

SYSTEMS AND METHODS FOR IDENTIFYING COMPOUNDS IN COMBINATORIAL LIBRARIES HAVING SPECIFIC MOLECULAR PROPERTIES

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority to United States Provisional Patent Application No. 63/587,417 entitled “SYSTEMS AND METHODS FOR CASTING VIRTUAL SCREENING AS A REINFORCEMENT LEARNING PROBLEM APPLIED TO LIBRARY-CONSTRAINED GENERATIVE MODELS,” filed October 2, 2023, which is hereby incorporated by reference.

[0002] This application also claims priority to United States Provisional Patent Application No. 63/587,002 entitled “SYSTEMS AND METHODS FOR CASTING VIRTUAL SCREENING AS A REINFORCEMENT LEARNING PROBLEM APPLIED TO LIBRARY-CONSTRAINED GENERATIVE MODELS,” filed September 29, 2023, which is hereby incorporated by reference.

TECHNICAL FIELD

[0003] The present disclosure is directed to systems and methods for identifying compounds in a combinatorial synthesis library that satisfy a plurality of molecular properties.

BACKGROUND

[0004] Commodity chemistry has the potential to rapidly advance pre-clinical drug discovery pipelines, from hit discovery to lead optimization. Commercially available virtual, synthesis-on-demand chemical catalogs show Moore-like growth (Figure 3A). This rapid growth in virtual compound databases is spurred largely by advances in combinatorial chemistry, in which chemical building blocks are transformed into synthons with experimentally validated reactivity features ensuring high synthetic success rates [1]. Importantly, a recent evaluation of several combinatorial synthesis libraries (CSLs) found that more than half of all FDA approved drugs had an analog in the library with maximum common substructure (MCS) similarity exceeding 0.82 [2]. The close similarity to approved drugs means those analogs likely share biologically active scaffolds and exhibit comparable physicochemical properties and toxicological safety profiles. These results suggest that CSLs can provide value

throughout pre-clinical drug development, beyond hit discovery into hit-to-lead and lead optimization.

[0005] However, massive CSLs, containing trillions of compounds, present computational challenges in hit discovery. First, ultra-large CSLs no longer permit explicit enumeration of chemical structures using, for example, simplified molecular input line entry system (SMILES) or self-referencing embedded strings (SELFIES). Instead, retrieving a compound relies on dedicated cheminformatics algorithms using fingerprint patterns [3, 4] or molecular substructure searches [5–7]. These algorithms rely on computational representation of the full compounds.

[0006] Second, the fraction of active or promising molecules in the CSL may not grow at the same rate as the full library. For example, in-stock screening libraries can contain 19,000-fold more bio-like molecules than virtual libraries [8] with important implications for the success of virtual screening. The likelihood of finding high-scoring but inactive compounds with predictive AI increases as large swaths of chemical space are now out-of-distribution for the models. Novel functional groups may have unknown bearings on ADME and molecular properties that may come to light only in later stages of the drug discovery pipeline. Finding developable hits, *e.g.*, potent and soluble compounds is more challenging as libraries continue to grow much faster than the set of known active compounds.

[0007] Third, exhaustive, structure-based scoring of a tera-size or larger library against a therapeutic target site is beyond today's computational capabilities. Giga-sized vHTS docking campaigns have been reported [9, 10], but require vast resources and their techniques do not scale with the rapid growth of CSLs. Sampling-based methods that extrapolate scores using similarity measures or surrogate models from a docked subset of the CSL are resource-efficient but suffer from limitations related to increasingly sparse and incomplete representations of the CSL by the subset [11, 12]. This limitation will worsen as CSLs continue to grow exponentially. Alternatively, divide-and-conquer approaches can construct high-scoring compounds by incrementally adding synthons in subsequent docking iterations and pruning low-scoring fragments [13]. While this method relies on the same combinatorial scaling techniques that underlie the CSL, a fragment-based pruning approach has several shortcomings. For example, fragments often adopt different binding poses compared to their equivalents embedded in a larger compound. Alternatively, a large fragment can score favorably at an intermediate iteration, despite steric hindrance preventing its growth. That could eliminate a smaller fragment ultimately satisfying new and favorable interactions with

the addition of more fragments (Figure 3B). Another important limitation of this approach is that several key compound properties in hit discovery, notably solubility, are not additive across chemical groups, reducing predictive efficiency (Figure 3C). Traditional search-and-predict approaches unavoidably will remain challenged by growing CSLs.

[0008] In principle, generative AI has the potential to overcome the limitations associated with the size of CSLs. In drug discovery applications, a generative AI can design a compound that satisfies desired properties [14–16], for example bioactivity towards a target [17–19] or physicochemical properties [20]. Instead of explicitly enumerating and scoring molecular hypotheses as is common with predictive modeling, generative AI designs those hypotheses, solving the inverse problem [21, 22]. However, virtually none of the large body of generative studies presented in the literature experimentally verified their proposed compounds, with few exceptions [23]. Indeed, it is a major and widely recognized challenge to bias generative AI towards synthetic accessibility [24], despite efforts to include explicit synthetic accessibility cost functions [25], limit the model to fragments from known compounds [26, 27], or induce bias towards simple and known synthetic pathways [28–30]. Synthesis of the proposed compounds is therefore often time- and cost-intensive, hindering experimental validation and adoption in drug discovery [21]. Circumventing synthetic inaccessibility, prior work [23] identified a small subset of their generated compounds around a single scaffold in Enamine REAL enabling rapid and cost-effective experimental testing.

[0009] Given the above background, what is needed in the art are improved systems and methods for identifying a plurality of compounds in a combinatorial synthesis library that each satisfy a plurality of molecular properties.

SUMMARY

[0010] The present disclosure addresses the shortcomings disclosed above. The present disclosure overcome a critical limitation of unconstrained generative AI in medicinal chemistry applications: it ‘hallucinates’ compounds that cannot be made. The present disclosure prioritizes compounds that can be readily made in the lab, accelerating the path from discovery to potential drug. The present disclosure’s strength over other methods that search CSLs lies in 1) its scalability, encoding ever growing CSLs logarithmically in space and lookup time [32], and 2) carefully balancing exploration and exploitation by learning a mass-covering policy enabled by a new family of α -divergences. These divergences (i) are

invariant to the unknown normalizing constant Ψ_{τ} , (ii) have the mass-covering inductive bias for $\alpha \geq 1$, and (iii) admit low variance and numerically stable stochastic gradients. In a virtual high-throughput screen for MC2R, the present disclosure found potent, selective and diverse hits from a three trillion compound CSL despite the lack of bioactive prior compounds for this challenging target.

[0011] In one aspect of the present disclosure, computer-implemented methods for identifying a plurality of compounds in a combinatorial synthesis library that each satisfy a plurality of molecular properties are provided. The combinatorial synthesis library is accessible through a trained variational auto-encoder that comprises (i) an encoder module that maps a compound in the combinatorial synthesis library into a d -dimensional code in a learned d -dimensional latent space and (ii) a decoder module that retrieves a molecular structure of the compound, with a reconstruction likelihood greater than a threshold reconstruction likelihood, upon inputting the d -dimensional code into the decoder module.

[0012] In some embodiments, the threshold reconstruction likelihood is 50 percent or greater, 60 percent or greater, or between 60 percent and 80 percent.

[0013] In some embodiments, a first molecular property in the plurality of molecular properties is a bioactivity prediction against a therapeutic target, and the first molecular property is satisfied for a respective compound in the plurality of compounds when a bioactivity classification module determines, based on a chemical structure of the respective compound, that the respective compound is an active compound against the therapeutic target.

[0014] In some embodiments, a first molecular property in the plurality of molecular properties is a potency prediction against a therapeutic target, and the first molecular property is satisfied for a respective compound in the plurality of compounds when a potency prediction regression module determines, based on a chemical structure of the respective compound, that the respective compound has at least a threshold potency against the therapeutic target.

[0015] In some embodiments, the plurality of compounds comprises 500 or more compounds, 1000 or more compounds, 10,000 or more compounds, or 100,000 or more compounds.

[0016] In some embodiments, the combinatorial synthesis library represents more than 10,000 synthons and more than 10 three- or four- component reactions.

[0017] In some embodiments, the d -dimensional code has a latent space dimension d that is between 32 and 128.

[0018] In some embodiments, the d -dimensional code has a latent space dimension d that is 64.

[0019] A. *Selecting a current batch of d -dimensional codes.* A policy model, comprising a plurality of parameters, is used to select a current batch of d -dimensional codes in the learned d -dimensional latent space.

[0020] In some embodiments, the policy model is a normalizing flow generative model.

[0021] In some embodiments, the normalizing flow generative model is a neural spine flow.

[0022] In some embodiments, the neural spine flow includes between four and 20 rational-quadratic spline flow coupling layers that each have a plurality of knots.

[0023] In some embodiments, the neural spine flow consists of eight quadratic spline flow coupling layers that each have eight knots.

[0024] In some embodiments, a fixed but randomly-initialized permutation is applied after each coupling layer.

[0025] In some embodiments, the plurality of parameters comprises 500,000 or more parameters, or 1×10^6 or more parameters.

[0026] B. *Retrieve the corresponding molecular structure of each d -dimensional code using the decoder module.* Each d -dimensional code in the current batch of d -dimensional codes is applied to the decoder module, thereby retrieving the molecular structure of each compound in a current batch of compounds.

[0027] C. *Determining the molecular properties of the retrieved compounds.* A determination, using at least the molecular structure of each compound in the current batch of compounds, is made of a corresponding molecular property value of each respective molecular property in the plurality of molecular properties, thereby determining a corresponding plurality of molecular property values for each respective compound in the current batch of compounds.

[0028] In some embodiments, the plurality of molecular properties comprises any two or more rules, any three or more rules, or all four rules of the Lipinski's rule of Five: (i) not

more than five hydrogen bond donors, (ii) not more than ten hydrogen bond acceptors, (iii) a molecular weight under 500 Daltons, and (iv) a LogP under 5.

[0029] In some embodiments, the plurality of molecular properties comprises any two or more rules, any three or more rules, or all four rules of the Ghose filter: (i) a calculated log P that is between -0.4 and 5.6, (ii) a molecular weight that is between 160 and 480, (iii) a molar refractivity that is between 40 and 130, and (iv) a total number of atoms that is between 20 and 70

[0030] In some embodiments, the plurality of molecular properties comprises (i) a first requirement that each compound in the plurality of compounds has three or fewer hydrogen bond donors and (ii) a second requirement that each compound in the plurality of compounds has three or fewer hydrogen bond acceptors.

[0031] In some embodiments, the plurality of molecular properties comprises any two or more rules, any three or more rules, any four or more rules, any five or more rules, or all six rules selected from the group consisting of: (i) a molecular weight under 300 Daltons, (ii) a LogP less than 3, (iii) three or fewer hydrogen bond donors, (iv) three or fewer hydrogen bond acceptors, (v) a polar surface area of 60 Å², and (vi) three or fewer rotatable bonds.

[0032] In some embodiments, a respective compound in the plurality of compounds satisfies the plurality of molecular properties when it has a respective Quantitative Estimate of Druglikeness (QED) score of greater than QED threshold, wherein the respective QED score is within a range of zero to one, and is based on a weighted summation of a plurality of individual desirability functions that canvass the respective compound's molecular weight, LogP, number of hydrogen bond donors, number of hydrogen bond acceptors, polar surface area, number of rotatable bonds, and number of structural alerts.

[0033] D. *Calculating the tempered reward function of each of the compounds using their molecular property values.* A determination, for each respective compound in the current batch of compounds, is made of a corresponding tempered reward function value for the respective compound using the plurality of molecular property values corresponding to the respective compound.

[0034] E. *Determining the current probability density using the tempered reward function values.* A determination of a current probability density, under a target distribution in which *d*-dimensional codes are sampled through the auto-encoder module from the combinatorial synthesis library with probability proportional to their tempered reward function value, is

made using the corresponding tempered reward function value of each respective compound in the current batch of compounds.

[0035] F. *Updating the policy model using the current probability density.* The plurality of parameters of the policy model are updated through minimization of at least a difference between (i) the current probability density and (ii) the target distribution. The difference is evaluated using an α -divergence based objective function.

[0036] In some embodiments, the method is repeated until an optimization criterion is satisfied.

[0037] In some embodiments, the method is repeated a plurality of instances and the update uses a portion of the current probability density of each instance in the plurality of instances to minimize against the target distribution.

[0038] In some embodiments, the combinatorial synthesis library comprises a trillion compounds, the encoder module is configured to map each compound in the combinatorial synthesis library into a d -dimensional code in the learned d -dimensional latent space, and the policy model is configured to search any portion, or plurality of portions, of the entire combinatorial synthesis library in accordance with the updating of the plurality of parameters.

[0039] In some embodiments, compounds in the plurality of compounds that fail a pan-assay interference (PAINS) filter are filtered out of the plurality of compounds.

[0040] In some embodiments, compounds in the plurality of compounds that have three or more stereocenters are filtered out of the plurality of compounds.

[0041] In some embodiments, the plurality of compounds is clustered based on structural similarity thereby forming a plurality of clusters and retaining, in the plurality of compounds, a subset of compounds in each cluster in the plurality of clusters.

[0042] In some embodiments, the plurality of compounds is synthesized and the synthesized plurality of compounds is tested in a wet lab assay to determine whether they have bioactivity against a therapeutic target.

[0043] In some embodiments, the plurality of compounds is synthesized and the synthesized plurality of compounds is tested in a wet lab assay to determine whether they have potency against a therapeutic target.

[0044] In some embodiments, the therapeutic target is associated with a condition.

[0045] In some embodiments, the condition is inflammation or pain.

[0046] In some embodiments, the condition is a disease.

[0047] In some embodiments, the condition is a cancer, hematologic disorder, autoimmune disease, inflammatory disease, immunological disorder, metabolic disorder, neurological disorder, genetic disorder, psychiatric disorder, gastroenterological disorder, renal disorder, cardiovascular disorder, dermatological disorder, respiratory disorder, viral infection, or other disease or disorder.

[0048] The systems, methods, devices, and non-transitory computer readable storage medium of the present invention have other features and advantages that will be apparent from, or are set forth in more detail in, the accompanying drawings, which are incorporated herein, and the following Detailed Description, which together serve to explain certain principles of exemplary embodiments of the present invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[0049] Fig. 1 illustrates an exemplary system topology including a computer system for identifying a plurality of compounds in a combinatorial synthesis library that each satisfy a plurality of molecular properties, in accordance some embodiments of the present disclosure.

[0050] Figs. 2A, 2B, 2C, 2D, 2E, and 2F collectively illustrate computer implemented methods for identifying a plurality of compounds in a combinatorial synthesis library that each satisfy a plurality of molecular properties in which optional elements are indicated by dashed boxes, in accordance with an embodiment of the represent disclosure.

[0051] Fig. 3A illustrates exponential growth of CSLs, which has unlocked quadrillions of compounds that are synthetically and cost-effectively accessible, with short lead times in accordance with the prior art.

[0052] Fig. 3B illustrates how combinatorial, synthon-growing searches rely on scoring partial molecules (top: triangle + rectangle; bottom: triangle + square) to prune the search, which can lead to suboptimal selection, in accordance with the prior art

[0053] Fig. 3C illustrates how methods for combinatorically growing compounds are incapable of optimizing biochemical properties that exhibit non-additivity across chemical groups, for example solubility, in accordance with the prior art.

[0054] Fig. 4 illustrates a computer implemented method for identifying a plurality of compounds in a combinatorial synthesis library that each satisfy a plurality of molecular properties in which optional elements are indicated by dashed boxes, in accordance with an embodiment of the present disclosure.

[0055] Fig. 5A illustrates how NeuralGenThesis (NGT) starts by training a combinatorial synthesis library variational auto-encoder (*e.g.*, CSLVAE), which is comprised of (i) an encoder $q_\phi(z|x)$, parameterized as a graph neural network, mapping molecular graphs, $x \in \mathcal{X}$, into a learned d -dimensional latent space, and (ii) a specially architected decoder $p_\theta(x|z, \mathcal{D})$, which efficiently retrieves compounds from a combinatorial synthetic library (CSL), $\mathcal{D}$, given a query vector, $z \in \mathbb{R}^d$. The architecture of the decoder module 114 ensures that all compounds generated by CSLVAE are contained in the library and acts as a “neural database” for the CSL, permitting probabilistic access to compounds in the library through queries in the latent space. CSLVAE is trained end-to-end with backpropagation.

[0056] Fig. 5B illustrates how NGT subsequently trains a policy $\pi_\eta(z)$, parameterized as a normalizing flow, to identify positions in the latent space which, when decoded with the trained CSLVAE decoder $p_\theta(x|z, \mathcal{D})$, retrieve compounds from the CSL that satisfy a set of user-specified black-box constraints in accordance with an embodiment of the present disclosure.

[0057] Fig. 5C illustrates how NGT seeks a policy such that $\pi_\eta(z) \propto \mathbb{E}_{p_\theta(x|z, \mathcal{D})}[R_0(x)]$, where the binarized reward $[R_0(x)]$ is equal to one if molecule x satisfies all of the specified constraints and is equal to zero otherwise. Once trained, latent codes are sampled from the policy and decoded by the decoder module 114 to find compounds that meet the search criteria.

[0058] Fig. 6 illustrates summary statistics from sampling 10,000 molecules from each policy using the UnREAL library in accordance with an embodiment of the present disclosure. Means and standard deviations are calculated using results from five distinct attempts. Fig. 11 illustrates the constraints for each of the rules illustrated in Fig. 6.

[0059] Figs. 7A and 7B illustrate enrichment and compound diversity in virtual screening with a synthetic oracle. The number of unique reactions (r1) and unique synthons at each reaction position (s1-4) for ASAHI (7A, top) and CDK5 (7B, top) compounds that satisfy NGT constraints. UMAP plots representing Tanimoto distances for ASAHI (7A, bottom)

and CDK5 (7B, bottom) compounds, in accordance with an embodiment of the present disclosure.

[0060] Figs. 7C, 7D, and 7E illustrate enrichment plots for bioactivity (7C), potency (7D), and rank-averaged bioactivity and potency (7E), in accordance with an embodiment of the present disclosure.

[0061] Fig. 8A illustrates an MC2R conformational ensemble that includes activated MC2R domain of the ACTH:MC2R:MRAP1:Gs complex with bound ACTH (spheres), an MC2R AlphaFold model, and a homology model derived from an MC4R crystal structure.

[0062] Fig. 8B illustrates AtomNet bioactivity classification scores and predicted potency for the top 400 compounds selected by rank-averaging.

[0063] Fig. 8C illustrates activity measured from single concentration (left) and concentration response curves (right). Note that for NGT not all single concentration active compounds were measured in concentration response assays.

[0064] Fig. 8D illustrates chemical structures and concentration response curves for the most potent hits from NGT (left) and HE (right).

[0065] Fig. 8E illustrates diversity of hits.

[0066] Fig. 8F illustrates MCXR selectivity of compounds in percent inhibition at 50 uM.

[0067] Fig. 8G illustrates a representative chemical structure for a MC2R antagonist from the patent literature (prior art).

[0068] Fig. 9 illustrates a function for numerically stable Monte Carlo estimation of $D_{\alpha}(r_{\tau} \parallel \pi_{\eta})$ given $\log \pi_{\eta}(z|\tau)$, $\log \beta(z|\tau)$, and $\log R_{\tau}(z)$ evaluated over a minibatch of n samples drawn from $\beta(z|\tau)$, in accordance with an embodiment of the present disclosure.

