Supplementary Material for *"Several Birds with One Stone": Exploring the Potential of AI Methods for Multi-Target Drug Design*

Muhetaer Mukaidaisi[1†], Madiha Ahmed[1,3†], Karl Grantham[1], Aws Al-Jumaily[1], Shoukat Dedhar[4], Michael Organ[5], Alain Tchagang[6], Jinqiang Hou[7,8], Ejaz Ahmed[3], Renata Dividino[1*], Yifeng Li[1,2*]

[1*]Department of Computer Science, Brock University, 1812 Sir Isaac Brock Way, St. Catharines, L2S 3A1, Ontario, Canada.
[2]Department of Biological Sciences, Brock University, 1812 Sir Isaac Brock Way, St. Catharines, L2S 3A1, Ontario, Canada.
[3]Department of Mathematics and Statistics, Brock University, 1812 Sir Isaac Brock Way, St. Catharines, L2S 3A1, Ontario, Canada.
[4]British Columbia Cancer Research Centre, Department of Biochemistry and Molecular Biology, University of British Columbia, 675 West 10th Avenue, Vancouver, V5Z 1G1, British Columbia, Canada.
[5]Department of Chemistry and Biomolecular Sciences, University of Ottawa, 75 Laurier Avenue E, Ottawa, K1N 6N5, Ontario, Canada.
[6]Digital Technologies Research Centre, National Research Council Canada, 1200 Montreal Road, Ottawa, K1A 0R6, Ontario, Canada.
[7]Department of Chemistry, Lakehead University, 955 Oliver Road, Thunder Bay, P7B 5E1, Ontario, Canada.
[8]Thunder Bay Regional Health Research Institute, 980 Oliver Road, Thunder Bay, P7B 6V4, Ontario, Canada.

*Corresponding author(s). E-mail(s): rdividino@brocku.ca; yli2@brocku.ca;
Contributing authors: mm20ul@brocku.ca; ma21jw@brocku.ca; kg16wk@brocku.ca; aa17qz@brocku.ca; sdedhar@bccrc.ca; organ@uottawa.ca; alain.tchagang@nrc-cnrc.gc.ca; jhou3@lakeheadu.ca; sahmed5@brocku.ca;

†These authors contributed equally to this work.

Abstract

This file contains supplementary information for the main text.

1 DEL Details

The evolutionary operators, non-dominated ranking method, and crowding distance calculation in DEL are described below.

1.1 Evolutionary Operators

To produce new samples of the next generation, latent representations of populations undergo evolutionary operations in accordance with classical genetic algorithms, which include individual selection, crossover, and mutation. Individual selection utilizes binary tournament selection to select one individual (parent) from every two random samples by comparing their partial relation, until the drawn mating pool size matches the population size M, namely, if $\boldsymbol{z}_1 \prec_n \boldsymbol{z}_2$ then $\boldsymbol{z}_1$ will be selected as the parent solution. Selected M parents are then paired up as $\boldsymbol{z}_{p1}$ and $\boldsymbol{z}_{p2}$ for crossover operator to generate children $\hat{\boldsymbol{z}}_{c1}$ and $\hat{\boldsymbol{z}}_{c2}$. All experiments use blending linear crossover (BLX) [1, 2], which formulates

$$\hat{\boldsymbol{z}}_{c1} = \boldsymbol{z}_{p1} + r_1(\boldsymbol{z}_{p2} - \boldsymbol{z}_{p1}), \tag{1}$$

$$\hat{\boldsymbol{z}}_{c2} = \boldsymbol{z}_{p1} + r_2(\boldsymbol{z}_{p2} - \boldsymbol{z}_{p1}), \tag{2}$$

where

$$r_1 = -d + (1 + 2d)\alpha_1, \tag{3}$$

$$r_2 = -d + (1 + 2d)\alpha_2. \tag{4}$$

The recommended value of $d = 0.25$ (from [1]) is used in this work and $\alpha_1, \alpha_2 \sim \text{Uniform}(0, 1)$. Recombinant samples $\hat{\boldsymbol{z}}_m$ are carried on to the mutation operations over a predefined mutation rate $p_m = 0.01$, meaning for every sample, if a random value l from $\text{Uniform}(0, 1)$ is smaller p_m, the l-th position of $\hat{\boldsymbol{z}}_m$ will be overwritten by value drawn from standard Gaussian distribution $\mathcal{N}(0, \mathbf{I})$ Grantham et al. [3].

1.2 Non-dominated Ranking and Crowding Distance

In preparation for evolutionary operations, DEL analyzes non-dominated rankings and crowding distances of multi-objective solutions to identify advantageous parent solutions in selection for generating offspring, as well to produce distinct populations with competitive and diverse molecules from the molecule pool of previous populations in the subsequent step.

In [4], crowding is defined as the density value of individuals around a feasible solution within Pareto fronts. It may be intuitively seen as a cuboid surrounding an individual and excluding others. An individual's crowding distance is calculated by summing the differences between the values of the sub-objective functions of two individuals before and after this individual. In general, the crowding distance of individual $\boldsymbol{z}_i$ with K objective is given as:

$$d(\boldsymbol{z}_i) = \sum_{k=1}^{K} \frac{f_k(\boldsymbol{z}_{i+1}) - f_k(\boldsymbol{z}_{i-1})}{f_k^{\max} - f_k^{\min}} \tag{5}$$

where $\boldsymbol{z}_{i+1}$ denotes the closest individual above $\boldsymbol{z}_i$ and $\boldsymbol{z}_{i-1}$ is the closest individual below $\boldsymbol{z}_i$. $f_k^{\max}$ and $f_k^{\min}$ are the the maximum and minimum value of the k-th objective, respectively.

Non-dominated ranking with k molecular properties and potentially conflicting objectives is defined in the latent space:

$$\min_{\boldsymbol{z}}(f_1(\boldsymbol{z}), f_2(\boldsymbol{z}), f_3(\boldsymbol{z}), ..., f_k(\boldsymbol{z})) \tag{6}$$

where $\boldsymbol{z}$ represents the feasible solutions. $\boldsymbol{z_1} \prec \boldsymbol{z_2}$ denotes $\boldsymbol{z}_1$ dominates $\boldsymbol{z}_2$, and samples from training data or DGMs can be sorted into Pareto fronts $\mathcal{F} = \{\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, ...\}$. For instance, the first front $\mathcal{F}_1$ dominates $\mathcal{F}_2$ but solutions in $\mathcal{F}_1$ do not dominate each other.

Two solutions are compared by combining non-dominated ranking and crowding distance using partial order $\prec_n$ i.e., given two solutions, $\boldsymbol{z}_1 \prec_n \boldsymbol{z}_2$ holds if $\boldsymbol{z_1} \prec \boldsymbol{z_2}$, or $\boldsymbol{z_1}$ do not dominate $\boldsymbol{z_2}$ but $d(\boldsymbol{z}_1) > d(\boldsymbol{z}_2)$. This procedure is performed to rank the top M samples for selection with regard to evolutionary operations, or to form the new population for the next generation [3].

2 Evaluation Metrics

In this work, we used 1-Wasserstein distance and hypervolume to evaluate the performance of multi-objective solutions.

2.1 1-Wasserstein Distance

In the space $\mathbb{R}$, there are many ways to describe the distance between two probability distributions $q(\boldsymbol{x})$ and $p(\boldsymbol{x})$. One of the more popular ones is Kullback-Leibler (KL) divergence:

$$\mathrm{KL}(p \parallel q) = \int_{\mathbb{R}^n} p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{q(\boldsymbol{x})} \mathrm{d}\boldsymbol{x}. \tag{7}$$

As defined above, KL does not measure the geometric properties of $\mathbb{R}^n$, because the comparison between $q(\boldsymbol{x})$ and $p(\boldsymbol{x})$ is made at the same points. This motivates us to utilize Wasserstein distance (WD) as a distance metric for a pair of distributions.

For the probability distributions μ and υ defined on $\mathbb{R}$, the p-th is given as:

$$W_p(\mu,\upsilon) := \inf_{\gamma\in\Gamma(\mu,\upsilon)} \Big(\int_{\mathbb{R}\times\mathbb{R}} d(x,y)^p \mathrm{d}\gamma(x,y) \Big)^{\frac{1}{p}}, \tag{8}$$

where $\Gamma(\mu,\upsilon)$ is the collection of joint probability measures γ on $\mathbb{R}\times\mathbb{R}$ with marginal distributions μ and υ. A measure with marginals μ and υ is also called the coupling of μ and υ. d can be any distance on $\mathbb{R}$, such as Euclidean distance, l_1 distance, etc. In the case of $p = 1$ and $d(x, y) = |x - y|$, the one-dimensional Wasserstein distance (1-Wasserstein distance) is explicitly formulates as:

$$W_1(\mu,\upsilon) = \int_{\mathbb{R}} |F_\mu(x) - F_\upsilon(x)| \mathrm{d}x, \tag{9}$$

where F is a cumulative distribution function.

2.2 Hypervolume Measure

The condition of algorithm convergence is an extremely critical aspect of multi-objective evolutionary algorithms. DEL retains the elite solution set of the previous generation and adds it to the evolutionary process of the new generation. The solution set of the evolutionary population continues to converge to the real Pareto frontier, and reaches a satisfactory optimization solution. Usually, when analyzing the performance of a multi-objective optimization algorithm, we hope that the algorithm can advance in three aspects. (1) The distance between the real Pareto front surface and the one obtained by the algorithm should be as small as possible. (2) Although the obtained individual solution points are only partial solutions, they should be distributed on the Pareto front as uniformly as possible. And, (3) a sufficient number of solution points should be able to cover the entire front, that is each region on the front should be represented by solution points unless this region is missing on the actual optimal Pareto front.

The hypervolume (HV) index measures the volume of the dimensional region in the target space bounded by the non-dominated solution set obtained by a multi-objective optimization algorithm and a pre-specified reference point. The mathematical representation of the HV calculation is given in Equation (10):

$$\mathrm{HV} = \delta(\cup_{i=1}^{|S|} v_i), \tag{10}$$

where δ represents the Lebesgue measure, which is used to compute volume, $|S|$ represents the number of non-dominated solutions, and v_i represents the hypercube formed by the reference point and the i-th solution in the solution set. HV is an effective quantitative scalar metric, which is strictly monotonic in terms of Pareto dominance. The larger the value of HV, the better the set of solutions covers the objective space and trade-offs between objectives. Especially, the calculation of the HV index does not require the ideal Pareto front of the test problem, which greatly facilitates the use of HV in practical applications.

3 Protein Targets

- Carbonic anhydrase IX protein (CA9/CAIX) [5]: CA9/CAIX has emerged as a significant marker of tumor hypoxia, holding potential as a diagnostic biomarker, prognostic indicator, and therapeutic target for cancer. Solid tumors depend on blood supply for oxygen and nutrient delivery. As the tumor expands, the blood vessels struggle to adequately supply these necessities, leading to regional hypoxia. Over time, this hypoxic environment causes acid accumulation within tumor cells. To counteract this stress, cancer cells release enzymes that neutralize the acidity, promoting their survival and potential metastasis. CA9, a major neutralizing enzyme, plays a pivotal role in cancer cell survival, invasion, and metastasis.
- Phospholipid hydroperoxide glutathione peroxidase or glutathione peroxidase 4 (GPX4) [6]: GPX4 is the fourth member of the selenium-containing GPX family. Various signaling pathways can induce cell death, with ferroptosis being an iron-dependent form. Research has shown that directly targeting GPX4 triggers ferroptosis in cancer cells, positioning GPX4 inhibitors as potent anticancer agents. Several GPX4 inhibitors, including RSL3, ML162, DPI compounds, FIN56, and FINO2, have been identified. However, their clinical applications are limited due to their suboptimal pharmacokinetic properties.
- Lysophosphatidic acid receptor 1 (LPA1) [7]: LPA1 is a bioactive lipid mediator primarily derived from membrane phospholipids. LPA signaling drives cell migration and proliferation, cytokine production, thrombosis, fibrosis, angiogenesis and lymphangiogenesis. Its signaling pathway is one of the most investigated mechanisms because over-expression of one or more of the LPA Receptors (LPARs) was found in several types of cancer, and therefore, the concept to modulate cancer by agonizing or antagonizing LPARs is naturally generated. In particular, LPAR 1 promotes the development of intra-tumoral heterogeneity by regulating PI3K/AKT signaling, and enhances metastasis and tumor motility. Aberrant expressions of LPAR1 were observed in many cancer cell lines and primary tumors, including ovarian cancer, breast cancer, liver cancer, gastric cancer, pancreatic cancer, lung cancer, glioblastoma (GBM) and osteosarcoma. Despite the apparent relevance of LPA signaling in cancer initiation, progression, metastasis and development of resistance against chemo- and radio-induced cancer cell death, no inhibitors targeting LPARs have progressed to cancer-related clinical trials this far.

4 Related Work

4.1 Deep Learning for Drug Discovery

There are a variety of deep generative models (DGM) for molecular discovery. Precursor DGM methods convert a molecule structure into a string, such as the simplified molecular-input line-entry system (SMILES) string. Generative autoencoders (e.g., variational autoencoder (VAE) [8]) have been used as a character-based language generative model for SMILES strings in order to enable efficient molecular generation. CVAE [9] converts a discrete SMILES representation of a molecule to a continuous multidimensional representation. This allows for the efficient search and optimization of

compounds in chemical space. SD-VAE (Syntax-directed VAE) [10] proposes stochastic lazy attributes that integrate the syntactic and semantic checking of SMILES strings into VAE to constrain the decoder directly. Sattarov et al. [11] introduced the Generative Topographic Mapping (GTM) to the sequence-to-sequence VAE in order to shape the latent space around the molecular properties of interest. Other generative models are used in SMILES-based molecular generation tasks, such as LatentGAN [12], ORGAN [13] which trained a GAN to generate SMILES strings. Although these methods are efficient in terms of training and sampling, they generate a large number of chemically invalid and repetitive molecules. Fragment-based drug design has been proposed as an alternative approach to the design of novel compounds. As a modification of the SMILES fragment-based drug design approach originally presented in [14], FragVAE [3] utilizes the Breaking of Retro-synthetically Interesting Chemical Substructures algorithm (BRICS) [15] to collect sequences of fragments from molecules. FragVAE employs a gated recurrent unit (GRU) [16, 17] encoder to map embeddings into a stochastic latent representation space. A latent vector (either generated by the encoder or sampled from the prior distribution) is used as the initial hidden state of a GRU-based autoregressive decoder to calculate the probability of the next possible fragments. FragVAE is currently integrated within DEL.

Subsequent techniques represented molecule structure as graphs with nodes and edges corresponding to atoms and bonds. Graph-based methods [18–21] leverage molecular topology information explicitly in the generator to learn a molecular structure as a natural two-dimensional or three-dimensional graph. Li et al. [22] incorporated graph neural network (GNN) into VAE by adding new structures to account for the sequential composition of molecular graphs. CGVAE (Constrained Graph VAE) [23] further investigates the VAE architecture with gated graph neural networks (GGNNs), in which gradient ascent is used to optimize the graph properties locally. Chen et al. [24] developed a novel deep generative model named Modof (modifier with one fragment) that predicts a single cleavage site in a molecule and removes and/or adds fragments at that site to modify molecules. However, solving the problem of graph isomorphism is computationally expensive and, as a result, graph reconstruction is extremely ineffective and inaccurate without imposing constraints. The maneuvering of other generative models in drug design persists such as adversarial learning based models [25–27] and flow-based models [28–32]. FAME [33] proposes an innovative generative model specifically for phenotypic drug design that is capable of learning the conditional distribution $P_\theta(molecule|phenotype)$ on gene expression profiles derived from molecular graphs. Using an encoder-decoder scheme, it consists of a conditional graph encoder and a fragment-based decoder which assembles fragments in an autoregressive manner. Motifs-based models [34, 35] can incorporate domain knowledge for controlling the size of motif vocabulary. JTVAE [36], a graph-fragment based approach, decomposes training molecules into a set of molecular substructures including rings, functional groups, and atoms. In order to optimize molecular properties, JTVAE was built on top of the junction tree-based encoder-decoder neural network and refined with graph-to-graph transformation [37, 38] and autoregressive techniques [39, 40]. In later work, the authors of JTVAE have extended their graph VAE architecture by introducing larger graph motifs instead of small substructures.

