Dynamic Beam Enumeration: A Bridge Between Generative Molecular Design and Library Screening

Anonymous Author(s)
Affiliation
Address
email

Abstract

Previous work reporting Beam Enumeration showed that *probable* substructures extracted from a generative model contains chemically meaningful information and can act as a source of explainability. In this work, we propose **Dynamic Beam Enumeration** as an extension to extract *larger* substructures. We show that this extracted insight can be made *actionable* and used to filter compounds in ultra-large make-on-demand libraries (10^{9-12}). The resulting molecules possess properties more aligned with the target objective than random sampling. Importantly, the results suggest that Dynamic Beam Enumeration can act as a bridge between generative design and library screening, such that even if generated molecules cannot be easily synthesized, extracted knowledge from the model can be used to find promising molecules that are make-on-demand.

1 Introduction

Molecular generative models have designed *experimentally validated* small molecule inhibitors[1] and catalysts[2]. However, *synthesizability* remains a challenge as many generative approaches do not explicitly enforce a notion of synthesizability. To this end, existing solutions include manual prioritization by expert chemists, which is not scalable. Algorithmic solutions include synthesizability heuristics based on historic data[3–6] or synthesizability-constrained generation which enforces chemically feasible transformation during generation[7–15]. Alternatively, retrosynthesis models[16–28], which predict feasible synthetic routes given a target molecule, can be included as an optimization objective during molecular generation[29,30] or for *post-hoc* filtering[31]. On the other side of the spectrum (Fig. 1b), virtual screening (VS), which aims to identify promising candidate molecules from *fixed* datasets containing synthesizable molecules continues to be a productive approach to molecular discovery in light of ultra-large molecular libraries (10^{9-12})[32–37].

Previous work introduced Beam Enumeration[38] which extracts *probable* molecular substructures during a language-based generative model's optimization trajectory (towards the target objective). These substructures were shown to be aligned with the target property profile. In this work, we propose **Dynamic Beam Enumeration (DBE)** as an extension of Beam Enumeration to extract probable *large* substructures from a model checkpoint. This contrasts Beam Enumeration which uses extracted substructures only *during* optimization. These *large* substructures form a considerable part of a full molecule and can be used to screen make-on-demand molecular libraries, such that molecules with matching substructures are enriched in the properties of interest. These *initial* results suggest the potential for DBE to act as a bridge between generative molecular design and VS, demonstrating how extracted knowledge can be made *actionable*. Importantly, matched compounds are presumably synthetically accessible, overcoming potential synthesizability challenges of *de novo* generated molecules.

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

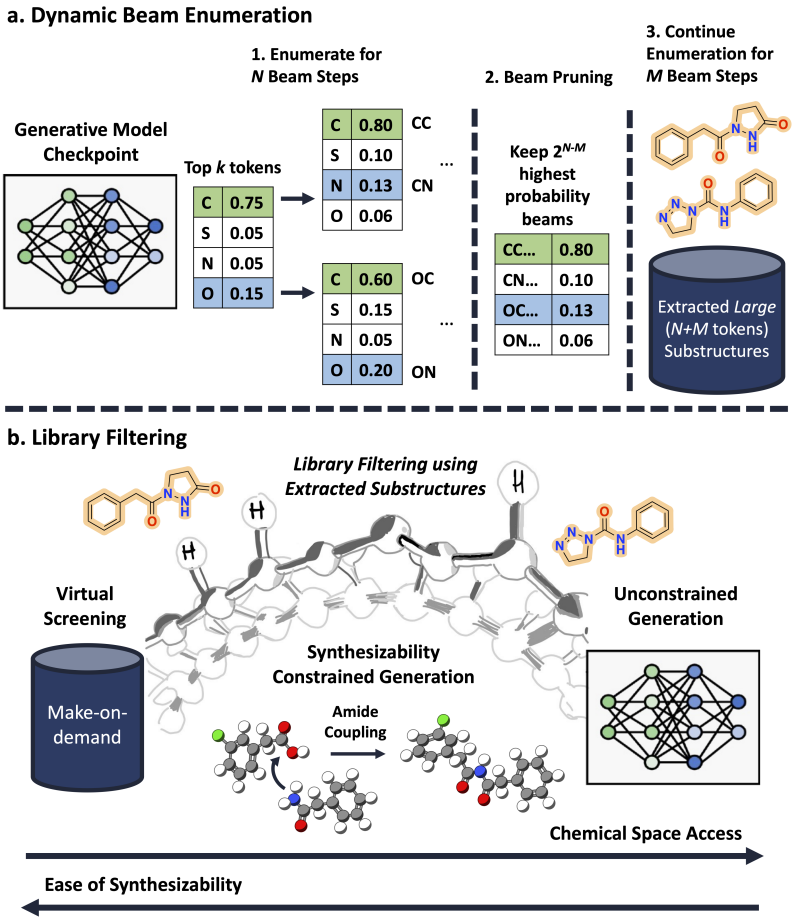

Figure 1: Dynamic Beam Enumeration. **a.** To enable the extraction of larger substructures, enumerated beams are pruned and only those with the highest probability are kept and further enumerated. **b.** Extracted large substructures can be used to filter make-on-demand libraries to *directly* identify promising molecules, acting as a bridge between generative design and screening.

