such sequences that:

$$\lim_{n\to\infty} \tau_n = 0, \quad \tau_n \in (0, 1), \tag{16}$$

$$(\theta_n, \phi_n) \in \operatorname*{Arg\,max}_{\theta\in\Theta, \phi\in\Phi} \mathcal{L}_{\tau_n}(\theta, \phi), \tag{17}$$

sequence $\{\phi_n\}$ converges to $\widetilde{\phi}$ and for any ϕ such that $\Delta(\phi) = 0$ exists θ such that $\Delta(\widetilde{x}_\theta, \phi) = 0$. Let $\widetilde{\theta}$ be:

$$\widetilde{\theta} \in \operatorname*{Arg\,max}_{\theta\in\Theta} \mathcal{L}_*(\theta, \widetilde{\phi}). \tag{18}$$

Then the sequence-wise reconstruction error rate decreases asymptotically as

$$\Delta(\widetilde{x}_{\theta_n}, \phi_n) = \mathcal{O}\left(\frac{1}{\log(1/\tau_n)}\right), \tag{19}$$

Parameters $(\widetilde{\theta}, \widetilde{\phi})$ solve the optimization problem for $\mathcal{L}_*$:

$$\mathcal{L}_*(\widetilde{\theta}, \widetilde{\phi}) = \sup_{\theta\in\Theta, \phi\in\Phi} \mathcal{L}_*(\theta, \phi). \tag{20}$$

This theorem shows that if we optimize the model and anneal the temperature, we will obtain an optimal encoder. If we then fine-tune the decoder, we will get an optimal encoder-decoder pair. In the experiments, we optimized $\mathcal{L}_\tau$ for gradually decreasing τ and trained a model on molecular data. We considered distribution learning problem and molecular property optimization tasks. On distribution learning, proposed training technique and bounded support proposals improve Fréchet ChemNet Distance (FCD) and similarity to the nearest neighbor (SNN) on MOSES dataset (Table 5). On molecular property optimization we optimized a commonly used penalized octanol-water partition coefficient [3]. With standard setup, we obtained better molecules than standard VAE and other baselines (Table 6). The model also showed better predictive performance of the target property from the latent codes compared to the baselines.

Table 5: Distribution learning with deterministic decoding on MOSES dataset for different reconstruction accuracies. We report generative modeling metrics: FCD/Test (lower is better) and SNN/Test (higher is better). Mean ± std over multiple runs. G = Gaussian proposal, T = Triweight proposal.

Method	FCD/Test (↓)			SNN/Test (↑)		
	70%	80%	90%	70%	80%	90%
VAE (G)	0.205 ± 0.005	0.344 ± 0.003	0.772 ± 0.007	0.550 ± 0.001	0.525 ± 0.001	0.488 ± 0.001
VAE (T)	0.207 ± 0.004	0.335 ± 0.005	0.753 ± 0.019	0.550 ± 0.001	0.526 ± 0.001	0.490 ± 0.000
DD-VAE (G)	0.198 ± 0.012	0.312 ± 0.011	0.711 ± 0.020	**0.555 ± 0.001**	0.531 ± 0.001	0.494 ± 0.001
DD-VAE (T)	**0.194 ± 0.001**	**0.311 ± 0.010**	**0.690 ± 0.010**	**0.555 ± 0.000**	**0.532 ± 0.001**	**0.495 ± 0.001**

Table 6: Reconstruction accuracy (sequence-wise) and validity of samples on ZINC dataset; Predictive performance of sparse Gaussian processes on ZINC dataset: Log-likelihood (LL) and Root-mean-squared error (RMSE); Scores of top 3 molecules found with Bayesian Optimization. G = Gaussian proposal, T = Tricube proposal.

Method	Reconstruction	Validity	LL	RMSE	top1	top2	top3
CVAE	44.6%	0.7%	-1.812 ± 0.004	1.504 ± 0.006	1.98	1.42	1.19
GVAE	53.7%	7.2%	-1.739 ± 0.004	1.404 ± 0.006	2.94	2.89	2.80
SD-VAE	76.2%	43.5%	-1.697 ± 0.015	1.366 ± 0.023	4.04	3.50	2.96
JT-VAE	76.7%	100.0%	-1.658 ± 0.023	1.290 ± 0.026	5.30	4.93	4.49
VAE (G)	87.01%	78.32%	-1.558 ± 0.019	1.273 ± 0.050	5.76	5.74	**5.67**
VAE (T)	90.3%	73.52%	-1.562 ± 0.022	1.265 ± 0.051	5.41	5.38	5.35
DD-VAE (G)	89.39%	63.07%	-1.481 ± 0.020	1.199 ± 0.050	5.13	4.84	4.80
DD-VAE (T)	89.89%	61.38%	**-1.470 ± 0.022**	**1.186 ± 0.053**	**5.86**	**5.77**	5.64

4 Conclusion

In the final section, we summarize the main contributions of this work.

