Through the looking glass: navigating in latent space to optimize over combinatorial synthesis libraries

Aryan Pedawi
Atomwise
aryan@atomwise.com

Saulo de Oliveira
Atomwise
saulo@atomwise.com

Henry van den Bedem
Atomwise & University of California, San Francisco
Dept. of Bioengineering & Therapeutic Sciences
vdbedem@atomwise.com

Abstract

Commercially available, synthesis-on-demand virtual libraries contain trillions of readily synthesizable compounds and can serve as a bridge between *in silico* property optimization and *in vitro* validation. However, as these libraries continue to grow exponentially in size, traditional enumerative search strategies that scale linearly with the number of compounds encounter significant limitations. Hierarchical enumeration approaches scale more gracefully in library size, but are inherently greedy and implicitly rest on an additivity assumption of the molecular property with respect to its sub-components. In this work, we present a reinforcement learning approach to retrieving compounds from ultra-large libraries that satisfy a set of user-specified constraints. Along the way, we derive what we believe to be a new family of α-divergences that may be of general interest in density estimation. Our method first trains a library-constrained generative model over a virtual library and subsequently trains a normalizing flow to learn a distribution over latent space that decodes constraint-satisfying compounds. The proposed approach naturally accommodates specification of multiple molecular property constraints and requires only black box access to the molecular property functions, thereby supporting a broad class of search problems over these libraries.

1 Introduction

Recent advances in combinatorial chemistry have vastly increased the size of commercially available, synthesis-on-demand virtual catalogs. Ultra-large[1] combinatorial synthesis libraries (CSL) now contain trillions of compounds, and library sizes are expected to continue to grow. These virtual catalogs are prohibitively large for any exhaustive experimental screen, and thus require virtual screening protocols to select a subset of molecules that satisfy multiple desired properties for synthesis and testing. However, the current size and growth of CSLs poses ongoing challenges for existing computational approaches.

Ultra-large CSLs cannot be explicitly enumerated using common molecular representations such as the string-based SMILES [16] encoding. Instead, retrieving a compound from these libraries relies on dedicated cheminformatics algorithms using fingerprint patterns [3, 13] or molecular substructure

[1]Today, the largest [15] commercially available CSLs contain on the order of 10^{12} synthesizable molecules, which is "ultra-large" in comparison to conventional chemical libraries used in virtual screening, but is of course still a round-off error with respect to the vastness of chemical space, estimated to be on the order of 10^{60}.

NeurIPS 2023 Generative AI and Biology Workshop.

searches [14]. These approaches rely on explicit representations of the full compound and encounter scaling challenges with modern library sizes. Hierarchical enumeration [12] approaches that exploit the structure of these libraries are a practical compromise, but have a tendency to be greedy and rest on an implicit assumption that the molecular properties of interest are additive with respect to a molecule's sub-components.

To accommodate this rapid growth in library size, recent work, like the combinatorial synthesis library variational auto-encoder (CSLVAE) [11], has proposed specially designed auto-encoders that give way to efficient decoding schemes for retrieving products from CSLs, which allows for navigation of these libraries through traversals in the latent space. This further presents us with the opportunity to formulate multi-parameter optimization over ultra-large libraries as a reinforcement learning problem over the latent space, which could serve as a useful template in supporting a broad class of search problems applied to CSLs.

In this work, we present an approach for retrieving compounds from CSLs given a set of user-specified constraints on molecular properties by applying reinforcement learning in the latent space of a trained CSLVAE. A key contribution is the derivation of a new objective function for policy optimization that admits a numerically stable, low variance gradient estimator for a new family of mass-covering divergences. We explore multiple molecular property constraints that are commonly used in drug discovery and show that our method is significantly more effective at retrieving compounds that satisfy those constraints than enumerative approaches.

2 Methodology

Let X denote the set of potentially pharmacologically active molecules, colloquially referred to as *chemical space*. The set of all molecules contained in a CSL $\mathcal{D}$ is denoted by $X_{\mathcal{D}} \subset X$. We omit details about the construction of CSLs as well as specifics about the featurization of molecules in this manuscript, but interested readers can refer to [11] for more information.