2 Methods

Dynamic Beam Enumeration. We start from the Augmented Memory[39] language-based (LSTM[40] RNN) molecular generative model which generates molecules as SMILES[41]. Previous work extended Augmented Memory with Beam Enumeration[38] which exhaustively enumerates the top k (2 in this work) tokens for N (18 in this work) beam steps, resulting in k^N token sub-sequences. These are *sub-sequences* which map to *substructures* because they are incomplete SMILES and do not map to a full molecule yet. The Beam Enumeration work showed that larger substructures are more meaningful and the maximum size that can be extracted is directly controlled by the number of beam steps. The authors stated that 18 beam steps is the limit (on a 24GB GPU) due to exponentially increasing memory requirements. In this work, we propose DBE to prune the sub-sequences set and enable the remaining beams to continue enumeration (Fig. 1a). Specifically, at the end of the N beam steps, we keep only the top (by probability) $N - M$ beams (M is 12 in this work), resulting in k^{N-M} beams left. These remaining beams are enumerated for M more steps, resulting in sub-sequences $N + M$ tokens long, which map to larger substructures (Appendix A).

Goal-directed Generative Design. In this work, the use case of DBE is to extract large substructures from a model checkpoint and then use it to filter a make-on-demand library. Therefore, the first step is to task Augmented Memory[39] with optimizing a target objective. The case study is to generate molecules with good QuickVina2-GPU-2.1[42–44] docking scores to ATP-dependent Clp protease proteolytic subunit (ClpP) which is implicated in cancer[45]. The objective function is:

$$R(x) = \left(Docking\ Score(x)^5 \times QED(x)\right)^{\frac{1}{6}} \in [0, 1] \quad (1)$$

where x is a generated SMILES and the exponential terms apply more weighting to docking, i.e., its contribution to the reward is greater than QED score[46] which is an empirical measure of "drug-likeness" (see Appendix C for more details). All experiments were run across 10 seeds (0-9 inclusive) with 3,000 oracle calls.

Ultra-large Make-on-demand Library. WuXi GalaXi[47] contains billions of make-on-demand molecules and the sheer size makes screening (even with active learning[48,49]) computationally expensive. We consider two pseudo-randomly constructed subsets which we call WuXi (84,243,879) and WuXi-Large (756,642,169). See Appendix B for details. Extracted substructures from DBE are used to find match WuXi compounds that contain the substructure. The rationale for considering subsets differing by an order of magnitude is to increase the chances of a sufficient number of matches (1,000 in this work), as *large* substructures are necessarily specific.

Experimental Setup. Augmented Memory was tasked to generate molecules satisfying the objective function (Eq. 1) across 10 replicates (10 seeds, 0-9 inclusive). DBE was run on all 10 model checkpoints, extracting substructures with either a minimum token length of 20 or 25. These substructures were then used to find 1,000 matching compounds in the WuXi subsets, which were then assessed according to the same objective function (Eq. 1) and compared to random sampling. We note that experiments that did not match 1,000 compounds could be rescued by screening a larger WuXi subset, as WuXi-Large, despite containing >700M molecules, is a small fraction of WuXi GalaXi.

3 Results and Discussion

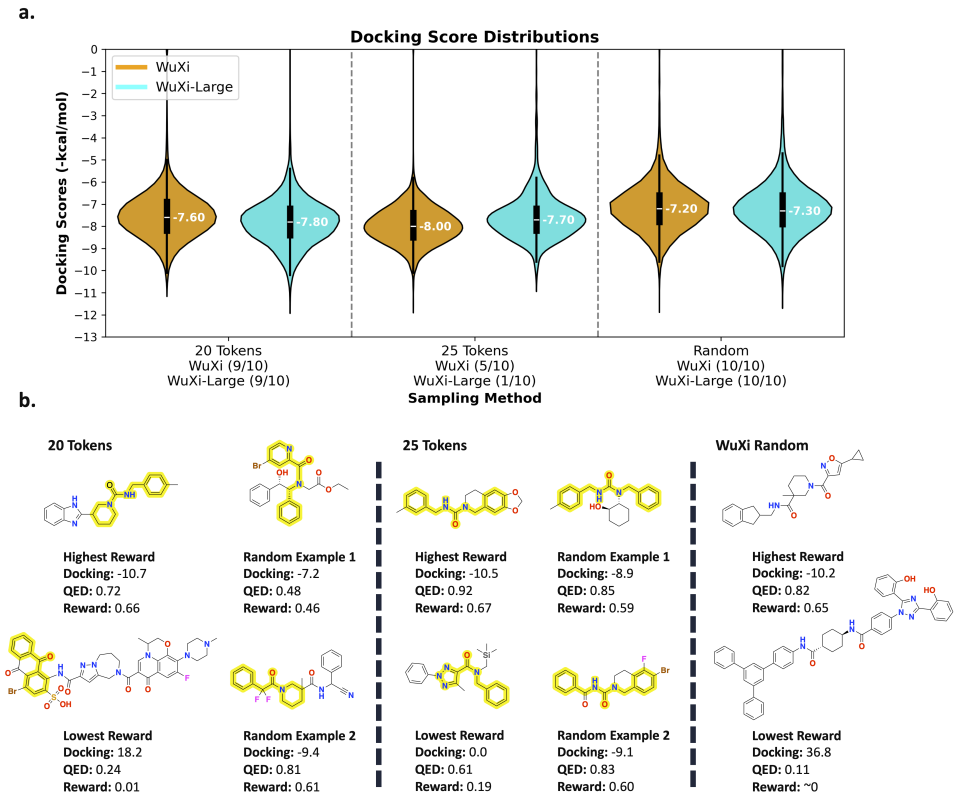

Figure 2: Docking scores distribution and example extracted substructures. **a.** Docking scores distribution of matched compounds vs. random sampling across WuXi and Wuxi-Large. **b.** Example extracted substructures using Dynamic Beam Enumeration and also showing the best and worst (by reward) and randomly matched compounds.