The method is referred to as the motif-based hierarchical encoder-decoder (HierG2G) model [34], which allows for more flexible substructure representation and more efficient molecular graph reconstruction. Previous work has integrated JTVAE into the DEL framework, whereby JTVAE provides a latent representation space for the multi-objective evolutionary algorithm to explore, while meanwhile elite samples from each generation of multi-objective evolutionary algorithm continue to learn JTVAE.

Further developments include 3D-graph and geometric deep learning methods to exploit the geometric properties of molecular structures [41]. Luo et al. [42], Li et al. [43] generate molecules in 3D space with high probability of the specific protein binding pocket targets from sampled atoms learned from an order-dependent distribution. Powers et al. [44] expands molecule fragments with specific physicochemical properties leveraging a 3D atomic point cloud representation and E(3) equivariant graph neural networks. Most recently, several equivariant diffusion-based models for molecule generation in 3D have been proposed [45–49]. These variants can be seen as generative autoencoders, where diffusion processes are employed as encoders and decoders. Hoogeboom et al. [45] introduces an equivariant diffusion model (EDM) for molecule generation that employs an equivariant E(3) graph neural network (EGNN) [50] to train flow-based model and diffusion model, and generate valid molecules in 3D space. EDM learns to denoise a diffusion process, which has shown to scale better than previous works during training. Vignac et al. [46] introduced DiGress, a denoising diffusion model for graph generation with categorical node and edge attributes that operates on a discrete space. Trippe et al. [49] defines a diffusion model for molecule generation in 3D that adapts E(3)-equivariant graph convolutional neural networks for efficient protein design and returning a diverse set of motif-supporting scaffold. These approaches essentially solve a point cloud generation task as they learn distributions over the Euclidean space with 3 coordinates per atom. Xu et al. [51], Jing et al. [52] employ diffusion model for conformation generation (i.e., for predicting coordinates from molecular graphs). GeoDiff [51] treats each atom as a particle and learns to directly reverse the diffusion process (i.e., transforming from a noise distribution to stable conformations) as a Markov chain (evolving with equivariant Markov kernels). Jing et al. [52] proposes a torsional diffusion framework for generating molecular conformers based on a diffusion process that acts only on the torsion angles and leaves the other degrees of freedom fixed.

Transformers, which replace recurrence with a multi-head self-attention mechanism, are shown to be superior to recurrent neural networks (RNNs) (GRU, LSTM) in terms of convergence time, featurization on long sequences and complicated problems [53], and stability of the training process [54]. The long-range syntactical dependencies learned by attention models have shown to be greatly beneficial for molecular generative tasks [55]. Grechishnikova [56] used a complete Transformer architecture to generate lead compounds for specific proteins with only sequence information, treating molecular generation as a translation task for protein sequences and molecular SMILES. A Transformer-encoder-based generative model with transfer learning and reinforcement learning was developed in [54] for designing new drugs with desirable activity against the protein target BRAF. This method was comprehensively compared with a GRU-based generative model and has exhibited improvement in the

percentage of generating chemically valid molecules, the structural diversity and activity of the generated molecules, the feasibility of molecular synthesis, the calculation speed and precision. Shrivastava and Kell [57] established a successful, novel and disentangled latent space by coupling transformers with contrastive learning, which allows similar molecules to cluster together in an effective and interpretable way. Another work, ReLSO [58], combines the powerful encoding ability of the transformer with an autoencoder-type bottleneck that produces information-rich, interpretable and low dimensional latent representations and jointly generates protein sequences as well as predict fitness from latent representations. However, since the vanilla transformer does not have an obvious latent space at the bottleneck such as that produced by a VAE, it cannot be directly integrated in the DEL framework and therefore, this combination is one of our future explorations for drug design. There has been recent development and success with regard to creating well-defined latent encodings in transformers.

4.2 Reinforcement Learning for Drug Discovery

In recent years, deep reinforcement learning (DRL) frameworks have emerged as effective methods for molecular generation and optimization [59–69]. Some of these methods are discussed below.

Olivecrona et al. introduced REINVENT [62], a sequence-based model for molecular generation. An RNN is pre-trained on the ChEMBL database to understand SMILES syntax and the distribution of ChEMBL's molecular structure. The RNN, acting as a prior, is then fine-tuned through an RL policy network that makes use of augmented episodic likelihood, which combines the prior likelihood with the RL reward function. The task of the RL agent is to generate SMILES strings in a character by character fashion. As proof of concept, three separate experiments were conducted: learning to generate molecules (1) without sulphur, (2) similar to any given structure (Celecoxib), and (3) active against dopamine receptor D2, with success in all three experiments.

In Objective-Reinforced Generative Adversarial Network for Inverse-design Chemistry (ORGANIC), Sanchez-Lengeling and colleagues [65] employed a combination of Generative Adversarial Network (GAN) and reinforcement learning to generate molecules in the form of SMILES strings. Given an initial distribution of molecules, the generator, an RNN that acts as a policy gradient agent, forms SMILES strings character by character. Generated molecules are of similar structures to molecules in the original distribution but are inclined towards satisfying properties of interest. The CNN-based discriminator is tasked with distinguishing between molecules produced by the generator and those from the original distribution. The generator is updated through feedback from the discriminator, along with the reward function. This framework successfully developed molecules optimized for target properties such as melting point, QED and organic photovoltaic material design.

Zhou and coauthors developed Molecule Deep Q-Networks (MolDQN) [69], a drug design framework that utilizes double deep Q-learning (DDQN) to optimize molecules represented as SMILES strings. Optimization of molecules is achieved through performing atom additions, bond additions, and bond removals. This framework was

successful in developing molecules that resemble the initial molecules, while being optimized in terms of Quantitative Estimate of Druglikeness (QED).

Deep Fragment-based Multi-Parameter Optimization (DeepFMPO), presented in [67], uses an actor-critic algorithm to optimize molecules in the form of SMILES strings. After fragmenting an initial set of lead molecules using the fragmentation method proposed in HierVAE [34], an agent in the form of an RNN is trained to improve these molecules in terms of LogP, polar surface area, and molecular weight by replacing their fragments with ones that are of similar molecular structures from the fragment library. As a result, produced molecules are optimized, while possessing structural resemblance to the original lead molecules.

Simm et al. [66] proposed a generative DRL framework guided by quantum mechanics to develop valid, diverse and stable molecular structures in Cartesian coordinates. To generate molecules, the agent uses proximal policy optimization (PPO) to learn the location of atom additions and what element atoms should be. The utilization of Cartesian coordinates allows for detailed molecular representations that cannot be achieved through graph or SMILES-based frameworks.

In RationaleRL [61], the concept of a molecular rationale, a substructure within a molecule that is expected to be accountable for a specific target property, is employed to guide the generation of molecules that are diverse, novel and inhibitors against specified proteins (GSK3β and JNK3). Given a dataset of molecules represented by graphs, a Monte Carlo tree search is applied to identify single-property rationales. These rationales are then merged to produce a library of multi-property rationales. A graph generator, in the form of a VAE, is pre-trained on a large set of real molecules, with the goal of learning how to form a full molecule from a given sub-graph. The generator is then fine-tuned using a policy gradient algorithm to generate complete molecules by sequentially inserting atoms along with their corresponding connecting edges to multi-property rationales. Generated molecules are rewarded based on whether or not they achieve target properties.

Introduced by Yang and coauthors, Fragment-Based Generative RL with Explorative Experience replay for Drug design using Predictive Error (FREED(PE)) [68] is a graph-based generative model that deploys soft actor-critic (SAC), a DRL algorithm that emphasizes exploration, while maximizing the expected reward. Following the preparation of a fragment library using the CReM fragmentation method [70], this model learns to generate molecules with favourable protein-ligand binding affinity score (BAS). The agent produces molecules sequentially by selecting the next fragment to attach, the location of the attachment, and the location of the chemical bond between the attachment and current molecular state.

While DRL methods continue to advance in the field of drug discovery, they frequently encounter limitations such as lengthy training durations, inefficient use of training data, and significant energy and computational requirements [71, 72]. Continuous improvements of these methods are warranted to enhance their efficacy and achieve higher degrees of success.

5 Extended Experimental Results

5.1 Hyperparameter Settings

During the experiments, we used the same evolutionary learning hyperparameters for evaluating the DEL framework. However, for FragVAE and JTVAE, we tuned the hyperparameters differently to optimize their performance.

5.1.1 FragVAE+DEL

Unless otherwise indicated in the main text, the subsequent values of hyperparameters are utilized in the FragVAE+DEL experimental proceedings.

Hyperparameter setting for FragVAE:

- size of the embedding layer: 128
- mask low-frequency fragments: Yes
- masking frequency: 10
- number of low frequency clusters: 10
- window for word2vec embedding: 3
- number of hidden layers: 2
- dimension of hidden state: 128
- size of the VAE latent space: 64
- batch size for training: 128
- optimizer learning rate: 0.001
- maximum length of the sampled sequence: 10
- shuffle batches: Yes
- dropout for the hidden layers: 0.3
- threshold to clip the gradient norm: 5
- number of layers of the MLP: 2
- dimension of hidden state of MLP: 64
- weight on property regression loss (α): 1
- weight on KL divergence (β): 0.1

Hyperparameter setting for DEL:

- number of evolutionary generations: 20
- population size: 20000
- number of epochs in the initial training of FragVAE: 20
- number of epochs in the subsequent training of FragVAE: 5
- scheduler annealing step size for the initial training of FragVAE: 4
- scheduler annealing rate: 0.8
- learning rate in the initial training: 0.0001
- crossover method: linear
- mutation rate: 0.01
- probability in tournament selection: 0.95

5.1.2 JTVAE+DEL

Unless otherwise indicated in the main text, the subsequent values of hyperparameters are utilized in the JTVAE+DEL experimental proceedings.

Hyperparameter Setting for JTVAE:

- size of the embedding layer: 128
- number of hidden layers: 1
- dimension of hidden state: 450
- size of the VAE latent space: 32
- batch size for training: 32
- maximum length of the sampled sequence: 10
- shuffle batches: Yes
- depth of tree message passing network: 20
- depth of graph message passing network: 3
- threshold to clip the gradient norm: 50
- number of layers of the MLP: 2
- dimension of hidden state of MLP: 64
- weight on KL divergence (β): 0.1

Hyperparameter Setting for DEL:

- number of evolutionary generations: 20
- population size: 20000
- number of epochs in the initial training of JTVAE: 20
- number of epochs in the subsequent training of JTVAE: 5
- scheduler annealing step size for the initial training of JTVAE: 4
- scheduler annealing rate: 0.8
- learning rate in the initial training: 0.0001
- crossover method: linear
- mutation rate: 0.01
- probability in tournament selection: 0.95

5.1.3 AAE+DEL

Unless otherwise indicated in the main text, the subsequent values of hyperparameters are utilized in the AAE+DEL experimental proceedings.

Hyperparameter setting for AAE:

- size of the embedding layer: 32
- number of hidden layers: 2
- dimension of hidden state: 512
- size of the VAE latent space: 128
- discriminator layer sizes: [640, 256]
- property predictor layer sizes: [640, 256]
- discriminator step: 3
- batch size for training: 64
- dropout for the hidden layers: 0.3

- optimizer learning rate: 0.001

Hyperparameter Setting for DEL:

- number of evolutionary generations: 20
- population size: 20000
- number of epochs in the initial training of AAE: 20
- number of epochs in the subsequent training of AAE: 5
- scheduler annealing step size for the initial training of AAE: 4
- scheduler annealing rate: 0.8
- learning rate in the initial training: 0.0001
- crossover method: linear
- mutation rate: 0.01
- probability in tournament selection: 0.95

5.2 Examples of Top Novel Drugs Binding Protein Targets

In our comprehensive study, we have utilized various computational methods to design novel compounds, showcasing their docking potential with target proteins.

First of all, we have also included docking visualizations in Fig. 1, 2, and 3, depicting the highest-ranked molecules generated by all three single-target methods docked to CA9, GPX4, and LPA1 protein surfaces, respectively. Furthermore, the binding interactions of FragVAE+DEL and JTVAE+DEL designed molecules with both CA9 and GPX4 proteins are detailed in Fig. 4 and 5. Fig. 6, 7, and 8 illustrate molecules designed by AAE+DEL, FragVAE+DEL, and JTVAE+DEL, highlighting their bifunctional characteristics with effective binding both CA9 and LPA1 proteins. This is further expanded with Fig. 9, 10, and 11, showcasing molecules with dual binding capabilities to GPX4 and LPA1 proteins. Interestingly, Fig. 12 and 13 present examples of high-ranking molecules generated by FragVAE+DEL and JTVAE+DEL, respectively, each demonstrating docking capabilities to CA9, GPX4, and LPA1 protein surfaces. These visualizations highlight the molecules' unique binding interactions and the versatility and specificity of our computational approaches, demonstrating their efficacy in generating compounds with diverse binding affinities.

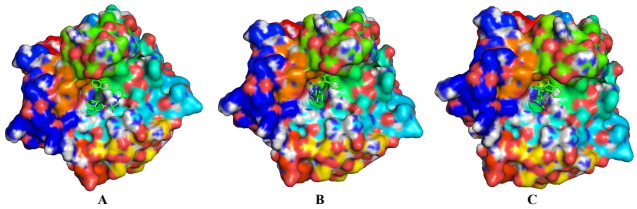

Fig. 1 Visualization of the highest-ranked novel molecules generated by single-target methods on {SAS, logP, BCA9} docked to CA9 protein surface. **(A)** shows the molecule `O=C(Cn1nc(-c2ccc3cccc3c2)ccc1=O)Nc1nc2ccccc2c(=O)n1-c1ccc2nnnn2c1` from **AAE+DEL** with the CA9 binding affinity score of -11.5 (SAS: 2.848, logP: 2.839), **(B)** shows the molecule `CC(N)NC(=O)C(=O)NN=C1C(=O)N(Cc2cccc3ccccc23)c2ccccc21` from **FragVAE+DEL** with the CA9 binding affinity score of -9.5 (SAS: 2.915, logP: 1.628), while **C** shows the molecule `Cc1ccc(C2C3=C(Nc4[nH]c(N)nc(=O)c42)c2ccccc2C3=O)cc1` from **JTVAE+DEL** with the CA9 binding affinity score of -9.1 (SAS: 2.971, logP: 2.826).

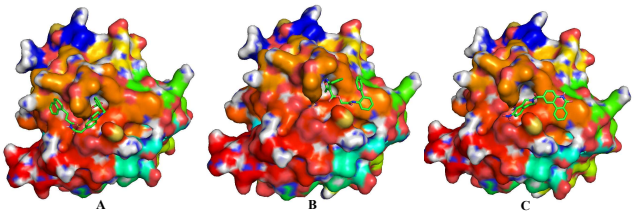

Fig. 2 Visualization of the highest-ranked novel molecules generated by single-target methods on {SAS, logP, BGPX4} docked to CA9 protein surface. **(A)** shows the molecule `O=C(Cc1ccc2c(c1)CCN2Cc1cccc(C(=O)[O-])c1)NC(=O)CCc1cccc2ccccc12` from **AAE+DEL** with the GPX4 binding affinity score of -10.3 (SAS: 2.650, logP: 3.584), **(B)** shows the molecule `Cc1onc(-c2ccccc2)c1C(=O)OCC(=O)Nc1ccccc1-c1ccccc1` from **FragVAE+DEL** with the GPX4 binding affinity score of -8.8 (SAS: 1.994, logP: 5.112), while **C** shows the molecule `O=C(c1ccc2[nH]c(=O)c3ccccc3c2c1)N1CCc2ccc([N+](=O)[O-])cc2C1` from **JTVAE+DEL** with the GPX4 binding affinity score of -9.1 (SAS: 2.270, logP: 3.788).