Define $a : X \to \mathbb{R}^k$ to be a vector-valued function that evaluates k molecular properties of a given query molecule. We assume, without loss of generality, that higher values are preferable to lower values for each of the k molecular properties. Let $b \in \mathbb{R}^k$ be a vector of acceptability thresholds. A molecule x is said to be constraint-satisfying if $a(x) \geq b$. Our objective is to identify and sample from the constraint-satisfying part of the CSL, denoted $X^*_{\mathcal{D}} \equiv \{x : x \in X_{\mathcal{D}}, a(x) \geq b\}$.

Let $R_0 : X \to \{0, 1\}$ denote the binarized reward function, which takes on a value of one if a molecule $x \in X$ jointly satisfies the constraints and zero if any of the constraints are violated,

$$R_0(x) = \begin{cases} 1, & \text{if } a(x) \geq b; \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Binarized rewards are challenging optimization targets for reinforcement learning algorithms due to the so-called "sparse reward" problem. We therefore introduce a temperature parameter $\tau > 0$ and consider a continuous family of tempered reward relaxations $R_\tau : X \to [0, 1]$,

$$R_\tau(x) = \prod_{i=1}^{k} \sigma\left(\frac{a_i(x) - b_i}{\tau}\right), \tag{2}$$

where $\sigma(x) = 1/(1 + \exp(-x))$ is the sigmoid function. Note that the binarized reward emerges in the zero temperature limit of the tempered reward, $R_0(x) = \lim_{\tau \to 0^+} R_\tau(x)$.

CSLVAEs [11] are comprised of a probabilistic encoder $q_\psi(z|x)$ and decoder $p_\theta(x|z, \mathcal{D})$, where $z \in Z \subseteq \mathbb{R}^d$ is a so-called *latent code* and Z is the *latent space*. Importantly, the architecture of the decoder guarantees that $\text{supp}(p_\theta) \subseteq X_{\mathcal{D}}$ by construction, i.e., that only compounds in $X_{\mathcal{D}}$ are reachable via the decoder.

In a sufficiently expressive and well-trained variational auto-encoder (i.e., one which attains a high reconstruction accuracy), the decoder $p_\theta(x|z, \mathcal{D})$ behaves like a probabilistic pseudo-inverse of the encoder $q_\psi(z|x)$ within the essential support[2] of the aggregated variational posterior $q_\psi(z|\mathcal{D}) \equiv$

[2]The ϵ-essential support of a distribution $p(z)$ is defined as $\text{supp}_\epsilon(p) = \{z : p(z) > \epsilon\}$, i.e., it includes all $z \in Z$ where the probability density $p(z)$ exceeds $\epsilon > 0$.

$|X_{\mathcal{D}}|^{-1} \sum_{x \in X_{\mathcal{D}}} q_\psi(z|x)$. In particular, high reconstruction accuracy suggests that (1) for distinct pairs of molecules $x \neq x'$ from $X_{\mathcal{D}}$, the latent space conditionals $q_\psi(z|x)$ and $q_\psi(z|x')$ have mostly disjoint essential supports (i.e., the probability of collisions in the latent space for distinct molecules is low), and (2) for latent codes sampled from the aggregated variational posterior $z \sim q_\psi(z|\mathcal{D})$, the essential support of $p_\theta(x|z, \mathcal{D})$ should concentrate on a relatively small subset of $X_{\mathcal{D}}$.

Given such an approximate bijectivity, it is natural to consider framing the search in the continuous latent space and using the learned decoder p_θ to retrieve the associated molecules from the library, effectively turning what is otherwise a discrete problem into policy optimization in $\mathbb{R}^d$ with a non-differentiable reward, which is a rather routine setting in reinforcement learning. In a slight abuse of notation, define the tempered reward induced by z as follows:

$$R_\tau(z) = \mathbb{E}_{p_\theta(x|z,\mathcal{D})} \left[R_\tau(x) \right]. \tag{3}$$

