

Systemic Evolutionary Chemical Space Exploration For Drug Discovery

Preprint, compiled December 5, 2021

Chong Lu ^{1*}, Shien Liu ^{1*}, Weihua Shi¹, Jun Yu¹, Zhou Zhou¹, Xiaoxiao Zhang¹, Xiaoli Lu¹, Faji Cai¹, Ning Xia², and Yikai Wang¹

¹Keen Therapeutics Co., Ltd.

²Chemical.AI

*Contribute equally.

ABSTRACT

Chemical space exploration is a major task of the hit-finding process during the pursuit of novel chemical entities. Compared with other screening technologies, computational *de novo* design has become a popular approach to overcome the limitation of current chemical libraries. Here, we reported a *de novo* design platform named systemic evolutionary chemical space explorer (SECSE). The platform was conceptually inspired by fragment-based drug design, that miniaturized a “lego-building” process within the pocket of a certain target. The key of virtual hits generation was then turned into a computational search problem. To enhance search and optimization, human intelligence and deep learning were integrated. Application of SECSE against PHGDH, proved its potential in finding novel and diverse small molecules that are attractive starting points for further validation. This platform is open-sourced and the code is available at <http://github.com/KeenThera/SECSE>.

Keywords Chemical space exploration · Fragment-based drug discovery · Deep learning · *De novo* drug design · PHGDH

1 INTRODUCTION

Developing a new drug is an enduring process that is estimated to take 10-15 years with a cost of 1.5 billion US dollars or more. At the early drug discovery stage, the hit-finding program is crucial for a successful R&D campaign, especially for the challenging targets, which usually yield meager hit rates. There are many options for hit-finding, such as high-throughput screening (HTS), affinity selection-mass spectrometry (AS-MS), fragment-based drug design (FBDD), DNA encoded library techniques (DELTA), and virtual screening (VS). However, all the above approaches suffer from the requirement of a predefined (real or virtual) compound library. To address the limitation, make-on-demand libraries [1–3] have gained some recent popularity in expanding the chemical space. Nevertheless, even the most extensive collection of compounds claimed so far with the size of 10^{26} [4, 5] is still a very tiny fraction of the estimated chemical space in the order of 10^{63} [6]. Therefore, a systemic chemical space searching strategy is needed to provide optimal starting points against the target of interest.

De novo design is one such strategy that is conceptually able to overcome the limitation of existing compound libraries, which produces novel compounds based on the 3D crystal structure of a given target from scratch. A comprehensive summary [7–11] of the recent development in *de novo* design is out of the scope of this paper though, several seminal works that inspire us will be briefly reviewed in the following section.

LUDI [12] was an example of early attempts, where fragments from a predefined library were positioned into sub-pockets of the target. Then the fitted fragments were bridged together to form a new compound that better occupied the pocket. A

similar approach called LigBuilder [13] used module POCKET to analyze and parameterize protein pockets and then applied module GROW or LINK to build up new molecules. A genetic algorithm was implemented in the growing and linking steps to avoid the combinatorial explosion of the molecular generating process. Subsequently, module SCORE predicted the binding affinity of the molecules. Synthesis accessibility analysis and more druglike filters were incorporated in the upgraded program LigBuilder v2 [14]. While in the latest version LigBuilder v3 [15], the authors began to consider the flexibility of pockets by including several samples from a particular target or different targets with similar binding pockets in the generation workflow. OpenGrowth [16] was an open-sourced *de novo* design program which also based on the fragment-based growing strategy. The 3-mers screening method required that generated molecules be made by defined fragments derived from the drug library, which warranted druglike properties. Like LigBuilder v3, different conformations of the target were considered to address the protein flexibility issue. Durrant *et al.* developed AutoGrow [17] to integrate fragment-based growing and docking with an evolutionary algorithm. The latest version is AutoGrow4 [18], which employed reaction-based rules for growing as mutation operators in the genetic algorithm and merging two molecules with maximum common substructure as crossover operators. Substructure or property filters (like the rule of 5, PAINS [19]) were used to control the quality of generated molecules. At the same time, open-source docking programs were invoked to evaluate the binding affinity. Although AutoGrow4 performed well in some cases, reaction-based molecular generation is intrinsically limited for constructing novel chemical entities. Polishchuk published an open-sourced tool called CREM [20] to produce highly diverse structures by fragment manipulation (mutate, grow and link). Nigam *et al.* proposed

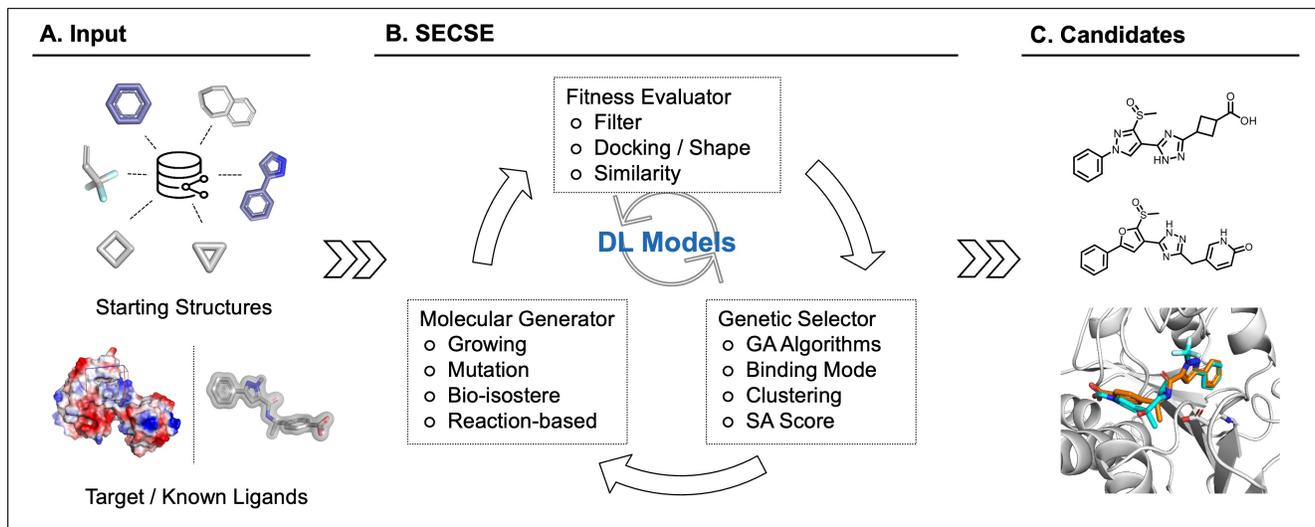


Figure 1. The general workflow of SECSE. A, Fragment library or preferred structures can be used as starting point for molecule evolution. Either binding pocket of 3D protein structures (structure-based) or a set of known active ligands (ligand-based) can be used for fitness evaluation. B, SECSE has three basic modules, molecular generator, fitness evaluator, and genetic selector. C, Examples of generated structures and binding poses can be analyzed for virtual candidate prioritization. Protein structure is shown in white cartoon. A selected candidate is shown in cyan stick, while reference compound is shown in orange stick.

STONED for efficient search of chemical space using a SELFIES modification method [21]. Recently, Steinmann and Jensen reported a non-fragment-based approach [22], which used a set of reaction-like rules to build up chemical structures, yielding molecules with acceptable glide docking scores and synthetic feasibility by genetic algorithm.