Quantitative Results. Fig. 2a shows the distribution of pooled (all runs that successfully matched 1,000 compounds) docking scores comparing DBE-matched compounds vs. random sampling. The docking scores of the matched compounds possess better docking scores, on average, than random sampling (statistically significant). For plots of QED and reward, see Appendix D. Next, Table 1 shows the mean and standard deviation of all properties. Note that reward is the aggregated "goodness" of a molecule, calculated based on Eq. 1. We make the following observations: firstly, on the WuXi subset, DBE outperforms random sampling and extracting *larger* substructures (25 vs. 20) further improves property values. Secondly, *both* docking scores and QED are improved because both properties are part of the objective function (Eq. 1). Thirdly, matched compounds maintain diversity which is often desirable in screening campaigns (Table 2). However, we note that this improvement is inconsistent as matching compounds on the WuXi-Large subset can result in worse QED. For future work, it would be straightforward to add additional physico-chemical property checks to guard against this by discarding matched molecules with poor QED, since it is cheap to compute. We further note that the reason for inconsistent results between WuXi and WuXi-Large may be due to the significantly decreased diversity of WuXi-Large (Table 2). Consequently, matched compounds may be *too* similar, rather than being a set of more distinctive compounds that simply share common substructures. The straightforward solution is to further increase the subset size and diversity. This would also increase the chances of matching 1,000 compounds.

Qualitative Results. Fig. 2b shows the best and worst (by reward) and two random examples of matched compounds. We make the following observations: firstly, the best randomly sampled molecule is comparable to the best DBE molecules, but on average, they are worse (Table 1). Secondly, the substructures extracted from different model checkpoints via DBE can converge, as evidenced by identical substructures highlighted (Fig. 2) Thirdly, molecules with poor docking scores can be particularly large or contain undesirable atoms, e.g., Si, and it would be straightforward for future work to add a filter for these when matching. Moreover, it is well known that docking scores can be artificially inflated for molecules with high molecular weight and logP[50]. Consequently, we analyzed select physico-chemical property distributions of the DBE matched vs. randomly sampled compounds (Fig. D5). Interestingly, DBE-matched compounds may not only possess better docking and QED values, but can also be smaller with less heavy atoms and polar surface area. This supports the efficacy of our workflow and shows promise for future development. Finally, the results show that information extracted from a generative model can be made *actionable* and that the substructures themselves are chemically meaningful, otherwise, such heavily biased matching would not result in a statistically significant improvement over random sampling.

Table 1: Extracted substructures are used to find compounds with the matching substructure in the WuXi datasets. The numbers in parenthesis denote the number of replicates out of 10 that were successful in matching 1,000 compounds. All compounds were pooled and the mean and standard deviation for docking, QED, and reward are reported.

Method	**WuXi (84,243,879)**			**WuXi-Large (756,642,169)**		
	Docking	QED	Reward	Docking	QED	Reward
Random (10, 10)	-7.13 ± 1.23	0.55 ± 0.19	0.47 ± 0.06	-7.17 ± 1.29	0.50 ± 0.16	0.46 ± 0.06
Dynamic Beam Enumeration						
20 Tokens (9, 9)	-7.52 ± 1.15	0.60 ± 0.22	0.49 ± 0.06	-7.82 ± 1.04	0.41 ± 0.15	0.47 ± 0.05
25 Tokens (5, 1)	-7.96 ± 0.83	0.74 ± 0.16	0.53 ± 0.04	-7.56 ± 1.08	0.36 ± 0.15	0.45 ± 0.05

4 Conclusion

In this work, we introduced **Dynamic Beam Enumeration (DBE)** as an extension to Beam Enumeration [38]. By extracting *larger* substructures from a generative model checkpoint, they can be used to find matching compounds in ultra-large make-on-demand datasets. Our results show that matched compounds have properties more aligned with the desired property profile than random sampling (Table 1), while maintaining diversity (Table 2). Future work will investigate different beam pruning methods, add filtering checks to discard matched compounds with poor physico-chemical properties, and extract larger substructures (>25 tokens). Our method demonstrates the potential for DBE to act as a bridge between generative design and library screening, showing how extracted chemical insights can be made *actionable*.

References

1. Yuanqi Du, Arian R Jamasb, Jeff Guo, Tianfan Fu, Charles Harris, Yingheng Wang, Chenru Duan, Pietro Liò, Philippe Schwaller, and Tom L Blundell. Machine learning-aided generative molecular design. *Nature Machine Intelligence*, pages 1–16, 2024.

2. Julius Seumer, Jonathan Kirschner Solberg Hansen, Mogens Brøndsted Nielsen, and Jan H Jensen. Computational evolution of new catalysts for the morita–baylis–hillman reaction. *Angewandte Chemie International Edition*, 62(18):e202218565, 2023.

3. Peter Ertl and Ansgar Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of cheminformatics*, 1:1–11, 2009.

4. Shuan Chen and Yousung Jung. Estimating the synthetic accessibility of molecules with building block and reaction-aware sascore. *Journal of Cheminformatics*, 2024.