Hence, $R_0(z) = \lim_{\tau \to 0^+} R_\tau(z)$ can be viewed as the probability that a molecule $x \sim p_\theta(x|z, \mathcal{D})$ decoded from z conditionally at random is constraint-satisfying, i.e.,

$$R_0(z) = \Pr_{p_\theta(x|z,\mathcal{D})}[x \in X^*_{\mathcal{D}}]. \tag{4}$$

We frame our objective as that of learning a policy that samples latent codes with probability proportional to their tempered reward $\pi_\eta(z|\tau) \propto R_\tau(z)$, which bears similarity to recent work on Boltzmann generators [10] and generative flow networks [1]. The policy together with the decoder $p_\theta(x|z, \mathcal{D})$ induces a distribution over $X_{\mathcal{D}}$:

$$\pi_\eta(x|\tau, \mathcal{D}) = \int_z p_\theta(x|z, \mathcal{D}) \, \pi_\eta(z|\tau) \, \mathrm{d}z. \tag{5}$$

In other words, the policy $\pi_\eta(x|\tau = 0, \mathcal{D})$ can be viewed as an approximation to uniformly sampling over the constraint-satisfying subset of the library, i.e.,

$$\pi_\eta(x|\tau = 0, \mathcal{D}) \overset{\propto}{\sim} \text{Uniform}(x|X^*_{\mathcal{D}}). \tag{6}$$

Optimization of the policy parameters η can be formulated as a divergence minimization problem,

$$\min_\eta \quad \mathbb{E}_{p(\tau)} \left[D(\pi_\eta \| r_\tau) \right], \tag{7}$$

where $D(\pi_\eta \| r_\tau) \geq 0$ is a divergence from the policy $\pi_\eta(z|\tau)$ to the target distribution $r_\tau(z) = \Psi_\tau^{-1} R_\tau(z)$ and $\Psi_\tau = \int_z R_\tau(z) \, \mathrm{d}z$ is the (unknown) normalizing constant of the tempered reward function. Here, we amortize optimization [4] with respect to $\tau > 0$ by specifying a temperature distribution $p(\tau)$, e.g., an exponential distribution with rate parameter $\lambda > 0$, and minimize the expected divergence from the policy to the tempered reward marginalized over $p(\tau)$.

Valid divergences are non-negative and attain a value of zero if and only if $\pi_\eta = r_\tau$, but otherwise vary in the manner by which they penalize differences between the source and target distributions [7–9]. So-called *mass-covering* divergences, such as the forward KL divergence $D_{\text{KL}}(r_\tau \| \pi_\eta) = \mathbb{E}_{r_\tau(z)}[\log r_\tau(z) - \log \pi_\eta(z|\tau)]$, prefer policies where $r_\tau(z) > 0 \implies \pi_\eta(z|\tau) > 0$, thereby steering the policy iterates towards solutions that cover the bulk of the essential support of the target r_τ. On the other hand, *mode-seeking* divergences, such as the reverse KL divergence $D_{\text{KL}}(\pi_\eta \| r_\tau) = \mathbb{E}_{\pi_\eta(z|\tau)}[\log \pi_\eta(z|\tau) - \log r_\tau(z)]$, prefer policies where $r_\tau(z) = 0 \implies \pi_\eta(z|\tau) = 0$, which can often have the pathological effect of steering the policy iterates towards the most salient and prominent nearby mode(s), leading to training instabilities that fail to consider the bulk of the target distribution (i.e., mode collapse). Policies optimized according to mass-covering (cf. mode-seeking) divergences show bias in favor of having a high recall (cf. precision) with respect to its coverage of the essential support of the target distribution.

We derive a new family of α-divergences related to the Rényi divergences [7] with the useful properties of (i) being invariant to the unknown normalizing constant Ψ_τ, (ii) having the mass-covering inductive bias for $\alpha \geq 1$, and (iii) admitting low variance and numerically stable stochastic gradients:

$$D_\alpha(\pi_\eta \| r_\tau) = \frac{\mathbb{E}_{\pi_\eta(z|\tau)} \left[\left(\frac{R_\tau(z)}{\pi_\eta(z|\tau)} \right)^\alpha \log \frac{R_\tau(z)}{\pi_\eta(z|\tau)} \right]}{\alpha \mathbb{E}_{\pi(z|\tau)} \left[\left(\frac{R_\tau(z)}{\pi_\eta(z|\tau)} \right)^\alpha \right]} - \frac{1}{\alpha^2} \log \mathbb{E}_{\pi_\eta(z|\tau)} \left[\left(\frac{R_\tau(z)}{\pi_\eta(z|\tau)} \right)^\alpha \right]. \tag{8}$$

Details pertaining to the derivation of this divergence family can be found in the Appendix. Once we have adequately trained the policy using the objective (7), we can sample from the learned distribution over the relevant constraint-satisfying subset of the library $\pi_\eta(x|\tau = 0, \mathcal{D})$.