In addition to rule-based generators, deep generative models have also been extensively explored. MolAICal [23] used generative deep learning models for 2D structure construction and classical methods for 3D evaluation and simulation. Recently, Ma *et al.* [24] developed SBMolGen, which contained an RNN based SMILES generator called ChemTS, a Monte Carlo tree search, and docking simulations. Lai *et al.*, the authors of LigBuilder, developed DeepLigBuilder [25] to generate 3D molecules directly from deep generative models. Several other new approaches utilizing deep learning methods to generate 3D molecules have been reported [26–29]. Compared with 1D/2D generative models or rule-based methods [30, 31], the competitive advantage of these 3D models is speed. However, it is not easy to directly converge when training deep learning models end to end. Researchers have to introduce some special treatments for the data type and model architecture to terminate the training process, which is usually difficult to interpret.

Inspired by previous attempts in the field, we present our work setting up a platform to explore the chemical space against a given target systemically. Analogous to other programs, the SECSE platform consists of a molecular generator, a fitness evaluator, and a genetic selector. In the molecular generator module, we have created more than 3,000 rules for molecular transformations based on knowledge and expertise from the literature domain and our internal medicinal chemists. These rules are comprehensively curated and strategically categorized for optimal output. In the fitness evaluator module, molecular docking is utilized for compound assessment, which can also be replaced by shape-/pharmacophore-based evaluation methods.

In the genetic selector module, genetic algorithm is used given the similarity between the triage strategy of fragment growing and the genetic rule “fitness to survive”.

The workflow of SECSE is described in Fig. 1. In the first place, fragments/groups are docked/positioned into the pocket, from which the ones with high docking scores or ligand efficiency are picked as elites. It is noteworthy to point out that fragments with less than 13 heavy atoms are exhaustively enumerated as initial input, yet any given structures or functional groups can be used as starting points. Then all the elites are applied the rules to generate new molecules. The daughter molecules are clustered and sampled to represent the pool. The sampled molecules are docked into the pocket again. Highly scored molecules adopting hereditary or reasonable 3D orientation are chosen as new elites. This process concludes one cycle. After multiple cycles of iteration, a considerable number of compounds are generated and accumulated. To comprehensively evaluate all compounds, we introduced a graph-based machine learning module to speed up elite selection in each generation. Finally, hit compounds are visually inspected and selected before wet lab synthesis.

PHGDH is chosen to demonstrate the potential of the SECSE platform. Virtually generated molecules are shown, and the corresponding structure-activity relationship is analyzed for this target. Their high docking scores and reasonable binding poses, in addition to structural novelty and patentability, warrant further exploration.

2 METHODS

2.1 Fragments Collection

As starting points of the entire workflow, the quality of fragments collection would determine the final output to some extent. Although fragments from co-crystal structures or based on

hypothesis can be used as proprietary input whenever available, it would inevitably be limited by the real fragment collection around the size of 10^3 or human bias. To ensure the diversity of the starting fragment library, we proposed an algorithm that can potentially enumerate all possible molecules containing up to 12 heavy atoms with MW ranging from 50 to 210. To the best of our knowledge, it is the first attempt to systemically explore such fragment chemical space.

As described in Fig. S1, sequential carbon strings, such as “CC-CCCC”, are the starting point of fragment generation with fixed heavy atom numbers. The SMILES string is then modified to construct aliphatic rings, which are subsequently submitted for structure transformations (aromatic ring formation, sidechain rearrangement, and atom/bond replacement, etc.). A series of filters (the same filters rules in SECSE) are applied to remove fragments with undesired architecture/topology or functional groups. Final structures of 121,860,917 fragments are stored in an SQLite database.

2.2 Input Preparation

In the workflow, chemical structures and protein structures are the primary inputs. Depending on different purposes, the chemical input can be an atom, a fragment structure, or a fragment library in the format of a tab-split file containing structure SMILES and ID. If needed, the provided SMILES can be converted into a 3D structure using ETKDG v2 built in RDKit. Tautomer and spiro centers are also enumerated on demand. For AutoDock Vina docking, the ligands are converted from SDF format to PDBQT format using Open Babel v3.1.1. Fragment libraries are recommended for hypothesis-driven hit discovery, especially when limited binders against the target of interest are reported. Protein 3D structures are prepared from crystal structures from the Protein Data Bank (PDB). Homology models or predicted structures from AlaphFold2/RoseTTAFold are also acceptable although with compromised accuracy and predicting power. In our demo case, protein structures are prepared for docking with ADFR v1.2 [32, 33].

2.3 Molecular Generator

The molecular generator we have developed provides a rule-based generation approach. There are four types of transformation rules (growing, mutation, bioisostere, and reaction) in our database. Some representative cases of each class are shown in Fig. 2.

- 1) In the grow rule, any of the replaceable hydrogen atoms on the seed compound can be replaced with a new substructure, such as an atom, a functional group, a ring, or a ring with a linking spacer.
- 2) The mutation rule includes the following three categories: atom replacing, insertion, and deletion; ring-closing, ring-open, ring modification (expansion, reduction, contraction); as well as aromatic-aliphatic exchange.
- 3) The bioisostere rule refers to classical or non-classical bioisosteric replacements, which are commonly used by medicinal chemists. Scaffold hopping is currently not included.

- 4) The reaction-based rule contains common organic reactions confined to one or two steps. A library of commercial building blocks is used as starting materials. Applying the chemical reaction rule is beneficial to efficiently increase the scaffold diversity of the resulting molecules, although they can be generated from multiple rounds of rules from the previous three categories.

All the rules are represented in the reaction-like format using the SMARTS definition in RDKit. A few examples from each category are provided in the SQLite database.

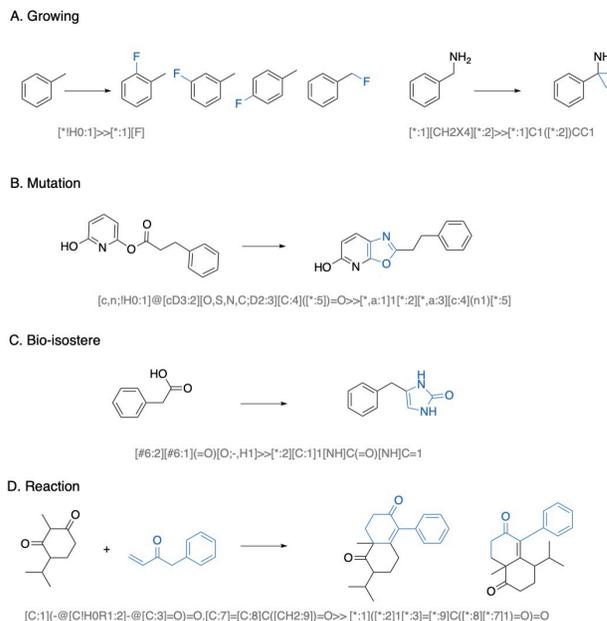


Figure 2. Categories of rules in SECSE and illustrative examples.

A. The Growing rule means applicable hydrogen can be replaced by a defined functional group. **B.** The Mutation rule contains a large set of structural transformations commonly practiced by medicinal chemists such as ring-closing-ring-opening, insertion or deletion of atoms, and etc. **C.** The Bio-isostere rule allows the interconversion of isosteric groups/atoms. **D.** The Reaction rule identifies functional groups/moieties that can react with building blocks (BB) from a pre-defined BB library and hypothetically generates all possible products. Changed atoms are highlighted in dark blue color.

2.4 Property & Structure Filter

To ensure that the platform generates molecules with decent chemical beauty [34], we construct several filters that define molecular properties, ring system count, and substructures.