5. Milan Voršilák, Michal Kolář, Ivan Čmelo, and Daniel Svozil. Syba: Bayesian estimation of synthetic accessibility of organic compounds. *Journal of cheminformatics*, 12:1–13, 2020.

6. Connor W Coley, Luke Rogers, William H Green, and Klavs F Jensen. Scscore: synthetic complexity learned from a reaction corpus. *Journal of chemical information and modeling*, 58 (2):252–261, 2018.

7. John Bradshaw, Brooks Paige, Matt J Kusner, Marwin Segler, and José Miguel Hernández-Lobato. A model to search for synthesizable molecules. *Advances in Neural Information Processing Systems*, 32, 2019.

8. John Bradshaw, Brooks Paige, Matt J Kusner, Marwin Segler, and José Miguel Hernández-Lobato. Barking up the right tree: an approach to search over molecule synthesis dags. *Advances in neural information processing systems*, 33:6852–6866, 2020.

9. Ksenia Korovina, Sailun Xu, Kirthevasan Kandasamy, Willie Neiswanger, Barnabas Poczos, Jeff Schneider, and Eric Xing. Chembo: Bayesian optimization of small organic molecules with synthesizable recommendations. In *International Conference on Artificial Intelligence and Statistics*, pages 3393–3403. PMLR, 2020.

10. Sai Krishna Gottipati, Boris Sattarov, Sufeng Niu, Yashaswi Pathak, Haoran Wei, Shengchao Liu, Simon Blackburn, Karam Thomas, Connor Coley, Jian Tang, et al. Learning to navigate the synthetically accessible chemical space using reinforcement learning. In *International conference on machine learning*, pages 3668–3679. PMLR, 2020.

11. Julien Horwood and Emmanuel Noutahi. Molecular design in synthetically accessible chemical space via deep reinforcement learning. *ACS omega*, 5(51):32984–32994, 2020.

12. Wenhao Gao, Rocío Mercado, and Connor W Coley. Amortized tree generation for bottom-up synthesis planning and synthesizable molecular design. *Proc. 10th International Conference on Learning Representations*, 2022.

13. Kyle Swanson, Gary Liu, Denise B Catacutan, Autumn Arnold, James Zou, and Jonathan M Stokes. Generative ai for designing and validating easily synthesizable and structurally novel antibiotics. *Nature Machine Intelligence*, 6(3):338–353, 2024.

14. Miruna Cretu, Charles Harris, Julien Roy, Emmanuel Bengio, and Pietro Liò. Synflownet: Towards molecule design with guaranteed synthesis pathways. *arXiv preprint arXiv:2405.01155*, 2024.

15. Michał Koziarski, Andrei Rekesh, Dmytro Shevchuk, Almer van der Sloot, Piotr Gaiński, Yoshua Bengio, Cheng-Hao Liu, Mike Tyers, and Robert A Batey. Rgfn: Synthesizable molecular generation using gflownets. *arXiv preprint arXiv:2406.08506*, 2024.

16. Marwin HS Segler and Mark P Waller. Neural-symbolic machine learning for retrosynthesis and reaction prediction. *Chemistry–A European Journal*, 23(25):5966–5971, 2017.

17. Marwin HS Segler, Mike Preuss, and Mark P Waller. Planning chemical syntheses with deep neural networks and symbolic ai. *Nature*, 555(7698):604–610, 2018.

18. Krzysztof Maziarz, Austin Tripp, Guoqing Liu, Megan Stanley, Shufang Xie, Piotr Gaiński, Philipp Seidl, and Marwin Segler. Re-evaluating retrosynthesis algorithms with syntheseus. *arXiv preprint arXiv:2310.19796*, 2023.

19. Samuel Genheden, Amol Thakkar, Veronika Chadimová, Jean-Louis Reymond, Ola Engkvist, and Esben Bjerrum. Aizynthfinder: a fast, robust and flexible open-source software for retrosynthetic planning. *Journal of cheminformatics*, 12(1):70, 2020.

20. Lakshidaa Saigiridharan, Alan Kai Hassen, Helen Lai, Paula Torren-Peraire, Ola Engkvist, and Samuel Genheden. Aizynthfinder 4.0: developments based on learnings from 3 years of industrial application. *Journal of Cheminformatics*, 16(1):57, 2024.

21. Ian A Watson, Jibo Wang, and Christos A Nicolaou. A retrosynthetic analysis algorithm implementation. *Journal of cheminformatics*, 11:1–12, 2019.

22. Connor W Coley, Luke Rogers, William H Green, and Klavs F Jensen. Computer-assisted retrosynthesis based on molecular similarity. *ACS central science*, 3(12):1237–1245, 2017.

23. Connor W Coley, Dale A Thomas III, Justin AM Lummiss, Jonathan N Jaworski, Christopher P Breen, Victor Schultz, Travis Hart, Joshua S Fishman, Luke Rogers, Hanyu Gao, et al. A robotic platform for flow synthesis of organic compounds informed by ai planning. *Science*, 365(6453): eaax1566, 2019.

24. Philippe Schwaller, Riccardo Petraglia, Valerio Zullo, Vishnu H Nair, Rico Andreas Haeuselmann, Riccardo Pisoni, Costas Bekas, Anna Iuliano, and Teodoro Laino. Predicting retrosynthetic pathways using transformer-based models and a hyper-graph exploration strategy. *Chemical science*, 11(12):3316–3325, 2020.