	Lipinski	Ghose	Lee	Rule-of-3	Macrocycles	QED
Random (uniform)						
# satisfied clusters	223.4±9.5	189.8±21.9	2.2±1.3	0.2±0.4	1.6±1.0	7.2±2.6
% satisfied	2.27±0.09	1.92±0.21	0.02±0.01	0.01±0.01	0.02±0.01	0.07±0.03
Random (stratified)						
# satisfied clusters	1956.2±23.5	2075.8±23.2	30.6±7.3	0.8±0.6	0.8±0.7	121.8±10.5
% satisfied	29.47±0.43	31.51±0.46	0.40±0.03	0.02±0.02	0.01±0.01	1.36±0.10
Hierarchical (*k*=2000, *m*=20)						
# satisfied clusters	924.4±8.65	812.2±7.16	20.8±1.3	**382.8±5.4**	0.0±0.0	149.8±7.8
% satisfied	52.01±0.26	37.37±0.12	0.58±0.08	**21.05±0.54**	0.00±0.00	2.09±0.13
# evals	108,260	117,199	96,664	76,812	61,560	124,699
# capless evals	44,260	50,238	30,365	32,951	20,560	55,872
Trained policy (α=2, *i.i.d.*)						
# satisfied clusters	1504.6±29.2	1537.6±24.9	1251.4±47.5	119.6±2.6	**57.8±4.1**	992.8±26.7
% satisfied	26.61±0.37	20.46±0.33	17.57±0.44	11.15±0.18	10.86±0.19	23.87±0.36
% unique	98.40±0.21	99.82±0.02	99.56±0.03	67.34±0.11	72.16±0.33	97.80±0.14
Trained policy (α=2, w/ *d.s.*)						
# satisfied clusters	**2705.8±21.7**	**4182.4±10.0**	**2490.4±17.5**	156.2±6.8	**61.2±3.4**	**1404.4±14.6**
% satisfied	**66.79±0.20**	**67.75±0.48**	**53.01±0.53**	12.66±0.25	**70.12±0.26**	**56.55±0.57**
% unique	99.61±0.04	99.98±0.01	99.92±0.01	88.46±0.14	43.95±0.36	95.20±0.38

Table 1: Summary statistics from sampling 10,000 molecules from each policy using the 3T compound library. Means and standard deviations are calculated using results from five distinct attempts.

3 Satisfying multiple molecular property constraints in an ultra-large CSL

We demonstrate that the proposed approach can efficiently retrieve constraint-satisfying compounds from ultra-large CSLs using a library comprised of 74,232 Enamine [6] building blocks and 21 three- and four-component reactions, resulting in a library with nearly three trillion products.

We trained a CSLVAE model on this library using the same architecture and hyperparameters described in [11], which has a latent space dimension of $d = 64$. For the policy, we use a neural spline flow [5] with eight rational quadratic spline flow coupling layers with eight knots each, with a fixed but randomly-initialized permutation applied after each coupling layer to mix units.

To demonstrate the generality of the proposed approach, we consider multiple commonly (and less commonly) used molecular property filters. We compare against two random baseline policies illustrative of naive enumeration as well as a hierarchical enumeration policy [12]. Details concerning the baseline methods and the molecular property filters are discussed in the Appendix.

For each molecular property filter, we train the policy for 100 episodes, where each episode is comprised of 1,000 samples from the most recent policy iterate, reflecting a total of 100,000 function evaluations over the course of training. The policy parameters are updated off-policy for 50 iterations given data sampled *i.i.d.* from a replay buffer of the last 32 episodes.

For each of the fitted policies, we sample 10,000 compounds and evaluate the satisfiability rate—the proportion of sampled compounds that satisfied the specified constraints—as well as the number of distinct compound clusters found among the constraint-satisfying retrievals as a measure of diversity. For this latter statistic, we use sequential Butina clustering [2] with the ECFP4 similarity and a cutoff of 0.35 and count the number of distinct clusters.