- 1) The default parameters of the molecular property filters are shown as follows: molecular weight (MW) $\in [81, 450]$; LogP $\in [0, 5]$; the number of hydrogen bond donors (HBD) ≤ 5 ; the number of hydrogen bond acceptor (HBA) ≤ 10 ; the number of rotatable bonds (RB) ≤ 4 ; and topological polar surface area (TPSA) ≤ 200 . All properties here are calculated by RDKit v2021.03.5 [35]. The definition of RB was rephrased as '[C^3!D1;!\$(C(F)(F)F)]-!@[!Br!F!Cl!I!H3&!\$(***)!D1;!\$([!Br!F!Cl!I](F)(F)F)]'
- 2) The default constraints for ring systems are: total ring system count ≤ 4 ; the max ring members of one ring system

≤ 3 ; the max size of a single ring ≤ 7 ; the max count of fused rings ≤ 3 ; the max count of bridged rings ≤ 1 ; and the max count of spiro ring ≤ 1 .

- 3) Undesirable structures are also filtered by identity filters (sulfur, phosphorus, or structure alert), and count filters (e.g., max number of carboxylic acid or alkyne in one compound). PAINS [19] filters are also included.

One thing worthy of note is that the filters are arbitrarily set depending on project requirements, which can be adjusted if the output is not ideal.

2.5 Fitness Evaluator

Structure-based virtual screening engines such as molecular docking or pharmacophore-based screening methods are optional for fitness evaluation. Docking is the first choice for fitness evaluation. The default docking software in our platform is AutoDock Vina v1.2.0 [36, 37]. We also provide a Glide interface for users with commercial licenses. Additionally, we offer shape-based screening and similarity scoring functions to evaluate fitness for ligand-based drug design (i.e., the initial input is not a protein structure but one or more ligands with known activity).

Several scoring functions are optimized to achieve the evaluator function for different scenarios. If the docking mode is selected, both docking score and \ln ligand efficiency [38] are considered as ranking criteria, where

$$\ln LE = \frac{\text{Docking score}}{1 + \ln(\text{Num of heavy atoms})}$$

The docking score tends to favor larger molecules in our previous tests. In contrast, \ln ligand can efficiency correct the issue by preventing premature enrichment of large molecules before reaching the upper molecular weight cutoff. Root Mean Square Deviation (RMSD) of aligned atoms between docking poses of the previous and current generation is calculated to determine whether the binding mode has changed in the two consecutive generations. If the similarity search mode is selected, the optional scoring functions will be a Tanimoto index of different molecular fingerprints from the generated molecules and reference compounds. In addition, the retrosynthesis module from Chemical.AI [39] is invoked to assess the synthetic availability.

2.6 Seed Selector

After scoring, molecules with RMSD less than 2 Å or with significantly decreased docking scores are selected as seeds for the next generation. The purpose of the selector is to make sure compounds with consistent binding modes are maintained while compounds with much better binding modes won't be carved out. Then we apply a genetic algorithm [40–42] to select seeds from all eligible molecules. In our platform, the default GA operator is the tournament selection which is the most widely used selection strategy. Consequently, it can quickly converge to the optimal solution within noisy environments and introduce some randomness to avoid the limitations caused by local optimization.

Because of the limited computing resources, we sample data from the molecules generated by all the rules. Likewise, we use

a partition clustering algorithm (see details in Fig. S2) before sampling to ensure the diversity of the selected molecules. We calculate the molecular fingerprint and Tanimoto index to evaluate the distance/dissimilarity between generated molecules, based on which the sampling is executed.

2.7 Deep Learning-based Fitness Prediction

Although SECSE can generate a significant number of molecules, most of them are not evaluated due to limitations in computing and storage resources. Therefore, we apply deep learning (DL) modeling to reduce computational costs and make it possible to evaluate the fitness of all molecules. We use the data generated after each generation to train the model and then predicts the fitness of unsampled molecules. Docking score or ligand efficiency can be considered as target for prediction if the docking mode was selected. Fitness prediction models are constructed using package Chemprop v1.3.1. Chemprop builds a directed message passing neural network and learns to predict molecular properties directly from the graph representation of molecules. [43] Two strategies are provided here for the integration of DL technology. One is the combined mode, where top ranked molecules prioritized by predicted scores were evaluated by the fitness module. These molecules were applied for seed selection together with docked molecules from sampling procedure. Moreover, in the combined mode deep learning models will be updated with each round of molecular generation. The other one is called clean mode. The DL model is trained based on the docking results after a SECSE campaign is finished. Data from each generation can be trained independently or together. The model can then be applied on undocked molecules for fitness prediction. Molecules with good performance from DL models can be subjected for further inspection. Additionally, these two modes can be used alone or in combination.

The platform uses some open-source packages: RDKit v2021.03.5, OpenBabel v3.1.1, AutoDock Vina v1.2.0, Chemprop v1.3.1, and GNU parallel v20190922 [44].

3 RESULTS

3.1 Properties of Generated Molecules

We constructed a random library using SECSE without any other evaluation constraint to estimate the molecular properties of generated compounds. Benzene was assigned as the only input fragment. During each iteration, one hundred molecules that passed the filter were randomly selected as seeds for the next iteration. The final random library after ten rounds of iteration contains 2,042,863 molecules, which are included in the GitHub repository.

We calculated the physicochemical properties of the random library, such as molecular weight (MW), LogP, and the fraction of *sp*³ hybridized carbons (Fsp3), as well-illustrated in Fig. 3. Despite the upper limitation of MW in the filters, we could find that the peak falls around 450 Da. The distribution of LogP showed that the majority of molecules have a value between 0 and 5. Molecules with a high Fsp3 tended to be more three-dimensional in shape. The Fsp3 of the random library was well-distributed from 0 to 0.8. In addition, five thousand molecules

were randomly sampled to plot the principal moments of inertia (PMI), a more direct descriptor for assessing the distribution of molecular geometry (rod-shaped, disc-shaped, and sphere-shaped). We used the MMFF94 force field in RDKit to optimize the conformers of sampled molecules. To our surprise, as presented in Fig. 3 D, the molecules scattered more towards the top-right vertex (sphere-shaped) in comparison with traditional HTS compounds that predominately dropped between the top-left vertex (rod-shaped) and bottom vertex (disc-shaped) (data not shown). The results presented here indicate that SECSE can generate molecules with suitable druglike properties and diverse geometry.

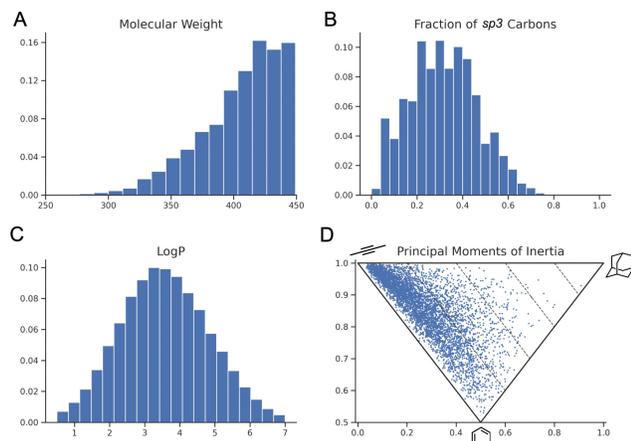


Figure 3. Properties of the randomly generated library. A-C, shows the distribution of MW, Fsp3, and LogP of molecules in the random library, respectively. D, The PMI plot illustrates the shape of sampled molecules from the randomly generated library. The top-left vertex represents rod-shaped, the top-right vertex represents sphere-shaped, and the bottom vertex represents disc-shaped.