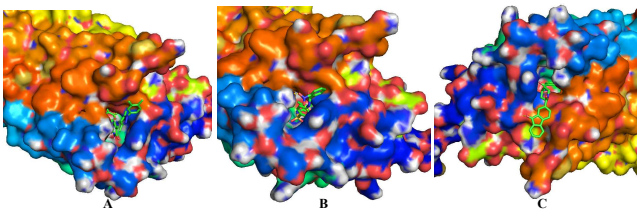

Fig. 3 Visualization of the highest-ranked novel molecules generated by single-target methods on {SAS, logP, BLPA1} docked to CA9 protein surface. **(A)** shows the molecule `Cc1ccc(N2C(=O)NC(=O)C(=CNc3c(C)nn(-c4ccc(C)c(C)c4)c3C)C2=O)cc1C` from **AAE+DEL** with the LPA1 binding affinity score of -9.8 (SAS: 2.670, logP: 4.302), **(B)** shows the molecule `Cc1noc([N-]S(=O)(=O)c2ccc(NC(=O)c3ccccc3[N+](=O)[O-])cc2)c1C` from **FragVAE+DEL** with the LPA1 binding affinity score of -9.7 (SAS: 2.950, logP: 3.846), while **C** shows the molecule `O=C(NC1CCN(C(=O)c2ccc3c(c2)C(=O)c2ccccc2-3)CC1)c1ccccc1` from **JTVAE+DEL** with the LPA1 binding affinity score of -9.9 (SAS: 2.007, logP: 3.932).

For a more comprehensive visualization of the generated molecular structures, Fig. 14 to 34 display a selection of 2D representations of high-quality samples from from AAE+DEL, FragVAE+DEL, and JTVAE+DEL. These are ranks in accordance with CA9, GPX4 and LPA1 binding scores. The illustrations prominently display the top 48 samples, thus elucidating the successful execution of our methodology.

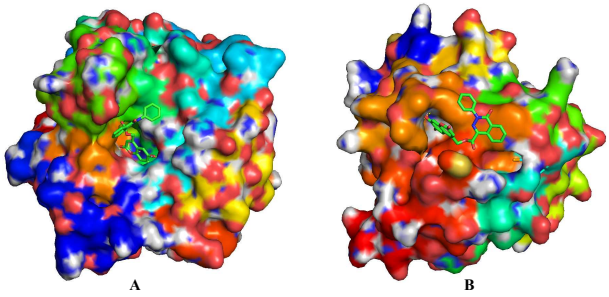

Fig. 4 Visualization of the highest-ranked novel molecule `O=C(OCc1cc(=O)oc2cc(O)c(O)cc12)c1nn(-c2ccccc2)c(=O)c2ccccc12` from dual-target **FragVAE+DEL** on {SAS, logP, BCA9, BGPX4} docked to both CA9 and GPX4 protein surfaces. This molecule has a CA9 binding affinity score of -9.3 and GPX4 score of -7.9 (SAS: 2.453, logP: 3.260). **(A)** shows the molecule binding the target binding site of CA9 protein, while **(B)** shows the molecule docked to the target site of GPX4.

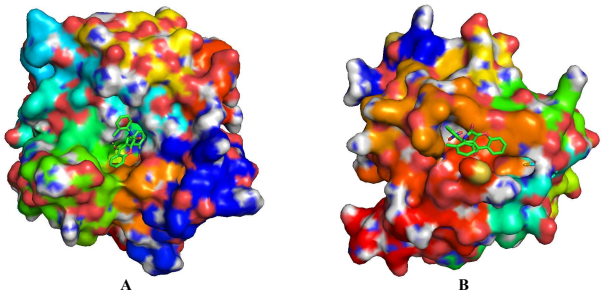

Fig. 5 Visualization of the highest-ranked molecule `Cn1nc(NC(=O)Cc2-cccc3ccccc23)c2c1NC(=O)CC2c1ccccc1` generated by dual-target **JTVAE+DEL** on {SAS, logP, BCA9, BGPX4} docked to both CA9 and GPX4 protein surface. This molecule has a CA9 binding affinity score of -9.6 and GPX4 score of -8.8 (SAS: 2.749, logP: 4.759). **(A)** shows the molecule binding the target binding site of CA9 protein, while **(B)** shows the molecule docked to the target site of GPX4.

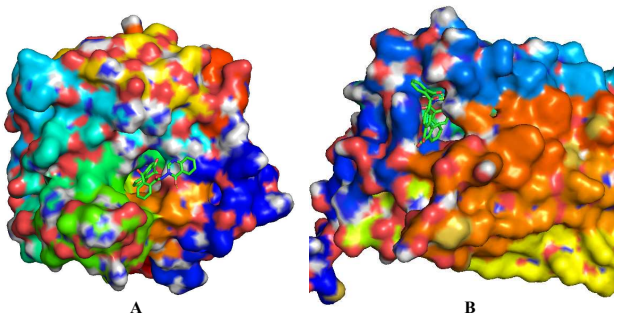

Fig. 6 Visualization of acquired highest-ranked molecule `O=C(Nn1cnc2ccccc2c1=O)c1ccccc1C(=O)c1ccc2c(c1)-c1ccccc1C(=O)C2=O` (CA9: -10.3, LPA1: -10.8, SAS: 2.519, logP: 4.057) from dual-target **AAE+DEL** on {SAS, logP, BCA9, BLPA1} docked to both CA9 and LPA1 protein surfaces. This molecule possesses bifunctional characteristic, where one end demonstrates affinity towards the CA9 protein, while the another end effectively binds the LPA1 protein. **(A)** shows the molecule binded to the target binding site of CA9 protein, while **(B)** shows the molecule docked to the target site of LPA1 with different conformation.

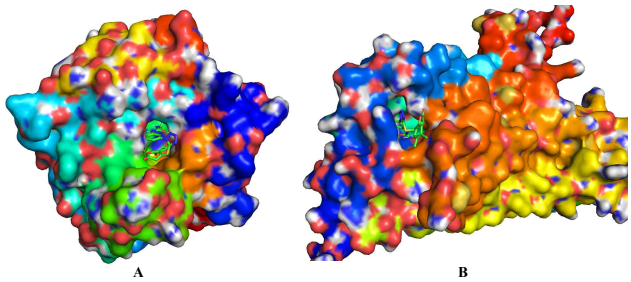

Fig. 7 Visualization of the highest-ranked novel molecule `C=C(C)NC(=O)Nc1cccc(-n2nnnc2SCc2cc(=O)oc3cc(C)c(C)cc23)c1` from dual-target **FragVAE+DEL** on {SAS, logP, BCA9, BLPA1} docked to both CA9 and LPA1 protein surfaces. This molecule has a CA9 binding affinity score of -9.1 and LPA1 score of -10.0 (SAS: 2.744, logP: 4.333). **(A)** shows the molecule binding the target binding site of CA9 protein, while **(B)** shows the molecule docked to the target site of LPA1.

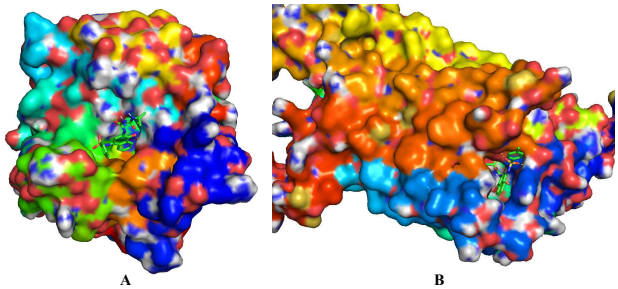

Fig. 8 Visualization of the highest-ranked novel molecule `C=C(C)NC(=O)NC(=O)NC1=CC(=NS(=O)(=O)c2ccc(C)cc2)c2ccccc2C1=O` from dual-target **JTVAE+DEL** on {SAS, logP, BCA9, BLPA1} docked to both CA9 and LPA1 protein surfaces. This molecule has a CA9 binding affinity score of -8.7 and LPA1 score of -9.4 (SAS: 2.874, logP: 2.795). **(A)** shows the molecule binding the target binding site of CA9 protein, while **(B)** shows the molecule docked to the target site of LPA1.

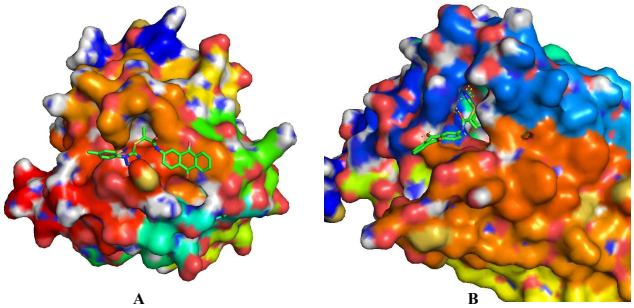

Fig. 9 Visualization of acquired highest-ranked molecule `Cc1ccc(-n2nnnc2SC(C)C(=O)Nc2ccc3c(c2)C(=O)c2ccccc2C3=O)cc1` (GPX4: -8.6, LPA1: -9.4, SAS: 2.740, logP: 3.865) from dual-target **AAE+DEL** on {SAS, logP, BGPX4, BLPA1} docked to both GPX4 and LPA1 protein surfaces. **(A)** shows the molecule binding the target binding site of GPX4 protein, while **(B)** shows the molecule docked to the target site of LPA1.

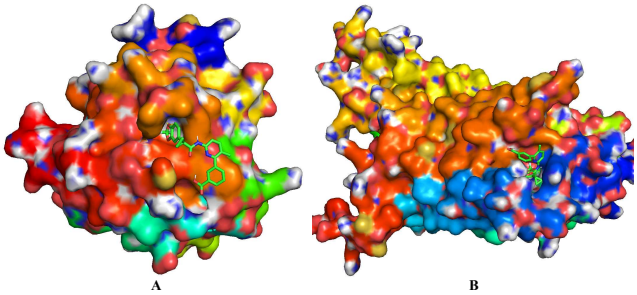

Fig. 10 Visualization of the highest-ranked novel molecule `Cc1ccc(NC(=O)C2(c3ccc4c(c3)OC(F)(F)O4)CC2)nc1-c1cccc(C(=O)O)c1` from dual-target **FragVAE+DEL** on {SAS, logP, BGPX4, BLPA1} docked to both GPX4 and LPA1 protein surfaces. This molecule has a GPX4 binding affinity score of -8.4 and LPA1 score of -9.4 (SAS: 2.756, logP: 4.747). **(A)** shows the molecule binding the target binding site of GPX4 protein, while **(B)** shows the molecule docked to the target site of LPA1.

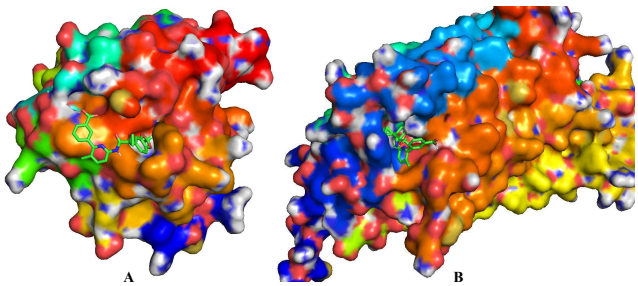

Fig. 11 Visualization of the highest-ranked novel molecule `Cc1ccc(N2C(=O)c3oc4ccc(F)cc4c(=O)c3C2c2cccc([N+](=O)[O-])c2)cc1` from dual-target **JTVAE+DEL** on {SAS, logP, BGPX4, BLPA1} docked to both GPX4 and LPA1 protein surfaces. This molecule has a GPX4 binding affinity score of -8.5 and LPA1 score of -9.2 (SAS: 2.787, logP: 4.899). **(A)** shows the molecule binding the target binding site of GPX4 protein, while **(B)** shows the molecule docked to the target site of LPA1.

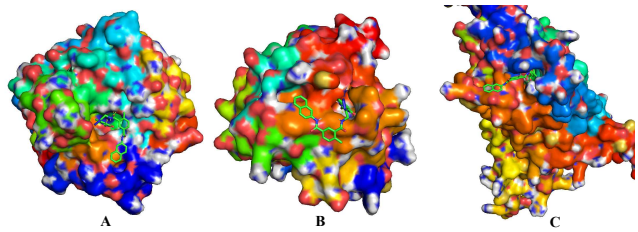

Fig. 12 Visualization of the highest-ranked molecule `Cc1ccc(NC(=O)C2(c3ccc4c(c3)OC(F)(F)O4)CC2)nc1-c1cccc(C(=O)O)c1` generated by triple-target **FragVAE+DEL** on {SAS, logP, BCA9, BGPX4, BLPA1} docked CA9, GPX4, and LPA1 protein surfaces. This molecule has CA9 binding affinity score of -8.8, GPX4 score of -8.4 and LPA1 score of -9.4 (SAS: 2.756, logP: 4.747). **(A)** shows the molecule binding the target binding site of CA9 protein, **(B)** shows the molecule docked to the target site of GPX4, while **(C)** shows the molecule docked to the target site of LPA1.

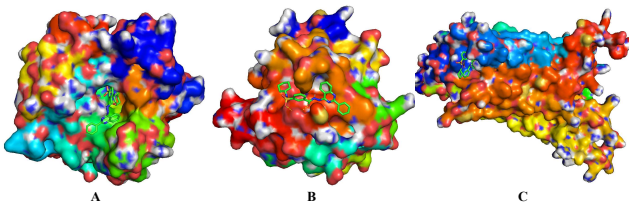

Fig. 13 Visualization of the highest-ranked molecule `O=C(NNc1nc(-c2ccccc2)nc2ccccc12)c1cccc(S(=O)(=O)NC2CCCCC2)c1` generated by triple-target **JTVAE+DEL** on {SAS, logP, BCA9, BGPX4, BLPA1} docked to CA9, GPX4, and LPA1 protein surfaces. This molecule has CA9 binding affinity score of -9.1, GPX4 score of -8.3 and LPA1 score of -9.4 (SAS: 2.331, logP: 4.665). **(A)** shows the molecule binding the target binding site of CA9 protein, **(B)** shows the molecule docked to the target site of GPX4, while **(C)** shows the molecule docked to the target site of LPA1.

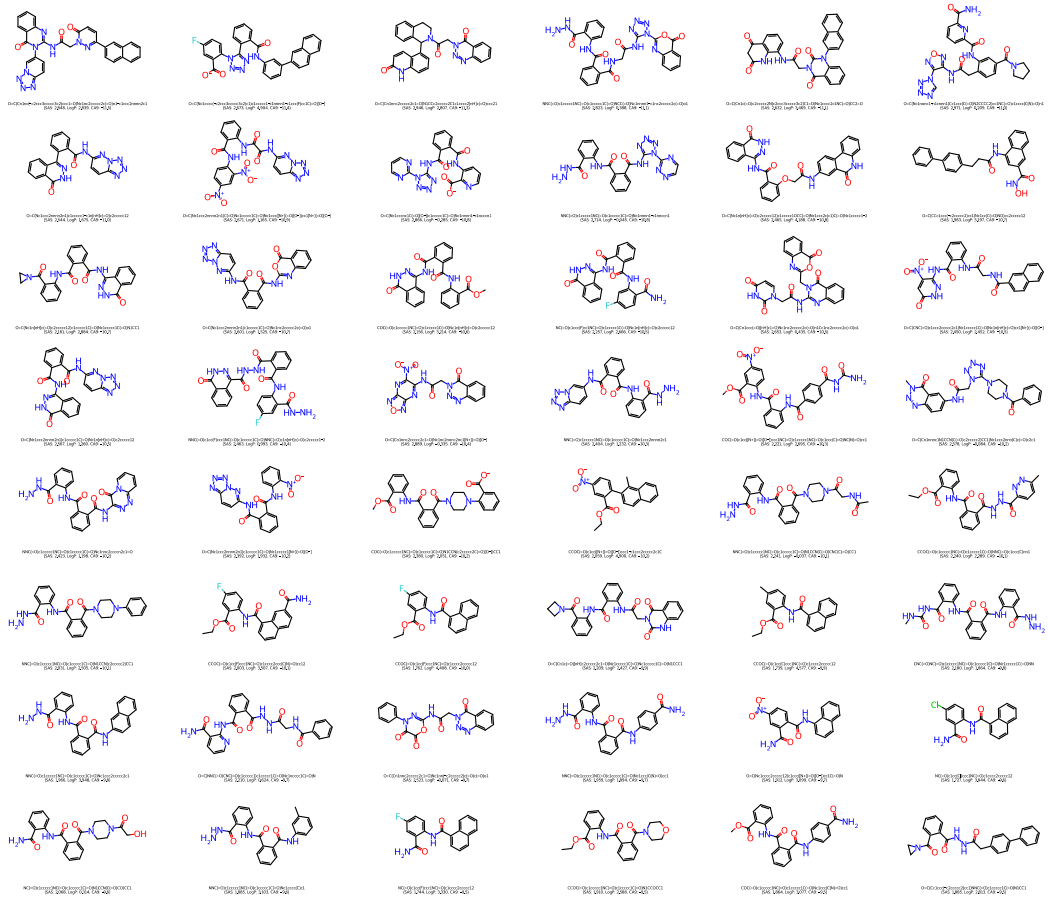

Fig. 14 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives {SAS, logP, CA9} in combination with **AAE**. Samples are ranked by CA9 binding scores.