[0069] Fig. 10 illustrates the number of compounds and clusters retrieved by NGT and ASAH1, in accordance with an embodiment of the present disclosure.

[0070] Fig. 11 illustrates drug-likeness constraints for various rules. The column QED represents the weighting factors.

[0071] Fig. 12 illustrates concentration response EC50s (MC2R) and EC50 estimates from single concentration (MC3/4/5R) cell based antagonist cAMP screens, in accordance with an embodiment of the present disclosure.

[0072] Fig. 13 illustrates surrogate gaps across NeuralGenThesis outer training iterations from a CDK5 experiment, in accordance with an embodiment of the present disclosure.

[0073] It should be understood that the appended drawings are not necessarily to scale, presenting a somewhat simplified representation of various features illustrative of the basic principles of the invention.

[0074] Figs. 14A, 14B, 14C, 14D, 14E, and 14F illustrate compounds selected in one embodiment to the present disclosure, in which column (1) is ATMW compound identifier number (*e.g.*, 1 is ATMW-2, 2 is ATMW-3, *etc.*), column (2) is the explorer method (HE = hierarchical enumeration (HE) described in Example 4; NGT = NeuralGenThesis, *e.g.*, Figure 4) used to identify each compound, column (3) is the simplified molecular input line entry system (SMILES) descriptor for each compound, column (4) is percent inhibition of control value for each compound, column (5) is whether each compound is active (T) or not (F), column (6) is CRC status (CRA A = CRC active; CRC I = CRC inactive), and column (7) is cluster number of each compound in accordance with an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0075] The present disclosure makes use of combinatorial synthesis libraries (CSLs) directly as large chemical universes for training a generative artificial intelligence with time- and cost-effective synthesizability guarantees. In some embodiments an autoencoder, which includes (i) an encoder module 110 for encoding chemical compounds in the CSL into d -dimensional codes in latent space on a probabilistic basis and (ii) a decoder module 114 for decoding d -dimensional codes into chemical structure, is coupled with a normalizing flow reinforcement learning agent that learns to sample from a distribution over the CSL induced by a reward function representing desirable molecular properties. When there are few or no known compounds interacting with a therapeutic target, obtaining a diverse set of hits is important for subsequent development. Therefore, the policy is specifically designed for efficient exploration and exploitation, avoiding mode concentration. Because the disclosed framework utilizes neural network based generative models to search over synthetically accessible chemical spaces, some embodiments of the claimed approach are termed NeuralGenThesis (NGT, Figure 4).

[0076] As disclosed in the examples section, NGT efficiently identified novel, potent antagonists for Melanocortin Type 2 Receptor (MC2R) in a CSL containing over three trillion compounds. MC2R is a G-protein coupled receptor primarily found in the adrenal cortex. Modulating the activity of MC2R impacts the production of cortisol, which could be beneficial in medical conditions associated with dysregulated cortisol levels, such as adrenal insufficiency or Cushing's syndrome [33, 34].

[0077] Reference will now be made in detail to embodiments, examples of which are illustrated in the accompanying drawings. In the following detailed description, numerous specific details are set forth in order to provide a thorough understanding of the present disclosure. However, it will be apparent to one of ordinary skill in the art that the present disclosure may be practiced without these specific details. In other instances, well-known methods, procedures, and components have not been described in detail so as not to unnecessarily obscure aspects of the embodiments.

[0078] It will also be understood that, although the terms first, second, *etc.* may be used herein to describe various elements, these elements should not be limited by these terms. These terms are only used to distinguish one element from another. For instance, a first subject could be termed a second subject, and, similarly, a second subject could be termed a first subject, without departing from the scope of the present disclosure. The first subject and the second subject are both subjects, but they are not the same subject.

[0079] The terminology used in the present disclosure is for the purpose of describing particular embodiments only and is not intended to be limiting of the invention. As used in the description of the invention and the appended claims, the singular forms "a," "an," and "the" are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will also be understood that the term "and/or" as used herein refers to and encompasses any and all possible combinations of one or more of the associated listed items. It will be further understood that the terms "comprises" and/or "comprising," when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof.

[0080] The foregoing description included example systems, methods, techniques, instruction sequences, and computing machine program products that embody illustrative implementations. For purposes of explanation, numerous specific details are set forth in order to provide an understanding of various implementations of the inventive subject matter.

It will be evident, however, to those skilled in the art that implementations of the inventive subject matter may be practiced without these specific details. In general, well-known instruction instances, protocols, structures, and techniques have not been shown in detail.

[0081] The foregoing description, for purpose of explanation, has been described with reference to specific implementations. However, the illustrative discussions below are not intended to be exhaustive or to limit the implementations to the precise forms disclosed. Many modifications and variations are possible in view of the above teachings. The implementations are chosen and described in order to best explain the principles and their practical applications, to thereby enable others skilled in the art to best utilize the implementations and various implementations with various modifications as are suited to the particular use contemplated.

[0082] In the interest of clarity, not all of the routine features of the implementations described herein are shown and described. It will be appreciated that, in the development of any such actual implementation, numerous implementation-specific decisions are made in order to achieve the designer's specific goals, such as compliance with use case- and business-related constraints, and that these specific goals will vary from one implementation to another and from one designer to another. Moreover, it will be appreciated that such a design effort might be complex and time-consuming, but nevertheless be a routine undertaking of engineering for those of ordinary skill in the art having the benefit of the present disclosure.

[0083] As used herein, the term "if" may be construed to mean "when" or "upon" or "in response to determining" or "in response to detecting," depending on the context. Similarly, the phrase "if it is determined" or "if [a stated condition or event] is detected" may be construed to mean "upon determining" or "in response to determining" or "upon detecting [the stated condition or event]" or "in response to detecting [the stated condition or event]," depending on the context.

[0084] As used herein, the term "about" or "approximately" can mean within an acceptable error range for the particular value as determined by one of ordinary skill in the art, which can depend in part on how the value is measured or determined, *e.g.*, the limitations of the measurement system. For example, "about" can mean within 1 or more than 1 standard deviation, per the practice in the art. "About" can mean a range of $\pm 20\%$, $\pm 10\%$, $\pm 5\%$, or $\pm 1\%$ of a given value. Where particular values are described in the application and claims,

unless otherwise stated, the term “about” means within an acceptable error range for the particular value. The term “about” can have the meaning as commonly understood by one of ordinary skill in the art. The term “about” can refer to $\pm 10\%$. The term “about” can refer to $\pm 5\%$.

[0085] As used interchangeably herein, the terms “classifier,” “model,” or “regressor” interchangeably refer to a machine learning model.

[0086] In some embodiments, a model is an unsupervised learning model. One example of an unsupervised learning model is cluster analysis. In some embodiments, a model includes supervised machine learning. Nonlimiting examples of supervised learning models include, but are not limited to, logistic regression models, neural networks, support vector machines, Naive Bayes models, nearest neighbors models, random forest models, decision trees, boosted trees, multinomial logistic regression models, linear models, linear regression models, Gradient Boosting models, mixture models, hidden Markov models, Gaussian NB models, linear discriminant analysis models, or any combinations thereof. In some embodiments, a model is a multinomial classifier algorithm. In some embodiments, a model is a 2-stage stochastic gradient descent (SGD) model. In some embodiments, a model is a deep neural network (*e.g.*, a deep-and-wide sample-level model).

[0087] *Neural networks.* In some embodiments, a model is a neural network (*e.g.*, a convolutional neural network and/or a residual neural network). Neural networks, also known as artificial neural networks (ANNs), include convolutional and/or residual neural networks (deep learning models). In some embodiments, neural networks are machine learning models that are trained to map an input dataset to an output dataset, where the neural network includes an interconnected group of nodes organized into multiple layers of nodes. For example, in some embodiments, the neural network architecture includes at least an input layer, one or more hidden layers, and an output layer. In some embodiments, the neural network includes any total number of layers, and any number of hidden layers, where the hidden layers function as trainable feature extractors that allow mapping of a set of input data to an output value or set of output values. In some embodiments, a deep learning model is a neural network including a plurality of hidden layers, *e.g.*, two or more hidden layers. In some instances, each layer of the neural network includes a number of nodes (or “neurons”). In some embodiments, a node receives input that comes either directly from the input data or the output of nodes in previous layers, and performs a specific operation, *e.g.*, a summation operation. In some embodiments, a connection from an input to a node is associated with a

parameter (*e.g.*, a weight and/or weighting factor). In some embodiments, the node sums up the products of all pairs of inputs, x_i , and their associated parameters. In some embodiments, the weighted sum is offset with a bias, b . In some embodiments, the output of a node or neuron is gated using a threshold or activation function, f , which, in some instances, is a linear or non-linear function. In some embodiments, the activation function is, for example, a rectified linear unit (ReLU) activation function, a Leaky ReLU activation function, or other function such as a saturating hyperbolic tangent, identity, binary step, logistic, arcTan, softsign, parametric rectified linear unit, exponential linear unit, softPlus, bent identity, softExponential, Sinusoid, Sine, Gaussian, or sigmoid function, or any combination thereof.

[0088] In some implementations, the weighting factors, bias values, and threshold values, or other computational parameters of the neural network, are “taught” or “learned” in a training phase using one or more sets of training data. For example, in some implementations, the parameters are trained using the input data from a training dataset and a gradient descent or backward propagation method so that the output value(s) that the ANN computes are consistent with the examples included in the training dataset. In some embodiments, the parameters are obtained from a back propagation neural network training process.

[0089] Any of a variety of neural networks are suitable for use in accordance with the present disclosure. Examples include, but are not limited to, feedforward neural networks, radial basis function networks, recurrent neural networks, residual neural networks, convolutional neural networks, residual convolutional neural networks, and the like, or any combination thereof. In some embodiments, the machine learning makes use of a pre-trained and/or transfer-learned ANN or deep learning architecture. In some implementations, convolutional and/or residual neural networks are used, in accordance with the present disclosure.

[0090] For instance, a deep neural network model includes an input layer, a plurality of individually parameterized (*e.g.*, weighted) convolutional layers, and an output scorer. The parameters (*e.g.*, weights) of each of the convolutional layers as well as the input layer contribute to the plurality of parameters (*e.g.*, weights) associated with the deep neural network model. In some embodiments, at least 50 parameters, at least 100 parameters, at least 1000 parameters, at least 2000 parameters or at least 5000 parameters are associated with the deep neural network model. As such, deep neural network models require a computer to be used because they cannot be mentally solved. In other words, given an input to the model, the model output needs to be determined using a computer rather than mentally in such embodiments. *See*, for example, Krizhevsky *et al.*, 2012, “Imagenet classification with deep

convolutional neural networks,” in *Advances in Neural Information Processing Systems 2*, Pereira, Burges, Bottou, Weinberger, eds., pp. 1097-1105, Curran Associates, Inc.; Zeiler, 2012 “ADADELTA: an adaptive learning rate method,” CoRR, vol. abs/1212.5701; and Rumelhart *et al.*, 1988, “Neurocomputing: Foundations of research,” ch. Learning Representations by Back-propagating Errors, pp. 696-699, Cambridge, MA, USA: MIT Press, each of which is hereby incorporated by reference.

[0091] Neural networks, including convolutional neural networks, suitable for use as models in the present disclosure are disclosed in, for example, Vincent *et al.*, 2010, “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion,” J Mach Learn Res 11, pp. 3371-3408; Larochelle *et al.*, 2009, “Exploring strategies for training deep neural networks,” J Mach Learn Res 10, pp. 1-40; and Hassoun, 1995, *Fundamentals of Artificial Neural Networks*, Massachusetts Institute of Technology, each of which is hereby incorporated by reference. Additional example neural networks suitable for use as models are disclosed in Duda *et al.*, 2001, *Pattern Classification*, Second Edition, John Wiley & Sons, Inc., New York; and Hastie *et al.*, 2001, *The Elements of Statistical Learning*, Springer-Verlag, New York, each of which is hereby incorporated by reference in its entirety. Additional example neural networks suitable for use as models in the present disclosure are also described in Draghici, 2003, *Data Analysis Tools for DNA Microarrays*, Chapman & Hall/CRC; and Mount, 2001, *Bioinformatics: sequence and genome analysis*, Cold Spring Harbor Laboratory Press, Cold Spring Harbor, New York, each of which is hereby incorporated by reference in its entirety.

[0092] As used herein, the term “parameter” refers to any coefficient or, similarly, any value of an internal or external element (*e.g.*, a weight and/or a hyperparameter) in a model, regressor, and/or classifier that can affect (*e.g.*, modify, tailor, and/or adjust) one or more inputs, outputs, and/or functions in the model, regressor and/or classifier. For example, in some embodiments, a parameter refers to any coefficient, weight, and/or hyperparameter that can be used to control, modify, tailor, and/or adjust the behavior, learning, and/or performance of a model, regressor, and/or classifier. In some instances, a parameter is used to increase or decrease the influence of an input (*e.g.*, a feature) to the model, regressor, and/or classifier. As a nonlimiting example, in some embodiments, a parameter is used to increase or decrease the influence of a node (*e.g.*, of a neural network), where the node includes one or more activation functions. Assignment of parameters to specific inputs, outputs, and/or functions is not limited to any one paradigm for a given model, regressor,

and/or classifier but can be used in any suitable model, regressor, and/or classifier architecture for a desired performance. In some embodiments, a parameter has a fixed value. In some embodiments, a value of a parameter is manually and/or automatically adjustable. In some embodiments, a value of a parameter is modified by a validation and/or training process for an algorithm, model, regressor, and/or classifier (*e.g.*, by error minimization and/or backpropagation methods). In some embodiments, a model, regressor, and/or classifier of the present disclosure includes a plurality of parameters. In some embodiments, the plurality of parameters is n parameters, where: $n \geq 2$; $n \geq 5$; $n \geq 10$; $n \geq 25$; $n \geq 40$; $n \geq 50$; $n \geq 75$; $n \geq 100$; $n \geq 125$; $n \geq 150$; $n \geq 200$; $n \geq 225$; $n \geq 250$; $n \geq 350$; $n \geq 500$; $n \geq 600$; $n \geq 750$; $n \geq 1,000$; $n \geq 2,000$; $n \geq 4,000$; $n \geq 5,000$; $n \geq 7,500$; $n \geq 10,000$; $n \geq 20,000$; $n \geq 40,000$; $n \geq 75,000$; $n \geq 100,000$; $n \geq 200,000$; $n \geq 500,000$, $n \geq 1 \times 10^6$, $n \geq 5 \times 10^6$, or $n \geq 1 \times 10^7$. As such, some embodiments of the models, regressors, and/or classifiers of the present disclosure cannot be mentally performed. In some embodiments n is between 10,000 and 1×10^7 , between 100,000 and 5×10^6 , or between 500,000 and 1×10^6 . In some embodiments, the models, regressors, and/or classifiers of the present disclosure operate in a k -dimensional space, where k is a positive integer of 5 or greater (*e.g.*, 5, 6, 7, 8, 9, 10, *etc.*). As such, some embodiments of the models, regressors, and/or classifiers of the present disclosure cannot be mentally performed.

[0093] *Ensembles of models and boosting.* In some embodiments, an ensemble (two or more) of models is used. In some embodiments, a boosting technique such as AdaBoost is used in conjunction with many other types of learning algorithms to improve the performance of the model. In some such approaches, the output of any of the models disclosed herein, or their equivalents, is combined into a weighted sum that represents the final output of the boosted model. In some embodiments, the plurality of outputs from the models is combined using any measure of central tendency known in the art, including but not limited to a mean, median, mode, a weighted mean, weighted median, weighted mode, *etc.* In some embodiments, the plurality of outputs is combined using a voting method. In some embodiments, a respective model in the ensemble of models is weighted or unweighted.

[0094] Several aspects are described below with reference to example applications for illustration. It should be understood that numerous specific details, relationships, and methods are set forth to provide a full understanding of the features described herein. One having ordinary skill in the relevant art, however, will readily recognize that the features described herein can be practiced without one or more of the specific details or with other

methods. The features described herein are not limited by the illustrated ordering of acts or events, as some acts can occur in different orders and/or concurrently with other acts or events. Furthermore, not all illustrated acts or events are required to implement a methodology in accordance with the features described herein.

[0095] *Exemplary System Embodiments*

[0096] Details of an exemplary system are now described in conjunction with Fig. 1. Fig. 1 is a block diagram illustrating a system 100 in accordance with some implementations. The device 100 in some implementations includes at least one or more processing units CPU(s) 52 (also referred to as processors), one or more network interfaces 54, a user interface 56, having a display 58, an input device 60, a memory 61/62, and one or more communication buses 64 for interconnecting these components. The one or more communication buses 64 optionally include circuitry (sometimes called a chipset) that interconnects and controls communications between system components. The memory may be a non-persistent memory 61, a persistent memory 62, or any combination thereof. The non-persistent memory 61 typically includes high-speed random access memory, such as DRAM, SRAM, DDR RAM, ROM, EEPROM, flash memory, whereas the persistent memory typically includes CD-ROM, digital versatile disks (DVD) or other optical storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, magnetic disk storage devices, optical disk storage devices, flash memory devices, or other non-volatile solid-state storage devices. Regardless of its specific implementation, the non-persistent memory 61 and/or the persistent memory 62 comprises at least one non-transitory computer-readable storage medium, and it stores thereon computer-executable instructions which can be in the form of programs, modules, and data structures.

[0097] In the present disclosure, unless expressly stated otherwise, descriptions of devices and systems will include implementations of one or more computers, such as one or more processing units (*e.g.*, one or more central processing units and/or one or more graphics processing units). For instance, and for purposes of illustration in Fig. 1, a computer system 100 is represented as a single device that includes all the functionality of the computer system 100. However, the present disclosure is not limited thereto. For instance, in some embodiments, the functionality of the computer system 100 is spread across any number of networked computers and/or resides on each of several networked computers and/or is hosted on one or more virtual machines and/or containers at a remote location accessible across a communications network 55. One of skill in the art will appreciate that a wide array of

different computer topologies is possible for the computer system 100, and other devices and systems of the present disclosure, and that all such topologies are within the scope of the present disclosure. Moreover, rather than relying on a physical communications network 55, the illustrated devices and systems may wirelessly transmit information between each other. As such, the exemplary topology shown in Figure 1 merely serves to describe the features of an embodiment of the present disclosure in a manner that will be readily understood to one of skill in the art.

[0098] Fig. 1 depicts a block diagram of a distributed computer system (*e.g.*, computer system 100) according to some embodiments of the present disclosure. The computer system 100 at least facilitates communicating one or more instructions for identifying a plurality of compounds in a combinatorial synthesis library that each satisfy a plurality of molecular properties.

[0099] In some embodiments, the communication network 55 optionally includes the Internet, one or more local area networks (LANs), one or more wide area networks (WANs), other types of networks, or a combination of such networks.