1. We proposed a new molecular graph generative model that produces molecular structures hierarchically—starting with a single node, it iteratively increases graph size by splitting each node into two. We built this model aiming to achieve good performance simultaneously in molecular property optimization and distribution learning. We discovered that modern node-level graph generators produce poor distribution learning results, significantly worse than existing string-based and substructure-based generators. Our model significantly improves distribution learning metrics for node-level graph generators and discovers high scoring molecules on two common molecular property optimization tasks.

2. We studied conditional generative models for drug discovery and proposed a new model—Entangled Conditional Adversarial Autoencoder. This model can handle multiple conditions and extrapolate beyond the condition's training range. This paper was one of the first to demonstrate *in vitro* activity and specificity of a generated molecular structure against a selected target protein. The proposed direction of conditional generation was later combined with reinforcement learning techniques in our later publications [39, 16].

3. We studied a widely used combination of variational autoencoders and Bayesian optimization and discovered potential issues with using deterministic decoding during sampling and stochastic decoding during training. We constructed a deterministic decoding procedure and proposed an intuitive training scheme using relaxed objective function. In the experiments, we showed that training the model with deterministic decoding improves molecular property optimization quality.

4. We proved a theorem connecting the relaxed objective function of a deterministic decoder and the original training objective. We also observed that lossless decoding is impossible with full support proposals. Hence, we proposed to use bounded support proposals to improve the model.

References

[1] 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, February 2018.

[2] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2323–2332, Stockholmsmässan, Stockholm Sweden, 2018. PMLR.

[3] Matt J Kusner, Brooks Paige, and José Miguel Hernández-Lobato. Grammar variational autoencoder. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1945–1954. JMLR. org, 2017.

[4] 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.

[5] Daniil Polykovskiy, Alexander Zhebrak, Dmitry Vetrov, Yan Ivanenkov, Vladimir Aladinskiy, Polina Mamoshina, Marine Bozdaganyan, Alexander Aliper, Alex Zhavoronkov, and Artur Kadurin. Entangled conditional adversarial autoencoder for de novo drug discovery. *Molecular pharmaceutics*, 15(10):4398–4405, 2018.

[6] Alex Zhavoronkov, Bogdan Zagribelnyy, Alexander Zhebrak, Vladimir Aladinskiy, Victor Terentiev, Quentin Vanhaelen, Dmitry S Bezrukov, Daniil Polykovskiy, Rim Shayakhmetov, Andrey Filimonov, et al. Potential non-covalent sars-cov-2 3c-like protease inhibitors designed using generative deep learning approaches and reviewed by human medicinal chemist in virtual reality. 2020.

[7] Daniel Merk, Lukas Friedrich, Francesca Grisoni, and Gisbert Schneider. De novo design of bioactive small molecules by artificial intelligence. *Molecular informatics*, 37(1-2):1700153, 2018.

[8] Daniel Merk, Francesca Grisoni, Lukas Friedrich, and Gisbert Schneider. Tuning artificial intelligence on the de novo design of natural-product-inspired retinoid x receptor modulators. *Communications Chemistry*, 1(1):68, 2018.

[9] Maxim Kuznetsov and Daniil Polykovskiy. Molgrow: A graph normalizing flow for hierarchical molecular generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, Feb. 2021.

[10] Daniil Polykovskiy and Dmitry Vetrov. Deterministic decoding for discrete data in variational autoencoders. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 3046–3056, Online, 26–28 Aug 2020. PMLR. URL `http://proceedings.mlr.press/v108/polykovskiy20a.html`.

[11] 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. *Chem. Sci.*, 9(2):513–530, January 2018.

[12] Stefano Curtarolo, Gus L W Hart, Marco Buongiorno Nardelli, Natalio Mingo, Stefano Sanvito, and Ohad Levy. The high-throughput highway to computational materials design. *Nat. Mater.*, 12(3):191–201, March 2013.

[13] Xiangqian Hu, David N Beratan, and Weitao Yang. Emergent strategies for inverse molecular design. *Sci. China B*, 52(11):1769–1776, November 2009.

[14] Tu C Le and David A Winkler. Discovery and optimization of materials using evolutionary approaches. *Chem. Rev.*, 116(10):6107–6132, May 2016.