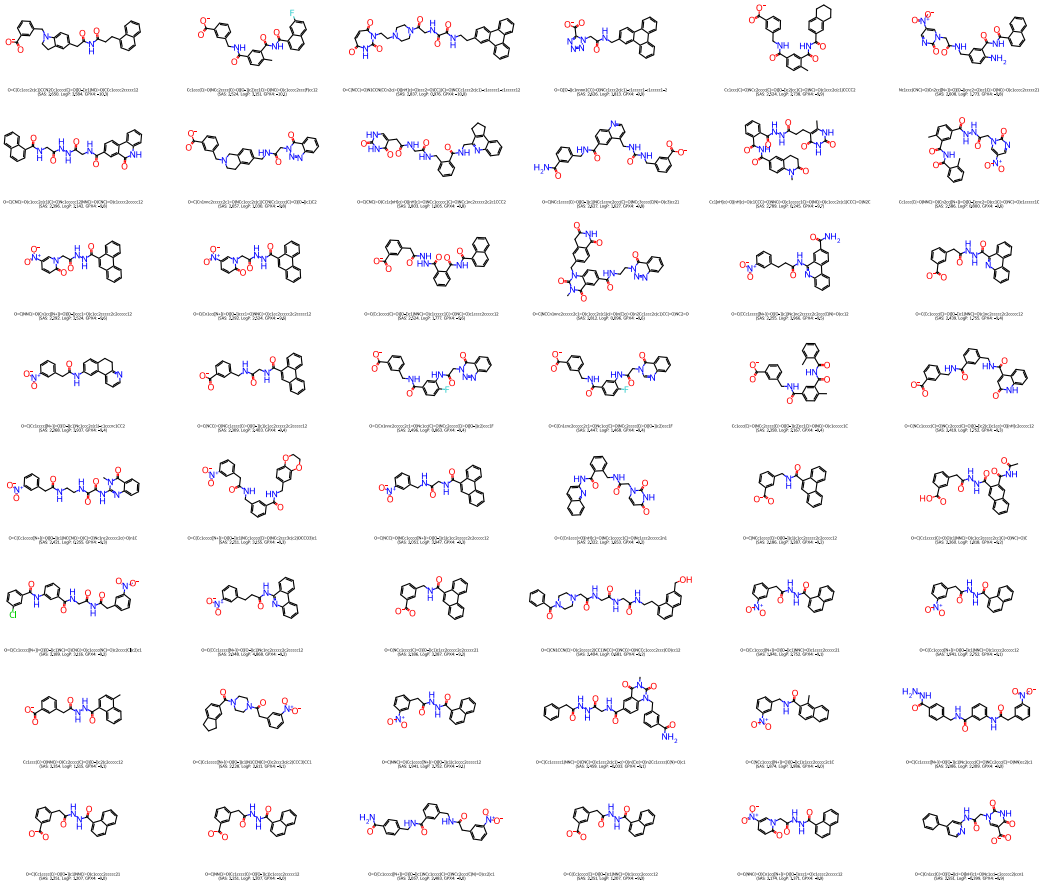

Fig. 15 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives {SAS, logP, GPX4} in combination with **AAE**. Samples are ranked by GPX4 binding scores.

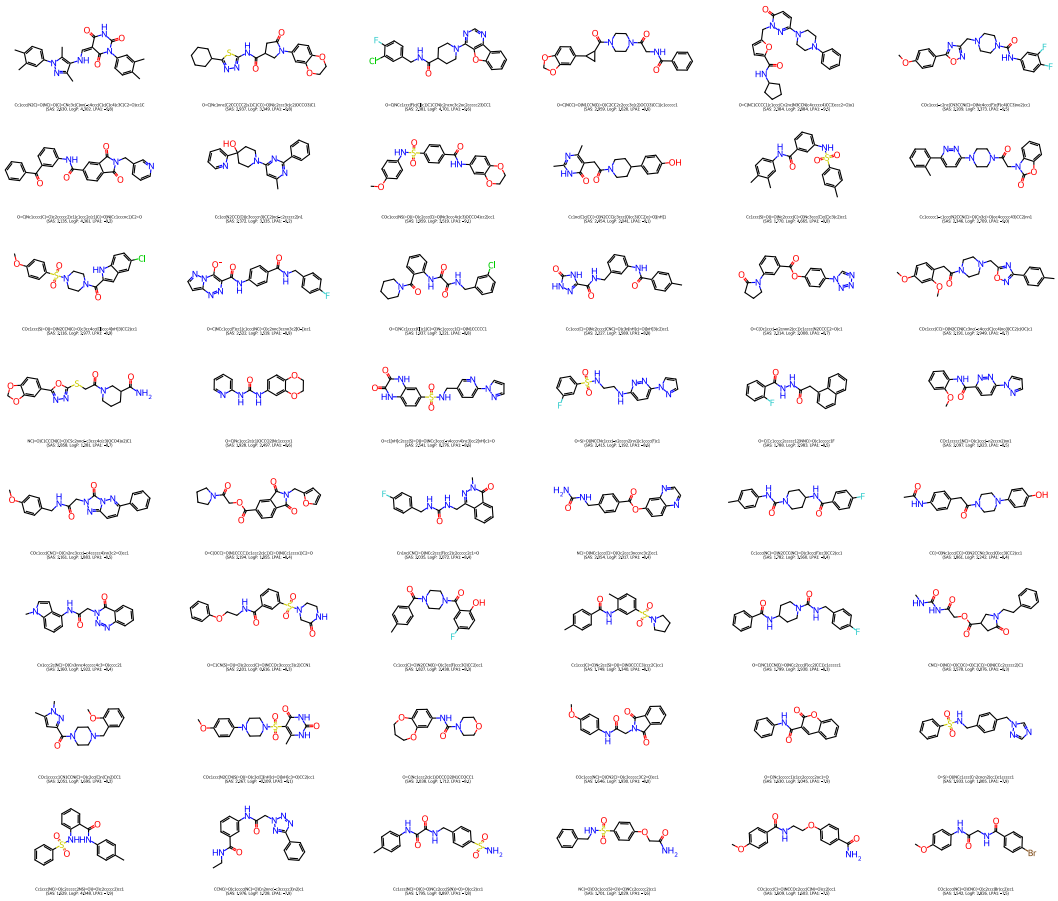

Fig. 16 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives {SAS, logP, LPA1} in combination with **AAE**. Samples are ranked by LPA1 binding scores.

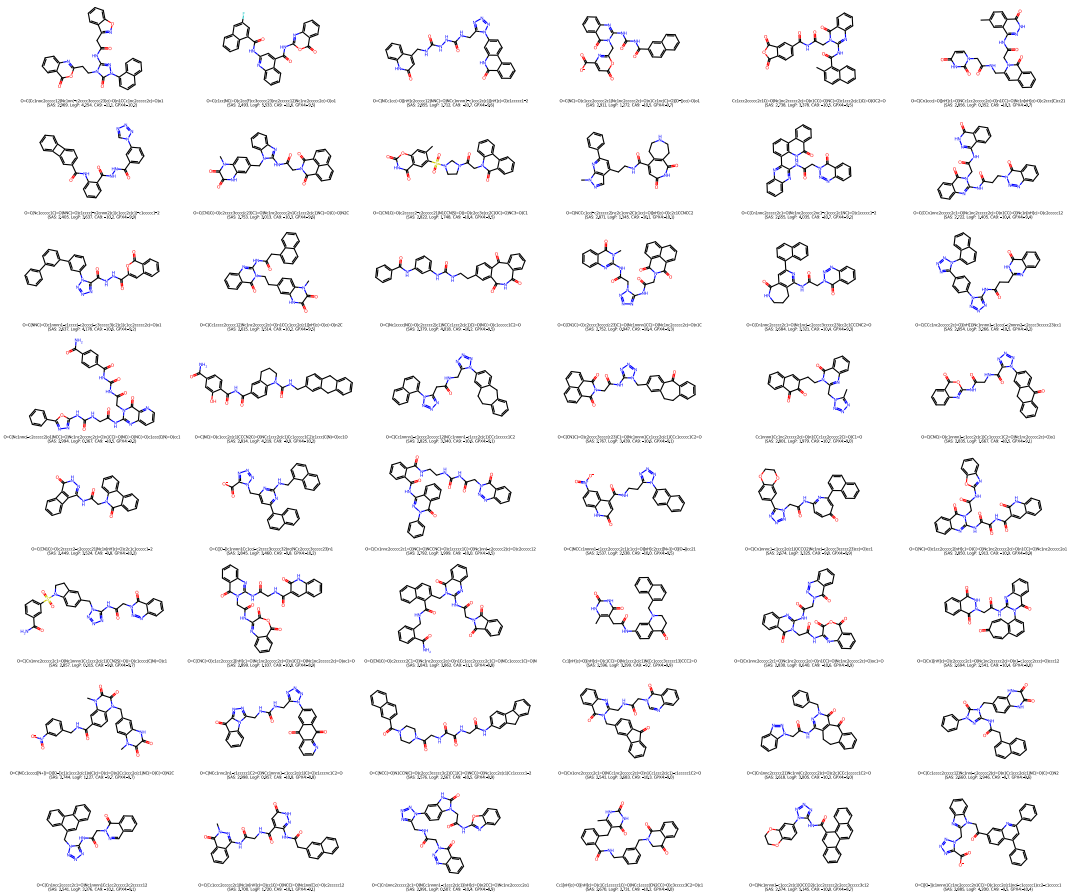

Fig. 17 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BCA9}, \text{BGPX4}\}$ in combination with **AAE**. Samples are ranked by CA9 and GPX4 binding scores respectively, then sorted by the sum of these ranks.

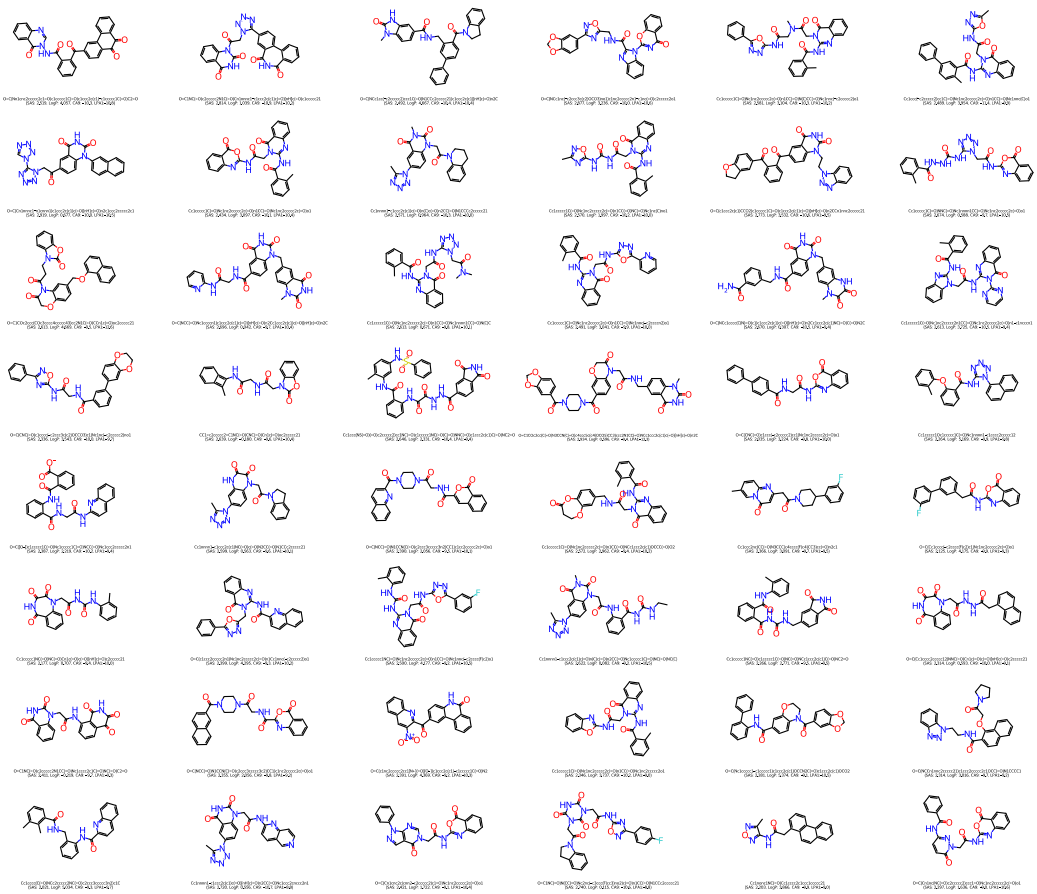

Fig. 18 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BCA9}, \text{BLPA1}\}$ in combination with **AAE**. Samples are ranked by CA9 and LPA1 binding scores respectively, then sorted by the sum of these ranks.