[00100] Examples of communication networks 55 include the World Wide Web (WWW), an intranet and/or a wireless network, such as a cellular telephone network, a wireless local area network (LAN) and/or a metropolitan area network (MAN), and other devices by wireless communication. The wireless communication optionally uses any of a plurality of communications standards, protocols and technologies, including Global System for Mobile Communications (GSM), Enhanced Data GSM Environment (EDGE), high-speed downlink packet access (HSDPA), high-speed uplink packet access (HSUPA), Evolution, Data-Only (EV-DO), HSPA, HSPA+, Dual-Cell HSPA (DC-HSPDA), long term evolution (LTE), near field communication (NFC), wideband code division multiple access (W-CDMA), code division multiple access (CDMA), time division multiple access (TDMA), Bluetooth, Wireless Fidelity (Wi-Fi) (*e.g.*, IEEE 802.11a, IEEE 802.11ac, IEEE 802.11ax, IEEE 802.11b, IEEE 802.11g and/or IEEE 802.11n), voice over Internet Protocol (VoIP), Wi-MAX, a protocol for e-mail (*e.g.*, Internet message access protocol (IMAP) and/or post office protocol (POP)), instant messaging (*e.g.*, extensible messaging and presence protocol (XMPP), Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions (SIMPLE), Instant Messaging and Presence Service (IMPS)), and/or Short Message Service (SMS), or any other suitable communication protocol, including communication protocols not yet developed as of the filing date of this document.

[00101] In various embodiments, the computer system 100 includes one or more processing units, such as one or more central processing units (CPUs) and/or one or more graphics processing units (GPUs), a network or other communications interface 54, and memory 61/62.

[00102] In some embodiments, the computer system 100 includes a user interface 56. The user interface 56 typically includes a display 58 for presenting media. In some embodiments, the display 58 is integrated within the computer systems (*e.g.*, housed in the same chassis as the CPU 52 and memory 61/62). In some embodiments, the computer system 100 includes one or more input device(s) 60, which allow a subject to interact with the computer system 100. In some embodiments, input devices 60 include a keyboard, a mouse, and/or other input mechanisms. Alternatively, or in addition, in some embodiments, the display 58 includes a touch-sensitive surface (*e.g.*, where display 58 is a touch-sensitive display or computer system 100 includes a touch pad).

[00103] In some embodiments, the computer system 100 presents media to a user through the display 58. Examples of media presented by the display 58 include one or more images, a video, audio (*e.g.*, waveforms of an audio sample), or a combination thereof. In typical embodiments, the one or more images, the video, the audio, or the combination thereof is presented by the display 58 through a client application. In some embodiments, the audio is presented through an external device (*e.g.*, speakers, headphones, input/output (I/O) subsystem, *etc.*) that receives audio information from the computer system 100 and presents audio data based on this audio information. In some embodiments, the user interface 56 also includes an audio output device, such as speakers or an audio output for connecting with speakers, earphones, or headphones.

[00104] Memory 61/62 includes high-speed random access memory, such as DRAM, SRAM, DDR RAM, or other random access solid state memory devices, and optionally also includes non-volatile memory, such as one or more magnetic disk storage devices, optical disk storage devices, flash memory devices, or other non-volatile solid state storage devices. Memory 61/62 may optionally include one or more storage devices remotely located from the CPU(s) 52. Memory 61/62, or alternatively the non-volatile memory device(s) within memory 61/62, includes a non-transitory computer readable storage medium. Access to memory 61/62 by other components of the computer system 100, such as the CPU(s) 52, is, optionally controlled by a controller. In some embodiments, the memory 61/62 include mass storage that is remotely located with respect to the CPU(s) 52. In other words, some data stored in

memory 61/62 may in fact be hosted on devices that are external to the computer system 100, but that can be electronically accessed by the computer system 100 over an Internet, intranet, or other form of network 55 or electronic cable using network interface 54.

[0001] In some embodiments, as shown in Fig. 1, the non-persistent memory 61 stores the following:

- optional instructions, programs, data or information associated with optional operating system 102, which includes procedures for handling various basic system services and for performing hardware-dependent tasks;
- programs, data or information associated with a combinatorial synthesis library 104 including compounds 106-1, ..., 106-N, wherein N is a positive integer;
- programs, data or information associated with an autoencoder 108, where the autoencoder includes an encoder 110 that encodes a molecule into a latent space 112, and a decoder module 114 that decodes the molecule from the latent space ;
- programs, data or information associated with a policy model 116 that comprises parameters 118-1, ..., 118-M;
- Iterations 120-1, ..., 120-L, each interaction comprising a batch of compounds 122 (e.g., 122-1, 122-2, ..., 122-L), each batch of compounds including a plurality of compounds (e.g., 124-1-1, 124-1-2, ..., 124-1-X) and, for each such compound, a molecular property value 126 (e.g., 126-1-1-1, ..., 126-1-1-Q) for each molecular property in a plurality of molecular properties, and, for each such compound, a tempered reward function value 128 (e.g., 128-1-1);
- programs, data or information associated with a current probability density 130;
- programs, data or information associated with a target distribution 132; and
- programs, data or information associated with an α -divergence based objection function 134.

[00105] It should be appreciated that the computer system of Fig. 1 is only one example of a computer system 100, and that the computer system 100 optionally has more or fewer components than shown, optionally combines two or more components, or optionally has a different configuration or arrangement of the components. The various components shown in Fig. 1 are implemented in hardware, software, firmware, or a combination thereof, including one or more signal processing and/or application specific integrated circuits.

[00106] In various implementations, one or more of the above-identified elements are stored in one or more of the previously mentioned memory devices, and correspond to a set of instructions for performing various methods described herein. The above-identified modules, data, or programs (*e.g.*, sets of instructions) may not be implemented as separate software programs, procedures, datasets, or modules, and thus various subsets of these modules and data may be combined or otherwise re-arranged in various implementations. In some implementations, the non-persistent memory 61 optionally stores a subset of the modules and data structures identified above. Furthermore, in some embodiments, the memory stores additional modules and data structures not described above. In some embodiments, one or more of the above-identified elements are stored in a computer system, other than that of the system 100, that is addressable by the system 100 so that the system 100 may retrieve all or a portion of such data.

[00107] Although Figure 1 depicts a “system 100,” the figure is intended more as functional description of the various features which may be present in computer systems than as a structural schematic of the implementations described herein. In practice, and as recognized by those of ordinary skill in the art, items shown separately could be combined and some items can be separate. Moreover, although Fig. 1 depicts certain data and modules in the memory (which can be non-persistent memory 61 or persistent memory 62), it should be appreciated that these data and modules, or portion(s) thereof, may be stored in more than one memory.

[00108] Various modules in a memory of a computer system and/or a memory of a client device perform certain processes of the methods of the present disclosure, unless expressly stated otherwise. Furthermore, it will be appreciated that the processes of a method of the present disclosure can be encoded in a single module or any combination of modules.

[00109] Now that a general topology of a system 100 has been described in accordance with various embodiments of the present disclosures, details regarding some processes and methods of the present disclosure will be described in conjunction with Fig. 2.

[00110] Referring to block 200 of Fig. 2, computer-implemented methods for identifying a plurality of compounds in a combinatorial synthesis library 104 that each satisfy a plurality of molecular properties are provided.

[00111] In some embodiments, the systems and methods of the present disclosure define $a : X \rightarrow \mathbb{R}^k$ to be a vector-valued function that evaluates k molecular properties of a given query

molecule. In some embodiments, k is 1, 2, 3, 4, 5, 6, 7, 8, 9 or 10, or more than 10. In some embodiments, it is assumed, 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, with respect to the plurality of molecular properties if $a(x) \geq b$.

[00112] In some embodiments, X denotes a set of potentially pharmacologically active compounds, colloquially referred to as a “chemical space.” The set of all compounds contained in a CSL $\mathcal{D}$ 104 is denoted by $X_{\mathcal{D}} \subset X$. Exemplary details on the construction of CSLs 104 as well as specifics about the featurization of compounds in some embodiments of the present disclosure are found in in [32].

[00113] The combinatorial synthesis library is accessible through a trained variational auto-encoder 108 that comprises (i) an encoder module 110 that maps a compound 106/124 in the combinatorial synthesis library into a d -dimensional code in a learned d -dimensional latent space and (ii) a decoder module 114 that retrieves a molecular structure of the compound 106/124, with a reconstruction likelihood greater than a threshold reconstruction likelihood, upon inputting the d -dimensional code into the decoder module.

[00114] In some embodiments, the combinatorial synthesis library (CSL) variational auto-encoder 108 is trained. In some embodiments the variational auto-encoder 108 is CSLVAE which is further described in [32], which is hereby incorporated by reference. In Pedawi *et al.*, 2022 [32], the combinatorial synthesis variational auto-encoder comprises an 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*. In some embodiments, the architecture of the decoder module 114 guarantees that $\text{supp}(p_{\theta}) \subseteq X_{\mathcal{D}}$ by construction, *e.g.*, that only compounds in $X_{\mathcal{D}}$ are reachable via the decoder module 114.

[00115] As outlined in Figures 1, 4 and 5A, the variational auto-encoder 108 automaps molecular graphs $x \in X$ into a learned d -dimensional latent space 112 with an encoder 110 $q_{\psi}(z|x)$ parameterized, in some embodiments, by a graph neural network. Given a query vector z (d -dimensional code) $\in \mathbb{R}^d$, the decoder $p_{\theta}(x|z, \mathcal{D})$ 114 efficiently retrieves a compound from the CSL $\mathcal{D}$ 104. The architecture of the decoder 114 ensures that all compounds generated by the autoencoder 110 are contained in the CSL 104. The auto-encoder 108 thereby permits probabilistic access to compounds in the CSL 104 through

queries in the latent space 112, efficiently encoding the CSL as a neural database. In some embodiments, the auto-encoder 108 is trained end-to-end with backpropagation.

[00116] Referring to block 202, in some embodiments, the threshold reconstruction likelihood is 40 percent or greater, 50 percent or greater, 60 percent or greater, or between 60 percent and 80 percent. In some embodiments, the threshold reconstruction likelihood is 40 percent, 50 percent, 60 percent, 70 percent, 80 percent, 90 percent, or between 60 percent and 95 percent. In some embodiments, the threshold reconstruction likelihood is a percentage that is below 40 percent, a percentage that is below 50 percent, a percentage that is below 60 percent, a percentage that is below 70 percent, a percentage that is below 80 percent, or a percentage that is below 90 percent.

[00117] *Generative molecular reconstruction.* Whether the auto-encoder 108 faithfully reconstructs queries from the CSL [32] was evaluated. All compounds generated through use of the encoder module 110 by the policy model 116, by construction, were chemically valid. To quantitatively understand a measure of recall for use of the encoder module 110 by the policy model 116, the average per-compound reconstruction likelihoods were calculated across a large number of diverse products sampled from the library 104. This metric assesses the decoder module's 114 ability to accurately reconstruct the complete molecular graph of a compound and approximates the probability of successfully reproducing a query compound through the decoder module 114. The average top-1 reconstruction accuracy is in excess of 62%, suggesting that the policy model 116, in conjunction with the auto-encoder 108, adequately covers UnREAL (Table 1):

[00118] Table 1 – Recall statistics for the encoder module 110 by the policy model 116 on the three trillion compound UnREAL library. Ten thousand compounds were sampled uniformly random over the library, encoded with encoder module 110, and then decoded with the decoder module 114 using the argmax decision rule for 1, 10, 100, and 1000 retrievals. The percent of compounds that were accurately reconstructed in this way is reported.

Top 1	Top 10	Top 100	Top 1000
62.7%	80.2%	91.2%	95.0%

[00119] *Calculated Properties.* The present disclosure allows for the search for compounds in CSL 104 that simultaneously satisfy a plurality of molecular properties, such as those listed in Fig. 6. In some embodiments, this is done by uniformly sampling from the subset of

a CSL 104 that satisfies the constraints imposed by the plurality of molecular properties. An optimization procedure for training an autoencoder 108 with a normalizing flow policy to approximate such a target distribution is disclosed in [35], which is hereby incorporated by reference

[00120] In one example, 100 training episodes were conducted, each consisting of 1,000 samples from a most recent policy iteration 120 (described in further detail below) and employing off-policy training. This example makes use of a replay buffer containing the 32 most recent episodes and a replay ratio of 50. For each of the policies, 10,000 compounds were sampled and the satisfiability rate—the proportion of sampled compounds that satisfied the specified constraints—was evaluated as well as the number of distinct compound clusters found among the constraint-satisfying retrievals as a measure of diversity. For this latter statistic, sequential Butina clustering with the ECFP4 metric and a cutoff of 0.35 was used and the number of distinct clusters was counted.

[00121] The results were compared with two baseline scenarios that reflect the complexities associated with enumeration in large CSLs (Fig. 6). The random (uniform) baseline randomly samples from the CSL 104, while random (stratified) randomly samples reactions first and subsequently samples a product induced by that reaction. These results were compared to the implementation of hierarchical enumeration (HE) described in Example 4.

[00122] The random and stratified random baselines revealed that certain sets of constraints, such as Lee, Rule-of-3, and QED, exhibit very low (< 2%) satisfiability. HE (Example 4) generally showed higher satisfiability across most constraint sets compared to random baselines, although Lee and QED constraints still demonstrate low satisfiability (< 2%). HE combines fragments, which is ideal for optimizing additive properties like molecular weight. The low satisfiability observed when using HE for more complex constraint sets, such as QED, could stem from their non-additivity. Additionally, as described in Example 6, HE exhibited less diversity compared to the stratified random baseline, indicating a trade-off between satisfiability and diversity. The tendency of HE to produce similar compounds, indicating a bias toward exploitation over exploration, may harm virtual screening scenarios that often require diversity to help overcome attrition during later stages of drug discovery.

[00123] In all cases the trained policy with diversity sampling (Example 3) achieved the highest satisfiability rate and the largest number of distinct clusters. For filters where the prevalence in the library is low, significant enrichment was observed in both of these

quantities. Notably, for two filters with relatively high prevalence in the library (Lipinski and Ghose), sampling i.i.d. from the trained policy model 116 attained a lower satisfiability rate and cluster count than the stratified random baseline. With diversity sampling described in Example 6, the number of satisfied clusters increases two to four-fold, substantially outperforming the baselines and HE.

[00124] Referring to block 204, in some embodiments, a first molecular property in the plurality of molecular properties is a bioactivity prediction against a therapeutic target, and the first molecular property is satisfied for a respective compound 106/124 in the plurality of compounds when a bioactivity classification module determines, based on a chemical structure of the respective compound 106/124, that the respective compound 106/124 is an active compound 106/124 against the therapeutic target. For example, in some embodiments the bioactivity classification module is AtomNet®, a target-agnostic (“global”) system of models that predicts bioactivity and potency from docked receptor-ligand complexes [36–38]. The AtomNet® bioactivity module is a classification model and has been trained on active and inactive compounds. Using the AtomNet® bioactivity module to compute the first molecular property in the plurality of molecular properties in the methods summarized in Fig. 2 is computationally costly in some embodiments. Accordingly, in some embodiments the bioactivity classification module is a computationally tractable surrogate model that predicts “oracle” AtomNet® bioactivity scores from ligands alone. Example 5 provides an example of using a computationally tractable surrogate model that predicts “oracle” AtomNet® bioactivity scores from ligands alone.

[00125] Referring to block 206, in some embodiments, a first molecular property in the plurality of molecular properties is a potency prediction against a therapeutic target, and the first molecular property is satisfied for a respective compound 106/124 in the plurality of compounds when a potency prediction regression module determines, based on a chemical structure of the respective compound 106/124, that the respective compound 106/124 has at least a threshold potency against the therapeutic target. For example, in some embodiments the potency prediction regression module is AtomNet®, a target-agnostic (“global”) system of models that predicts bioactivity and potency from docked receptor-ligand complexes [36–38]. The AtomNet® potency module is a regression model trained on active compounds only. Using the AtomNet® potency module to compute the first molecular property in the plurality of molecular properties in the methods summarized in Figure 2 is computationally costly in some embodiments. Accordingly, in some embodiments the potency prediction

regression module is a computationally tractable surrogate model that predicts “oracle” AtomNet® potency scores from ligands alone. Example 5 provides an example of using a computationally tractable surrogate model that predicts “oracle” AtomNet® potency scores from ligands alone.

[0001] In some embodiments, a first molecular property in the plurality of molecular properties is a binding coefficient of the query compound to a target polymer. In some such embodiments the binding coefficient is an IC₅₀, EC₅₀, K_d, K_I, or pK_I for the query compound with respect to the target polymer. IC₅₀, EC₅₀, K_d, K_I, and pK_I are generally described in Huser *ed.*, 2006, *High-Throughput-Screening in Drug Discovery*, Methods and Principles in Medicinal Chemistry 35; and Chen *ed.*, 2019, *A Practical Guide to Assay Development and High-Throughput Screening in Drug Discovery*, each of which is hereby incorporated by reference.

[0002] In some embodiments, a first molecular property in the plurality of molecular properties is a binary score, where a first value for the binary score represents an IC₅₀, EC₅₀, K_d, K_I, or pK_I for the query compound with respect to a target polymer that is above a threshold, and a second value for the binary score represents an IC₅₀, EC₅₀, K_d, K_I, or pK_I for the test compound with respect to the target polymer that is below the threshold. In some embodiments, the threshold is between 1 picomolar and 100 micromolar. In some embodiments, the threshold is between 10 picomolar and 10 micromolar. In some embodiments, the threshold is between 0.01 micromolar and 1 micromolar.

[00126] Referring to block 208, in some embodiments, the plurality of compounds comprises 2, 3, 4, 5, 6, 7, 8, 9 or 10 compounds. In some embodiments, the plurality of compounds comprises 20, 30, 40, 50, 60, 70, 80, 90 or 100 or more compounds. In some embodiments, the plurality of compounds comprises 500 or more compounds, 1000 or more compounds, 10,000 or more compounds, or 100,000 or more compounds. In some embodiments, the plurality of compounds comprises 1×10^6 , 5×10^6 , or 1×10^7 or more compounds.

[00127] Referring to block 210, in some embodiments, the combinatorial synthesis library 102 represents more than 10,000 synthons and more than 10 three- or four- component reactions.

[00128] A combinatorial synthesis library 102 represents a collection of compounds that are synthesized using diverse building blocks (synthons) and reaction methods to produce a wide array of complex compounds. Such libraries focus on creating a diverse set of compounds

that can be used for various applications such as drug discovery, material science, or other research purposes. Such libraries include a variety of synthons, which are building blocks or molecular fragments that can be combined in different ways to produce a wide range of compounds. In some embodiments the combinatorial synthesis library represents more than 100, 500, 1000, 2000, 5000, 10,000, 11,000, 12,000, 13,000, 14,000, 15,000, or 20,000 synthons. In some embodiments these synthons might include various functional groups, heterocycles, and other structural motifs. In some embodiments, the CSL 104 makes use of multicomponent reactions, which are chemical reactions where three or more reactants combine to form a single product. Multicomponent reactions are efficient and versatile, allowing the creation of complex compounds in fewer steps and with high atom economy. In some embodiments, the CSL 104 includes a range of reaction types such as condensation reactions, cycloadditions, and cross-coupling reactions, that can be used in multicomponent reactions to produce different classes of compounds. By using different combinations of synthons and multicomponent reactions, the CSL 104 generates a structurally diverse set of compounds. This diversity is used in the present disclosure for applications such as drug discovery, where a variety of chemical structures is needed to explore different biological targets. In some embodiments, the CSL 104 is not based explicitly on specific synthons or particular molecular reactions.