3.2 Case: 3-Phosphoglycerate Dehydrogenase (PHGDH)

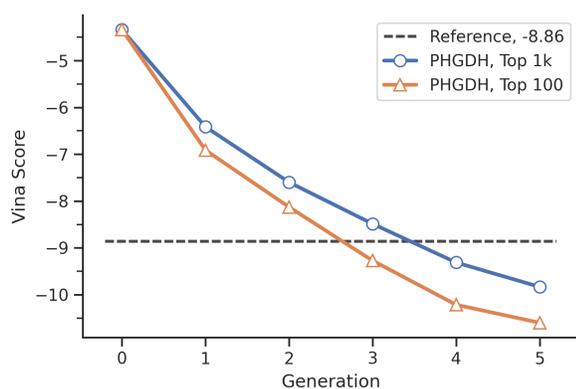


Figure 4. Average docking scores of top N compounds in each generation. The unit of vina score is kcal/mol. The compound in generation 0 is the initial fragment. The vina score of reference compound is -8.86 kcal/mol. The blue line represents the average vina score of top 1000 compounds. The orange line represents the average vina score of top 100 compounds.

PHGDH is a crucial enzyme that catalyzes the first committed step of the *de novo* serine synthesis pathway. It con-

verts 3-phosphoglycerate to 3-phosphopyruvate in a reduced nicotinamide adenine dinucleotide (NADH)/nicotinamide adenine dinucleotide (NAD⁺)-dependent oxidation reaction. Many reports [45, 46] have indicated that overexpression of PHGDH is associated with short-term survival and aggressive disease that are common in many patients. Inhibition of PHGDH may be a promising strategy for cancer therapy [47–50].

AutoDock Vina was selected for molecular docking in this case. The upper limitation of molecular weight was set to 500 Da, and the starting fragment was benzene. The crystal structure of PHGDH (PDB code:6RJ3) was prepared using previous descriptions in **Methods**. Together, 502,226 poses were collected after five generations. The docking scores were gradually decreased (Fig. 4). Not surprisingly, compared with the average docking score of the top 1,000 of each generation, that of the top 100 molecules was improved more rapidly. After three generations, the average docking scores of either the top 100 or top 1,000 compounds were better than that of the reference compound (-8.8 kcal/mol). Additionally, it was observed that the scores started to converge at later generations indicating the pocket occupancy was quickly approaching its optimum. It is plausible that the converging rate for different targets might be different. More rounds of iteration can be performed. Yet in this case, we stopped here for further analysis.

Finally, 14,413 poses with AutoDock Vina score less than -9 kcal/mol were obtained. Then, similarity distance cutoff was set to 0.15 to cluster these molecules according to the RDKit fingerprint. The one with the lowest docking score of each cluster was chosen for further binding pose inspection. Afterwards, we retrieved analogs of molecules of interest from the original docking pose pool. To keep the long-range electrostatic (LRE) interactions in the phosphate channel [50], the generated molecules with electron-rich functional groups are preferably selected. Table 1 below includes some selected examples.

All the molecules shared similar binding modes with the reference compound from 6RJ3. Compounds **1-7** share a common topology. Similar to the phenylpyrazole-5-carboxamide part of the reference compound, the phenylpyridopyrazin-5-one occupies the same position of the adenine pocket. The nitrogen atom in the pyrazine ring forms a polar interaction with the side chain of D174 to stabilize the lipophilic aromatic fragment. The cyclopropanecarboxylic acid motif, which mimics the benzoic acid in the reference compound, has long-range VDW interactions with the basic residues in the phosphate channel. The hydroxy-pyrrol-2-yl acetic acid moiety of compounds **8** and **9** act as the same role for VDW interactions, as well as oxazolone of compounds **11** and **12**.

The step-by-step elaboration of compound **2** from the benzene ring in the NADH/NAD⁺ binding pocket was presented in Fig. 5 A to F. The final pose of compound **2** was aligned to the reference compound. The common structure between current and previous generations showed nearly identical orientations. The vina docking scores decreased from -4.3 kcal/mol to -10.8 kcal/mol, while the MW increased from 78 Da to 485 Da. Despite the decent docking scores, conformation of compounds in D, E, F generated by AutoDock Vina may not be energetically favorable. The oxygen atom of the carbonyl group is near the nitrogen atom of pyrazine, whereas they should stay away in the

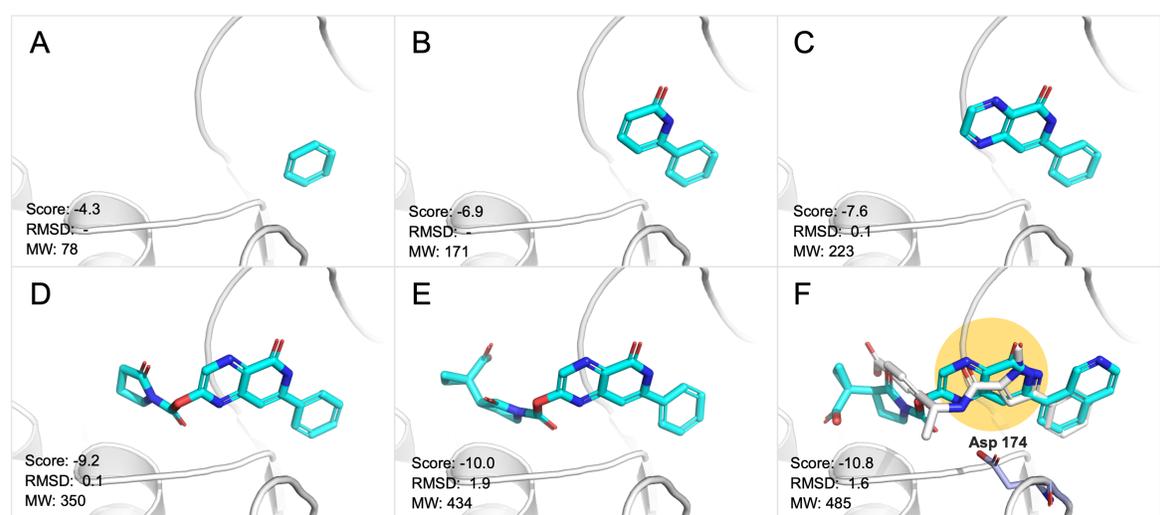


Figure 5. Evolutionary path of compound 2. A-E, fragment binding poses of Compound 2 in generation 1-5. F, binding pose of compound 2 and reference compound in the NADH/NAD⁺ binding pocket. The protein structure is shown in white cartoon. Compound 2 is shown in cyan stick. The reference compound is represented as white stick. Asp 174 is shown as lightblue stick. The adenine pocket is highlighted in a yellow circle. Vina score, RMSD(Å) of shared structure between two consecutive generations, and MW(Da) are provided.

lowest energy conformations. More accurate docking programs are needed for better outcomes.

Subsequently, the Synthetic Accessibility Module from Chemical.AI was used to estimate accessibility of these proposed molecules. The Synthetic Accessibility Module provides a primary estimation for many organic compounds under restricted computing resources. The predicted routes may not be the best choice, but it gives a quick estimate that can be used to assess whether the compound is easy to make or not. Generally speaking, majority of compounds can be made within 15 synthetic steps with no more than 7 linear steps. Unfortunately, no synthetic routes of compound 5 are suggested under the default setting of Synthetic Accessibility Module. In such cases, or when dealing with a short list of candidates that are of high interest, more accurate predictions can be done using the Synthesis Plan Module, which performs extensive search for all possible synthetic routes.