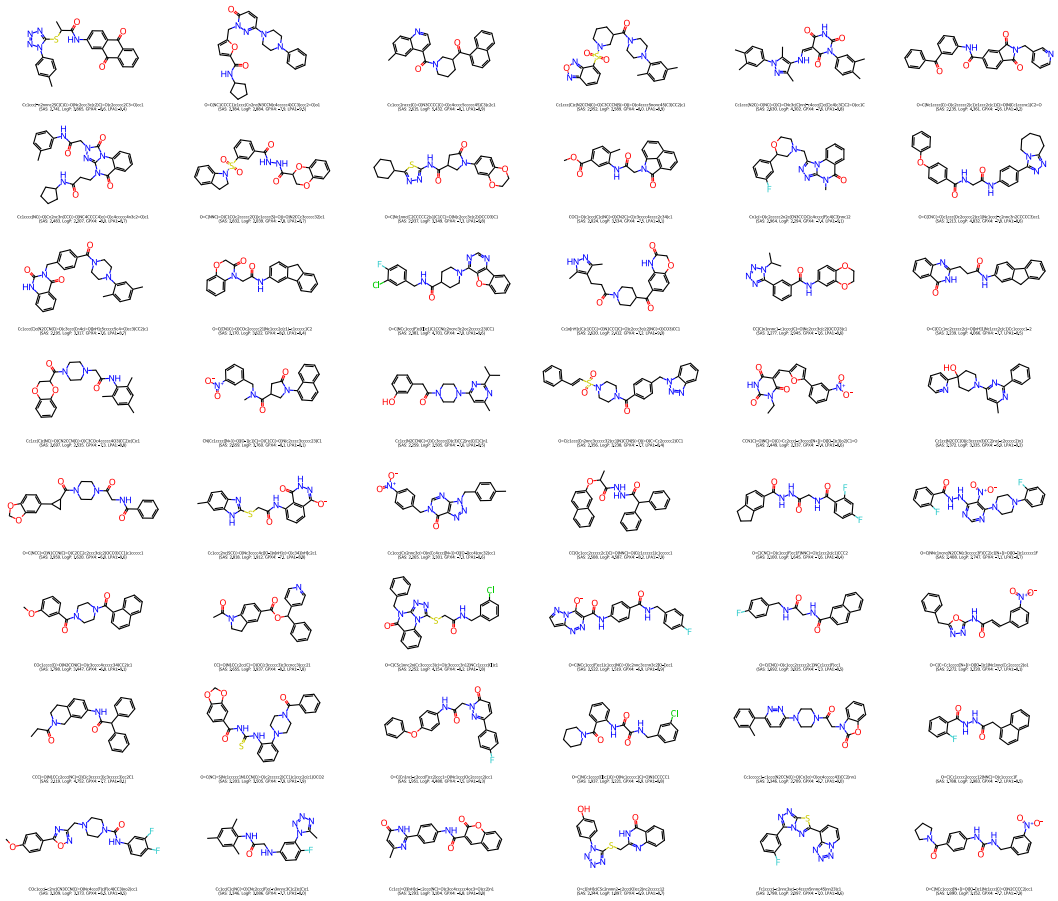

Fig. 19 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives {SAS, BlogP, BGPX4, BLPA1} in combination with **AAE**. Samples are ranked by GPX4 and LPA1 binding scores respectively, then sorted by the sum of these ranks.

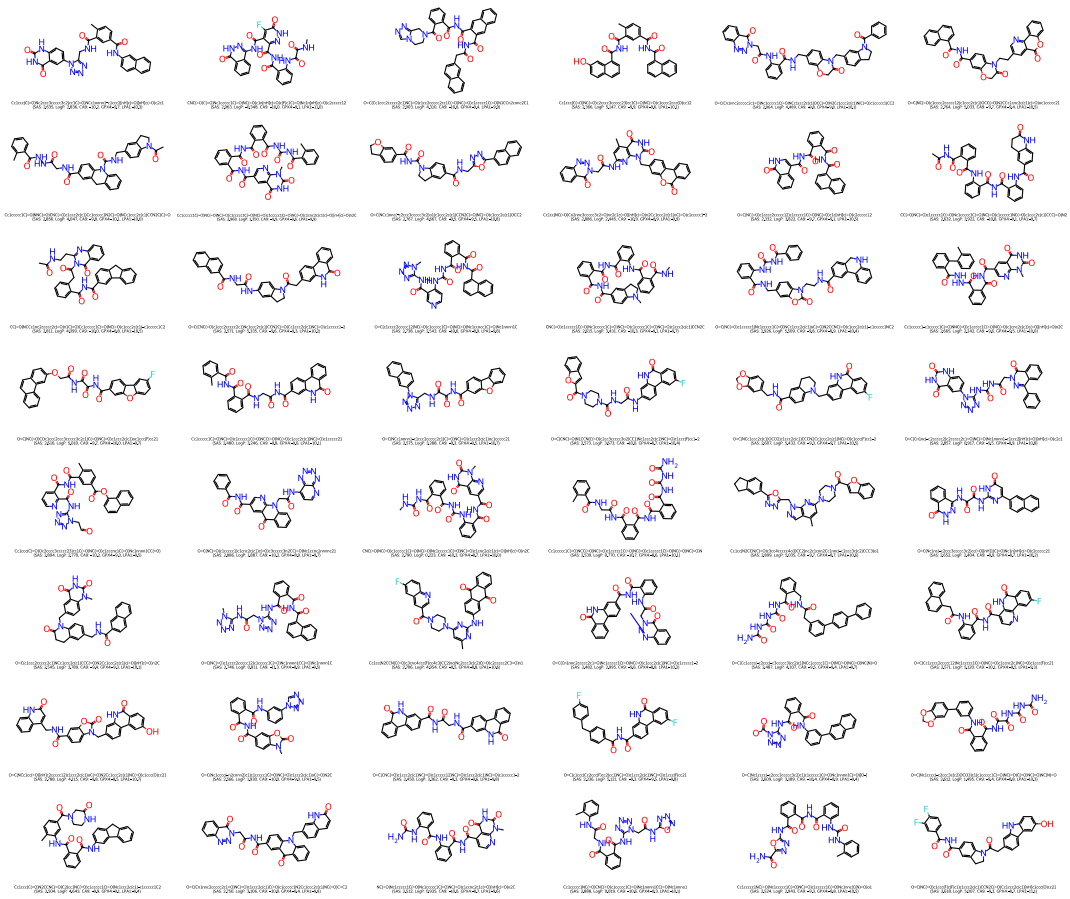

Fig. 20 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives {SAS, logP, BCA9, BGPX4, BLPA1} in combination with **AAE**. Samples are ranked by CA9, GPX4 and LPA1 binding scores respectively, then sorted by the sum of these ranks.

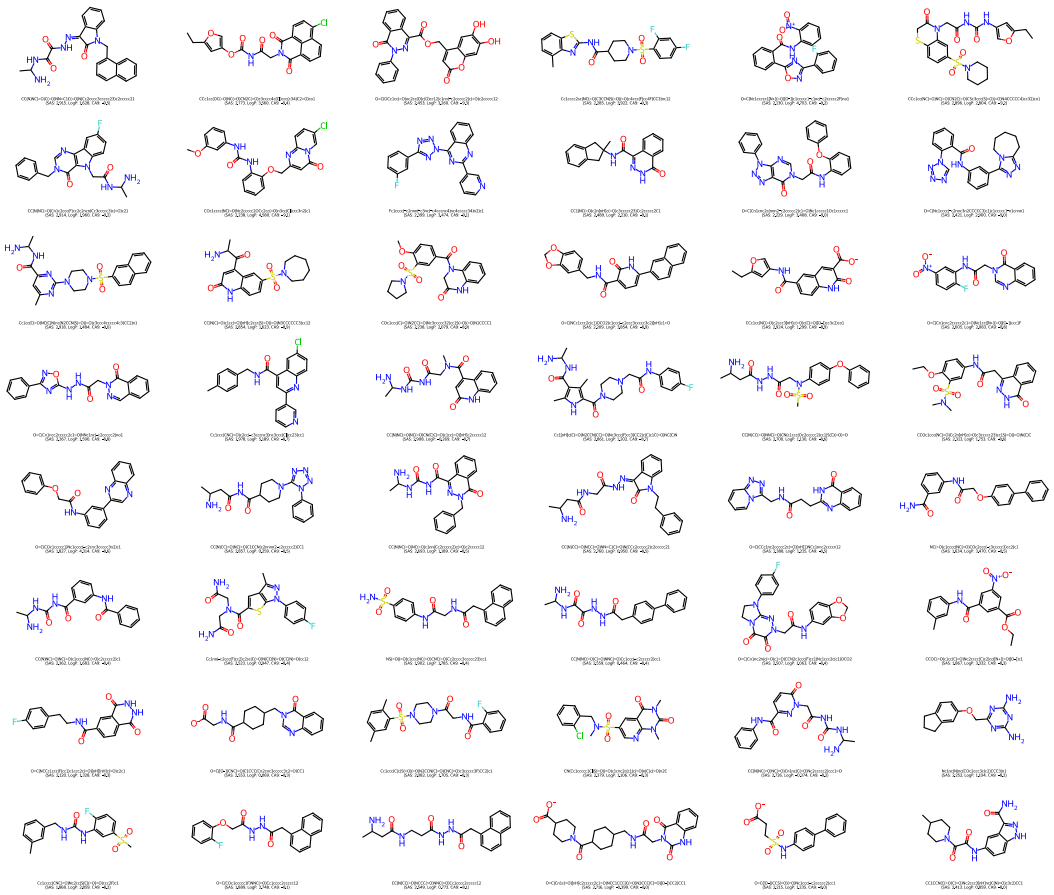

Fig. 21 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BCA9}\}$ in combination with **FragVAE**. Samples are ranked by CA9 binding scores.

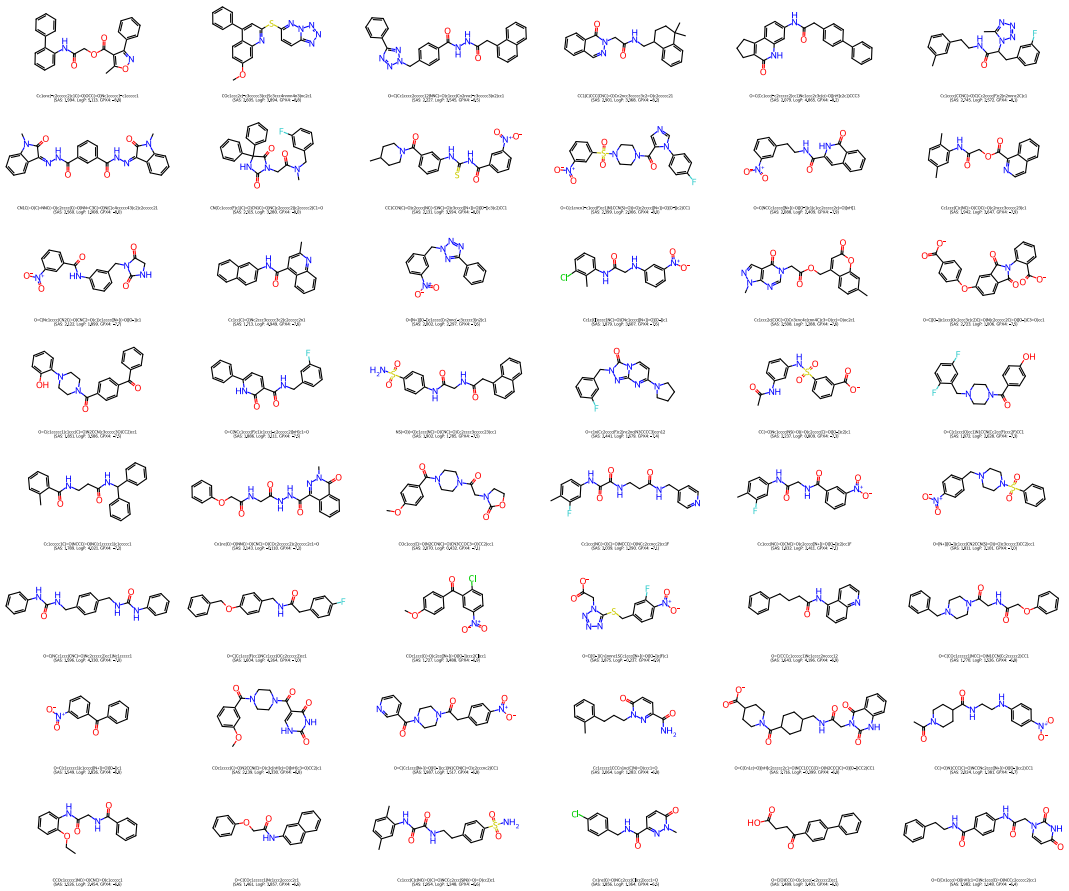

Fig. 22 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BGPX4}\}$ in combination with **FragVAE**. Samples are ranked by GPX4 binding scores.

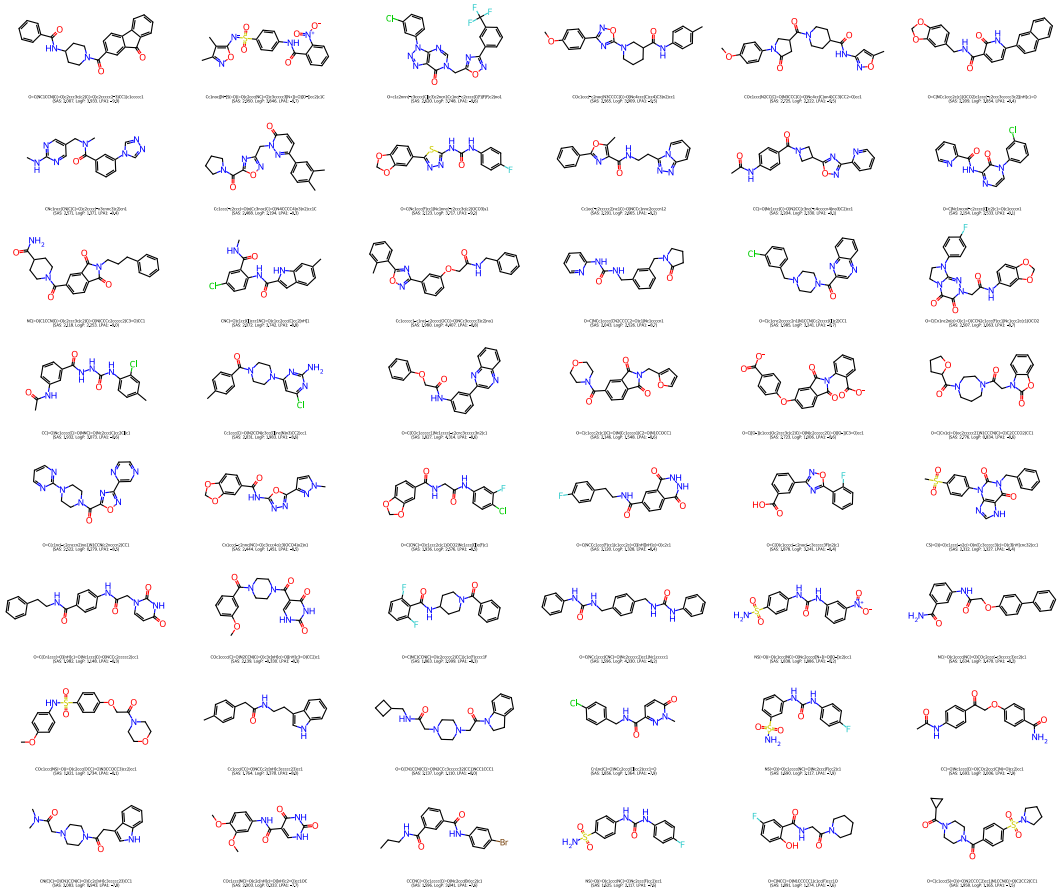

Fig. 23 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BLPA1}\}$ in combination with **FragVAE**. Samples are ranked by LPA1 binding scores.