[00129] One example of a CSL 104 is the PubChem database (*see*, Kim *et al.*, 2021, “PubChem in 2021: new data content and improved web interfaces,” Nucleic Acids Research 8(49)(D1), pp. D1388-D1395, which is hereby incorporated by reference). Another example of a CSL 104 is the ZINC database (*see*, Irwin *et al.*, 2020, “ZINC20—A Free Ultralarge-Scale Chemical Database for Ligand Discovery,” J. Chem. Inf. Model 60(12), 6065-6073, which is hereby incorporated by reference). Another example of a CSL 104 database is the ChEMBL database. ChEMBL is a manually curated database of bioactive molecules with drug-like properties. *See*, Gaulton *et al.*, 2019, “ChEMBL: towards direct deposition of bioassay data,” Nucleic Acids Res. 47(D1), D930-D940, which is hereby incorporated by reference. Another example of a CSL 104 is MCULE. *See*, Kiss *et al.*, 2012, “Http://Mcule.Com: A Public Web Service for Drug Discovery,” J. Cheminformatics 4 (1), p.17). Another example of a CSL 104 is ENAMINE REAL. *See*, 6. Shivanyuk *et al.* Enamine real database: Making chemical diversity real. Chem Today [Internet]. 2007 [cited 2024 Apr 11];25(6). Available from: <https://elibrary.ru/item.asp?id=27792199>.

[00130] Referring to block 212, in some embodiments, the d -dimensional code has a latent space dimension d that is between 32 and 128. Referring to block 214, in some embodiments, the d -dimensional code has a latent space dimension d that is 64.

[00131] In some embodiments the encoder module 110 turns a graph representation of a compound into d -dimensional codes in the latent space 112, where such codes of d dimensions. This involves several steps that transform the molecular graph into a numerical representation (d -dimensional code). First, the graph representation of the compound includes a plurality of nodes connected by edges. The nodes represent the atoms of the compound and the edges represent the chemical bonds between atoms in the compound. In some embodiments, each node (atom) has associated features such as element type, hybridization state, and other atomic properties. In some embodiments, each edge (bond) has associated features such as bond type (single, double, *etc.*) and bond stereochemistry. In some embodiments, the encoder module 110 takes the graph as input and learns, through optimization of the parameters of the autoencoder module 110, to propagate information from neighboring nodes and edges through the graph. After processing the graph, a readout function of the encoder module 110 is applied to aggregate the node embeddings into a single graph-level embedding (d -dimensional code).

[00132] *A. Selecting a current batch of d -dimensional codes.*

[00133] Referring to block 216, a policy model 116, comprising a plurality of parameters 118, is used to select a current batch 122 of d -dimensional codes in the learned d -dimensional latent space.

[00134] Referring to block 218, in some embodiments, the policy model 116 is a normalizing flow. Normalizing flows are a family of methods in the field of machine learning that involve learning complex probability distributions using a series of invertible transformations. A normalizing flow models data $\mathbf{x}$ as the output of an invertible, differentiable transformation $\mathbf{f}$ of noise $\mathbf{u}$:

$$\mathbf{x} = \mathbf{f}(\mathbf{u}) \text{ where } \mathbf{u} \sim \pi(\mathbf{u}).$$

The probability density of $\mathbf{x}$ under the flow is obtained by a change of variables:

$$p(\mathbf{x}) = \pi(\mathbf{f}^{-1}(\mathbf{x})) \left| \det \left(\frac{\partial \mathbf{f}^{-1}}{\partial \mathbf{x}} \right) \right|.$$

Intuitively, the function $\mathbf{f}$ compresses and expands the density of the noise distribution $\pi(\mathbf{u})$, and this change is quantified by the determinant of the Jacobian of the transformation. The noise distribution $\pi(\mathbf{u})$ is typically chosen to be simple, such as a standard normal, whereas the transformation $\mathbf{f}$ and its inverse $\mathbf{f}^{-1}$ are often implemented by composing a series of invertible neural-network modules. Given a dataset $\mathcal{D} = \{\mathbf{x}^{(n)}\}_{n=1}^N$, the flow is trained by maximizing the total log likelihood $\sum_n \log p(\mathbf{x}^{(n)})$ with respect to the parameters of the transformation. See, Durken *et al.*, “Neural Spline Flows,” Advances in Neural Information Processing Systems 32 (NeurIPS 2019), which is hereby incorporated by reference.

[00135] Referring to block 220, in some embodiments, the normalizing flow is a neural spline flow. Neural spline flows have flexibility and relative ease of implementation. However any normalizing flow can be implemented through various embodiments of the policy model 116. See, for example, Kobyzev *et al.*, 2019, “Normalizing Flows: Introduction and Ideas,” arXiv:1908.09257v1 [stat.ML] 25 Aug 2019.

[00136] For instance, in some embodiments rather than a neural spline flow, the policy model 116 implements Real-valued Non-Volume Preserving (Real NVP) Transformations. This approach uses a combination of affine coupling layers and permutation layers to model distributions. Each coupling layer transforms half of the input dimensions at a time using an invertible function. See Dinh *et al.*, 2017, “Density Estimation Using REAL NVP,” arXiv:1605.08803v3 [cs.LG] 27 Feb 2017, which is hereby incorporated by reference.

[00137] As another example in some embodiments, rather than a neural spline flow, the policy model 116 implements Glow, which is a type of normalizing flow that builds upon Real NVP and introduces additional improvements, such as 1×1 invertible convolutions for mixing data and an efficient way of computing the Jacobian determinant. Glow uses affine coupling layers and invertible linear transformations. See Kingma and Dhariwal, 2018, “Glow: Generative Flow with Invertible 1×1 Convolutions,” 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada, which is hereby incorporated by reference.

[00138] As another example in some embodiments, rather than a neural spline flow, the policy model 116 implements masked autoregressive flow (MAF). MAF is a flow-based model that uses a series of masked autoregressive layers (similar to MADE, a masked autoencoder model). Each layer transforms the input in an autoregressive manner, allowing for the modeling of complex dependencies between dimensions. See Papamakarios *et al.*,

2017, “Masked Autoregressive Flow for Density Estimation,” 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, which is hereby incorporated by reference.

[00139] As another example in some embodiments, rather than a neural spline flow, the policy model 116 implements another of the various forms of spline flows. Besides neural spline flows, other spline-based flows include piecewise linear and rational-quadratic flows, which use piecewise functions for the transformations. These flows use splines (piecewise functions) to create invertible transformations with flexible shapes. *See*, for example, Kobyzev *et al.*, 2019, “Normalizing Flows: Introduction and Ideas,” arXiv:1908.09257v1 [stat.ML] 25 Aug 2019.

[00140] As another example in some embodiments, rather than a neural spline flow, the policy model 116 implements Continuous Normalizing Flows (CNF). CNF, also known as neural ODE flows, use ordinary differential equations (ODEs) to define the flow transformations. By treating the flow as a continuous transformation, CNFs can efficiently model complex distributions. *See*, for example, Onken, 2021, “OT-Flow: Fast and Accurate Continuous Normalizing Flows via Optimal Transport. Proceedings of the AAAI Conference on Artificial Intelligence, 35(10), 9223-9232, which is hereby incorporated by reference.

[00141] Referring to block 222, in some embodiments, the neural spine flow includes between four and 20 rational-quadratic spline flow coupling layers that each have a plurality of knots.

[00142] A coupling layer is a layer in the architecture of a normalizing flow model that contains a coupling transform. The coupling layer thus includes the transformation as well as other components such as a neural network for generation transformation parameters.

[00143] A rational-quadratic spline flow is a type of transformation used in neural flow models to transform the input data into a new, often more straightforward, distribution while keeping the transformation invertible.

[00144] In a coupling transform, the data (input $\mathbf{x}$) is split into two parts: $\mathbf{x} = [\mathbf{x}_{1:d-1}, \mathbf{x}_{d:D}]$. Then the parameters $\boldsymbol{\theta} = \text{NN}(\mathbf{x}_{1:d-1})$ are computed, where NN is a neural network. Then, $y_i = g_{\boldsymbol{\theta}_i}(x_i)$ for $i = d, \dots, D$ is computed in parallel, where $g_{\boldsymbol{\theta}_i}$ is an invertible function parameterized by $\boldsymbol{\theta}_i$. Then $\mathbf{y}_{1:d-1} = \mathbf{x}_{1:d-1}$ is set and $\mathbf{y} = [\mathbf{y}_{1:d-1}, \mathbf{y}_{d:D}]$ is returned. *See*, Durken *Id.*

[00145] In some embodiments, the transformation is a rational-quadratic function that takes the form of a quotient of two quadratic polynomials. Rational-quadratic functions are differentiable, and when only monotonic segments of these functions are considered, they are also analytically invertible. They are strictly more flexible than quadratic functions, and allow direct parameterization of the derivatives and heights at each knot. In some embodiments, the method of Gregory and Delbourgo, 1982, “Piecewise rational quadratic interpolation to monotonic data,” IMA Journal of Numerical Analysis 2(2):123–130, is used to parameterize a monotonic rational-quadratic spline. The spline itself maps an interval $[-B, B]$ to $[-B, B]$. A transformation outside this range is defined in some embodiments as the identity, resulting in linear ‘tails’, so that the overall transformation can take unconstrained inputs. In some embodiments the spline uses K different rational-quadratic functions, with boundaries set by $K+1$ coordinates $\{(x^{(k)}, y^{(k)})\}_{k=0}^K$ known as knots. In some embodiments, the knots monotonically increase between $(x^{(0)}, y^{(0)}) = (-B, B)$ and $(x^{(K)}, y^{(K)}) = (B, B)$. See, Durken *et al.*, “Neural Spline Flows,” Advances in Neural Information Processing Systems 32 (NeurIPS 2019), which is hereby incorporated by reference.

[00146] Referring to block 224, in some embodiments, the neural spine flow consists of eight quadratic spline flow coupling layers that each have eight knots. In some embodiments the neural spine flow comprises 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, or 20 quadratic spline flow coupling layers that each have a plurality of knots. In some such embodiments, each such quadratic spline flow coupling layer makes use of 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, or 20 knots.

[00147] Referring to block 226, in some embodiments, a fixed but randomly-initialized permutation is applied after each coupling layer to mix units.

[00148] Referring to block 228, in some embodiments, the plurality of parameters 118 of the policy model comprises 500,000 or more parameters, or 1×10^6 parameters. That is, in some embodiments the neural network used to parameterize $\theta = \text{NN}(\mathbf{x}_{1:d-1})$ a coupling transform of a coupling layer of the neural spine flow comprises 500,000 or more parameters, or 1×10^6 or more parameters. In some embodiments, each respective coupling layer in the policy model has its own respective neural network to parameterize the respective $\theta = \text{NN}(\mathbf{x}_{1:d-1})$ of the coupling transform of the respective coupling layer and each such respective neural network has 500,000 or more parameters, or 1×10^6 or more parameters.

[00149] B. *Retrieve the corresponding molecular structure of each d -dimensional code using the decoder module.*

[00150] Referring to block 230, each d -dimensional code in the current batch 122 of d -dimensional codes is applied to the decoder module 114, thereby retrieving the molecular structure of each compound 106/124 in a current batch 122 of compounds. In some embodiments, 50 or more d -dimensional codes are applied to the decoder module 114 in an instance of block 230 thereby retrieving 50 or more compounds. In some embodiments, 100 or more d -dimensional codes are applied to the decoder module 114 in an instance of block 230 thereby retrieving 100 or more compounds. In some embodiments, 200 or more d -dimensional codes are applied to the decoder module 114 in an instance of block 230 thereby retrieving 200 or more compounds. In some embodiments, 500 or more d -dimensional codes are applied to the decoder module 114 in an instance of block 230 thereby retrieving 500 or more compounds. In some embodiments, 1000 or more d -dimensional codes are applied to the decoder module 114 in an instance of block 230 thereby retrieving 1000 or more compounds. In some embodiments, 5000 or more d -dimensional codes are applied to the decoder module 114 in an instance of block 230 thereby retrieving 5000 or more compounds.

[00151] C. *Determining the molecular properties of the retrieved compounds.*

[00152] Referring to block 232, there is determined, using at least the molecular structure of each compound 106/124 in the current batch 122 of compounds, a corresponding molecular property value 126 of each respective molecular property in the plurality of molecular properties, thereby determining a corresponding plurality of molecular property values 126 for each respective compound 106/124 in the current batch 122 of compounds.

[00153] Referring to block 234, in some embodiments, the plurality of molecular properties comprises any two or more rules, any three or more rules, or all four rules of the Lipinski's rule of Five: (i) not more than five hydrogen bond donors, (ii) not more than ten hydrogen bond acceptors, (iii) a molecular weight under 500 Daltons, and (iv) a LogP under 5.

[00154] Referring to block 236, in some embodiments, the plurality of molecular properties comprises any two or more rules, any three or more rules, or all four rules of the Ghose filter: (i) a calculated log P that is between -0.4 and 5.6, (ii) a molecular weight that is between 160 and 480, (iii) a molar refractivity that is between 40 and 130, and (iv) a total number of atoms that is between 20 and 70.

[00155] Referring to block 238, in some embodiments, the plurality of molecular properties comprises (i) a first requirement that each compound 106/124 in the plurality of compounds has three or fewer hydrogen bond donors and (ii) a second requirement that each compound 106/124 in the plurality of compounds has three or fewer hydrogen bond acceptors.

[00156] Referring to block 240, in some embodiments, the plurality of molecular properties comprises any two or more rules, any three or more rules, any four or more rules, any five or more rules, or all six rules selected from the group consisting of: (i) a molecular weight under 300 Daltons, (ii) a LogP less than 3, (iii) three or fewer hydrogen bond donors, (iv) three or fewer hydrogen bond acceptors, (v) a polar surface area of 60 Å², and (vi) three or fewer rotatable bonds.

[00157] In some embodiments, the plurality of molecular properties includes a calculated compound molecular weight.

[00158] In some embodiments, the plurality of molecular properties includes a compound distribution coefficient (logD), a compound octanol-water partition coefficient (logP), a penalized logP, a compound acid dissociation constant (pK_A), a compound acid dissociation constant (k_A), and/or compound topological polar surface area (TPSA).

[00159] LogD is a distribution constant of the lipophilicity of a compound. In some embodiments LogD is calculated from the chemical structure of a compound (cLogD). Generally, cLogD refers to the calculated distribution constant of the lipophilicity of a molecule. This is typically determined for a compound in an aqueous phase, which can be adjusted to a specific pH using a buffer. This parameter can serve as an estimate of a compound's overall lipophilicity, a value that influences the behavior of the compound in a range of biological processes relevant to a drug discovery, such as solubility, permeability through biological membranes, hepatic clearance, lack of selectivity and/or non-specific toxicity. In some instances, lipophilicity can be used as an estimate of druglikeness. *See*, Cambridge MedChem Consulting, 2019, "Lipophilicity," available online at [cambridgemedchemconsulting.com/ resources/physiochem/logD.html](https://cambridgemedchemconsulting.com/resources/physiochem/logD.html), which is hereby incorporated herein by reference in its entirety.

[00160] The logP is a measure of compound hydrophilicity. The partition coefficient, abbreviated P, is defined as a particular ratio of the concentrations of a solute between the two solvents (a biphasic of liquid phases), specifically for un-ionized solutes, and the logarithm of the ratio is thus log P. As used here, following convention, the numerator of this

ratio is concentration of the compound in octanol and the denominator is concentration of the compound in water.

[00161] Penalized logP is a logP score that also accounts for ring size and synthetic accessibility. *See*, for example, Ertl, 2009, “Estimation of synthetic accessibility score of drug-like molecules,” J. Cheminform, which is hereby incorporated by reference.

[00162] In some embodiments, the pK_A is estimated using the structure of a molecule. *See*, for example, Seybold and Shields, 2015, “Computational estimation of pK_a values,” WIREs Comput Mol Sci 5, pp. 290–297, which is hereby incorporated by reference.

[00163] Compound topological polar surface area (TPSA) makes use of functional group contributions based on a large database of structures, is a measure of the polar surface area of a compound that avoids the need to calculate compound three-dimensional structure or to decide which is the relevant biological conformation or conformations. *See*, for example, Prasanna and Doerksen *et al.*, 2009, “Topological Polar Surface Area: A Useful Descriptor in 2D-QSAR,” Current Medicinal Chemistry 16(1), pp. 21-41, which is hereby incorporated by reference.

[0002] Referring to block 242, in some embodiments, a respective compound 106/124 in the plurality of compounds satisfies the plurality of molecular properties when it has a respective Quantitative Estimate of Druglikeness (QED) score of greater than QED threshold, where the respective QED score is within a range of zero to one, and is based on a weighted summation of a plurality of individual desirability functions that canvass the respective compound’s molecular weight, LogP, number of hydrogen bond donors, number of hydrogen bond acceptors, polar surface area, number of rotatable bonds, and number of structural alerts. QED scores are described in Bickerton *et al.*, 2012, “Quantifying the chemical beauty of drugs,” Nature Chemistry 4(2), pp. 90-98, which is hereby incorporated by reference.

[00164] D. *Calculating the tempered reward function of each of the compounds using their molecular property values.*

[00165] Referring to block 244, there is determined, for each respective compound 106/124 in the current batch 122 of compounds, a corresponding tempered reward function value for the respective compound 106/124 using the plurality of molecular property values 126 corresponding to the respective compound 106/124.

[00166] Let $R_0 : X \rightarrow \{0, 1\}$ denote a binarized reward function, which, for example, takes on a value of “1” when a molecule $x \in X$ jointly satisfies the specified constraints and, for example, takes on a value of “0” zero if any of the constraints are violated,

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

[00167] Binarized rewards are challenging optimization targets for reinforcement learning algorithms due to the so-called “sparse reward” problem. Accordingly, in some embodiments, the systems and methods of the present disclosure therefore introduce a temperature parameter $\tau > 0$ and consider a continuous family of tempered reward relaxations $R_\tau : X \rightarrow [0, 1]$,

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

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, $R_0(x) = \lim_{\tau \rightarrow 0+} R_\tau(x)$. That is, as noted in block 200 above, $a : X \rightarrow \mathbb{R}^k$ is a vector-valued function that evaluates the k molecular properties of a given query molecule against their corresponding thresholds b_i in order to compute the corresponding tempered reward function value for the respective compound. If k is 5, then there are five molecular properties a_i and five thresholds b_i that are used in equation 2 to computer the tempered reward function of a respective compound 106/124.

[00168] E. *Determining the current probability density using the tempered reward function values.*

[00169] Referring to block 245, a current probability density 130, under a target distribution 132 in which d -dimensional codes are sampled through the auto-encoder module 110 from the combinatorial synthesis library with probability proportional to their tempered reward function value, is determined using the corresponding tempered reward function value of each respective compound 106/124 in the current batch 122 of compounds. Computation of probability densities is discussed above in conjunction with block 218 with reference to Durken *et al.*, “Neural Spline Flows,” Advances in Neural Information Processing Systems 32 (NeurIPS 2019), which is hereby incorporated by reference.