25. Amol Thakkar, Alain C Vaucher, Andrea Byekwaso, Philippe Schwaller, Alessandra Toniato, and Teodoro Laino. Unbiasing retrosynthesis language models with disconnection prompts. *ACS Central Science*, 9(7):1488–1498, 2023.

26. IBM. Rxn for chemistry.

27. Sara Szymkuć, Ewa P Gajewska, Tomasz Klucznik, Karol Molga, Piotr Dittwald, Michał Startek, Michał Bajczyk, and Bartosz A Grzybowski. Computer-assisted synthetic planning: the end of the beginning. *Angewandte Chemie International Edition*, 55(20):5904–5937, 2016.

28. Bartosz A Grzybowski, Sara Szymkuć, Ewa P Gajewska, Karol Molga, Piotr Dittwald, Agnieszka Wołos, and Tomasz Klucznik. Chematica: a story of computer code that started to think like a chemist. *Chem*, 4(3):390–398, 2018.

29. Jeff Guo and Philippe Schwaller. Saturn: Sample-efficient generative molecular design using memory manipulation. *arXiv preprint arXiv:2405.17066*, 2024.

30. Jeff Guo and Philippe Schwaller. Directly optimizing for synthesizability in generative molecular design using retrosynthesis models. *arXiv preprint arXiv:2407.12186*, 2024.

31. Jason D Shields, Rachel Howells, Gillian Lamont, Yin Leilei, Andrew Madin, Christopher E Reimann, Hadi Rezaei, Tristan Reuillon, Bryony Smith, Clare Thomson, et al. Aizynth impact on medicinal chemistry practice at astrazeneca. *RSC Medicinal Chemistry*, 15(4):1085–1095, 2024.

32. Jiankun Lyu, Sheng Wang, Trent E Balius, Isha Singh, Anat Levit, Yurii S Moroz, Matthew J O'Meara, Tao Che, Enkhjargal Algaa, Kateryna Tolmachova, et al. Ultra-large library docking for discovering new chemotypes. *Nature*, 566(7743):224–229, 2019.

33. Christoph Gorgulla, Andras Boeszoermenyi, Zi-Fu Wang, Patrick D Fischer, Paul W Coote, Krishna M Padmanabha Das, Yehor S Malets, Dmytro S Radchenko, Yurii S Moroz, David A Scott, et al. An open-source drug discovery platform enables ultra-large virtual screens. *Nature*, 580(7805):663–668, 2020.

34. Christoph Gorgulla, AkshatKumar Nigam, Matt Koop, Süleyman Selim Çınaroğlu, Christopher Secker, Mohammad Haddadnia, Abhishek Kumar, Yehor Malets, Alexander Hasson, Minkai Li, et al. Virtualflow 2.0-the next generation drug discovery platform enabling adaptive screens of 69 billion molecules. *bioRxiv*, pages 2023–04, 2023.

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

36. François Sindt, Anthony Seyller, Merveille Eguida, and Didier Rognan. Protein structure-based organic chemistry-driven ligand design from ultralarge chemical spaces. *ACS Central Science*, 10(3):615–627, 2024.

37. Fangyu Liu, Olivier Mailhot, Isabella S Glenn, Seth F Vigneron, Violla Bassim, Xinyu Xu, Karla Fonseca-Valencia, Matthew S Smith, Dmytro S Radchenko, James S Fraser, et al. The impact of library size and scale of testing on virtual screening. *bioRxiv*, pages 2024–07, 2024.

38. Jeff Guo and Philippe Schwaller. Beam enumeration: Probabilistic explainability for sample efficient self-conditioned molecular design. In *Proc. 12th International Conference on Learning Representations*, 2024.

39. Jeff Guo and Philippe Schwaller. Augmented memory: Sample-efficient generative molecular design with reinforcement learning. *JACS Au*, 2024.

40. Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

41. David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1): 31–36, 1988.

42. Oleg Trott and Arthur J Olson. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of computational chemistry*, 31(2):455–461, 2010.

43. Amr Alhossary, Stephanus Daniel Handoko, Yuguang Mu, and Chee-Keong Kwoh. Fast, accurate, and reliable molecular docking with quickvina 2. *Bioinformatics*, 31(13):2214–2216, 2015.

44. Shidi Tang, Ji Ding, Xiangyu Zhu, Zheng Wang, Haitao Zhao, and Jiansheng Wu. Vina-gpu 2.1: towards further optimizing docking speed and precision of autodock vina and its derivatives. *bioRxiv*, pages 2023–11, 2023.

45. Mark F Mabanglo, Keith S Wong, Marim M Barghash, Elisa Leung, Stephanie HW Chuang, Afshan Ardalan, Emily M Majaesic, Cassandra J Wong, Shen Zhang, Henk Lang, et al. Potent clpp agonists with anticancer properties bind with improved structural complementarity and alter the mitochondrial n-terminome. *Structure*, 31(2):185–200, 2023.

46. G Richard Bickerton, Gaia V Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L Hopkins. Quantifying the chemical beauty of drugs. *Nature chemistry*, 4(2):90–98, 2012.

47. WuXi. Virtual screening. URL `https://wuxibiology.com/drug-discovery-services/hit-finding-and-screening-services/virtual-screening/`.

48. Antonio Lavecchia and Carmen Di Giovanni. Virtual screening strategies in drug discovery: a critical review. *Current medicinal chemistry*, 20(23):2839–2860, 2013.