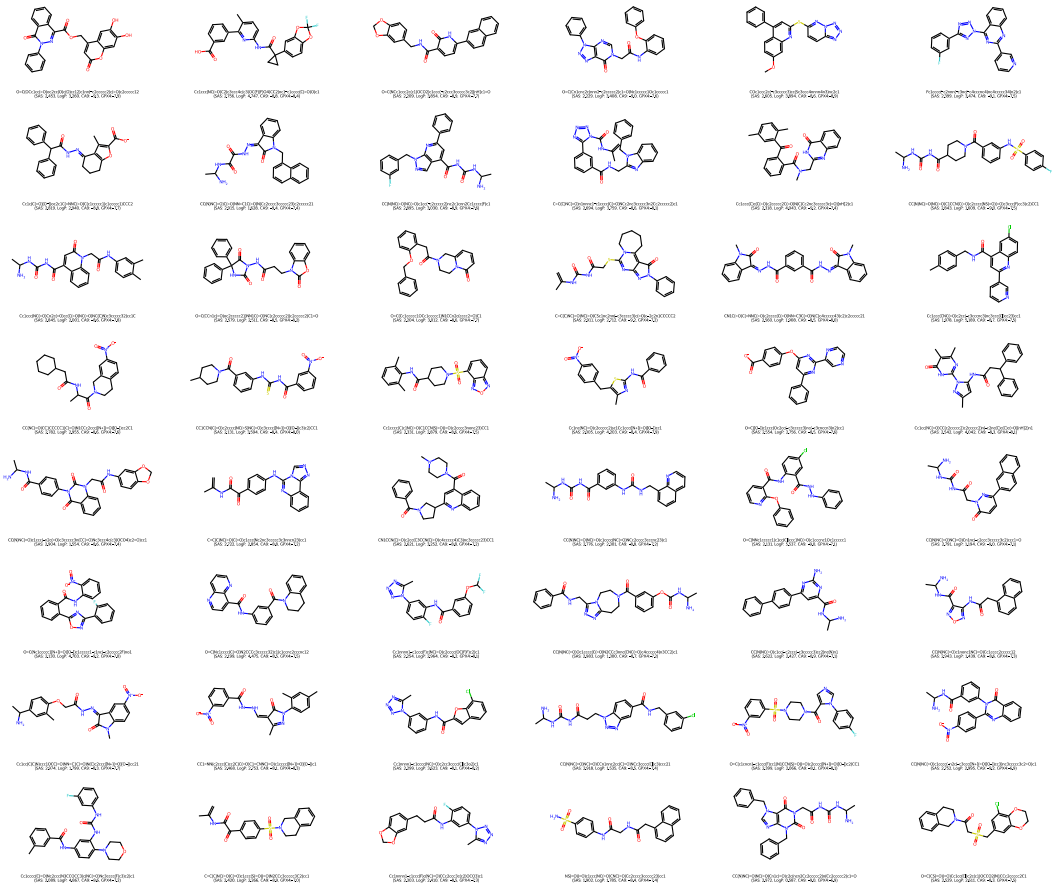

Fig. 24 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BCA9}, \text{BGPX4}\}$ in combination with **FragVAE**. Samples are ranked by CA9 and GPX4 binding scores respectively, then sorted by the sum of these ranks.

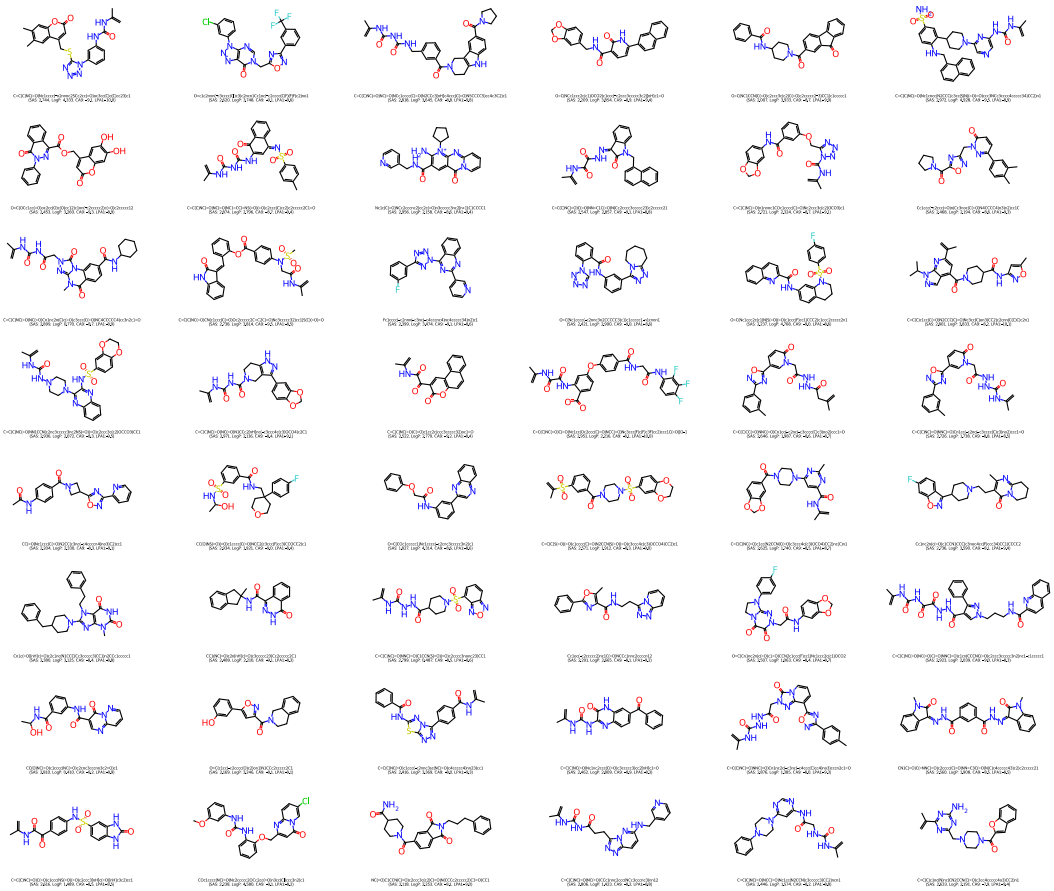

Fig. 25 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BCA9}, \text{BLPA1}\}$ in combination with **FragVAE**. Samples are ranked by CA9 and LPA1 binding scores respectively, then sorted by the sum of these ranks.

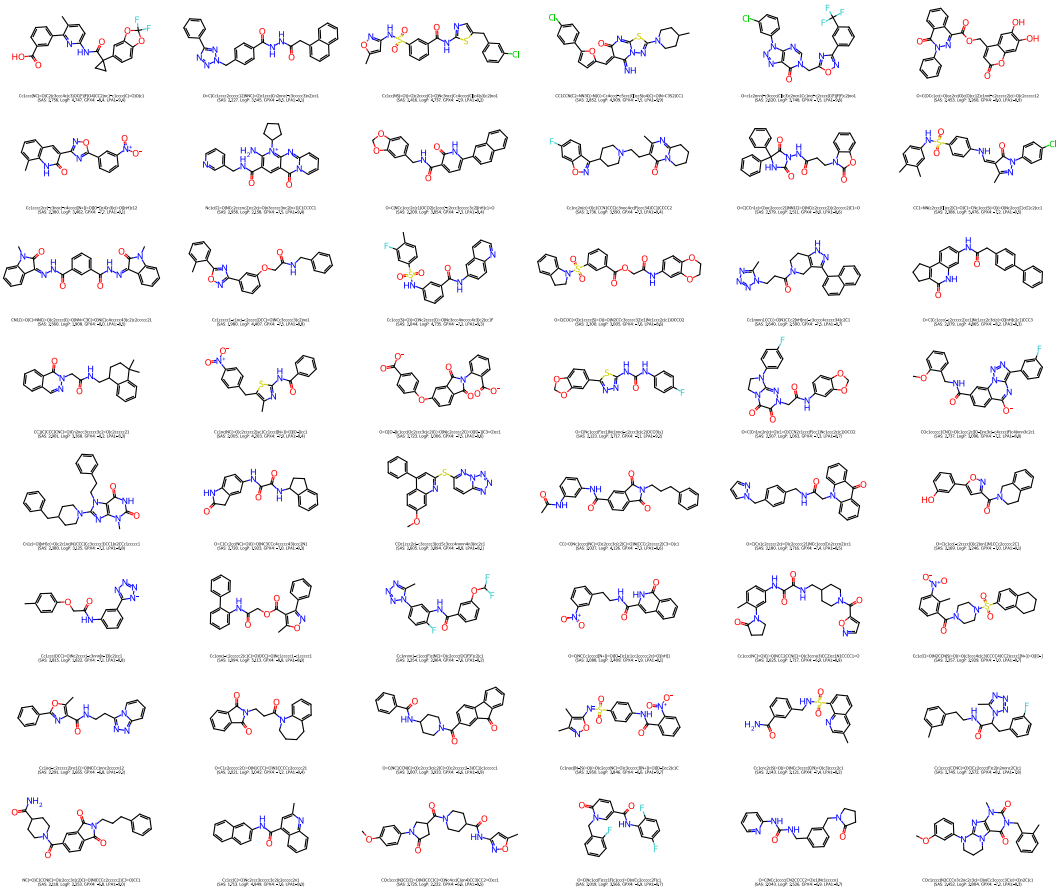

Fig. 26 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BGPX4}, \text{BLPA1}\}$ in combination with **FragVAE**. Samples are ranked by GPX4 and LPA1 binding scores respectively, then sorted by the sum of these ranks.

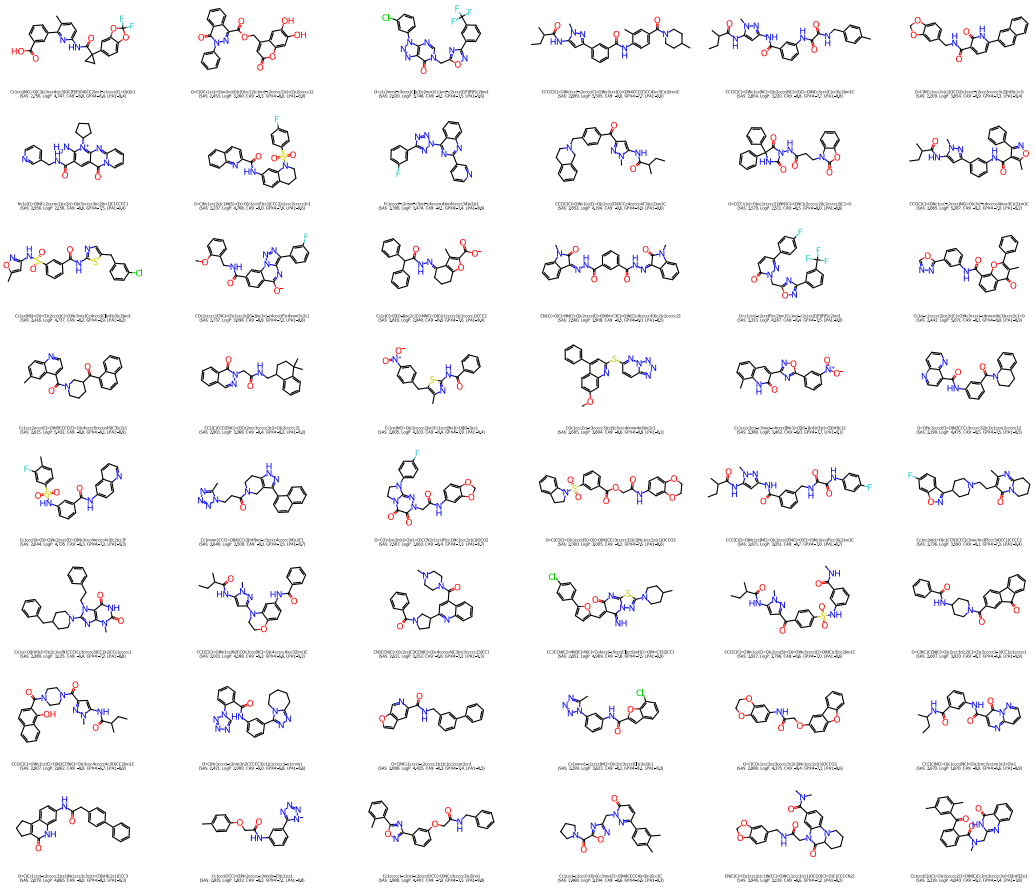

Fig. 27 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives {SAS, logP, BCA9, BGPX4, LPA1} in combination with **FragVAE**. Samples are ranked by CA9, GPX4 and LPA1 binding scores respectively, then sorted by the sum of these ranks.

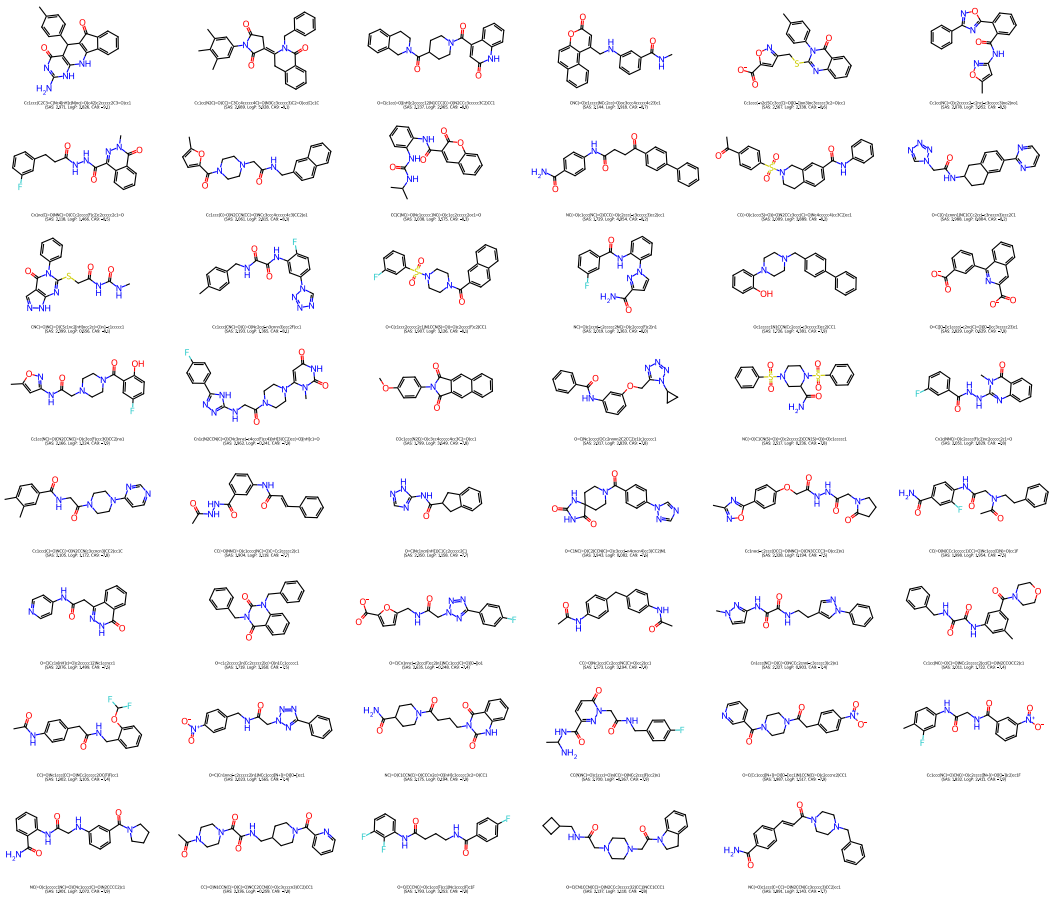

Fig. 28 2D graph visualization of top 47 high-quality novel samples in the final population of DEL with objectives {SAS, logP, BCA9} in combination with **JTVAE**. Samples are ranked by CA9 binding scores.