To address the sampling limitation of generated molecules before fitness evaluation, a new selection protocol combined deep learning method was developed. As described previously in the **Methods** section, the clean mode was used to build the DL model. Fig. S3 demonstrates the details of the model performance of each generation (A-E) and combined set (F), which includes data from all previous generations. The value of R square from Generation 1 to Generation 5 was gradually improved from 0.66 to 0.85. Furthermore, the R square of the DL model from the combined set was 0.85, slightly better than other models trained only by a single generation. It is reasonable to believe the model performance is sufficient for prediction [51–53]. The model F was then used for the prediction of the 66,687,173 molecules and 2,094 molecules with predicted scores less than -10.5 kcal/mol were subjected for redocking. The structures, MW, LogP, and synthetic accessibility analysis of representative molecules were listed in Table 2. It was pleased to see compounds that share similar scaffold with compound 2. Compounds that have completely different scaffolds

were also identified to provide diverse and valuable hypotheses for further validation. The superior performance of the deep learning model in the SECSE platform was speculated to result from the intrinsic logic. The 3D structural information of the parent compound was inherited in its daughter compounds, while the daughter generation would feedback rich SAR for model training. It also explained the model performance was improved in later generations while the combined set yielded the best.

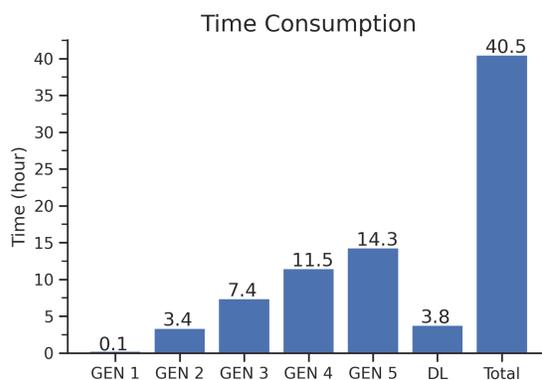
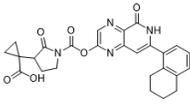
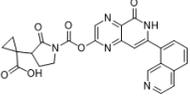
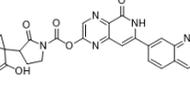
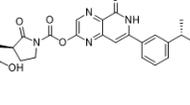
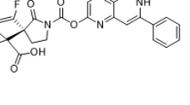
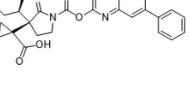
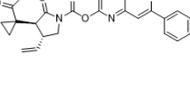
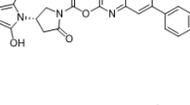
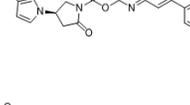
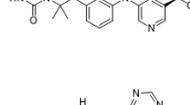
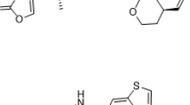
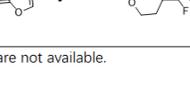


Figure 6. Time consumption. Running time of each generation (GEN 1-5). DL represents the time cost of a final search by deep learning model. With 80 cores, a five-generation computing cost was 40.5 hours.

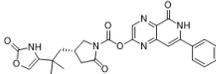
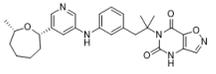
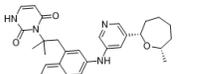
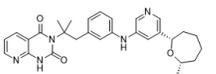
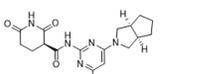
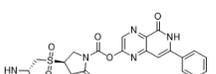
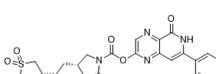
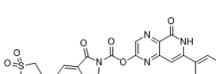
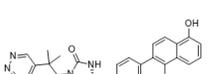
In this case, we used an 80-core computer, which took 40.5 hours in total. The Fig. 6 shows the calculation time of each generation. As the molecules grow larger and their complexity increases, the running time of each generation would gradually increase. Deep learning modeling significantly reduces search time and makes it possible to obtain estimated fitness scores for all generated molecules.

Table 1. List of selected candidates

ID	Name	Structure	MW	LogP	Vina Score	Total Steps	Liner Steps	SAScore
1	GEN_5_M_00769405-CC0		488	3.08	-11.51	11	7	3.32
2	GEN_5_M_00769403-CC0		485	2.75	-10.76	9	7	3.00
3	GEN_5_M_00769397-CC0		485	2.75	-10.25	10	7	3.16
4	GEN_5_M_00768747-CC3		490	3.71	-10.11	11	5	3.32
5	GEN_5_M_00769103-CC1		492	3.44	-10.21	N/A*	N/A	5.00
6	GEN_5_M_00768755-CC3		490	3.61	-10.08	7	5	2.65
7	GEN_5_M_00768685-CC3		460	2.61	-10.06	11	7	3.32
8	GEN_5_M_01041792-CC1		489	2.09	-10.35	8	5	2.83
9	GEN_5_M_01041787-CC0		489	2.09	-9.91	8	5	2.83
10	GEN_4_M_05182502-CC1		464	3.59	-9.54	14	10	3.74
11	GEN_5_M_67166829-CC1		433	4.09	-10.00	12	7	3.46
12	GEN_5_M_00132975-CC7		479	5.74	-9.20	13	9	3.61

* N/A means the predicted synthetic routes are not available.

Table 2. List of selected candidates based on the evaluation of DL models

ID	Name	Structure	MW	LogP	Predicted Score	Vina Score	Total Steps	Linear Steps	SAScore
13	GEN_4_M_00490117-CC1		489	2.98	-10.60	-11.34	14	9	3.74
14	GEN_5_M_02805023-CC3		489	4.81	-10.53	-10.12	12	10	3.46
15	GEN_5_M_02804877-CC0		498	5.83	-10.53	-10.12	16	8	4.00
16	GEN_5_M_02804883-CC2		499	5.22	-10.73	-10.10	15	8	3.87
17	GEN_3_M_26077238-CC1		419	2.37	-10.65	-10.06	5	4	2.24
18	GEN_4_M_00456833-CC1		498	0.23	-10.58	-10.06	10	7	3.16
19	GEN_4_M_00471055-CC0		496	2.55	-10.58	-10.21	12	7	3.46
20	GEN_4_M_00476893-CC7		494	2.47	-10.62	-10.21	17	7	4.12
21	GEN_5_M_05971075-CC1		484	5.62	-10.58	-10.18	15	13	3.87

4 DISCUSSION

As discussed earlier, the limitation in chemical space coverage of current chemical libraries is a common problem the field is facing. Intensively expanding the chemical space via brute-force exploration, which leads to ultra-large chemical libraries such as GDB [54–56] is explored. A more widespread attempt is the make-on-demand library [4], which comprises of structures from the enumeration of commercial building blocks based on reliable reaction schemes. The commercial providers also claim to have a relatively high synthesis success rate (at least 30%). The success of ultra-large compounds virtual screening contributes to the vigor of the make-on-demand library [2, 3]. Furthermore, people train machine learning models to accelerate the speed of virtual screening to balance the tradeoff between accuracy and speed [51]. However, it is still

a very tough task to do virtual screening of ultra-large libraries directly on the present hardware.

All these factors are considered and balanced in our own platform. It is probably unrealistic to enumerate all druglike molecules, but exhaustive enumeration of fragments with less than 13 heavy atoms is doable. Reaction rules are also included in our molecular generator to enrich the diversity of structures. To avoid the combinatorial explosion, protein pockets are constraints to direct the evolution. In addition, unexpected accuracy of the deep learning model allows us to evaluate large amount of compounds with minimal false positives or false negatives.

Recently, deep generative neural networks have become a promising approach for molecular generation. Many seminal reviews [10, 57] have summarized the development of these deep generative models with different generative architectures

(like recurrent neural networks, autoencoders, and generative adversarial networks) based on various molecular representations (SMILES, molecular graph). Despite the limitations of these generative models and the inaccuracy of current evaluation techniques for these models [58], they are indeed one choice for *de novo* molecular generation.

In parallel, rule-based molecular generation is also very popular such as AbbVies project Drug Guru [30, 31], the abbreviation of drug generation using rules. A data-driven method called matched molecular pairs (MMPs) [59–61] is another way to collect the experts knowledge from literature. Indeed, the rules of Drug Guru and MMPs are essentially the same method and nearly from the same source, that is molecular design thoughts of human beings. They can be stored as lines of reaction SMARTS code in RDKit. Scientists from SIMM constructed DrugSpaceX [62], a virtual compound library, using rules Nova and BIOSTER from StarDrop.