49. David E Graff, Eugene I Shakhnovich, and Connor W Coley. Accelerating high-throughput virtual screening through molecular pool-based active learning. *Chemical science*, 12(22): 7866–7881, 2021.

50. John A Arnott and Sonia Lobo Planey. The influence of lipophilicity in drug discovery and design. *Expert opinion on drug discovery*, 7(10):863–875, 2012.

51. Rdkit: Open-source cheminformatics. URL `http://www.rdkit.org`.

52. Daniil Polykovskiy, Alexander Zhebrak, Benjamin Sanchez-Lengeling, Sergey Golovanov, Oktai Tatanov, Stanislav Belyaev, Rauf Kurbanov, Aleksey Artamonov, Vladimir Aladinskiy, Mark Veselov, et al. Molecular sets (moses): a benchmarking platform for molecular generation models. *Frontiers in pharmacology*, 11:565644, 2020.

53. Yutong Xie, Ziqiao Xu, Jiaqi Ma, and Qiaozhu Mei. How much space has been explored? measuring the chemical space covered by databases and machine-generated molecules. In *Proc. 11th International Conference on Learning Representations*, 2023.

Appendix

The Appendix contains details on how Dynamic Beam Enumeration extracts substructures, the WuXi GalaXi[47] make-on-demand library, reward shaping details, and additional results.

A Dynamic Beam Enumeration: How are Substructures Extracted?

In this section, we discuss more in detail the formulation of the original Beam Enumeration[38], how Dynamic Beam Enumeration extends the method, and finally, how substructures are extracted from token sequences. Beam Enumeration exhaustively enumerates the top k highest probability tokens from autoregressive language-based molecular generative models. When k = 2, this results in 2^N sub-sequences, where N (18 in this work) is the number of beam steps. As the generative model is autoregressive, every sub-sequence has an associated probability given by the product of the individual token probabilities. For example, the sub-sequence "CN" has a probability of P(C) * P(N | C). Dynamic Beam Enumeration stores these probabilities for all enumerated sub-sequences, sorts them, then prunes the set and keeps only the highest 2^{N-M} subsequences. M is 12 in this work, therefore 2^6 = 64 were kept. Next, these 64 sub-sequences were further enumerated for 12 steps, resulting in token sub-sequences of length $M + N$ tokens = 30. Substructures are then extracted exactly as formulated in the original Beam Enumeration work: every consecutive sub-sub-sequence is considered and those that map to valid RDKit[51] molecules are stored. For example, consider the sub-sequence ABCDEF. The set of consecutive sub-sub-sequences are A, AB, ABC, ABCD, ABCDE, ABCDEF. Every single sub-sequence undergoes this process and the substructure frequencies are summed up. The top-4 most frequently appearing substructures are the set of extracted substructures. Finally, Dynamic Beam Enumeration enforces the minimum token length of the substructures to be either 20 or 25 in this work. This ensures that the subtructures are *larger* than the original Beam Enumeration work which only considered substructures with minimum token length as long as 15.

B WuXi GalaXi: Ultra-large Make-on-demand Library

WuXi GalaXi[47] contains billions of make-on-demand molecules. Dynamic Beam Enumeration extracted substructures are used to *match* compounds with these corresponding substructures. As searching through such a large database can be computationally prohibitive, we create two smaller sub-sets of WuXi in a pseudo-random manner. In this section, we detail the construction process of the two datasets which we call **WuXi** and **WuXi-Large**. Note that WuXi GalaXi is provided in **Phases** which can contain multiple **Parts**, each composed of a set of SMILES. Construction of the datasets involved extracting the raw zip files from WuXi GalaXi. Due to the size of the files, we performed the extraction for an arbitrary amount of time (not extracting the full file). This is why every file extraction as described below contains a seemingly completely random number of SMILES.

B.1 WuXi

WuXi was constructed by taking the entire **Phase 1** (24,093,421 SMILES) and part of **Phase 2 Part 3** (the first 60,150,458 SMILES). This was done arbitrarily and resulted in **84,243,879 total SMILES.**

B.2 WuXi-Large

WuXi-Large was constructed by taking some SMILES from all six parts of **WuXi Phase 2**.

- Part 1: First 125,821,918
- Part 2: First 122,345,617
- Part 3: First 133,357,788
- Part 4: First 132,610,455
- Part 5: First 120,735,841
- Part 6: First 121,770,550