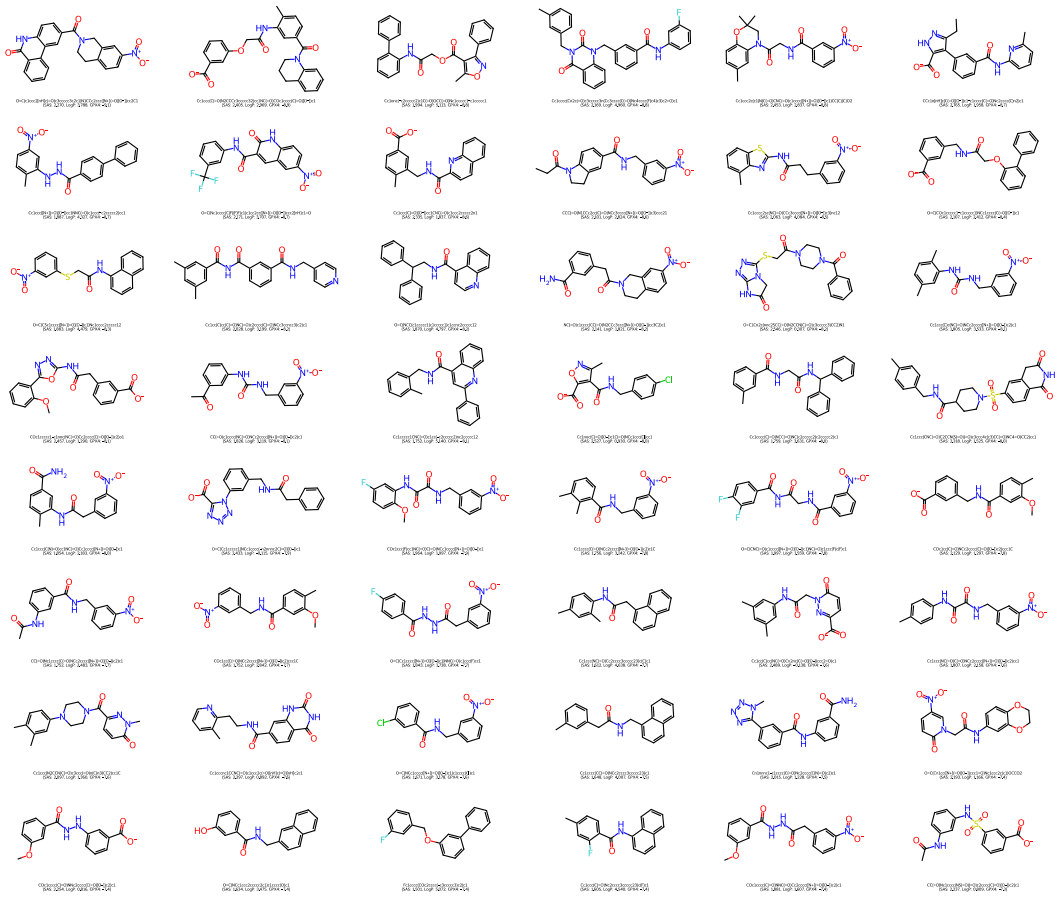

Fig. 29 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BGPX4}\}$ in combination with **JTVAE**. Samples are ranked by GPX4 binding scores.

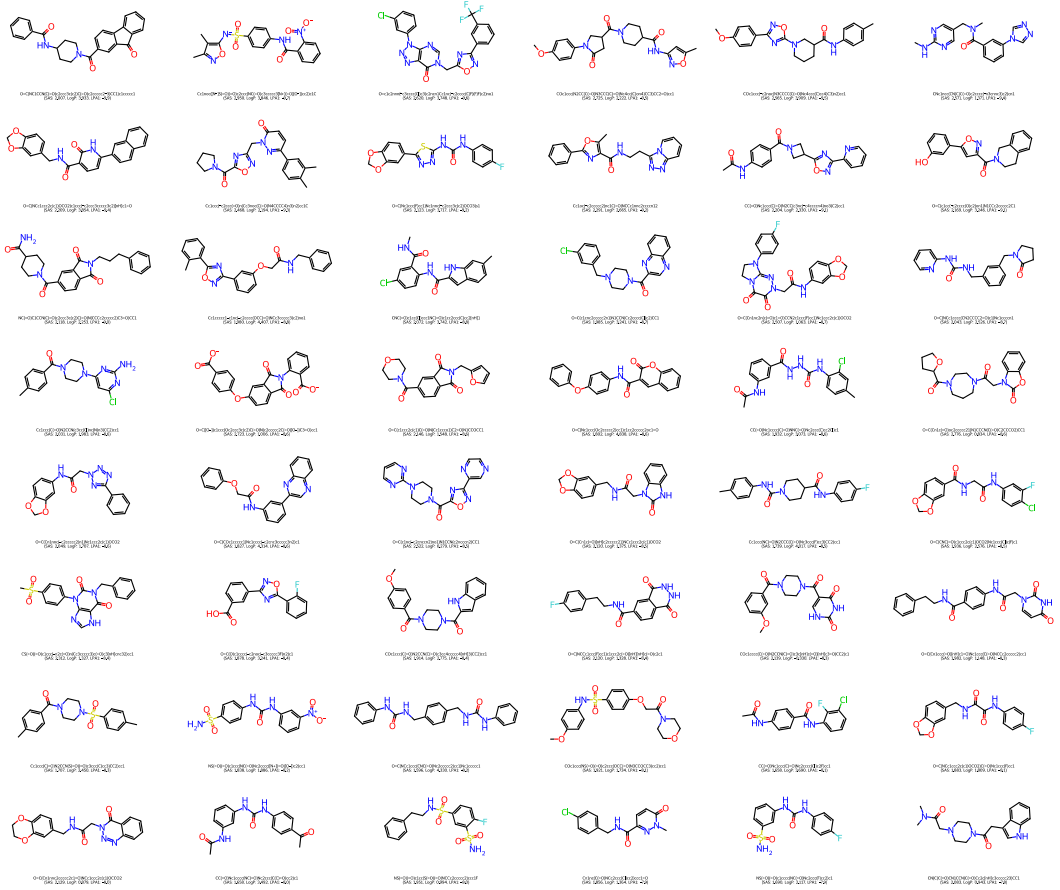

Fig. 30 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BLPA1}\}$ in combination with **JTVAE**. Samples are ranked by LPA1 binding scores.

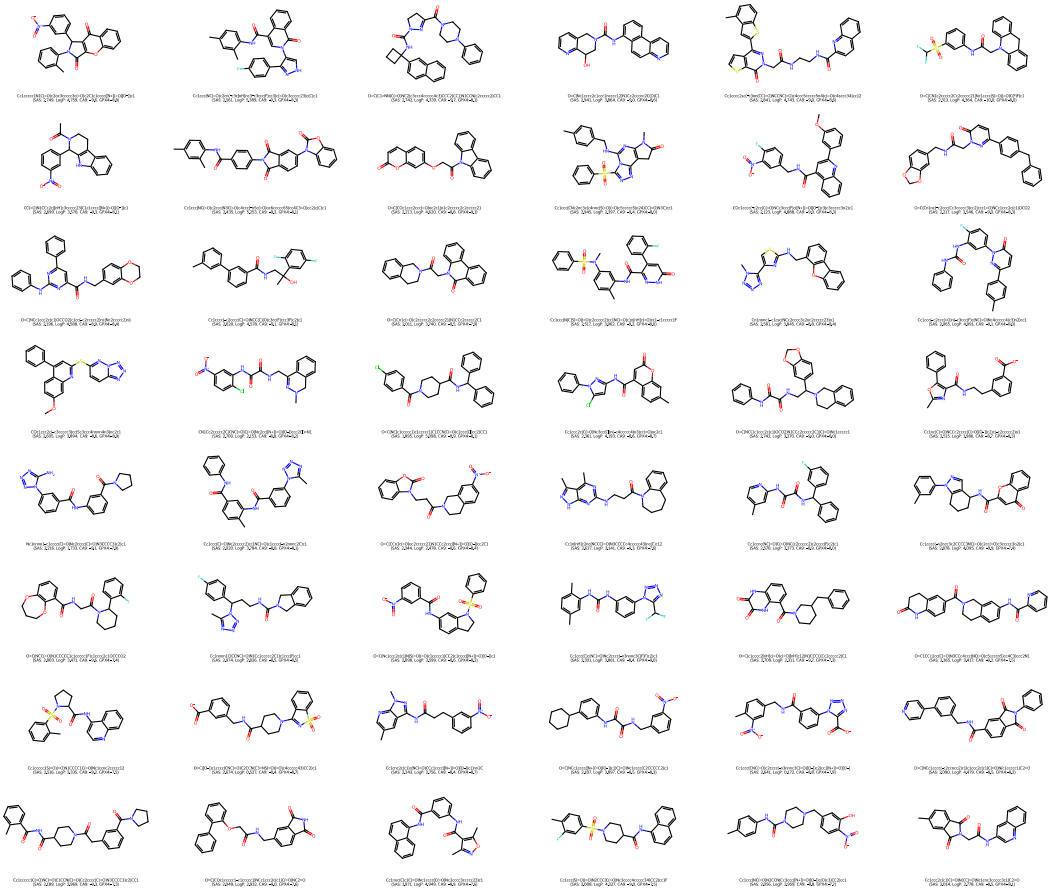

Fig. 31 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BCA9}, \text{BGPX4}\}$ in combination with **JTVAE**. Samples are ranked by CA9 and GPX4 binding scores respectively, then sorted by the sum of these ranks.

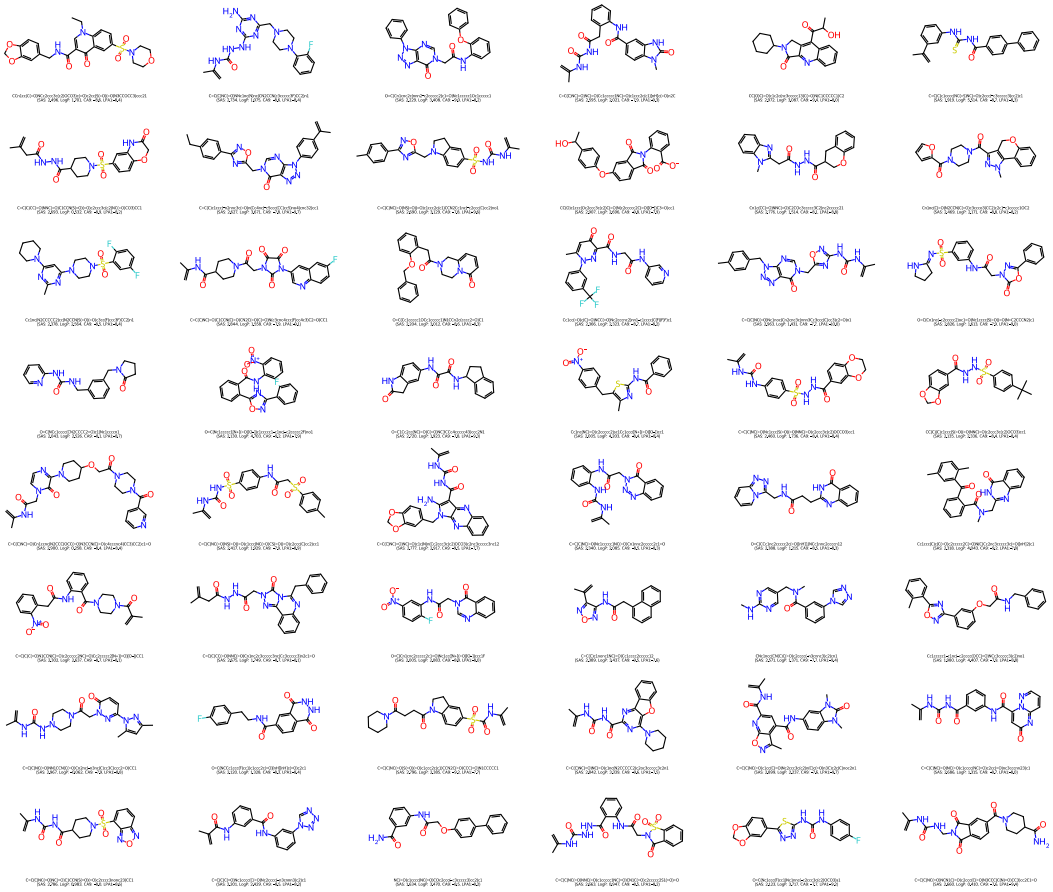

Fig. 32 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\text{SAS}, \text{logP}, \text{BCA9}, \text{BLPA1}\}$ in combination with **JTVAE**. Samples are ranked by CA9 and LPA1 binding scores respectively, then sorted by the sum of these ranks.

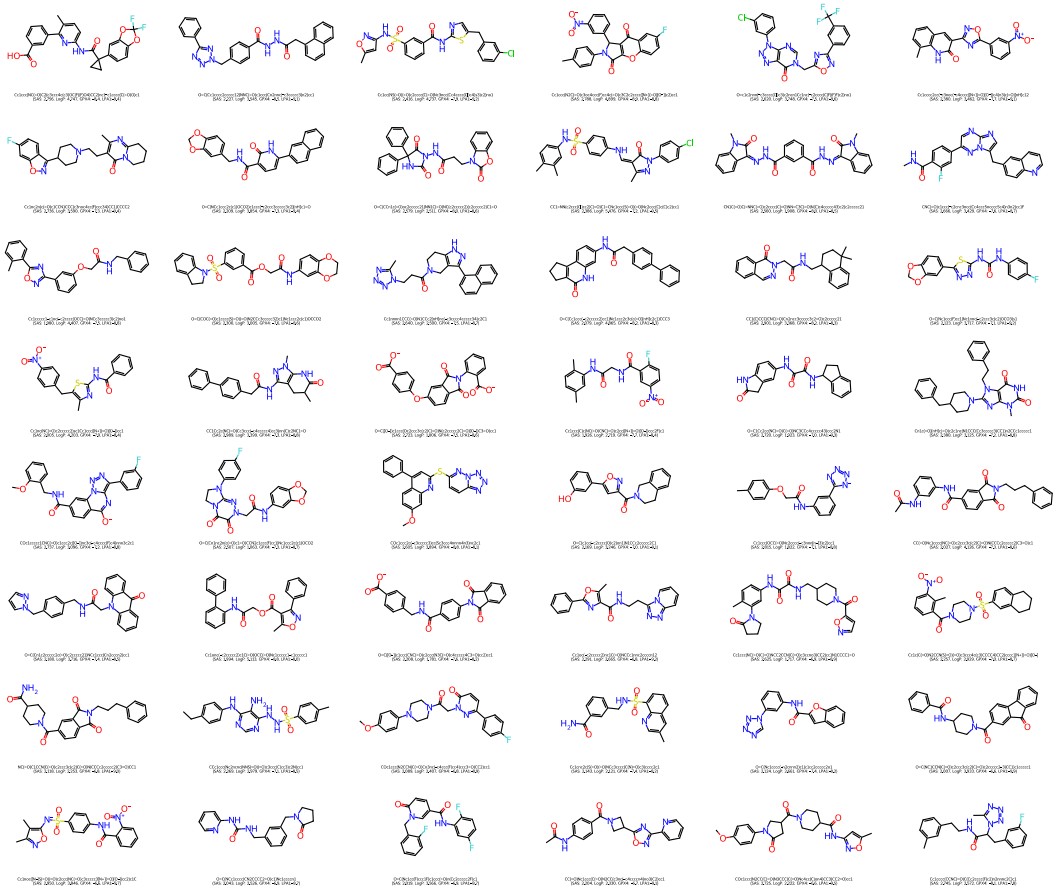

Fig. 33 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives $\{\mathrm{SAS}, \mathrm{logP}, \mathrm{BGPX4}, \mathrm{BLPA1}\}$ in combination with **JTVAE**. Samples are ranked by GPX4 and LPA1 binding scores respectively, then sorted by the sum of these ranks.