Scientists from GSK did a Turing test [63] for molecular generators by comparing three molecular generators in-house. The first one is BioDig, an MMPs-based algorithm. The second one is BRICS, a molecular generator by fragment recombination. The last one is RG2Smi, a deep generative model for generating molecules, which translates a molecule into a pharmacophore-based graph representation, then generates smiles string by deconvolution algorithm trained using natural language processing architecture. BioDig performed better than the other two methods across all tests in their report. Despite the fact that rule-based methods are somewhat limited to human knowledge or bias, we prefer rule-based methods for practical considerations.

Another challenge in computational *de novo* drug design is that compounds proposed by these tools are often hard to synthesize. Therefore, synthetic accessibility is a critical assessment for meaningful output. Previous retrosynthesis analysis tools were usually incapable of handling complex synthetic routes. Recently, as the development of deep learning and the availability of large reaction database, several new algorithms [64, 65] have been developed with improved capability for synthetic route planning. Yet increasing the evaluation throughput is challenging since it may take a few minutes to find practical routes for a single molecule. A batch mode that can evaluate thousands to millions of molecules at an affordable cost within a given timeframe is urgently needed.

SECSE mainly relies on structure-based computational design tools. Different tools will lead to different search directions, which might result in different chemical structure output. SECSE platforms are built to be compatible with various tools as fitness evaluators, like molecular docking, shape-based screening, and pharmacophore alignment, even ligand-based screening methods. Until now, docking has been the primary choice of SECSE because of its tradeoff between accuracy and speed. Despite the excellent performance of SECSE docking mode, there are still some inherent shortcomings in molecular docking methods, such as simplistic scoring with empirical energy function, rigid protein structures, ill-modeled poses. To enhance the prediction power, theoretically more accurate methods need to be introduced into the fitness assessment module. Claudio *et al.* [66] proposed a new QM-based docking program to replace the current docking methods based on molec-

ular mechanics (MM) force fields. The new scoring function has achieved excellent performance in most cases. However, their QM docking scoring function is ten times slower than traditional MM-based scoring functional per core. To explicitly consider the dynamic nature of proteins, molecular dynamics (MD) simulation is the best choice. Hugo Guterres *et al.* [67] reported a high throughput molecular dynamics (HTMD) simulations method to refine the docking results from AutoDock Vina. They calculated the RMSD of ligand by aligning protein structure from the initial docking pose and all protein structures in MD trajectory. They used a large set of 56 diverse target proteins and 560 ligands from the DUD-E dataset. The results show that short time MD simulations increase the area under the curve (AUC) of 0.8 from a value of 0.68 from AutoDock Vina. Enabled by the increasing computational power, attempts to add QM and MD concepts to the current docking program will be a promising way to improve the fitness evaluation module of SECSE.

5 CONCLUSION

We have developed a *de novo* design platform SECSE that integrates human intelligence for systemic evolutionary chemical space exploration against a specific protein pocket. The platform incorporated design rules of medicinal chemistry, computational evaluation methods, and deep learning models to efficiently speed up the search process of virtual hit compounds. The application in a demo target PHGDH proved its utility in finding diverse novel drug-like chemotypes. Further optimization considering high-precision evaluation methods and protein dynamics is currently underway. SECSE is released as an open-source project under the Apache License, Version 2.0. Any efforts and suggestions to improve its performance are welcomed.

ACKNOWLEDGEMENTS

We thank all of our colleagues at Keen Therapeutics, who have contributed to this work. We appreciate Chemical.AI for their help. Finally, we thank Dr. Yingxiao Cai for assistance with SA evaluation.

REFERENCES

- (1) MADE Building Blocks from Enamine, <https://enamine.net/building-blocks/make-on-demand-building-blocks>, Accessed December 1, 2021. Enamine.
- (2) Lyu, J.; Wang, S.; Balias, T. E.; Singh, I.; Levit, A.; Moroz, Y. S.; OMeara, M. J.; Che, T.; Alga, E.; Tolmachova, K., et al. *Nature* **2019**, *566*, 224–229.
- (3) Bender, B. J.; Gahbauer, S.; Lutten, A.; Lyu, J.; Webb, C. M.; Stein, R. M.; Fink, E. A.; Balias, T. E.; Carlsson, J.; Irwin, J. J., et al. *Nature protocols* **2021**, 1–34.
- (4) Warr, W. *ChemRxiv* **2021**, DOI: [10.26434/chemrxiv.14554803.v1](https://doi.org/10.26434/chemrxiv.14554803.v1).
- (5) Lemmen, C. Efficient 3D Exploration of Multi-Billion Compound Spaces, 2020.