[00170] F. *Updating the policy model using the current probability density.*

[00171] Referring to block 246, the plurality of parameters 118 of the policy model 116 are updated (optimized) through minimization of at least a difference between (i) the current probability density 130 and (ii) the target distribution 132. In some embodiments, the difference is evaluated using an α -divergence based objective function 134. In other words, to identify d -dimensional codes that decode into compounds satisfying the plurality of molecular properties, the systems and methods of the present disclosure iteratively train the parameters 118 of a policy model $\pi_\eta(z)$, parameterized as a normalizing flow (Figure 5B). Formally, this training seeks a policy model such that $\pi_\eta(z) \propto \mathbb{E}_{p_\theta(x|z,D)}[R_0(x)]$, where the binarized reward $R_0(x)$ is equal to one if molecule x satisfies all of the specified constraints and zero otherwise. Once trained, latent codes are sampled from the policy and decoded to find compounds that meet the search criteria.

[00172] In some embodiments, the systems and methods of the present disclosure determine a parameterization for the policy π_η that: (i) permits efficient sampling from $z \sim \pi_\eta(z | \tau)$; (ii) permits efficient evaluation of log densities $\log \pi_\eta(z | \tau)$; and (iii) permits efficient evaluation of log density gradients $\nabla_\eta \log \pi_\eta(z | \tau)$.

[00173] In some embodiments, a normalizing flow, a class of generative model in which noise is sampled from some base distribution $p(\epsilon)$ that admits straightforward sampling and log density evaluations (e.g., an isotropic standard Gaussian) and transformed through a series of coupling layers that are specially constructed so as to be invertible functions with efficient evaluation of the log absolute determinant of the Jacobian as discussed in further detail above in conjunction with block 222. In some embodiments, because the entire network is invertible, log densities for the output distribution can be computed via application of the change of variables formula. Let $g_\eta : \mathbb{R}^d \times \mathbb{R}_+ \rightarrow \mathbb{R}^d$ be a conditional normalizing flow with parameters η , such that g_η satisfies $g_\eta(g_\eta^{-1}(z, \tau), \tau) = z$ for all τ by construction. In some embodiments, the conditional log density for z under the policy is then evaluated via

$$\log \pi_\eta(z | \tau) = \log p(g_\eta^{-1}(z, \tau)) + \log \left| \det \frac{\partial g_\eta^{-1}(z, \tau)}{\partial z} \right|. \quad (A1)$$

[00174] In some embodiments, g_η is comprised of a stack of neural spline flow rational-quadratic spline affine coupling layers interleaved with shuffling, with an affine transformation applied as the final operation. Additional details and information is found in Durkan *et al.*, 2019, "Neural Spline Flow," Advances in Neural Information Processing

Systems 32, which is hereby incorporated by reference in its entirety for all purposes, as well as above in conjunction with block 222.

[00175] In a sufficiently expressive and well-trained variational auto-encoder 108 (*e.g.*, one that attains a high reconstruction accuracy), the decoder $p_\phi(x|z, \mathcal{D})$ 114 behaves like a probabilistic pseudo-inverse of the encoder $q_\psi(z|x)$ within the essential support of the aggregated variational posterior $q_\psi(z|\mathcal{D}) \equiv |X_{\mathcal{D}}|^{-1} \sum_{x \in X_{\mathcal{D}}} q_\psi(z|x)$. In particular, a high reconstruction accuracy suggests that Equation (1) of block 244 above for distinct pairs of molecules $x \neq x'$ from $X_{\mathcal{D}}$, the latent space conditionals $q_\psi(z|x)$ and $q_\psi(z|x')$ should have mostly disjoint essential supports (*e.g.*, 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 generally concentrate on a relatively small subset of $X_{\mathcal{D}}$. In some embodiments, the essential support of a distribution is $p(z)$ is defined as $\text{supp}, (p) = \{z: p(z) > \epsilon\}$, *e.g.*, it includes all $z \in Z$ where the probability density $p(z)$ exceeds $\epsilon > 0$.

[00176] Given such an approximate bijectivity, it is possible to frame conducting the search in the continuous latent space using the learned decoder p_θ to extract molecules from the library. This process transforms a discrete search problem into policy optimization in $\mathbb{R}^d$ with a non-differentiable reward. In some embodiments, the tempered reward induced by z is:

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

[00177] In some embodiments, hence, $R_0(z) = \lim_{\tau \rightarrow 0+} R_\tau(z)$ is viewed as the probability that a molecule $x \sim p_\theta(x|z, \mathcal{D})$ decoded from z conditionally at random is constraint-satisfying, in that:

$$R_0(z) = \Pr_{p_\theta(x|z, \mathcal{D})}[x \in x_{\mathcal{D}}^*] \quad (4)$$

[00178] In some embodiments, the systems and methods of the present disclosure learn a policy that samples latent codes with probability proportional to their tempered reward $\pi_\eta(z|\tau) \propto R_\tau(z)$. This approach bears similarity to recent work on Boltzmann generators [53] and generative flow networks [54]. The policy together with the decoder $p_\phi(x|z, \mathcal{D})$ includes a distribution over $X_{\mathcal{D}}$:

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

[00179] 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, *e.g.*,

$$\pi_\eta(x|\tau = 0, \mathcal{D}) \approx \text{Uniform}(x|X_{\mathcal{D}}^*). \quad (6)$$

[00180] Optimization of the policy parameters η (*e.g.*, parameters $\theta = \text{NN}(\mathbf{x}_{1:d-1})$ of block 222 above) can be formulated as a divergence minimization problem,

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

where $\mathcal{D}(\pi_\eta||r_\tau) \geq 0$ is a divergence from the policy $\pi_\eta(z|\tau)$ to the target distribution $r_\tau(z) = \Psi_\tau^{-1} R_\tau(z)$ and $\Psi_\tau = \int_z R_\tau(z) dz$ is the (unknown) normalizing constant of the tempered reward function. In some embodiments, optimization [55] is amortized with respect to $\tau > 0$ by specifying a temperature distribution $p(\tau)$, *e.g.*, an exponential distribution with rate parameter $\lambda > 0$, and the expected divergence is minimized from the policy to the tempered reward marginalized over $p(\tau)$.

[00181] In some embodiments, 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 [56–58]. 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 $r_\tau(z) > 0 \Rightarrow \pi_\eta(z|\tau) > 0$, 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 $r_\tau(z) = 0 \Rightarrow \pi_\eta(z|\tau) = 0$, 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.

[00182] The present disclosure provides a novel family of α -divergences related to the Rényi divergences [56] 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:

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

Details pertaining to the derivation of this divergence family are found in Example 1. Once the policy has been adequately trained using the objective (A18) of Example 1, the learned distribution can be sampled from over the relevant constraint-satisfying subset of the library $\pi_{\eta}(x|\tau = 0, \mathcal{D})$.

[00183] Referring to block 248, in some embodiments, the method is repeated until an optimization criterion is satisfied. For instance, in some embodiments, the optimization criterion is satisfied when the policy model 116 selects compounds from CSL 104 with a probability proportional to their tempered reward function values. In some embodiments, the optimization criterion is satisfied when the method has been repeated a certain number of times, such as 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, or 20 times. In some embodiments, the optimization criterion is satisfied when the method has been repeated at least 100, 200, 500, or 1000 times.

[00184] Referring to block 250, in some embodiments, the method is repeated a plurality of instances and the updating of the policy parameters uses a portion of the probability density 130 (*e.g.*, a subset of the compounds) of each instance of the plurality of instances to minimize against the target distribution 132. In some embodiments, the method is repeated a plurality of instances and the updating of the policy parameters uses a portion of the current probability density 130 (*e.g.*, based on a subset of the compounds) of each of the last 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, or 20 or more instances to minimize against the target distribution 132. In some embodiments, the method is repeated a plurality of instances and the updating of the policy parameters uses a portion of each current probability density 130 (*e.g.*, based on a subset of the compounds) of the last 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, or 200 or more instances to minimize against the target distribution 132. In some embodiments, the method is repeated a plurality of instances and the updating of the policy parameters uses a portion of each current probability density 130 (*e.g.*, based on a subset of the compounds) of the last 200, 300, 400, 500, 600, 700, 800, 900, 1000, 1100, 1200, 1300, 1400, 1500, 1600, 1700, 1800, 1900, or 2000 or more instances to minimize against the target distribution 132. In some embodiments, the method is repeated a plurality of instances and the updating of the policy parameters uses a portion of each current probability density 130 (*e.g.*, based on a subset of

the compounds) of each of the last 100 to 10,000 instances to minimize against the target distribution 132.

[00185] Referring to block 252, in some embodiments, the combinatorial synthesis library comprises a trillion compounds, the encoder module 110 is configured to map each compound 106/124 in the combinatorial synthesis library into a d-dimensional code in the learned d-dimensional latent space, and the policy model 116 is configured to search any portion, or plurality of portions, of the entire combinatorial synthesis library in accordance with the updating of the plurality of parameters 118.

[00186] Referring to block 254, in some embodiments, the plurality of compounds is filtered to remove those compounds that fail a pan-assay interference (PAINS) filter. The PAINS filter is described in Baell and Walters, 2014, "Chemistry: Chemical con artists foil drug discovery," *Nature*. 513 (7519): 481–3, which is hereby incorporated by reference.

[00187] Referring to block 256, in some embodiments, the plurality of compounds is filtered to remove those compounds that have three or more stereocenters.

[00188] Referring to block 258, in some embodiments, the plurality of compounds is clustered based on structural similarity thereby forming a plurality of clusters and retaining, in the plurality of compounds, a subset of compounds from each cluster in the plurality of clusters.

[00189] Referring to block 260, in some embodiments, the plurality of compounds is synthesized and the synthesized plurality of compounds is tested in a wet lab assay to determine whether they have bioactivity against a therapeutic target. An example of such a wet lab assay test is provided in Example 7 in conjunction with Fig. 8. Additional examples of wet lab assays are described in Huser ed., 2006, "High-Throughput-Screening in Drug Discovery," *Methods and Principles in Medicinal Chemistry* 35; and Chen ed., 2019, *A Practical Guide to Assay Development and High-Throughput Screening in Drug Discovery*, CRC Press, Boca Raton, each of which is hereby incorporated by reference.

[00190] Referring to block 262, in some embodiments, the plurality of compounds is synthesized and the synthesized plurality of compounds is tested in a wet lab assay to determine whether they have potency against a therapeutic target. An example of such a wet lab assay test is provided in Example 7 in conjunction with Fig. 8.

[00191] Referring to block 264, in some embodiments, the therapeutic target is associated with a condition.

[00192] Referring to block 266, in some embodiments, the condition is inflammation or pain.

[00193] Referring to block 268, in some embodiments, the condition is a disease.

[00194] Referring to block 270, in some embodiments, the condition is a cancer, hematologic disorder, autoimmune disease, inflammatory disease, immunological disorder, metabolic disorder, neurological disorder, genetic disorder, psychiatric disorder, gastroenterological disorder, renal disorder, cardiovascular disorder, dermatological disorder, respiratory disorder, viral infection, or other disease or disorder.

[00195] In some embodiments the wet lab assay test validates a compound in the plurality of compounds as being a suitable compound for alleviation of the condition. In some such embodiments the compound is used in *in vivo* assays such as animal models.

[00196] In some embodiments, a compound in the plurality of compounds is combined with one or more excipient and/or one or more pharmaceutically acceptable carrier and/or one or more diluent when administering to an animal model or a human.

[00197] Such excipients and/or carriers include all conventional solvents, dispersion media, fillers, solid carriers, coatings, antifungal and antibacterial agents, dermal penetration agents, surfactants, isotonic and absorption agents and the like.

[00198] An exemplary carrier is pharmaceutically “acceptable” in the sense of being compatible with the other ingredients of the composition (*e.g.*, the composition comprising the selected compound in the plurality of compounds) and not injurious to a subject. The compound may conveniently be presented in unit dosage form and may be prepared by any methods well known in the art of pharmacy. Such methods include the step of bringing into association the compound with the carrier that constitutes one or more accessory ingredients. In general, the compound is prepared by uniformly and intimately bringing into association the compound with liquid carriers or finely divided solid carriers or both.

[00199] As noted above, exemplary compounds formulated for intravenous, intramuscular or intraperitoneal administration, or a pharmaceutically acceptable salt, solvate or prodrug thereof may be administered by injection or infusion.

[00200] Injectables for such use can be prepared in conventional forms, either as a liquid solution or suspension or in a solid form suitable for preparation as a solution or suspension in a liquid prior to injection, or as an emulsion. Carriers can include, for example, water, saline (*e.g.*, normal saline (NS), phosphate-buffered saline (PBS), balanced saline solution

(BSS)), sodium lactate Ringer's solution, dextrose, glycerol, ethanol, and the like; and if desired, minor amounts of auxiliary substances, such as wetting or emulsifying agents, buffers, and the like can be added. Proper fluidity can be maintained, for example, by using a coating such as lecithin, by maintaining the required particle size in the case of dispersion and by using surfactants.

[00201] The compound may also be suitable for oral administration and may be presented as discrete units such as capsules, sachets or tablets each containing a predetermined amount of the test chemical compound; as a powder or granules; as a solution or a suspension in an aqueous or non-aqueous liquid; or as an oil-in-water liquid emulsion or a water-in-oil liquid emulsion. The compound may also be presented as a bolus, electuary or paste.

[00202] A tablet of the compound may be made by compression or molding, optionally with one or more accessory ingredients. Compressed tablets may be prepared by compressing in a suitable machine the test chemical compound in a free-flowing form such as a powder or granules, optionally mixed with a binder [*e.g.*, inert diluent, preservative disintegrant (*e.g.* sodium starch glycolate, cross-linked polyvinyl pyrrolidone, cross-linked sodium carboxymethyl cellulose) surface-active or dispersing agent]. Molded tablets may be made by molding in a suitable machine a mixture of the powdered compound moistened with an inert liquid diluent. The tablets may optionally be coated or scored and may be formulated so as to provide slow or controlled release of the compound therein using, for example, hydroxypropylmethyl cellulose in varying proportions to provide the desired release profile. Tablets may optionally be provided with an enteric coating, to provide release in parts of the gut other than the stomach.

[00203] The compound may be suitable for topical administration in the mouth including lozenges comprising the active ingredient in a flavored base, usually sucrose and acacia or tragacanth gum; pastilles comprising the active ingredient in an inert basis such as gelatine and glycerin, or sucrose and acacia gum; and mouthwashes comprising the active ingredient in a suitable liquid carrier.

[00204] The compound may be suitable for topical administration to the skin. In such instances, the compound may be dissolved or suspended in any suitable carrier or base and may be in the form of lotions, gel, creams, pastes, ointments and the like. Suitable carriers include mineral oil, propylene glycol, polyoxyethylene, polyoxypropylene, emulsifying wax, sorbitan monostearate, polysorbate 60, cetyl esters wax, cetearyl alcohol, 2-octyldodecanol,

benzyl alcohol and water. Transdermal patches may also be used to administer the compound.

[00205] The compound may be suitable for parenteral administration. In such embodiments, the compound may include aqueous and non-aqueous isotonic sterile injection solutions that may contain anti-oxidants, buffers, bactericides and solutes that render the compound isotonic with the blood of the intended recipient; and aqueous and non-aqueous sterile suspensions that may include suspending agents and thickening agents. The compound may be presented in unit-dose or multi-dose sealed containers, for example, ampoules and vials, and may be stored in a freeze-dried (lyophilized) condition requiring only the addition of the sterile liquid carrier, for example water for injections, immediately prior to use. Extemporaneous injection solutions and suspensions may be prepared from sterile powders, granules and tablets of the kind previously described.

[00206] It should be understood that in addition to the compound particularly mentioned above, the composition or combination of this present disclosure (*e.g.*, the compound selected from the plurality of compounds) may include other agents conventional in the art having regard to the type of composition or combination in question, for example, those suitable for oral administration may include such further agents as binders, sweeteners, thickeners, flavoring agents disintegrating agents, coating agents, preservatives, lubricants and/or time delay agents. Suitable sweeteners include sucrose, lactose, glucose, aspartame or saccharine. Suitable disintegrating agents include cornstarch, methylcellulose, polyvinylpyrrolidone, xanthan gum, bentonite, alginic acid or agar. Suitable flavoring agents include peppermint oil, oil of wintergreen, cherry, orange or raspberry flavoring. Suitable coating agents include polymers or copolymers of acrylic acid and/or methacrylic acid and/or their esters, waxes, fatty alcohols, zein, shellac or gluten. Suitable preservatives include sodium benzoate, vitamin E, alpha-tocopherol, ascorbic acid, methyl paraben, propyl paraben or sodium bisulphite. Suitable lubricants include magnesium stearate, stearic acid, sodium oleate, sodium chloride or talc. Suitable time delay agents include glyceryl monostearate or glyceryl distearate.

[00207] In some embodiments, the present disclosure informs the selection of one or more human subjects for treatment with the compound and/or selection of one or more human subjects for continuation or discontinuation of treatment with the compound.

[00208] In some embodiments, the present disclosure informs the dosing amount, duration, and/or frequency of the compound in one or more human subjects for treatment.

[00209] In some embodiments, the present disclosure informs the design of a clinical trial, the clinical trial comprising the use of the compound. In some embodiments, the present disclosure informs the design of an adaptive clinical trial, the adaptive clinical trial comprising the use of the compound.

[00210] In some embodiments, the present disclosure further comprises formulating the compound for use in a therapy. In some embodiments, this includes formulating the compound with any of the excipients, pharmaceutically acceptable carrier, diluents, or other pharmacological formulations described in the present disclosure or known in the art. In some embodiments, the therapy is to alleviate a condition such as inflammation. In some embodiments the therapy is to alleviate or treat a disease or disorder. In some embodiments the disease or disorder is cancer, a hematologic disorder, an autoimmune disease, an inflammatory disease, an immunological disorder, a metabolic disorder, a neurological disorder, a genetic disorder, a psychiatric disorder, a gastroenterological disorder, a renal disorder, a cardiovascular disorder, a dermatological disorder, a respiratory disorder, a viral infection, or other disease or disorder.

[00211] **Use Case** - The following is a sample use case provided for illustrative purposes only that describes an example application of some embodiments of the present disclosure. Other uses may be considered, and example use case provided below is non-limiting and may be subject to variations, omissions, or may contain additional elements.

[00212] *Hit discovery.* Drug discovery companies, such as pharmaceutical companies, spend millions of dollars on screening compounds to discover new prospective drug leads. Large compound collections, such as combinatorial synthesis libraries 104, are tested to find the small number of compounds that have any interaction with the disease target of interest. Unfortunately, wet lab screening suffers experimental errors and, in addition to the cost and time to perform the assay experiments, the gathering of large screening collections imposes significant challenges through storage constraints, shelf stability, or chemical cost. Even the largest pharmaceutical companies have only between hundreds of thousands to a few millions of compounds, versus the tens of millions of commercially available molecules and the hundreds of millions of simulate-able molecules.

[00213] A potentially more efficient alternative to physical experimentation is virtual high throughput screening using the systems and methods of the present disclosure. In the same manner that physics simulations can help an aerospace engineer to evaluate possible wing designs before a model is physically tested, computational screening of molecules can focus the experimental testing on a small subset of high-likelihood compounds. This may reduce screening cost and time, reduce false negatives, improve success rates, and/or cover a broader swath of chemical space.