This was done arbitrarily and resulted in **756,642,169 total SMILES.**

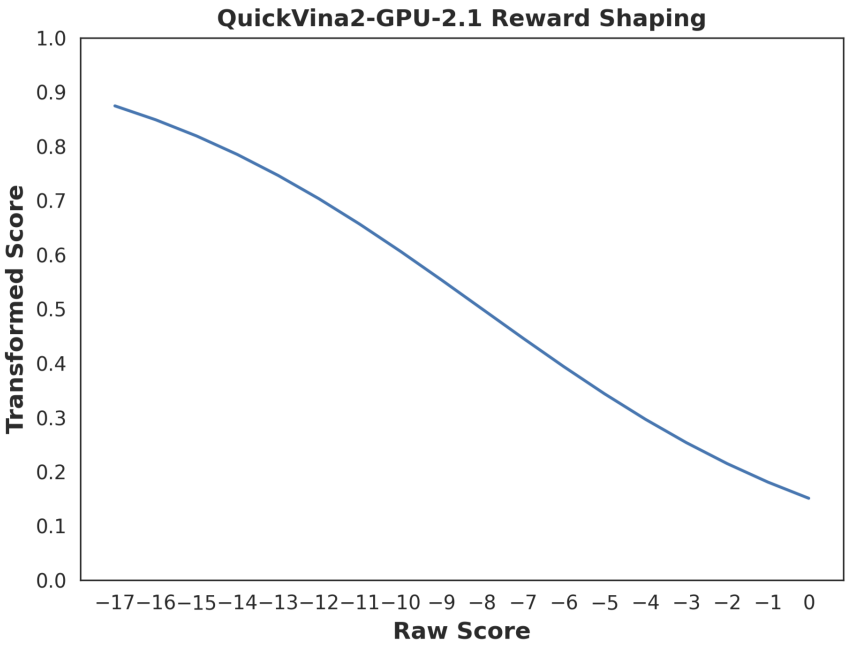

Figure C3: Reward shaping function for QuickVina2-GPU-2.1.

C Reward Shaping

This section contains details on the reward shaping functions used during Augmented Memory[39] goal-directed generation. Fig. C3 shows the function for QuickVina2-GPU-2.1[42–44] docking scores. For QED[46], raw values were used. Subsequently, the reward-shaped docking scores and raw QED values were aggregated to a single scalar following the equation below:

$$R(x) = \left[\prod_i p_i(x)^{w_i} \right]^{\frac{1}{\sum_i w_i}} \tag{2}$$

where x is a SMILES[41], i is the index of an oracle given many oracles (MPO objective), p_i is an oracle, and w_i is the weight assigned to the oracle. In this work, the oracles were QuickVina2-GPU-2.1[42–44] docking and QED[46] with weights of 5 and 1, respectively.

D Additional Results

This section contains supplementary plots comparing Dynamic Beam Enumeration matched compounds vs. random sampling in WuXi and WuXi-Large. Fig. D4 shows the distributions of docking scores, QED, and reward. Fig. D5 shows the distributions of molecular weight, topological polar surface area (tPSA), and number of heavy atoms. Table 2 contains statistics on the internal diversity (IntDiv1)[52] and #Circles[53] with threshold = 0.75 of the matched vs. randomly sampled compounds. We note that WuXi-Large is notably less diverse than the smaller WuXi.

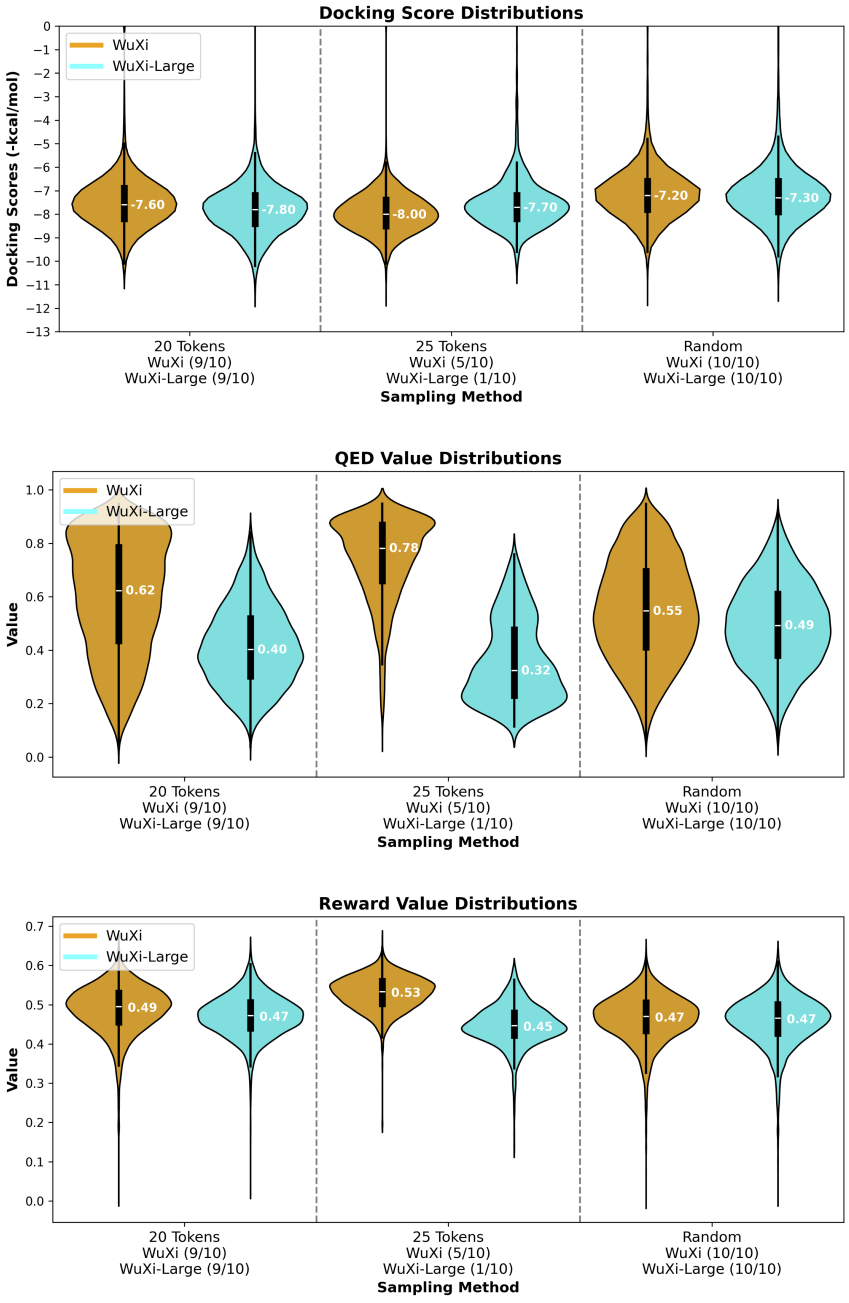

Figure D4: Distributions of oracle values.