- (6) Bohacek, R. S.; McMartin, C.; Guida, W. C. *Medicinal research reviews* **1996**, *16*, 3–50.
- (7) Schneider, G.; Fechner, U. *Nature Reviews Drug Discovery* **2005**, *4*, 649–663.
- (8) Hartenfeller, M.; Schneider, G. In *Cheminformatics and Computational Chemical Biology*, Bajorath, J., Ed.; Humana Press: Totowa, NJ, 2011, pp 299–323.
- (9) Schneider, G.; Clark, D. E. *Angewandte Chemie - International Edition* **2019**, *58*, 10792–10803.
- (10) Mouchlis, V. D.; Afantitis, A.; Serra, A.; Fratello, M.; Papadiamantis, A. G.; Aidinis, V.; Lynch, I.; Greco, D.; Melagraki, G. *International Journal of Molecular Sciences* **2021**, *22*, 1–22.
- (11) Meyers, J.; Fabian, B.; Brown, N. *Drug Discovery Today* **2021**.
- (12) Böhm, H.-J. *Journal of computer-aided molecular design* **1992**, *6*, 593–606.
- (13) Wang, R.; Gao, Y.; Lai, L. *Molecular modeling annual* **2000**, *6*, 498–516.
- (14) Yuan, Y.; Pei, J.; Lai, L. *Journal of chemical information and modeling* **2011**, *51*, 1083–1091.
- (15) Yuan, Y.; Pei, J.; Lai, L. *Frontiers in Chemistry* **2020**, *0*, 142.
- (16) Cheron, N.; Jasty, N.; Shakhnovich, E. I. *Journal of medicinal chemistry* **2016**, *59*, 4171–4188.
- (17) Durrant, J. D.; Amaro, R. E.; McCammon, J. A. *Chemical biology & drug design* **2009**, *73*, 168–178.
- (18) Spiegel, J. O.; Durrant, J. D. *Journal of Cheminformatics* **2020**, *12*, 1–16.
- (19) Baell, J. B.; Holloway, G. A. *Journal of medicinal chemistry* **2010**, *53*, 2719–2740.
- (20) Polishchuk, P. *Journal of Cheminformatics* **2020**, *12*, 28.
- (21) Nigam, A.; Pollice, R.; Krenn, M.; Gomes, G. D. P.; Aspuru-Guzik, A. *Chemical Science* **2021**, *12*, 7079–7090.
- (22) Steinmann, C.; Jensen, J. H. *PeerJ Physical Chemistry* **2021**, *3*, e18.
- (23) Bai, Q.; Tan, S.; Xu, T.; Liu, H.; Huang, J.; Yao, X. *Briefings in bioinformatics* **2021**, *22*, bbaa161.
- (24) Ma, B.; Terayama, K.; Matsumoto, S.; Isaka, Y.; Sasakura, Y.; Iwata, H.; Araki, M.; Okuno, Y. *Journal of Chemical Information and Modeling* **2021**, *61*, 3304–3313.
- (25) Li, Y.; Pei, J.; Lai, L. *Chemical Science* **2021**, DOI: [10.1039/d1sc04444c](https://doi.org/10.1039/d1sc04444c).
- (26) Gebauer, N. W.; Gastegger, M.; Schütt, K. T. *arXiv preprint arXiv:1906.00957* **2019**.
- (27) Imrie, F.; Bradley, A. R.; van der Schaar, M.; Deane, C. M. *Journal of Chemical Information and Modeling* **2020**, *60*, PMID: 32195587, 1983–1995.
- (28) Green, H.; Koes, D. R.; Durrant, J. D. *Chem. Sci.* **2021**, *12*, 8036–8047.
- (29) Nesterov, V.; Wieser, M.; Roth, V. *arXiv preprint arXiv:2010.06477* **2020**.
- (30) Stewart, K. D.; Shiroda, M.; James, C. A. *Bioorganic and Medicinal Chemistry* **2006**, *14*, 7011–7022.
- (31) Stewart, K. D.; Shanley, J.; Ahmed, K. B. A.; Bowen, J. P. In *Bioisosteres in Medicinal Chemistry*; John Wiley & Sons, Ltd: 2012; Chapter 11, pp 183–198.
- (32) Ravindranath, P. A.; Forli, S.; Goodsell, D. S.; Olson, A. J.; Sanner, M. F. *PLOS Computational Biology* **2015**, *11*, 1–28.
- (33) Ravindranath, P. A.; Sanner, M. F. *Bioinformatics* **2016**, *32*, 3142–3149.
- (34) Bickerton, G. R.; Paolini, G. V.; Besnard, J.; Muresan, S.; Hopkins, A. L. *Nature Chemistry* **2012**, *4*, 90–98.
- (35) RDKit: Open-source cheminformatics, <http://www.rdkit.org>, Accessed October 13, 2021.
- (36) Trott, O.; Olson, A. J. *Journal of computational chemistry* **2010**, *31*, 455–461.
- (37) Eberhardt, J.; Santos-Martins, D.; Tillack, A.; Forli, S. **2021**.
- (38) Kuntz, I.; Chen, K.; Sharp, K.; Kollman, P. *Proceedings of the National Academy of Sciences* **1999**, *96*, 9997–10002.
- (39) Chemical.AI, <https://chemical.ai/>, Accessed October 13, 2021. Wuhan Zhihua Technology Co., Ltd.
- (40) Goh, G. K.-M.; Foster, J. A. **2000**, 27–33.
- (41) Goldberg, D. E.; Holland, J. H. **1988**.
- (42) Michalewicz, Z.; Michalewicz, Z., *Genetic algorithms+ data structures= evolution programs*; Springer Science & Business Media: 1996.
- (43) Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M.; Palmer, A.; Settels, V.; Jaakkola, T.; Jensen, K.; Barzilay, R. *Journal of Chemical Information and Modeling* **2019**, *59*, PMID: 31361484, 3370–3388.
- (44) Tange, O. et al. *The USENIX Magazine* **2011**, *36*, 42–47.
- (45) Zhang, B.; Zheng, A.; Hydbning, P.; Ambroise, G.; Ouchida, A. T.; Goigny, M.; Vakifahmetoglu-Norberg, H.; Norberg, E. *Cell reports* **2017**, *19*, 2289–2303.
- (46) Rathore, R.; Schutt, C. R.; Van Tine, B. A. *Cancer drug resistance (Alhambra, Calif.)* **2020**, *3*, 762.
- (47) Zhao, J.-Y.; Feng, K.-R.; Wang, F.; Zhang, J.-W.; Cheng, J. F.; Lin, G.-Q.; Gao, D.; Tian, P. *European Journal of Medicinal Chemistry* **2021**, *217*, 113379.
- (48) Reid, M. A.; Allen, A. E.; Liu, S.; Liberti, M. V.; Liu, P.; Liu, X.; Dai, Z.; Gao, X.; Wang, Q.; Liu, Y., et al. *Nature communications* **2018**, *9*, 1–11.
- (49) Mullarky, E. et al. *Bioorganic & Medicinal Chemistry Letters* **2019**, *29*, 2503–2510.
- (50) Weinstabl, H. et al. *Journal of Medicinal Chemistry* **2019**, *62*, PMID: 31365252, 7976–7997.
- (51) Yang, Y.; Yao, K.; Repasky, M. P.; Leswing, K.; Abel, R.; Shoichet, B.; Jerome, S. *Journal of Chemical Theory and Computation* **2021**, *0*, DOI: [10.1021/acs.jctc.1c00810](https://doi.org/10.1021/acs.jctc.1c00810).
- (52) Gentile, F.; Agrawal, V.; Hsing, M.; Ton, A.-T.; Ban, F.; Norinder, U.; Gleave, M. E.; Cherkasov, A. *ACS Central Science* **2020**, *6*, PMID: 32607441, 939–949.
- (53) Choi, J.; Lee, J. *International Journal of Molecular Sciences* **2021**, *22*, DOI: [10.3390/ijms222111635](https://doi.org/10.3390/ijms222111635).
- (54) Fink, T.; Bruggesser, H.; Reymond, J.-L. *Angewandte Chemie International Edition* **2005**, *44*, 1504–1508.

- (55) Blum, L. C.; Reymond, J.-L. *Journal of the American Chemical Society* **2009**, *131*, PMID: 19505099, 8732–8733.
- (56) Ruddigkeit, L.; van Deursen, R.; Blum, L. C.; Reymond, J.-L. *Journal of Chemical Information and Modeling* **2012**, *52*, PMID: 23088335, 2864–2875.
- (57) Sousa, T.; Pereira, V.; Rocha, M. **2021**, 1–49.
- (58) Renz, P.; Van Rompaey, D.; Wegner, J. K.; Hochreiter, S.; Klambauer, G. On failure modes in molecule generation and optimization, 2019.
- (59) Warner, D. J.; Griffen, E. J.; St-Gallay, S. A. *Journal of Chemical Information and Modeling* **2010**, *50*, 1350–1357.
- (60) Hussain, J.; Rea, C. *Journal of Chemical Information and Modeling* **2010**, *50*, 339–348.
- (61) Awale, M.; Hert, J.; Guasch, L.; Riniker, S.; Kramer, C. *Journal of Chemical Information and Modeling* **2021**, DOI: [10.1021/acs.jcim.0c01143](https://doi.org/10.1021/acs.jcim.0c01143).
- (62) Yang, T.; Li, Z.; Chen, Y.; Feng, D.; Wang, G.; Fu, Z.; Ding, X.; Tan, X.; Zhao, J.; Luo, X.; Chen, K.; Jiang, H.; Zheng, M. *Nucleic Acids Research* **2021**, *49*, D1170–D1178.
- (63) Bush, J. T. et al. *Journal of Medicinal Chemistry* **2020**, *63*, 11964–11971.
- (64) Coley, C. W.; Rogers, L.; Green, W. H.; Jensen, K. F. *ACS Central Science* **2017**, *3*, PMID: 29296663, 1237–1245.
- (65) Segler, M. H.; Preuss, M.; Waller, M. P. *Nature* **2018**, *555*, 604–610.
- (66) Cavasotto, C. N.; Aucar, M. G. *Frontiers in Chemistry* **2020**, *8*, 246.
- (67) Guterres, H.; Im, W. *Journal of Chemical Information and Modeling* **2020**, *60*, 2189–2198.