[15] Edward O Pyzer-Knapp, Changwon Suh, Rafael Gómez-Bombarelli, Jorge Aguilera-Iparraguirre, and Alán Aspuru-Guzik. What is High-Throughput virtual screening? a perspective from organic materials discovery. *Annu. Rev. Mater. Res.*, 45(1):195–216, 2015.

[16] 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.

[17] David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. 17:1–14, 1970.

[18] David Weininger, Arthur Weininger, and Joseph L Weininger. Smiles. 2. algorithm for generation of unique smiles notation. *Journal of chemical information and computer sciences*, 29(2):97–101, 1989.

[19] Mario Krenn, Florian Häse, AkshatKumar Nigam, Pascal Friederich, and Alán Aspuru-Guzik. Selfies: a robust representation of semantically constrained graphs with an example application in chemistry. *arXiv preprint arXiv:1905.13741*, 2019.

[20] Noel O'Boyle and Andrew Dalke. DeepSMILES: An Adaptation of SMILES for Use in Machine-Learning of Chemical Structures. *ChemRxiv*, 2018.

[21] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *J. Chem. Inf. Model.*, 50(5):742–754, May 2010.

[22] S Vichy N Vishwanathan, Nicol N Schraudolph, Risi Kondor, and Karsten M Borgwardt. Graph kernels. *The Journal of Machine Learning Research*, 11:1201–1242, 2010.

[23] 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.

[24] Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. In *Advances in neural information processing systems*, pages 6410–6421, 2018.

[25] Jiaxuan You, Rex Ying, Xiang Ren, William Hamilton, and Jure Leskovec. GraphRNN: Generating realistic graphs with deep auto-regressive models. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5708–5717, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/you18a.html.

[26] Mariya Popova, Mykhailo Shvets, Junier Oliva, and Olexandr Isayev. Molecularrnn: Generating realistic molecular graphs with optimized properties. *arXiv preprint arXiv:1905.13372*, 2019.

[27] Kaushalya Madhawa, Katushiko Ishiguro, Kosuke Nakago, and Motoki Abe. GraphNVP: An invertible flow model for generating molecular graphs. *arXiv preprint arXiv:1905.11600*, 2019.

[28] Chengxi Zang and Fei Wang. Moflow: An invertible flow model for generating molecular graphs. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Aug 2020. doi: 10.1145/3394486.3403104. URL `http://dx.doi.org/10.1145/3394486.3403104`.

[29] Chence Shi, Minkai Xu, Zhaocheng Zhu, Weinan Zhang, Ming Zhang, and Jian Tang. Graphaf: a flow-based autoregressive model for molecular graph generation. *International Conference on Learning Representations*, 2020.

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

[31] Martin Simonovsky and Nikos Komodakis. Graphvae: Towards generation of small graphs using variational autoencoders. In Věra Kůrková, Yannis Manolopoulos, Barbara Hammer, Lazaros Iliadis, and Ilias Maglogiannis, editors, *Artificial Neural Networks and Machine Learning – ICANN 2018*, pages 412–422, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01418-6.

[32] Artur Kadurin, Alexander Aliper, Andrey Kazennov, Polina Mamoshina, Quentin Vanhaelen, Kuzma Khrabrov, and Alex Zhavoronkov. The cornucopia of meaningful leads: Applying deep adversarial autoencoders for new molecule development in oncology. *Oncotarget*, 8(7):10883–10890, 2016. ISSN 1949-2553. doi: https://doi.org/10.18632/oncotarget.14073. URL `https://www.oncotarget.com/article/14073/`.

[33] 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. *Mol. Pharm.*, 14(9):3098–3104, September 2017.

[34] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, and Ian Goodfellow. Adversarial autoencoders. 2016.

[35] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. pages 2672–2680, 2014.

[36] Nathan Brown, Marco Fiscato, Marwin HS Segler, and Alain C Vaucher. Guacamol: benchmarking models for de novo molecular design. *Journal of chemical information and modeling*, 59(3):1096–1108, 2019.

[37] Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. *International Conference on Learning Representations*, 2013.

[38] Edward Snelson and Zoubin Ghahramani. Sparse gaussian processes using pseudo-inputs. In *Advances in neural information processing systems*, pages 1257–1264, 2006.

[39] Maxim Kuznetsov, Daniil Polykovskiy, Dmitry P Vetrov, and Alex Zhebrak. A prior of a googol gaussians: a tensor ring induced prior for generative models. In *Advances in Neural Information Processing Systems*, pages 4102–4112, 2019.