[00214] In this application, a therapeutic target may be provided as input to the system. In some embodiments, the therapeutic target is a particular protein, a particular nucleic acid, a particular complex of proteins and/or nucleic acids that is implicated in a condition such as a disease. A large set of compounds, such as in the form of a combinatorial synthesis library 104, is also provided. The large set of compounds is accessed using an auto-encoder 108 comprising an encoder 110 and a decoder 112. The encoder 110 maps compounds in the large set of compounds into latent codes in a learned latent space. The decoder 114 retrieves molecular structures of compounds using such codes. A policy model 116 selects a batch of latent codes, from which a plurality of compounds is identified using the decoder. Molecular properties of these compounds are determined *in silico* and used to determine a reward function value for each compound. The molecular properties and the thresholds for each of the molecular properties are chosen that will enrich for compounds that are suitable candidates for either inhibiting or promoting an activity of the therapeutic target in order to alleviate the condition of interest. For instance, in some embodiments, inhibition of the therapeutic target is expected to alleviate a condition of interest and one of plurality of molecular properties that is used in the screening is a binding coefficient of each respective compound against the therapeutic target or a calculated bioactivity against the therapeutic target exhibited the respective compound as illustrated in nonlimiting Example 7. A probability density, under a target distribution in which latent codes are sampled through the auto-encoder from the large set of compounds with probability proportional to tempered reward function values, is determined using the reward function values. Policy parameters are updated using at least a difference between the probability density and the target distribution, using an α -divergence based objective function. This process is repeated until the policy model selects compounds with a probability that at least approximates their tempered reward values. In some embodiments the compounds that satisfied the molecular properties are filtered as discussed above in conjunction with blocks 254, and 256. In some

embodiments the selected compounds are also clustered based on structural similarity and only a select few compounds from each cluster are retained. Finally the remaining compounds are tested in a wet lab assay, such as the one described in Example 7 to determine which of the compounds have bioactivity against the therapeutic target.

[00215] EXAMPLES

[00216] Example 1 - Objective Function

[00217] In some embodiments, the Rényi α -divergence family generalizes the KL divergence and has been applied in variational inference via the variational Rényi bound (VRB) [56], which has the form

$$\mathcal{L}_\alpha(\pi_\eta, r_\tau) = \frac{1}{\alpha} \log \mathbb{E}_{\pi_\eta(\mathcal{Z}|\tau)} \left[\left(\frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right)^\alpha \right]. \quad (\text{A2})$$

[00218] In some embodiments, in variational inference, the VRB is a lower bound on the marginal log likelihood for $\alpha < 1$, and it is related to the negative of an upper bound on the marginal log likelihood for $\alpha > 1$, which motivates maximization (cf. minimization) of the VRB for $\alpha < 1$ (cf. $\alpha > 1$). In some embodiments, the VRB recovers the negative of the reverse KL divergence in the limit $\alpha \rightarrow 0$ and extrema of the log likelihood ratios in the left and right limits,

$$\lim_{\alpha \rightarrow 0} \mathcal{L}_\alpha(\pi_\eta, r_\tau) = \mathbb{E}_{\pi_\eta(\mathcal{Z}|\tau)} \left[\log \frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right] = -D_{\text{KL}}(\pi_\eta \parallel r_\tau); \quad (\text{A3})$$

$$\lim_{\alpha \rightarrow -\infty} \mathcal{L}_\alpha(\pi_\eta, r_\tau) = \min_{\pi_\eta(\mathcal{Z}|\tau)} \log \frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} = \max_{\pi_\eta(\mathcal{Z}|\tau)} \log \frac{\pi_\eta(\mathcal{Z}|\tau)}{r_\tau(\mathcal{Z})}; \quad (\text{A4})$$

$$\lim_{\alpha \rightarrow +\infty} \mathcal{L}_\alpha(\pi_\eta, r_\tau) = \max_{\pi_\eta(\mathcal{Z}|\tau)} \log \frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} = \min_{\pi_\eta(\mathcal{Z}|\tau)} \log \frac{\pi_\eta(\mathcal{Z}|\tau)}{r_\tau(\mathcal{Z})}. \quad (\text{A5})$$

[00219] In some embodiments, it follows that optimizing the VRB for $\alpha \rightarrow -\infty$ requires that $r_\tau(\mathcal{Z}) = 0 \Rightarrow \pi_\eta(\mathcal{Z}|\tau) = 0$ (otherwise, the VRB attains a value of $-\infty$), which is exactly the mode-seeking behavior described earlier. Similarly, in some embodiments, optimizing the VRB for $\alpha \rightarrow +\infty$ requires that $\pi_\eta(\mathcal{Z}|\tau) = 0 \Rightarrow r_\tau(\mathcal{Z}) = 0$, or equivalently $r_\tau(\mathcal{Z}) > 0 \Rightarrow \pi_\eta(\mathcal{Z}|\tau) > 0$ by contraposition, which is the mass-covering behavior.

[00220] In some embodiments, the systems and methods of the present disclosure determine that the unknown normalizing constant contributes a level shift to the VRB:

$$\mathcal{L}_\alpha(\pi_\eta, r_\tau) = \frac{1}{\alpha} \log \mathbb{E}_{\pi_\eta(\mathcal{Z}|\tau)} \left[\left(\frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right)^\alpha \right] \quad (\text{A6})$$

$$= \frac{1}{\alpha} \log \mathbb{E}_{\pi_\eta}(z|\tau) \left[\exp \left(\alpha \log \frac{r_\tau(z)}{\pi_\eta(z|\tau)} \right) \right] \quad (\text{A7})$$

$$= \frac{1}{\alpha} \log \mathbb{E}_{\pi_\eta}(z|\tau) \left[\exp \left(\alpha \log \frac{\psi_\tau^{-1} R_\tau(z)}{\pi_\eta(z|\tau)} \right) \right] \quad (\text{A8})$$

$$= \frac{1}{\alpha} \log \mathbb{E}_{\pi_\eta}(z|\tau) \left[\exp \left(\alpha \log \frac{R_\tau(z)}{\pi_\eta(z|\tau)} - \alpha \log \psi_\tau \right) \right] \quad (\text{A9})$$

$$= \frac{1}{\alpha} \log \mathbb{E}_{\pi_\eta}(z|\tau) \left[\exp \left(\alpha \log \frac{R_\tau(z)}{\pi_\eta(z|\tau)} \right) \right] - \log \psi_\tau \quad (\text{A10})$$

$$= \mathcal{L}_\alpha(\pi_\eta, R_\tau) - \log \psi_\tau \quad (\text{A11})$$

[00221] In some embodiments, since ψ_τ does not depend on η , the systems and methods of the present disclosure determine $\nabla_\eta \mathcal{L}_\alpha(\pi_\eta, r_\tau) = \nabla_\eta \mathcal{L}_\alpha(\pi_\eta, R_\tau)$, and thus optimization of the VRB with respect to the policy parameters η does not require knowledge of the normalizing constant. However, in some embodiments, for mass-covering $\alpha > 1$, optimization is challenged by pronounced variance in the estimated gradient, which is unfortunate since this is the regime that is generally of interest.

[00222] In some embodiments, the gradient of the VRB with respect to α is invariant to the unknown normalizing constant and has the following form:

$$\frac{\partial}{\partial \alpha} \mathcal{L}_\alpha(\pi_\eta, r_\tau) = \frac{\partial}{\partial \alpha} \mathcal{L}_\alpha(\pi_\eta, R_\tau) \quad (\text{A12A})$$

$$= \frac{\mathbb{E}_{\pi_\eta}(z|\tau) \left[\left(\frac{R_\tau(z)}{\pi_\eta(z|\tau)} \right)^\alpha \log \frac{R_\tau(z)}{\pi_\eta(z|\tau)} \right]}{\alpha \mathbb{E}_{\pi_\eta}(z|\tau) \left[\left(\frac{R_\tau(z)}{\pi_\eta(z|\tau)} \right)^\alpha \right]} - \frac{1}{\alpha} \mathcal{L}_\alpha(R_\tau, \pi_\eta). \quad (\text{A12B})$$

[00223] In some embodiments, the systems and methods of the present disclosure determine that (1) $\frac{\partial}{\partial \alpha} \mathcal{L}_\alpha(r_\tau, \pi_\eta) \geq 0$ for all $\alpha \in \mathbb{R}$, and (2) $\frac{\partial}{\partial \alpha} \mathcal{L}_\alpha(r_\tau, \pi_\eta) = 0$ if and only if $r_\tau = \pi_\eta$. In some such embodiments, the gradient of the VRB with respect to α , in fact, defines a valid (new) α -divergence family,

$$\begin{aligned}
D_\alpha(\pi_\eta \parallel r_\tau) &= \frac{\mathbb{E}_{\pi_\eta(\mathcal{Z}|\tau)} \left[\left(\frac{R_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right)^\alpha \log \frac{R_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right]}{\alpha \mathbb{E}_{\pi_\eta(\mathcal{Z}|\tau)} \left[\left(\frac{R_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right)^\alpha \right]} \\
&= \frac{1}{\alpha^2} \log \mathbb{E}_{\pi_\eta(\mathcal{Z}|\tau)} \left[\left(\frac{R_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right)^\alpha \right]. \quad (\text{A13})
\end{aligned}$$

[00224] Without intending to be limited to any particular theory, in some embodiments, the systems and methods of the present disclosure interpret these α -divergences as the difference between a weighted arithmetic mean of the log likelihood ratios and the log of a weighted geometric mean of the likelihood ratios, all scaled by α , and where the weights are proportional to the likelihood ratios exponentiated by α . Without intending to be limited to any particular theory it is postulated that such a formulation acts like a control variate, thereby helping to reduce gradient variance. In some such embodiments, these α -divergences have the useful property of being invariant to the unknown normalizing constant of the target distribution.

[00225] In some embodiments, the systems and methods of the present disclosure determine that $D_\alpha(\pi_\eta \parallel r_\tau)$ recovers the *forward* KL divergence for $\alpha = 1$,

$$D_1(\pi_\eta \parallel r_\tau) = \frac{\mathbb{E}_{\pi_\eta(\mathcal{Z}|\tau)} \left[\frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \log \frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right]}{\mathbb{E}_{\pi_\eta(\mathcal{Z}|\tau)} \left[\frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right]} - \log \mathbb{E}_{\pi_\eta(\mathcal{Z}|\tau)} \left[\left(\frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right) \right] \quad (\text{A14})$$

$$= \mathbb{E}_{\pi_\eta(\mathcal{Z}|\tau)} \left[\frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \log \frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right] \quad (\text{A15})$$

$$= \mathbb{E}_{r_\tau(\mathcal{Z})} \left[\log \frac{r_\tau(\mathcal{Z})}{\pi_\eta(\mathcal{Z}|\tau)} \right] \quad (\text{A16})$$

$$= D_{\text{KL}}(r_\tau \parallel \pi_\eta) \quad (\text{A17})$$

[00226] In some embodiments, interestingly, optimization of the VRB for $\alpha = 1$, by contrast, does not correspond to divergence minimization at all, since $\mathcal{L}_1(\pi_\eta, r_\tau) = \log \mathbb{E}_{\pi_\eta(\mathcal{Z}|\tau)} [r_\tau(\mathcal{Z})/\pi_\eta(\mathcal{Z}|\tau)] = 0$ is constant.

[00227] In some embodiments, for $\alpha \geq 1$, the derived α -divergence family has the mass-covering inductive bias. In some embodiments, the systems and methods of the present disclosure therefore minimize the objective

$$\min_{\eta} \mathbb{E}_{p(\tau)} [D(\pi_\eta \parallel r_\tau)] \quad (\text{A18})$$

with the divergence defined in (A13) for $\alpha \geq 1$, via Monte Carlo approximation and stochastic gradient optimization.

[00228] Example 2 - Policy Gradient

[00229] In some embodiments, Monte Carlo approximations are used for all expectations when minimizing the objective (A18). In some embodiments, the systems and methods of the present disclosure introduce a behavior policy $\beta(z|\tau)$ and re-write expectations using importance sampling, arriving at the following expression for the gradient of the temperature-marginalized α -divergence with respect to the policy parameters:

$$\nabla_{\eta} \mathbb{E}_{p(\tau)}[D_{\alpha}(\pi_{\tau} \parallel r_{\tau})] = \nabla_{\eta} \mathbb{E}_{p(\tau)} \left[\frac{\mathbb{E}_{\beta(z|\tau)} \left[\frac{\pi_{\eta}(z|\tau)}{\beta(z|\tau)} \left(\frac{R_{\tau}(z)}{\pi_{\eta}(z|\tau)} \right)^{\alpha} \log \frac{R_{\tau}(z)}{\pi_{\eta}(z|\tau)} \right]}{\alpha \mathbb{E}_{\beta(z|\tau)} \left[\frac{\pi_{\eta}(z|\tau)}{\beta(z|\tau)} \left(\frac{R_{\tau}(z)}{\pi_{\eta}(z|\tau)} \right)^{\alpha} \right]} - \frac{1}{\alpha^2} \log \mathbb{E}_{\beta(z|\tau)} \left[\frac{\pi_{\eta}(z|\tau)}{\beta(z|\tau)} \left(\frac{R_{\tau}(z)}{\pi_{\eta}(z|\tau)} \right)^{\alpha} \right] \right]. \quad (\text{A19})$$

[00230] In some embodiments, because expectations are taken with respect to $p(\tau)$ and $\beta(z|\tau)$, neither of which depend on η , the systems and methods of the present disclosure avoid the need to invoke a reparameterization trick in gradient estimation. This supports both on-policy, such as for $\beta(z|\tau) = \text{stop_grad}(\pi_{\eta}(z|\tau))$ and off-policy learning (*e.g.*, using a replay buffer to store past episodes).

[00231] In some embodiments, given an episode $\{(z_i, \tau_i, \log \beta(z_i|\tau_i), \log R_{\tau_i}(z_i))\}_{i=1}^n$ of n evaluations, the Monte Carlo objective is expressed in a numerically stable log-sum-exp style, making iteration on (A19) reliable in practice and avoiding issues of underflow/overflow.

[00232] Because (A19) does not depend on η , some embodiments of the systems and methods of the present disclosure do not require that the reward $R_{\tau}(z)$ be differentiable with respect to z for policy gradient estimation. This allows the systems and methods of the present disclosure to incorporate numerous molecular properties as constraints without needing to extend/complicate the approach or introduce approximations like surrogate models (although these can indeed be utilized, *e.g.*, in scenarios where evaluation of the underlying molecular property is expensive).

[00233] Fig. 9 provides a code listing for a function for numerically stable Monte Carlo estimation of $D_\alpha(r_\tau \parallel \pi_\eta)$ given $\log \pi_\eta(z|\tau)$, $\log \beta(z|\tau)$, and $\log R_\tau(z)$ evaluated over a minibatch of n samples drawn from $\beta(z|\tau)$.

[00234] Example 3 - Diversity Sampling

[00235] The constraint-satisfying set of compounds is referred to herein as X_D^* . Depending on the constraints, it is highly likely that sampling k compounds uniformly at random from X_D^* results in far fewer than k distinct compound clusters, *e.g.*, when using Butina clustering [40] with ECFP4 similarity. In some embodiments, this occurs when the library includes clusters that contain a very large number of analogues that are constraint-satisfying. For the policy, this can manifest in probability density being more heavily concentrated in regions where many latent codes map to compounds that earn a large (expected) reward, such that sampling *i.i.d.* from the policy induces compound sets that are surprisingly lacking in diversity, even if the policy has correctly identified other relevant modes.

[00236] In some embodiments, to address this, the systems and methods of the present disclosure sample from an alternative distribution informed by the policy, but defined over a set of latent codes such that (i) they mutually attain a high log likelihood under the policy, and (ii) they are sufficiently spread out to encourage coverage of distinct modes of the policy. Ideas similar to this have been proposed in prior work, for example [62, 63].

[00237] In some embodiments, the systems and methods of the present disclosure formalize these preferences with an energy function,

$$E(z_1, \dots, z_k) = -\sum_{i=1}^k \log \pi_\eta(z_i | \tau = 0) + \lambda \sum_{i=1}^{k-1} \sum_{j=i+1}^k \max(\delta - \|z_i - z_j\|_2^2, 0), \quad (\text{A20})$$

where $\delta > 0$ is a hyperparameter specifying the tolerated proximity between latent codes and $\lambda > 0$ is a hyperparameter that governs the penalty for violating the proximity constraint. In some embodiments, the systems and methods determine the diversified policy, which satisfies $\tilde{\pi}_\eta(z_1, \dots, z_k) \propto \exp(-E(z_1, \dots, z_k))$.

[00238] To sample from $\tilde{\pi}_\eta$, in some embodiments, the systems and methods of the present first randomly initialize k latent codes by sampling from the policy π_η and then run stochastic gradient Langevin dynamics (SGLD) [64] on the latent codes for T iterations using $-E(z_1, \dots, z_k)$ as the target unnormalized log density. In some embodiments, while the proximity term in the energy function requires $O(k^2)$ distance evaluations, the Monte Carlo approximation can

be formed using a subset of $\ell \ll (k-1)k/2$ pairs sampled randomly each SGLD step to ensure tractability when k is large.

[00239] Example 4 - Hierarchical enumeration (HE)

[00240] Hierarchical enumeration was based on the V-SYNTHES method. It consisted of generating a minimally-enumerated library (MEL) consisting of capped compounds. These capped compounds were derived by using a synthon for one of the reaction's positions, and using caps or minimal synthons for the remaining positions. In that sense, the MEL contained one compound per unique synthon in the library, expressed in compound with caps for the other positions. The hierarchical enumeration proceeded by iteratively performing three steps: scoring, filtering/selection, and expansion. In the scoring step, properties were calculated for the current set of molecules. These could correspond to the calculated properties described below or some prediction on the molecule's bioactivity. For the filtering and selection step, molecules that had either grown too large or greasy were removed. The cutoffs were relaxed for molecular weight and logP based on the number of caps still present in the molecule (the maximum cutoff is decreased by 100 Da per cap and 1 unit of logP per cap). The top scoring molecules were then selected according to a scoring system (*e.g.*, the log-reward based on satisfied constraints or the bioactivity predictions). Finally, a loose diversity cutoff ($T_c < 0.9$ to any other molecule in the set) was imposed to ensure sufficiently diverse molecules are selected for expansion. At each iteration, k molecules are selected for the expansion step (where k can be adjusted, *e.g.*, between 10 and 5000). In the expansion step, for each of the k molecules that were selected, each of the caps that were still present in the molecule are replaced with up to m synthons from the library (where m is a positive integer of between 1 and 2000 in some embodiments), using synthons that were available for that particular reaction and position. Synthons were selected based on the score of their corresponding MEL compound. Only one cap could be replaced at a time in some embodiments. The input for the enumeration was the MEL and, at each iteration, the method iteratively replaced one of the caps of the molecules in the pool with synthons from the library. The enumeration stops when all molecules have had all of their caps replaced - and the highest scoring molecules were output.