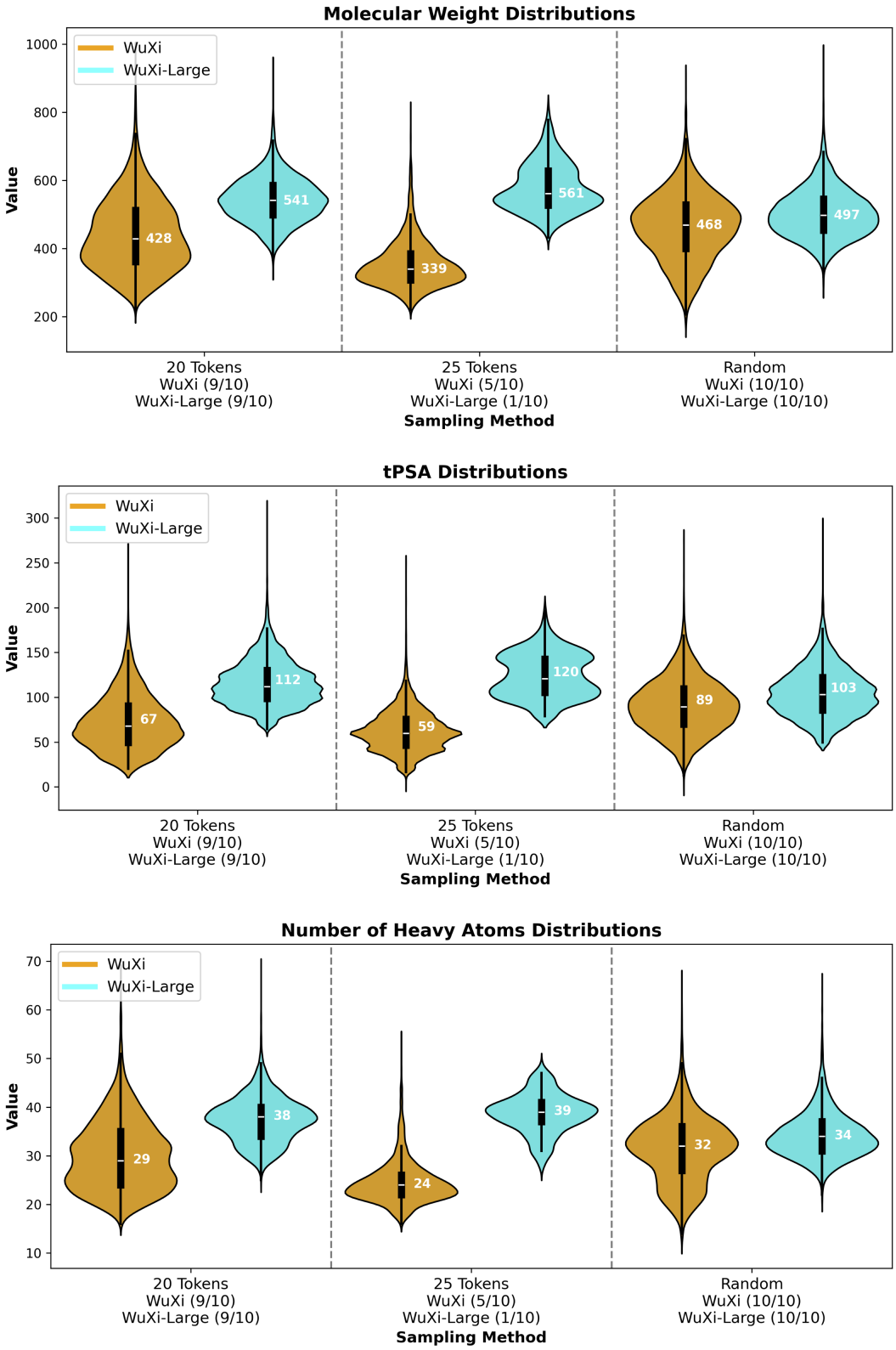

Figure D5: Distributions of property values.

Table 2: Diversity of WuXi matched compounds vs. randomly sampled. The numbers in parenthesis denote the number of replicates out of 10 that were successful in matching 1,000 compounds. All compounds were pooled and the mean and standard deviation for IntDiv1 and #Circles are reported.

Method	**WuXi (84,243,879)**		**WuXi-Large (756,642,169)**	
	IntDiv1	#Circles (Threshold = 0.75)	IntDiv1	#Circles (Threshold = 0.75)
Random (10, 10)	0.826 ± 0.001	90 ± 0.7	0.773 ± 0.002	5 ± 1
Dynamic Beam Enumeration				
20 Tokens (9, 9)	0.781 ± 0.035	20 ± 10	0.630 ± 0.039	1 ± 1
25 Tokens (5, 1)	0.743 ± 0.027	8 ± 3	0.741 ± 0.000	4 ± 0