Supplemental Materials

FRAGMENT LIBRARY GENERATION

As is shown in Fig. S1, a carbon string can be systemically closed to form rings with the same heavy-atom count. Sidechain replacement here means three connected non-aromatic atoms were rearranged to a center atom with two branches. The Enumerate Heterocycles function in RDKit is applied for heterocyclic ring generation if a ring structure exists. Reaction rules are used to achieve hetero atom replacement, bond replacement, and aromatic ring conversion. All the codes for fragment construction is available at http://github.com/KeenThera/fragment_generation.

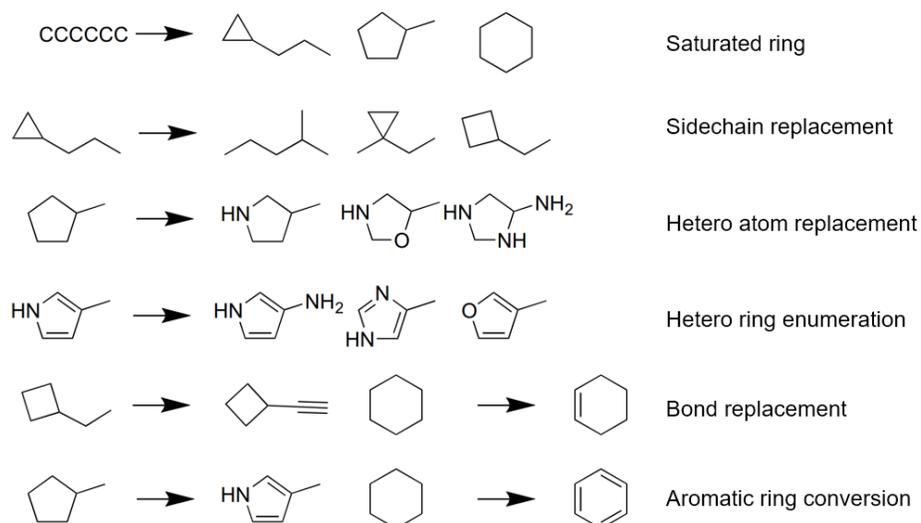


Figure S1. Examples of fragment construction rules.

CLUSTERING ALGORITHMS FOR HUGE DATASET

Fig. S2 presents the algorithm of clustering used in SECSE for selection of representative molecules. Clustering is necessary at this stage for efficiency considerations. In addition, a partition clustering method is introduced to reduce the computational cost significantly. Fingerprints calculated by RDKit (e.g., Morgan/Circular, MACCS keys) of all the molecules in the dataset are calculated as input features. Firstly, we randomly labeled one molecule in the dataset as the first cluster center C_1 . Then, we calculated the distance/dissimilarity ($1 - Tanimoto\ index$) of all the rest molecules with the first cluster center C_1 . The molecule with the largest distance is labeled as the second cluster center C_2 . At the same time, molecules that are pretty similar to the first cluster center will be masked. Next, the molecule with the largest distances to C_1 and C_2 would be considered the third cluster center C_3 ; molecules close to C_2 will be masked. Same iterations will continue until we find enough cluster centers or convergence is reached. Finally, we calculated the distance between all non-cluster center molecules and cluster centers and then assigned them based on the nearest cluster center id.

DEEP LEARNING MODELS PERFORMANCE

All the docking data are randomly split into three parts: training (80%), test (10%), and validation (10%) datasets. Fig. S3 presents the performance of deep learning models on the test dataset of PHGDH. A-E shows the performance of models using docking data set from generation 1-5, respectively. F shows the performance of the aggregate dataset, including data from the previous five generations.

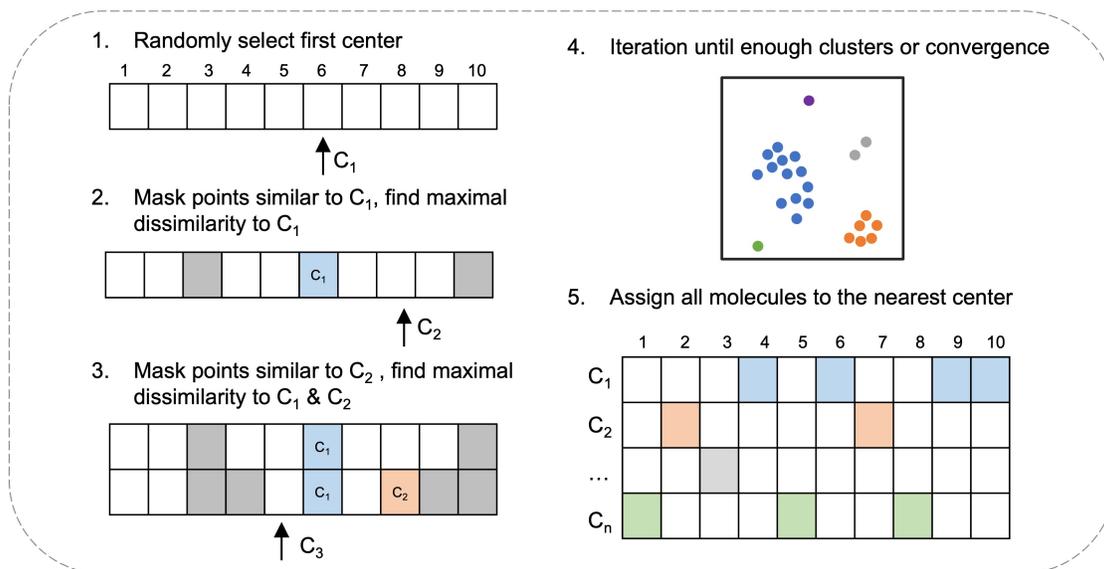
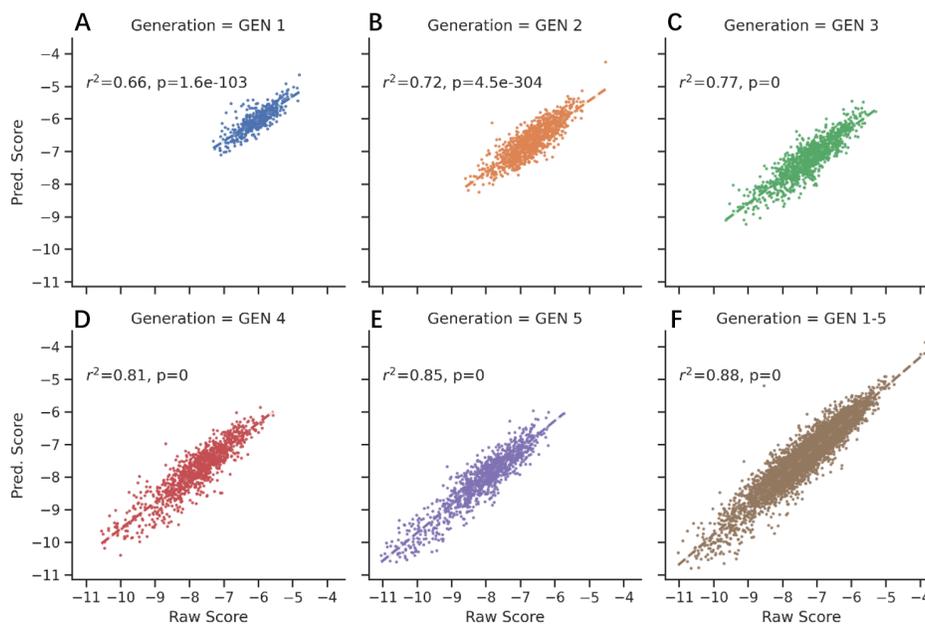


Figure S2. Details of clustering algorithms

Figure S3. Test data R^2 of deep learning models. A-E shows the performance of models using docking data set from generation 1-5, respectively. F shows the performance of the aggregate dataset, including data from the previous five generations.