[00241] Example 5 - Virtual screening with a synthetic oracle

[00242] Performance of the disclosed systems and methods in hit discovery was evaluated. In contrast to molecular properties calculated only from the ligand, biomolecular activity is

determined by the receptor-ligand complex. AtomNet® is a target-agnostic (“global”) system of models to predict bioactivity and potency from docked receptor-ligand complexes [36–38]. The AtomNet® bioactivity module is a classification model and was trained on active and inactive compounds. The AtomNet® potency module is a regression model trained on active compounds only. Docking and scoring concomitant with the methods disclosed above in conjunction with Figure 2 is computationally costly. Therefore, a computationally tractable surrogate model that predicts “oracle” AtomNet® bioactivity and potency scores from ligands alone was designed as described in Example 8. The screens for both the NGT (Figure 2) and HE (Example 4) methods start from a pre-scored Minimum Enumeration Library (MEL) [13]. A virtual high throughput screen against two established drug targets, cyclin-dependent kinase 5 (CDK5), a cyclin-dependent serine/threonine kinase, and N-acylsphingosine amidohydrolase 1 (ASAH1), an acid ceramidase was constructed. The virtual screens were designed such that the NGT (Figure 2) and HE (Example 4) approaches had an approximately similar computational budget for function evaluations. Consequently, the NGT (Figure 2) approach resulted in approximately three times as many final, scored compounds compared to the HE (Example 4) approach. Molecular property constraints were applied equally to the NGT (Figure 2) and HE (Example 4) approaches: the final compounds satisfy $MW < 550$ Da and $CLogP < 5$.

[00243] NGT (Figure 2) efficiently balances exploration and exploitation of ultra-large chemical spaces owing to the mass-covering design of its normalizing flow. For CDK5, NGT found 222,576 compounds satisfying the bioactivity and potency constraints compared to 191,551 compounds for HE (Fig. 10, second row). For the more challenging target ASAH1, NGT found 135,272 compounds, compared to just 24,951 for HE (Fig. 10, second row). The NGT constraint-satisfying compounds represent significantly more unique reactions and synthons at each position than those found by HE, with the exception of synthons unique to the 4-component reaction, suggesting HE takes a greedier approach (Figs. 7A, 7B, upper). To compare the extent of chemical space covered by both methods, Tanimoto distances for compounds from both methods were compared, visualized with a two-dimensional Uniform Manifold Approximation and Projection (UMAP) [39] Figs 7A, & 7B (lower). The two-dimensional distributions of compounds suggest that NGT compounds were widely distributed throughout chemical space whereas HE (Example 4) compounds tended to cluster.

[00244] NGT (Figure 2) demonstrated significant early enrichment of bioactive compounds compared to the prescored MEL (baseline) and to HE (Fig 7C). Enrichment was calculated as the number of NGT or HE compounds with AtomNet scores exceeding the top N score from the MEL baseline, normalized by the total number of compounds scored from each set. When chemical diversity among the compounds was ignored, enrichment was observed evenly distributed over the top ASAH1 and CDK5 compounds for both methods, illustrated by the generally flat dotted lines (Fig 7C). NGT enrichment in bioactivity ranged from up to 4-fold for the top ASAH1 compounds and up to 16-fold for the top CDK5 compounds. In contrast, while HE enrichment for CDK5 resembled NGT's, a significant decrease in HE enrichment occurred for ASAH1 compared to NGT (Fig 7C left panel, dotted lines). The reduced HE enrichment compared to NGT suggests that bioactivity scores that AtomNet assigned to partial ASAH1 fragments translated non-additively to full compounds in ASAH1. Similar performance degradation for HE in CDK5 was not observed, likely because abundant training data for kinases led to better recognition of common ATP-site binding motifs compared to hydrolase binding motifs.

[00245] Example 6 – Diversity

[00246] How enforcing chemical diversity, which reports on balancing exploration and exploitation, affects enrichment was evaluated. Top compounds generated by each method (Figure 2, as discussed above, versus hierarchical enumeration - HE - described in Example 4) - were clustered using the Taylor-Butina method [40] at different distance. By design, when controlled for chemical diversity, Figure 2 (NGT) enrichment for bioactivity degraded only mildly (Fig 7C, dashed and solid ngt lines). This is consistent with the normalizing flow policy approximating a distribution covering multiple modes that satisfy the constraints. In comparison, when controlled for chemical diversity in HE (Example 4), enrichment for bioactivity dramatically decreased in both ASAH1 and CDK5 (Fig 7C, dashed sbd lines), vanishing altogether for the most diverse selection threshold (Fig 7C, solid sbd lines). This indicates that HE adopts a greedier approach, concentrating on specific chemotypes when replacing the caps of the MEL compounds. Chemical diversity is important in virtual screens to hedge against false positives, and throughout early discovery to help overcome liabilities in pharmacokinetics.

[00247] Finally, to select bioactive compounds that performed well on the potency objective, compounds were rank ordered on their mean bioactivity and potency scores (Fig 7E). In the

absence of a chemical diversity threshold, NGT and HE had similar performance, with NGT maintaining more robust enrichment compared to HE when diversity was enforced.

[00248] Example 7 – Designing G-protein Associated Melanocortin Receptor

antagonists Melanocortin-2 receptor (MC2R, Fig 8A) is a G protein coupled receptor selectively agonized by the adrenocorticotrophic hormone (ACTH) and exclusively expressed in the adrenal gland [41]. Autosomal mutations of MC2R result in glucocorticoid deficiency [42, 43] and require exogenous glucocorticoid administration [44]. ACTH overexpression leads to Cushing's disease [45] and congenital adrenal hyperplasia (CAH) [46]. As a result, small molecules that antagonize MC2R serve as potential tool compounds for understanding Cushing's disease [34] and CAH biology, as well as points of departure for potential therapeutics. An MC2R conformational ensemble was used to represent the receptor consisting of the activated MC2R domain of the ACTH:MC2R:MRAP1:Gs complex⁴⁵, the MC2R AlphaFold [48] model, and a homology model based on an MC4R crystal structure (Suppl. Mat.). The screening site included residues I84, E80, K77, D103, I106, D107, S108, F110, L112, V111, L114, V166, F178, W232, F235, F258, I265. (Fig 8A). HE and NGT was used to screen the site. To improve the diversity of hits from HE the algorithm was slightly modified (Methods).

[00249] Selection and experimental characterization of compounds. Common chemical structure and physicochemical property filters were applied to avoid assay interference and improve drug-likeness. For a single concentration assay, 400 compounds were selected for the Figure 2 method and the HE method (Example 4). Pan-assay interference (PAINS) compounds, compounds containing unwanted functional groups, for example reactive moieties, and compounds that contained three or more stereocenters were removed to avoid assay interference and improve drug-likeness. HE compounds for which MEL caps were not replaced from the entire list of compounds returned by each method were also removed. The compounds were then clustered by applying a diversity cutoff of 0.4 Tanimoto similarity using 1024-bit ECFP4 fingerprints and up to three compounds per similarity cluster were selected. The bioactivity scores and predicted potency scores for the compounds were rank-average and the top scoring 400 compounds were selected from each list Fig 8B. Verification that both sets contained 400 distinct compounds was done by de-duplicating compounds. There were no inter-explorer overlaps. The maximum compound similarity across all 800 compounds was 0.93 from compounds belonging to the same high-scoring cluster. The list of

compounds is provided as Figure 14. From each list, Enamine was used to prioritize 125 compounds that synthesized fastest.

[00250] For these compounds, MC2R antagonism measured by cAMP accumulation was performed at Eurofins (catalog #2241) using MC2R stably expressed in HEK293 cells. Compounds were diluted from a stock concentration of 10 mM to 10 uM for single concentration screening and serially from 100 to 0.3 uM for subsequent concentration response screens described below. Reference antagonist ACTH (1-39) was added to all cells except the basal control at a concentration of 100nM. Cells were suspended in HBSS buffer (Invitrogen), 20 mM HEPES (pH 7.4), and 500 uM IBMX and then dispensed to microplates at a concentration of approximately 100 cells/well. Cells were then incubated for 5 minutes in the presence of the reference antagonist, the test compound, or HBSS as a basal control. After incubation for 10 minutes at room temperature, cells were lysed and the fluorescent donor and acceptor compounds were added (D2-labeled cAMP and anti-cAMP antibody labeled with europium cryptate, respectively) and allowed to incubate for 60 minutes at room temperature. Florescence transfer was then recorded (excitation: 337nm; emission: 620 and 665 nm) using a microplate reader (Envision, Perkin Elmer). Compounds were measured in duplicate, and inhibition was calculated by measuring the cAMP accumulation, which was determined by dividing the 655 nm signal by the 620 nm signal and expressed as percent inhibition relative to ACTH agonism. For the selectivity panel described below, EC50 was estimated from single concentration screens with a hyperbolic equation, $\%inh = ([I]/([I] + IC50))$, where [I] is the concentration of the inhibitor and %inh is the percent inhibition, assuming 100% inhibition at infinite [I] [59].

[00251] In the single concentration MC2R antagonism functional assay, 13 out of 121 NGT compounds showed more than 50% inhibition at 30 μ M concentration, compared to 10 out of 127 HE compounds (Suppl Mat) (Fig 8C, left). This corresponds to a 35% higher hit rate for NGT (10.7%) than for HE (7.9%). Moreover, AtomNet scores do not differentiate between antagonists and agonists; NGT identified three additional compounds that showed more than 50% activation compared to one with HE, resulting in a 55% higher combined hit rate for NGT (13.3%) than for HE (8.6%). These findings confirm a notable advantage for NGT to generate hits from vast chemical universes.

[00252] Compounds for a subsequent dose-response relationship assay were considered. Lipophilic ligand efficiency (cLLE) was calculated from single concentration-estimated pIC50 and cLogD, and compounds with cLLE > 1.9 or estimated pIC50 > 4.9 were selected

(Suppl Mat). This resulted in 5 compounds for each method. The two strongest agonists were also selected for each method. All selected antagonists validated in the concentration response assay (Fig 8C, right). The most potent NGT compound reported an $EC_{50} = 6.7 \mu M$ corresponding to a Ligand Efficiency (LE) of 0.49 (Fig 8D, left). One HE compound was slightly more potent ($EC_{50} = 4.1 \mu M$) at slightly lower LE = 0.51 (Fig 8D, right).

[00253] In Figure 14, compound 1 (ATMW-1) was determined to have an IC_{50} of $3.8E-05$ M and a pIC_{50} of 4.42; compound 2 (ATMW-2) was determined to have an IC_{50} of $4.10E-06$ M and a pIC_{50} of 5.38; compound 3 (ATMW-3) was determined to have an IC_{50} of $1.70E-05$ M and a pIC_{50} of 4.77; compound 4 (ATMW-4) was determined to have an IC_{50} of $9.4E-06$ M and a pIC_{50} of 5.03; compound 6 (ATMW-6) was determined to have an IC_{50} of $7.00E-06$ M and a pIC_{50} of 5.15; compound 7 (ATMW-7) was determined to have an IC_{50} of $6.70E-06$ M and a pIC_{50} of 5.17; compound 8 (ATMW-8) was determined to have an IC_{50} of $1.10E-05$ M and a pIC_{50} of 4.96; compound 10 (ATMW-10) was determined to have an IC_{50} of $2.80E-05$ M and a pIC_{50} of 4.55; compound 11 (ATMW-11) was determined to have an IC_{50} of $4.40E-05$ M and a pIC_{50} of 4.36; compound 12 (ATMW-12) was determined to have an IC_{50} of $2.10E-05$ M and a pIC_{50} of 4.68.

[00254] Both agonists were inactive in dose-response (Fig. 8D, left). Without intending to be limited to any particular theory, it is possible that more diversity among the NGT compounds compared to HE compounds translated into better selectivity for MC2R in the MCXR family. Designing a selective agonist or antagonist for MC2R is important and challenging because ACTH also activates all other MCXRs [49]. In a single concentration functional assay panel (Methods) the only compounds demonstrating selectivity for MC2R over MC1R, MC4R, and MC5R were generated by NGT.

[00255] Without intending to be limited to any particular theory, it is believed that compounds ATMW-1-12 (Fig. 8F) are the first potent and selective MC2R antagonists reported in the scientific literature. A Crinetics patent (WO2019236699A1) previously disclosed the only publicly available MC2R antagonists. These antagonists have undergone further chemical optimization to enhance potency and reduce ADMET liabilities. The crinetics compounds all contain an arylpiperazine functional group fused to at least one additional aromatic ring (Fig 8G). This motif is common among GPCR antagonists [50–52]. In contrast, the catalog compounds ATMW-1-12 represent structurally diverse starting points that do not contain the arylpiperazine moiety and provide new regions of chemical space to explore and optimize, differentiated from existing MCXR and other GPCR antagonists.

[00256] Example 8. Surrogate Model

[00257] Multi-layer perceptron (MLP) based surrogate models were trained on the latent space 112 over which the disclosed policy model 116 performs its search. The use of surrogate models is to avoid expensive function evaluations during policy training with Atomnet® endpoints. In particular, compounds scored at each outer iteration of NeuralGenThesis (NGT) that have been evaluated against the screening targets with AtomNet® are encoded into latent space 112 by encoder module 110. Then each MLP in an ensemble of MLP surrogates is trained minimize a mean squared error loss of the predicted AtomNet® scores. To encourage policy model 116 exploration in regions of the latent space 112 where the MLP ensemble members disagree, an upper confidence bound (UCB)-like approach is taken where the endpoint used for purposes of reward construction is the mean of the ensemble member predictions plus some important value times the standard deviation of their predictions. This encourages the policy model 116 to sample in regions of the latent space 112 where the average predicted AtomNet® score is high or sufficiently compensated by the disagreement between the ensemble members. Referring to Figure 13, it was observed that, across outer iterations of NGT, the surrogate models become increasingly more accurate proxies for predicting AtomNet® scores.

REFERENCES CITED AND ALTERNATIVE EMBODIMENTS

[00258] [1] Grygorenko *et al.*, “Generating multibillion chemical space of readily accessible screening compounds,” *iScience* 23, 101681 (2020).

[00259] [2] Neumann *et al.*, “Relevance of the trillion-sized chemical space “explore” as a source for drug discovery,” *ACS Med. Chem. Lett.* 14, 466–472 (2023).

[00260] [3] Arthor. Nextmove software, inc.: Cambridge, England (2020).

[00261] [4] Bellmann *et al.*, “Topological similarity search in large combinatorial fragment spaces,” *J. Chem. Inf. Model.* 61, 238–251 (2021).

[00262] [5] Rarey Dixon, “Feature trees: A new molecular similarity measure based on tree matching. *J. Comput. Aided Mol. Des.* 12, 471–490 (1998).

[00263] [6] Smallworld. Nextmove software, inc.: Cambridge, England (2020).

[00264] [7] Schmidt *et al.*, “Maximum common substructure searching in combinatorial make-on-demand compound spaces,” *J. Chem. Inf. Model.* 62, 2133–2150 (2022).

- [00265] [8] Lyu *et al.*, “Modeling the expansion of virtual screening libraries,” *Nature Chemical Biology* 19, 712–718 (2023).
- [00266] [9] OpenEye Software. Giga Docking Guide–Orion Programming Guide (2020).
- [00267] [10] Acharya *et al.*, “Supercomputer-based ensemble docking drug discovery pipeline with application to covid-19,” *J. Chem. Inf. Model.* 60, 5832–5852 (2020).
- [00268] [11] Warr, “Report on an nih workshop on ultralarge chemistry databases (2021). URL <https://chemrxiv.org/engage/chemrxiv/article-details/60c75883bdbb89984ea3ada5>.
- [00269] [12] Gentile *et al.*, “Artificial intelligence-enabled virtual screening of ultra-large chemical libraries with deep docking,” *Nat. Protoc.* 17, 672–697 (2022).
- [00270] [13] Sadybekov *et al.*, “Synthon-based ligand discovery in virtual libraries of over 11 billion compounds,” *Nature* 601, 452–459 (2022).
- [00271] [14] Zeng *et al.*, “Deep generative molecular design reshapes drug discovery,” *Cell Rep. Med.* 3, 100794 (2022).
- [00272] [15] Sousa *et al.*, “Generative deep learning for targeted compound design,” *J. Chem. Inf. Model.* 61, 5343–5361 (2021).
- [00273] [16] Xie *et al.*, “Advances and challenges in de novo drug design using three-dimensional deep generative models,” *J. Chem. Inf. Model.* 62, 2269–2279 (2022).
- [00274] [17] Jin *et al.*, “Multi-objective molecule generation using interpretable substructures,” Preprint at <http://arxiv.org/abs/2002.03244> (2020).
- [00275] [18] Wang *et al.*, “Relation: A deep generative model for structure-based de novo drug design,” *J. Med. Chem.* 65, 9478–9492 (2022).
- [00276] [19] Thomas *et al.*, “Integrating structure-based approaches in generative molecular design,” *Curr. Opin. Struct. Biol.* 79, 102559 (2023).
- [00277] [20] Jin *et al.*, “Junction tree variational autoencoder for molecular graph generation,” Preprint at <http://arxiv.org/abs/1802.04364> (2019).
- [00278] [21] Anstine and Isayev, “Generative models as an emerging paradigm in the chemical sciences,” *J. Am. Chem. Soc.* 145, 8736–8750 (2023).
- [00279] [22] Stanley and Segler, “Fake it until you make it? generative de novo design and virtual screening of synthesizable molecules,” *Curr. Opin. Struct. Biol.* 82, 102658 (2023).

- [00280] [23] Korshunova *et al.*, “Generative and reinforcement learning approaches for the automated de novo design of bioactive compounds,” *Commun. Chem.* 5, 129 (2022).
- [00281] [24] Gao and Coley, “The synthesizability of molecules proposed by generative models,” *J. Chem. Inf. Model.* 60, 5714–5723 (2020).
- [00282] [25] Griffiths and Hernández-Lobato, “Constrained Bayesian optimization for automatic chemical design using variational autoencoders,” *Chem. Sci.* 11, 577–586 (2020).
- [00283] [26] Gottipati *et al.*, “Learning to navigate the synthetically accessible chemical space using reinforcement learning,” *Proceedings of Machine Learning Research* 119, 3668–3679 (2020).
- [00284] [27] Polishchuk, “Control of synthetic feasibility of compounds generated with crem,” *J. Chem. Inf. Model.* 60, 6074–6080 (2020).
- [00285] [28] Bradshaw *et al.*, “A model to search for synthesizable molecules,” *NIPS'19: Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Article No. 713, pp. 7937–7949 (2019).
- [00286] [29] Bradshaw *et al.*, “Barking up the right tree: an approach to search over molecule synthesis DAGS,” *NIPS '20: Proceedings of the 34th International Conference on Neural Information Processing Systems*, Article 575, pp. 6852–6866 (2020).
- [00287] [30] Horwood and Noutahi, “Molecular design in synthetically accessible chemical space via deep reinforcement learning,” *ACS Omega* 5, 32984–32994 (2020).
- [00288] [31] Thorne *et al.*, “Neural databases,” (2020). URL <http://arxiv.org/abs/2010.06973>.
- [00289] [32] Pedawi *et al.*, “An efficient graph generative model for navigating ultra-large combinatorial synthesis libraries,” *36th Conference on Neural Information Processing Systems (NeurIPS 2022)* (2022).
- [00290] [33] Clark *et al.*, “Acth antagonists,” *Front. Endocrinol.* 7 (2016).
- [00291] [34] Forfar *et al.*, “Identification of a novel specific small-molecule melanocortin-2-receptor antagonist,” *Endocr. Connect.* 11, e220338 (2022).
- [00292] [35] Pedawi *et al.*, “Through the looking glass: navigating in latent space to optimize over combinatorial synthesis libraries,” (2023). URL <https://openreview.net/forum?id=ALsSka1db3>.

