

LING682 - Final Project

Controlled Molecular Generation using Sequential Monte Carlo Sampling with Small Language Models

260828214 Chih-Chao Hsu

McGill University, Department of Electrical and Computer Engineering

November 2025

Abstract

This report investigates the use of Sequential Monte Carlo (SMC) sampling for controlled molecular generation using small chemistry-focused language models. We demonstrate that SMC, equipped with potential-based rewards and SMILES validity constraints, substantially improves controllability and sampling efficiency. Our findings show that small models such as GPT2-ZINC, when guided by SMC, can achieve control performance that rivals or surpasses that of large-scale models such as Smiley Llama-8B, despite operating at a fraction of the computational cost.

<https://github.com/leohsuofnchu/Chem-SMC/>

1 Introduction

1.1 De Novo Molecular Discovery

De novo molecular discovery plays a crucial role in modern drug discovery and materials design. The ultimate goal is to be able to reveal novel, valid compounds that satisfy specified physicochemical or pharmacological properties, which could then possibly be potential precursors of ground-breaking new drugs and materials. Traditionally, rule-based, fragment-based design, combinatorial chemistry, and QSAR methods have been the major approaches. However, they are limited because predefined rules and fragment libraries restrict the accessible chemical space. With the rapid development of AI, various approaches including large language model pretraining [1], reinforcement learning [2], and more recently diffusion-based models [3], have all been explored for this purpose. A persistent challenge across these methods is to achieve reliable and robust control over properties such as molecular weight (MW), lipophilicity ($\log P$), synthetic accessibility, and drug-likeness (QED) for proposing useful candidates.

1.2 SMILES Representation and LLM-based Molecular Design

One popular direction of de novo molecular generation is to leverage the auto-regressive nature of large language models for generating SMILES strings. A SMILES string is a linear string representation of molecules, making it suitable to be modeled as a text generation task that follows certain rules. However, SMILES generation is fragile: tokenization errors or invalid syntactic structures can easily produce chemically impossible molecules. Most of the previous works focus on pre-training the models [1][4] with large amounts of SMILES strings in the scale of millions, so as to teach the model correct generation. The validness of generating molecules thus has higher guarantee, since the distribution of generated SMILES shares the similarity of the training dataset, but the ability of controlling the range of properties remains a challenge.

1.3 LLM-Constrained Generation

While large-scale pre-trained models can generate valid and diverse molecules, the generation process still needs the ability to specify precise and desired property range targets, which will accelerate the discovery process and make it more efficient for filtering out unwanted molecules early. Prior works to tackle this challenge have incorporated the instruction-following philosophy by supervised fine-tuning and direct preference optimization [5]. Although it manages to offer the ability to specify preferable ranges for generation, the effort for building datasets for fine-tuning and preference encoding is still time-consuming. On the other hand, the concept of LLM constraint decoding during inference time has become popular; multiple directions such as grammar-constrained decoding [6][7], and probability logit modification [8] have been proposed. Besides requiring less effort for computationally intensive pretraining, some even show a promising way for enhancing smaller models to have comparable performance to their bigger counterparts [9].

1.4 Motivation

The motivation for this project comes from one simple research question:

Can a small model outperform a much larger model in range-specific molecular generation via a constraint decoding mechanism?

Although the large pre-trained model SmileyLlama [5] has demonstrated strong ability for property-constrained generation through pure text prompts, the model itself is still huge, with 8 billion parameters. Based on the experiment in [9], it shows the potential of sequential Monte Carlo (SMC) constrained generation in molecular generation with smaller models. This work will extend this idea to explore how much performance can be squeezed out from a relatively smaller model compared to SmileyLlama. The hypothesis is that SMC has the potential to empower the model to have more fine-grained control over chemical property ranges, and possibly better performance under strict constraints.

2 Methodology

2.1 Importance Sampling

Importance sampling estimates expectations under an intractable target distribution $\pi(x)$ by drawing samples from a proposal $q(x)$ and reweighting them. The self-normalized estimator is

$$\mathbb{E}_\pi[f(x)] \approx \sum_{i=1}^N \tilde{w}^{(i)} f(x^{(i)}), \quad x^{(i)} \sim q(x),$$

with weights

$$w^{(i)} = \frac{\gamma(x^{(i)})}{q(x^{(i)})}, \quad \tilde{w}^{(i)} = \frac{w^{(i)}}{\sum_j w^{(j)}},$$

where $\pi(x) \propto \gamma(x)$ is known up to a constant. In molecular generation, π represents a property-aligned molecule distribution, while q corresponds to the language model's proposal distribution.