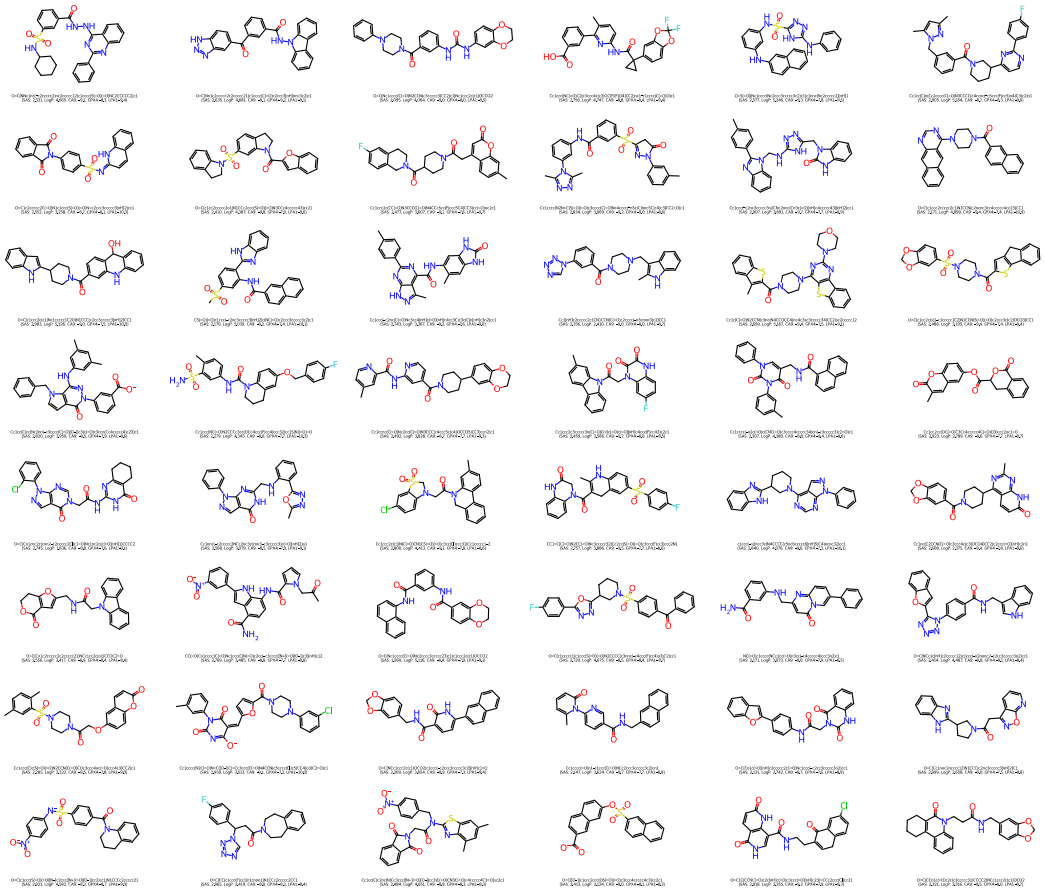

Fig. 34 2D graph visualization of top 48 high-quality novel samples in the final population of DEL with objectives {SAS, logP, BCA9, BGPX4, BLPA1} in combination with **JTVAE**. Samples are ranked by CA9, GPX4 and LPA1 binding scores respectively, then sorted by the sum of these ranks.

References

[1] Wang, X., Cao, L.: Genetic Algorithms - Theory, Applications, and Software Implementations. Xi'an Jiaotong University Press (2002)

[2] Herrera, F., Lozano, M., Sanchez, A.M.: A taxonomy for the crossover operator for real-coded genetic algorithms: An experimental study. International Journal of Intelligent Systems **18**, 309–338 (2003)

[3] Grantham, K., Mukaidaisi, M., Ooi, H.K., *et al.*: Deep evolutionary learning for molecular design. IEEE Computational Intelligence Magazine **17**(2), 14–28 (2022)

[4] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation **6**(2), 182–197 (2002)

[5] Span, P.N., Bussink, J., Manders, P., Beex, L.V., Sweep, C.G.: Carbonic anhydrase-9 expression levels and prognosis in human breast cancer: association with treatment outcome. British Journal of Cancer **89**(2), 271–276 (2003)

[6] Esworthy, R.S., Doan, K., Doroshow, J.H., Chu, F.-F.: Cloning and sequencing of the cDNA encoding a human testis phospholipid hydroperoxide glutathione peroxidase. Gene **144**(2), 317–318 (1994)

[7] Lin, Y.-H., Lin, Y.-C., Chen, C.-C.: Lysophosphatidic acid receptor antagonists and cancer: The current trends, clinical implications, and trials. Cells **10**(7), 1629 (2021)

[8] Kingma, D.P., Welling, M.: Auto-encoding variational Bayes. In: International Conference on Learning Representations (2014)

[9] Gómez-Bombarelli, R., Duvenaud, D., Hernández-Lobato, J., Aguilera-Iparraguirre, J., Hirzel, T., Adams, R., Aspuru-Guzik, A.: Automatic chemical design using a data-driven continuous representation of molecules. ACS Central Science **4**(2), 268–276 (2018) https://doi.org/10.1021/acscentsci.7b00572

[10] Dai, H., Tian, Y., Dai, B., *et al.*: Syntax-directed variational autoencoder for structured data. In: International Conference on Learning Representations (2018)

[11] Sattarov, B., Baskin, I.I., Horvath, D., *et al.*: De novo molecular design by combining deep autoencoder recurrent neural networks with generative topographic mapping. Journal of Chemical Information and Modeling **59**(3), 1182–1196 (2019)

[12] Prykhodko, O., Johansson, S.V., Kotsias, P.-C., Arús-Pous, J., Bjerrum, E.J., Engkvist, O., Chen, H.: A de novo molecular generation method using latent

vector based generative adversarial network. Journal of Cheminformatics **11**(1), 74 (2019) https://doi.org/10.1186/s13321-019-0397-9

[13] Guimaraes, G.L., Sánchez-Lengeling, B., Farias, P.L.C., Aspuru-Guzik, A.: Objective-reinforced generative adversarial networks (ORGAN) for sequence generation models. arXiv:1705.10843 (2017)

[14] Podda, M., Bacciu, D., Micheli, A.: A deep generative model for fragment-based molecule generation. In: International Conference on Artificial Intelligence and Statistics, pp. 2240–2250 (2020)

[15] Degen, J., Wegscheid-Gerlach, C., Zaliani, A., *et al.*: On the art of compiling and using 'drug-like' chemical fragment spaces. ChemMedChem **3**(10), 1503–1507 (2008)

[16] Chung, J., Gülçehre, Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv:1412.3555 (2014)

[17] Li, Y., Tarlow, D., Brockschmidt, M., Zemel, R.: Gated graph sequence neural networks. In: International Conference on Learning Representations (2016)

[18] Simonovsky, M., Komodakis, N.: GraphVAE: Towards generation of small graphs using variational autoencoders. In: International Conference on Artificial Neural Networks, pp. 412–422 (2018)

[19] Samanta, B., De, A., Jana, G., othersl: NeVAE: A deep generative model for molecular graphs. Journal of Machine Learning Research **21**, 1–33 (2020)

[20] Mitton, J., Senn, H.M., Wynne, K., Murray-Smith, R.: A graph VAE and graph transformer approach to generating molecular graphs. arXiv:2104.04345 (2021)

[21] Vignac, C., Frossard, P.: Top-N: Equivariant set and graph generation without exchangeability. arXiv:2110.02096 (2021)

[22] Li, Y., Vinyals, O., Dyer, C., et al.: Learning deep generative models of graphs. arXiv:1803.03324 (2018)

[23] Liu, Q., Allamanis, M., Brockschmidt, M., Gaunt, A.L.: Constrained graph variational autoencoders for molecule design. In: Neural Information Processing Systems, pp. 7806–7815 (2018)

[24] Chen, Z., Min, M.R., Parthasarathy, S., *et al.*: A deep generative model for molecule optimization via one fragment modification. Nature Machine Intelligence **3**(12), 1040–1049 (2021)

[25] Kadurin, A., Nikolenko, S., Khrabrov, K., *et al.*: druGAN: An advanced generative adversarial autoencoder model for de novo generation of new molecules with desired molecular properties in silico. Molecular Pharmaceutics **14**, 3098–3104

(2017)

[26] De Cao, N., Kipf, T.: MolGAN: An implicit generative model for small molecular graphs. arXiv:1805.11973 (2018)

[27] Ma, T., Chen, J., Xiao, C.: Constrained generation of semantically valid graphs via regularizing variational autoencoders. In: Neural Information Processing Systems (2018)

[28] Shi, C., Xu, M., Zhu, Z., et al.: GraphAF: A flow-based autoregressive model for molecular graph generation. arXiv preprint arXiv:2001.09382 (2020)

[29] Zang, C., Wang, F.: MoFlow: An invertible flow model for generating molecular graphs. In: ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 617–626 (2020)

[30] Madhawa, K., Ishiguro, K., Nakago, K., et al.: GraphNVP: An invertible flow model for generating molecular graphs. arXiv:1905.11600 (2019)

[31] Lippe, P., Gavves, E.: Categorical normalizing flows via continuous transformations. arXiv preprint arXiv:2006.09790 (2020)

[32] Luo, Y., Yan, K., Ji, S.: GraphDF: A discrete flow model for molecular graph generation. In: International Conference on Machine Learning, pp. 7192–7203 (2021). PMLR

[33] Pham, T.-H., Xie, L., Zhang, P.: FAME: Fragment-based conditional molecular generation for phenotypic drug discovery. In: SIAM International Conference on Data Mining, pp. 720–728 (2022)

[34] Jin, W., Barzilay, R., Jaakkola, T.: Hierarchical generation of molecular graphs using structural motifs. In: International Conference on Machine Learning, pp. 4839–4848 (2020)

[35] Maziarz, K., Jackson-Flux, H., Cameron, P., Sirockin, F., Schneider, N., Stiefl, N., Segler, M., Brockschmidt, M.: Learning to extend molecular scaffolds with structural motifs. arXiv:2103.03864 (2021)

[36] Jin, W., Barzilay, R., Jaakkola, T.: Junction tree variational autoencoder for molecular graph generation. In: International Conference on Machine Learning, pp. 2323–2332 (2018)

[37] Dai, H., Dai, B., Song, L.: Discriminative embeddings of latent variable models for structured data. In: International Conference on Machine Learning, pp. 2702–2711 (2016). PMLR

[38] Gilmer, J., Schoenholz, S.S., Riley, P.F., *et al.*: Neural message passing for quantum chemistry. In: International Conference on Machine Learning, pp. 1263–1272

(2017)

[39] Kusner, M.J., Paige, B., Hernández-Lobato, J.M.: Grammar variational autoencoder. In: International Conference on Machine Learning, pp. 1945–954 (2017)

[40] Mueller, J., Gifford, D., Jaakkola, T.: Sequence to better sequence: continuous revision of combinatorial structures. In: International Conference on Machine Learning, pp. 2536–2544 (2017). PMLR

[41] Atz, K., Grisoni, F., Schneider, G.: Geometric deep learning on molecular representations. Nature Machine Intelligence **3**, 1023–1032 (2021)

[42] Luo, S., Guan, J., Ma, J., *et al.*: A 3D generative model for structure-based drug design. In: Neural Information Processing Systems, vol. 34, pp. 6229–6239 (2021)

[43] Li, Y., Pei, J., Lai, L.: Structure-based de novo drug design using 3D deep generative models. Chemical Science **12**(41), 13664–13675 (2021)

[44] Powers, A.S., Yu, H.H., Suriana, P., et al.: Fragment-based ligand generation guided by geometric deep learning on protein-ligand structure. bioRxiv, 2022–03 (2022)

[45] Hoogeboom, E., Satorras, V.G., Vignac, C., *et al.*: Equivariant diffusion for molecule generation in 3d. In: International Conference on Machine Learning, pp. 8867–8887 (2022)

[46] Vignac, C., Krawczuk, I., Siraudin, A., et al.: DiGress: Discrete denoising diffusion for graph generation. arXiv:2209.14734 (2022)

[47] Anand, N., Achim, T.: Protein structure and sequence generation with equivariant denoising diffusion probabilistic models. arXiv:2205.15019 (2022)

[48] Schneuing, A., Du, Y., Harris, C., et al.: Structure-based drug design with equivariant diffusion models. arXiv:2210.13695 (2022)

[49] Trippe, B.L., Yim, J., Tischer, D., Broderick, T., Baker, D., Barzilay, R., Jaakkola, T.: Diffusion probabilistic modeling of protein backbones in 3D for the motif-scaffolding problem. arXiv preprint arXiv:2206.04119 (2022)

[50] Satorras, V.G., Hoogeboom, E., Welling, M.: E (n) equivariant graph neural networks. In: International Conference on Machine Learning, pp. 9323–9332 (2021). PMLR

[51] Xu, M., Luo, S., Bengio, Y., Peng, J., Tang, J.: Learning neural generative dynamics for molecular conformation generation. arXiv:2102.10240 (2021)

[52] Jing, B., Corso, G., Chang, J., et al.: Torsional diffusion for molecular conformer generation. arXiv:2206.01729 (2022)

[53] Honda, S., Shi, S., Ueda, H.R.: SMILES Transformer: Pre-trained molecular fingerprint for low data drug discovery. arXiv:1911.04738 (2019)

[54] Yang, L., Yang, G., Bing, Z., *et al.*: Transformer-based generative model accelerating the development of novel BRAF inhibitors. ACS Omega **6**(49), 33864–33873 (2021)

[55] Dollar, O., Joshi, N., Beck, D.A.C., Pfaendtner, J.: Attention-based generative models for de novo molecular design. Chemical Science **12**(24), 8362–8372 (2021) https://doi.org/10.1039/D1SC01050F

[56] Grechishnikova, D.: Transformer neural network for protein-specific de novo drug generation as a machine translation problem. Scientific Reports **11**(1), 1–13 (2021)

[57] Shrivastava, A.D., Kell, D.B.: FragNet, a contrastive learning-based transformer model for clustering, interpreting, visualizing, and navigating chemical space. Molecules **26**(7), 1–28 (2021)

[58] Castro, E., Godavarthi, A., Rubinfien, J., et al.: ReLSO: A Transformer-based model for latent space optimization and generation of proteins. arXiv:2201.09948 (2022)

[59] Atance, S.R., Diez, J.V., Engkvist, O., Olsson, S., Mercado, R.: De novo drug design using reinforcement learning with graph-based deep generative models. Journal of Chemical Information and Modeling **62**(20), 4863–4872 (2022)

[60] Goel, M., Raghunathan, S., Laghuvarapu, S., Priyakumar, U.D.: Molegular: Molecule generation using reinforcement learning with alternating rewards. Journal of Chemical Information and Modeling **61**(12), 5815–5826 (2021)

[61] Jin, W., Barzilay, R., Jaakkola, T.: Multi-objective molecule generation using interpretable substructures. In: International Conference on Machine Learning, pp. 4849–4859 (2020)

[62] Olivecrona, M., Blaschke, T., Engkvist, O., Chen, H.: Molecular de-novo design through deep reinforcement learning. Journal of Cheminformatics **9**(1), 1–14 (2017)

[63] Pereira, T., Abbasi, M., Ribeiro, B., Arrais, J.P.: Diversity oriented deep reinforcement learning for targeted molecule generation. Journal of Cheminformatics **13**(1), 1–17 (2021)

[64] Popova, M., Isayev, O., Tropsha, A.: Deep reinforcement learning for de novo drug design. Science Advances **4**(7), 7885 (2018)

[65] Sanchez-Lengeling, B., Outeiral, C., Guimaraes, G., Aspuru-Guzik, A.: Optimizing distributions over molecular space. an objective-reinforced generative adversarial network for inverse-design chemistry (organic). ChemRxiv (2017) https://doi.org/10.26434/chemrxiv.5309668.v2

[66] Simm, G., Pinsler, R., Hernández-Lobato, J.M.: Reinforcement learning for molecular design guided by quantum mechanics. In: International Conference on Machine Learning, pp. 8959–8969 (2020). PMLR

[67] Ståhl, N., Falkman, G., Karlsson, A., *et al.*: Deep reinforcement learning for multiparameter optimization in de novo drug design. Journal of Chemical Information and Modeling **59**(7), 3166–3176 (2019)

[68] Yang, S., Hwang, D., Lee, S., Ryu, S., Hwang, S.J.: Hit and lead discovery with explorative RL and fragment-based molecule generation. Neural Information Processing Systems **34**, 7924–7936 (2021)

[69] Zhou, Z., Kearnes, S., Li, L., Zare, R.N., Riley, P.: Optimization of molecules via deep reinforcement learning. Scientific Reports **9**(1), 1–10 (2019)

[70] Polishchuk, P.: CReM: chemically reasonable mutations framework for structure generation. Journal of Cheminformatics **12**(1) (2020) https://doi.org/10.1186/s13321-020-00431-w

[71] Kapturowski, S., Campos, V., Jiang, R., Rakićević, N., Hasselt, H., Blundell, C., Badia, A.P.: Human-level atari 200x faster. arXiv:2209.07550 (2022)

[72] Sokar, G., Mocanu, E., Mocanu, D.C., Pechenizkiy, M., Stone, P.: Dynamic sparse training for deep reinforcement learning. arXiv:2106.04217 (2021)