- [00293] [36] Wallach *et al.*, “Atomnet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery. arXiv preprint arXiv:1510.02855 (2015).
- [00294] [37] Gniewek *et al.*, “Learning physics confers pose-sensitivity in structure-based virtual screening,” ArXiv211015459 Q-Bio (2021).
- [00295] [38] Stafford *et al.* “Atomnet posranker: Enriching ligand pose quality for dynamic proteins in virtual high-throughput screens,” J. Chem. Inf. Model. 62, 1178–1189 (2022).
- [00296] [39] McInnes *et al.*, “Umap: Uniform manifold approximation and projection,” J. Open Source Softw. 3, 861 (2018).
- [00297] [40] 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,” J. Chem. Inf. Comput. Sci. 39, 747–750 (1999).
- [00298] [41] Clark and Metherell, “Mechanisms of disease: the adrenocorticotropin receptor and disease,” Nat. Clin. Pract. Endocrinol. Metab. 2, 282–290 (2006).
- [00299] [42] Clark *et al.*, “Familial glucocorticoid deficiency associated with point mutation in the adrenocorticotropin receptor,” The Lancet 341, 461–462 (1993).
- [00300] [43] Clark *et al.*, “The genetics of familial glucocorticoid deficiency,” Best Pract. Res. Clin. Endocrinol. Metab. 23, 159–165 (2009).
- [00301] [44] Dineen *et al.*, “The management of glucocorticoid deficiency: Current and future perspectives,” Clin. Chim. Acta 505, 148–159 (2020).
- [00302] [45] Guaraldi and Salvatori, “Cushing syndrome: Maybe not so uncommon of an endocrine disease,” J. Am. Board Fam. Med. 25, 199–208 (2012).
- [00303] [46] Mallappa and Merke, “Management challenges and therapeutic advances in congenital adrenal hyperplasia,” Nat. Rev. Endocrinol. 18, 337–352 (2022).
- [00304] [47] Fowler *et al.*, “Effects of crn04894, a nonpeptide orally bioavailable acth antagonist, on corticosterone in rodent models of acth excess,” J. Endocr. Soc. 5, A167–A167 (2021).
- [00305] [48] Jumper *et al.*, “Highly accurate protein structure prediction with alphafold,” Nature 596, 583–589 (2021).
- [00306] [49] Yang and Harmon, “Molecular determinants of acth receptor for ligand selectivity,” Mol. Cell. Endocrinol. 503, 110688 (2020).

- [00307] [50] Gaillard *et al.*, “Binding of Arylpiperazines, (Aryloxy) propanolamines, and Tetrahydropyridylindoles to the 5-HT_{1A} Receptor: Contribution of the Molecular Lipophilicity Potential to Three-Dimensional Quantitative Structure Affinity Relationship Models,” *Journal of Medicinal Chemistry* 39, 126–134 (1996).
- [00308] [51] Stroth *et al.*, “Arylpiperazine agonists of the serotonin 5-HT_{1A} receptor preferentially activate camp signaling versus recruitment of -arrestin-2” *Bioorganic Medicinal Chemistry* 23, 4824–4830 (2015).
- [00309] [52] Lacivita *et al.*, “Knowledge-based design of long-chain arylpiperazine derivatives targeting multiple serotonin receptors as potential candidates for treatment of autism spectrum disorder,” *ACS Chemical Neuroscience* 12, 1313–1327 (2021).
- [00310] [53] No’e *et al.*, “Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning,” *Science* 365 (2019).
- [00311] [54] Bengio *et al.*, “GFlowNet Foundations,” arXiv preprint arXiv:2111.09266 (2021).
- [00312] [55] Dosovitskiy and Djolonga, ICLR 2020 Conference Blind Submission, “You only train once: loss-conditional training of deep networks,” (2019).
- [00313] [56] Li and Turner, “Rényi divergence variational inference. *Advances in Neural Information Processing Systems* 29 (2016).
- [00314] [57] Midgley *et al.*, “M. Flow annealed importance sampling bootstrap,” arXiv preprint arXiv:2208.01893 (2022).
- [00315] [58] Minka, “Divergence measures and message passing,” Tech. Rep., Microsoft Research (2005).
- [00316] [59] Kenyon *et al.*, “Novel human lipoxigenase inhibitors discovered using virtual screening with homology models,” *Journal of Medicinal Chemistry* 49, 1356–1363 (2006).
- [00317] [60] Papamakarios *et al.*, “Normalizing flows for probabilistic modeling and inference,” *The Journal of Machine Learning Research* 22, 2617–2680 (2021).
- [00318] [61] Rezende and Mohamed, “Variational inference with normalizing flows,” 1530–1538, PMLR (2015).
- [00319] [62] Ma *et al.*, “Diverse sampling for normalizing flow based trajectory forecasting,” arXiv preprint arXiv:2011.15084 (2020).

[00320] All references cited herein are incorporated herein by reference in their entirety and for all purposes to the same extent as if each individual publication or patent or patent application was specifically and individually indicated to be incorporated by reference in its entirety for all purposes.

[00321] The present invention can be implemented as a computer program product that includes a computer program mechanism embedded in a non-transitory computer-readable storage medium. For instance, the computer program product could contain instructions for operating the user interfaces disclosed herein and described with respect to the Figures. These program modules can be stored on a CD-ROM, DVD, magnetic disk storage product, USB key, or any other non-transitory computer readable data or program storage product.

[00322] Many modifications and variations of this invention can be made without departing from its spirit and scope, as will be apparent to those skilled in the art. The specific embodiments described herein are offered by way of example only. The embodiments were chosen and described in order to best explain the principles of the invention and its practical applications, to thereby enable others skilled in the art to best utilize the invention and various embodiments with various modifications as are suited to the particular use contemplated. The invention is to be limited only by the terms of the appended claims, along with the full scope of equivalents to which such claims are entitled.

WHAT IS CLAIMED IS:

1. A computer-implemented method of identifying a plurality of compounds in a combinatorial synthesis library that each satisfy a plurality of molecular properties, wherein the combinatorial synthesis library is accessible through a trained variational auto-encoder that comprises (i) an encoder module that maps a compound in the combinatorial synthesis library into a d -dimensional code in a learned d -dimensional latent space and (ii) a decoder module that retrieves a molecular structure of the compound, with a reconstruction likelihood greater than a threshold reconstruction likelihood, upon inputting the d -dimensional code into the decoder module, the method comprising:

A) using a policy model, comprising a plurality of parameters, to select a current batch of d -dimensional codes in the learned d -dimensional latent space;

B) applying each d -dimensional code in the current batch of d -dimensional codes to the decoder module, thereby retrieving the molecular structure of each compound in a current batch of compounds;

C) determining, using at least the molecular structure of each compound in the current batch of compounds, a corresponding molecular property value of each respective molecular property in the plurality of molecular properties, thereby determining a corresponding plurality of molecular property values for each respective compound in the current batch of compounds;

D) determining, for each respective compound in the current batch of compounds, a corresponding tempered reward function value for the respective compound using the plurality of molecular property values corresponding to the respective compound;

E) determining a current probability density, under a target distribution in which d -dimensional codes are sampled through the auto-encoder module from the combinatorial synthesis library with probability proportional to their tempered reward function value, using the corresponding tempered reward function value of each respective compound in the current batch of compounds; and

F) updating the plurality of parameters of the policy model through minimization of at least a difference between (i) the current probability density and (ii) the target distribution, wherein the difference is evaluated using an α -divergence based objective function.

2. The computer-implemented method of claim 1, wherein the using A), applying B), determining C), determining D), determining E), and updating F) are repeated until an optimization criterion is satisfied.
3. The computer-implemented method of claim 1, wherein the using A), applying B), determining C), determining D), determining E), and updating F) are repeated a plurality of instances and the updating F) uses a portion of the current probability density of each instance in the plurality of instances to minimize against the target distribution.
4. The computer-implemented method of any one of claims 1-3, wherein
the combinatorial synthesis library comprises a trillion compounds,
the encoder module is configured to map each compound in the combinatorial synthesis library into a d -dimensional code in the learned d -dimensional latent space, and
the policy model is configured to search any portion, or plurality of portions, of the entire combinatorial synthesis library in accordance with the updating of the plurality of parameters in the updating F).
5. The computer-implemented method of any one of claims 1-4, wherein the threshold reconstruction likelihood is 50 percent or greater.
6. The computer-implemented method of any one of claims 1-4, wherein the threshold reconstruction likelihood is 60 percent or greater.
7. The computer-implemented method of any one of claims 1-4, wherein the threshold reconstruction likelihood is between 60 percent and 80 percent.
8. The computer-implemented method of any one of claims 1-7, wherein the plurality of molecular properties comprises any two or more rules, any three or more rules, or all four rules of the Lipinski's rule of Five: (i) not more than five hydrogen bond donors, (ii) not more than ten hydrogen bond acceptors, (iii) a molecular weight under 500 Daltons, and (iv) a LogP under 5.
9. The computer-implemented method of any one of claims 1-7, wherein the plurality of molecular properties comprises any two or more rules, any three or more rules, or all four

rules of the Ghose filter: (i) a calculated log P that is between -0.4 and 5.6, (ii) a molecular weight that is between 160 and 480, (iii) a molar refractivity that is between 40 and 130, and (iv) a total number of atoms that is between 20 and 70.

10. The computer-implemented method of any one of claims 1-7, wherein the plurality of molecular properties comprises (i) a first requirement that each compound in the plurality of compounds has three or fewer hydrogen bond donors and (ii) a second requirement that each compound in the plurality of compounds has three or fewer hydrogen bond acceptors.

11. The computer-implemented method of any one of claims 1-7, wherein the plurality of molecular properties comprises any two or more rules, any three or more rules, any four or more rules, any five or more rules, or all six rules selected from the group consisting of: (i) a molecular weight under 300 Daltons, (ii) a LogP less than 3, (iii) three or fewer hydrogen bond donors, (iv) three or fewer hydrogen bond acceptors, (v) a polar surface area of 60 Å², and (vi) three or fewer rotateable bonds.

12. The computer-implemented method of any one of claims 1-7, wherein a respective compound in the plurality of compounds satisfies the plurality of molecular properties when it has a respective Quantitative Estimate of Druglikeness (QED) score of greater than QED threshold, wherein the respective QED score is within a range of zero to one, and is based on a weighted summation of a plurality of individual desirability functions that canvass the respective compound's molecular weight, LogP, number of hydrogen bond donors, number of hydrogen bond acceptors, polar surface are, number of rotateable bonds, and number of structural alerts.

13. The computer-implemented method of any one of claims 1-7, wherein a first molecular property in the plurality of molecular properties is a bioactivity prediction against a therapeutic target, and wherein the first molecular property is satisfied for a respective compound in the plurality of compounds when a bioactivity classification module determines, based on a chemical structure of the respective compound, that the respective compound is an active compound against the therapeutic target.

14. The computer-implemented method of any one of claims 1-7, wherein a first molecular property in the plurality of molecular properties is a potency prediction against a therapeutic

target, and wherein the first molecular property is satisfied for a respective compound in the plurality of compounds when a potency prediction regression module determines, based on a chemical structure of the respective compound, that the respective compound has at least a threshold potency against the therapeutic target.

15. The computer-implemented method of any one of claims 1-14, wherein the plurality of compounds comprises 500 or more compounds.

16. The computer-implemented method of any one of claims 1-14, wherein the plurality of compounds comprises 1000 or more compounds.

17. The computer-implemented method of any one of claims 1-14, wherein the plurality of compounds comprises 10,000 or more compounds.

18. The computer-implemented method of any one of claims 1-14, wherein the plurality of compounds comprises 100,000 or more compounds.

19. The computer-implemented method of any one of claims 1-19, wherein the combinatorial synthesis library represents more than 10,000 synthons and more than 10 three- or four- component reactions.

20. The computer-implemented method of any one of claims 1-19, wherein the method further comprises filtering from the plurality of compounds those compounds that fail a pan-assay interference (PAINS) filter.

21. The computer-implemented method of any one of claims 1-20, wherein the method further comprises filtering from the plurality of compounds those compounds that have three or more stereocenters.

22. The computer-implemented method of any one of claims 1-21, wherein the method further comprises:

clustering the plurality of compounds based on structural similarity thereby forming a plurality of clusters; and

retaining, in the plurality of compounds, a subset of compounds in each cluster in the plurality of clusters.

23. The computer-implemented method of any one of claims 1-22, wherein the policy model is a normalizing flow.

24. The computer-implemented method of claim 23, wherein the normalizing flow is a neural spine flow.

25. The computer-implemented method of claim 24, wherein the neural spine flow includes between four and 20 rational-quadratic spline flow coupling layers that each have a plurality of knots.

26. The computer-implemented method of claim 24, wherein the neural spine flow consists of eight quadratic spline flow coupling layers that each have eight knots.

27. The computer-implemented method of claim 25 or 26 wherein a fixed but randomly-initialized permutation is applied after each rational-quadratic spline flow coupling layer.

28. The computer-implemented method of claim 23, wherein the normalizing flow comprises one or more real-valued non-volume preserving transformations.

29. The computer-implemented method of claim 23, wherein the normalizing flow is a masked autoregressive flow.

30. The computer-implemented method of claim 23, wherein the normalizing flow is a piecewise linear flow, a rational-quadratic flow, or a continuous normalizing flow.

31. The computer-implemented method of any one of claims 1-30, wherein the d -dimensional code has a latent space dimension d that is between 32 and 128.

32. The computer-implemented method of any one of claims 1-31, wherein the d -dimensional code has a latent space dimension d that is 64.

33. The computer-implemented method of any one of claims 1-32, wherein the plurality of parameters comprises 500,000 or more parameters.
34. The computer-implemented method of any one of claims 1-32, wherein the plurality of parameters comprises 1×10^6 or more parameters.
35. The computer-implemented method of any one of claims 1-34, the method further comprising:
- synthesizing the plurality of compounds; and
 - testing the synthesized plurality of compounds in a wet lab assay to determine whether they have bioactivity against a therapeutic target.
36. The computer-implemented method of any one of claims 1-35, the method further comprising:
- synthesizing the plurality of compounds; and
 - testing the synthesized plurality of compounds in a wet lab assay to determine whether they have potency against a therapeutic target.
37. The computer-implemented method of claim 35 or 36, wherein the therapeutic target is associated with a condition.
38. The computer-implemented method of claim 37, wherein the condition is inflammation or pain.
39. The computer-implemented method of claim 37, wherein the condition is a disease.
40. The computer-implemented method of claim 37, wherein the condition is a cancer, hematologic disorder, autoimmune disease, inflammatory disease, immunological disorder, metabolic disorder, neurological disorder, genetic disorder, psychiatric disorder, gastroenterological disorder, renal disorder, cardiovascular disorder, dermatological disorder, respiratory disorder, viral infection, or other disease or disorder.
41. A computer system for identifying a plurality of compounds in a combinatorial synthesis library that each satisfy a plurality of molecular properties, wherein the combinatorial

synthesis library is accessible through a trained variational auto-encoder that comprises (i) an encoder module that maps a compound in the combinatorial synthesis library into a d -dimensional code in a learned d -dimensional latent space and (ii) a decoder module that retrieves a molecular structure of the compound, with a reconstruction likelihood greater than a threshold reconstruction likelihood, upon inputting the d -dimensional code into the decoder module, the computer system comprising:

one or more processors; and

memory addressable by the one or more processors, the memory storing at least one program for execution by the one or more processors, the at least one program comprising instructions for:

A) using a policy model, comprising a plurality of parameters, to select a current batch of d -dimensional codes in the learned d -dimensional latent space;

B) applying each d -dimensional code in the current batch of d -dimensional codes to the decoder module, thereby retrieving the molecular structure of each compound in a current batch of compounds;

C) determining, using at least the molecular structure of each compound in the current batch of compounds, a corresponding molecular property value of each respective molecular property in the plurality of molecular properties, thereby determining a corresponding plurality of molecular property values for each respective compound in the current batch of compounds;

D) determining, for each respective compound in the current batch of compounds, a corresponding tempered reward function value for the respective compound using the plurality of molecular property values corresponding to the respective compound;

E) determining a current probability density, under a target distribution in which d -dimensional codes are sampled through the auto-encoder module from the combinatorial synthesis library with probability proportional to their tempered reward function value, using the corresponding tempered reward function value of each respective compound in the current batch of compounds; and

F) updating the plurality of parameters of the policy model through minimization of at least a difference between (i) the current probability density and (ii) the target distribution, wherein the difference is evaluated using an α -divergence based objective function.

42. A computer system for identifying a plurality of compounds in a combinatorial synthesis library that each satisfy a plurality of molecular properties, wherein the combinatorial

synthesis library is accessible through a trained variational auto-encoder that comprises (i) an encoder module that maps a compound in the combinatorial synthesis library into a d -dimensional code in a learned d -dimensional latent space and (ii) a decoder module that retrieves a molecular structure of the compound, with a reconstruction likelihood greater than a threshold reconstruction likelihood, upon inputting the d -dimensional code into the decoder module, the computer system comprising:

one or more processors; and

memory addressable by the one or more processors, the memory storing at least one program for execution by the one or more processors, the at least one program comprising instructions for implementing the computer-implemented method of any one of claims 1-34.

43. A non-transitory computer-readable medium storing one or more computer programs, executable by a computer, for identifying a plurality of compounds in a combinatorial synthesis library that each satisfy a plurality of molecular properties, wherein the combinatorial synthesis library is accessible through a trained variational auto-encoder that comprises (i) an encoder module that maps a compound in the combinatorial synthesis library into a d -dimensional code in a learned d -dimensional latent space and (ii) a decoder module that retrieves a molecular structure of the compound, with a reconstruction likelihood greater than a threshold reconstruction likelihood, upon inputting the d -dimensional code into the decoder module, the computer comprising one or more processors and a memory, the one or more computer programs collectively encoding computer executable instructions that perform a method comprising:

A) using a policy model, comprising a plurality of parameters, to select a current batch of d -dimensional codes in the learned d -dimensional latent space;

B) applying each d -dimensional code in the current batch of d -dimensional codes to the decoder module, thereby retrieving the molecular structure of each compound in a current batch of compounds;

C) determining, using at least the molecular structure of each compound in the current batch of compounds, a corresponding molecular property value of each respective molecular property in the plurality of molecular properties, thereby determining a corresponding plurality of molecular property values for each respective compound in the current batch of compounds;

D) determining, for each respective compound in the current batch of compounds, a corresponding tempered reward function value for the respective compound using the plurality of molecular property values corresponding to the respective compound;

E) determining a current probability density, under a target distribution in which d -dimensional codes are sampled through the auto-encoder module from the combinatorial synthesis library with probability proportional to their tempered reward function value, using the corresponding tempered reward function value of each respective compound in the current batch of compounds; and

F) updating the plurality of parameters of the policy model through minimization of at least a difference between (i) the current probability density and (ii) the target distribution, wherein the difference is evaluated using an α -divergence based objective function.

44. A non-transitory computer-readable medium storing one or more computer programs, executable by a computer, for identifying a plurality of compounds in a combinatorial synthesis library that each satisfy a plurality of molecular properties, wherein the combinatorial synthesis library is accessible through a trained variational auto-encoder that comprises (i) an encoder module that maps a compound in the combinatorial synthesis library into a d -dimensional code in a learned d -dimensional latent space and (ii) a decoder module that retrieves a molecular structure of the compound, with a reconstruction likelihood greater than a threshold reconstruction likelihood, upon inputting the d -dimensional code into the decoder module, the computer comprising one or more processors and a memory, the one or more computer programs collectively encoding computer executable instructions that perform the computer implemented method of any one of claims 1-34.