Table 1 summarizes the results. We observe that, in the majority of cases considered, the trained policy with diversity sampling attains both the highest satisfiability rate as well as the largest number of distinct clusters with a constraint-satisfying compound. For filters where the prevalence in the library is low, we observe significant enrichment in both of these quantities. Notably, for two filters considered where the prevalence in the library is relatively high (Lipinski and Ghose), sampling *i.i.d.* from the trained policy actually attains a lower satisfiability rate and cluster count than the stratified random baseline. Once we apply diversity sampling—a non-*i.i.d.* sampling procedure described in the Appendix to improve coverage of the policy modes—the number of identified satisfied clusters increases by 2-4x, substantially outperforming the baselines.

Acknowledgments

The authors would like to thank Greg Friedland for engineering support.

References

[1] Yoshua Bengio, Salem Lahlou, Tristan Deleu, Edward J Hu, Mo Tiwari, and Emmanuel Bengio. GFlowNet Foundations. *arXiv preprint arXiv:2111.09266*, 2021.

[2] Darko Butina. Unsupervised data base clustering based on daylight's fingerprint and tanimoto similarity: A fast and automated way to cluster small and large data sets. *Journal of Chemical Information and Computer Sciences*, 39(4):747–750, 1999.

[3] Adrià Cereto-Massagué, María José Ojeda, Cristina Valls, Miquel Mulero, Santiago Garcia-Vallvé, and Gerard Pujadas. Molecular fingerprint similarity search in virtual screening. *Methods*, 71:58–63, 2015.

[4] Alexey Dosovitskiy and Josip Djolonga. You only train once: loss-conditional training of deep networks. In *International conference on Learning Representations (ICLR)*, 2019.

[5] Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. *Advances in Neural Information Processing Systems*, 32, 2019.

[6] Oleksandr O Grygorenko. Enamine Ltd.: The Science and Business of Organic Chemistry and Beyond. *European Journal of Organic Chemistry*, 2021(47):6474–6477, 2021.

[7] Yingzhen Li and Richard E Turner. Rényi divergence variational inference. *Advances in Neural Information Processing Systems*, 29, 2016.

[8] Laurence Illing Midgley, Vincent Stimper, Gregor NC Simm, Bernhard Schölkopf, and José Miguel Hernández-Lobato. Flow annealed importance sampling bootstrap. *arXiv preprint arXiv:2208.01893*, 2022.

[9] Tom Minka. Divergence measures and message passing. Technical report, Microsoft Research, 2005.

[10] Frank Noé, Simon Olsson, Jonas Köhler, and Hao Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457), 2019.

[11] Aryan Pedawi, Pawel Gniewek, Chaoyi Chang, Brandon Anderson, and Henry van den Bedem. An efficient graph generative model for navigating ultra-large combinatorial synthesis libraries. *Advances in Neural Information Processing Systems*, 35:8731–8745, 2022.

[12] Arman A Sadybekov, Anastasiia V Sadybekov, Yongfeng Liu, Christos Iliopoulos-Tsoutsouvas, Xi-Ping Huang, Julie Pickett, Blake Houser, Nilkanth Patel, Ngan K Tran, Fei Tong, et al. Synthon-based ligand discovery in virtual libraries of over 11 billion compounds. *Nature*, 601(7893):452–459, 2022.

[13] RA Sayle, John Mayfield, and Noel O'Boyle. Recent advances in chemical & biological search systems: evolution vs revolution. In *11th International Conference on Chemical Structures*, 2018.

[14] Robert Schmidt, Raphael Klein, and Matthias Rarey. Maximum common substructure searching in combinatorial make-on-demand compound spaces. *Journal of Chemical Information and Modeling*, 62(9):2133–2150, 2021.

[15] Wendy A Warr, Marc C Nicklaus, Christos A Nicolaou, and Matthias Rarey. Exploration of ultra-large compound collections for drug discovery. *Journal of Chemical Information and Modeling*, 62(9):2021–2034, 2022.

[16] David Weininger. SMILES, a chemical language and information system: Introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, 1988.