2.2 Sequential Importance Sampling (SIS)

Sequential Importance Sampling extends importance sampling to sequence generation by updating particle weights as tokens are appended:

$$w_t^{(i)} = w_{t-1}^{(i)} \frac{\gamma_t(x_{1:t}^{(i)})}{\gamma_{t-1}(x_{1:t-1}^{(i)}) q_t(x_t^{(i)} | x_{1:t-1}^{(i)})}.$$

Although SIS can in principle sample from an intractable target distribution using a proposal q , the recursive update causes the weight variance to grow rapidly with t , leading to severe weight degeneracy where most particles collapse to negligible weight. This limitation motivates the introduction of resampling.

2.3 Sequential Monte Carlo (SMC)

Sequential Monte Carlo uses the same sequential importance weighting mechanism as SIS, but augments it with **adaptive resampling** to prevent particle collapse. At each step, particles propose a new token $x_t^{(i)} \sim q_t(\cdot | x_{1:t-1}^{(i)})$. When the effective sample size (ESS) falls below a threshold, SMC resamples—replicating high-weight particles and discarding low-weight ones—to maintain diversity and stabilize long-horizon sequence generation.

2.4 Potential-Based Preference Encoding

In SMC, preferences can be encoded into the target density by adding a potential function Φ_t that scores how well a partial sequence matches the desired property range so far. We define

$$\gamma_t(x_{1:t}) \propto p_\theta(x_{1:t}) \Phi_t(x_{1:t}),$$

so that the importance weights naturally become

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{\Phi_t(x_{1:t}^{(i)})}{\Phi_{t-1}(x_{1:t-1}^{(i)})}.$$

Particles whose predicted properties move toward the target range receive larger weights, while those deviating from it are down-weighted and eventually discarded during resampling. This allows SMC to steer generation according to property constraints during inference time.

2.5 GenLM Framework and Implementation

GenLM [10] provides a general framework for controlled sequence generation by recasting constraint satisfaction as probabilistic conditioning. Instead of relying on locally constrained decoding (LCD)—which masks the vocabulary at each step and often distorts the global sequence distribution—GenLM uses **Adaptive Weighted Rejection Sampling** (AWRS) to sample tokens that satisfy arbitrary black-box constraints while maintaining the correct global probability over strings. AWRS avoids enumerating the entire vocabulary by rejecting only non-conforming tokens, yielding substantial runtime improvements over masking-based approaches. Importantly, AWRS also produces unbiased estimates of the constraint marginal Z , which serve as incremental importance weights within a **Sequential Monte Carlo** (SMC) framework to correct LCD’s myopic behavior and preserve the target posterior distribution over sequences.

In our work, we build directly on this framework: SMC-guided molecular generation is implemented using GenLM’s **PromptedLLM**, AWRS proposal mechanism, and the **Potential** interface to encode both SMILES validity constraints and our property-based continuous potential. This enables efficient, globally-correct constraint-aware sampling for GPT2-ZINC-87M without modifying the underlying language model.

3 Experimental Setup

3.1 Models

Two models are chosen for experiments. One is GPT2-Zinc-87M, which was fine-tuned on around 480 million SMILES from ZINC, and SmileyLlama-8B mentioned in the previous section, fine-tuned with around 2 million molecules from the ChEMBL Dataset v33. The reasons for choosing these two models are due to the big difference in parameter scale, similar trained molecules, and that both output SMILES strings, which is convenient to compare. Table 1 is the comparison between the two models.

| Model | Parameters | Training Data | Prompt Format |
|----------------|------------|-----------------|----------------------------------|
| GPT2-ZINC | 87M | ZINC (SMILES) | Direct SMILES prefix |
| SmileyLlama-8B | 8B | ChEMBL (SMILES) | Natural-language property prompt |

Table 1: Comparison of the two molecular language models and their prompting interfaces.

3.2 Datasets

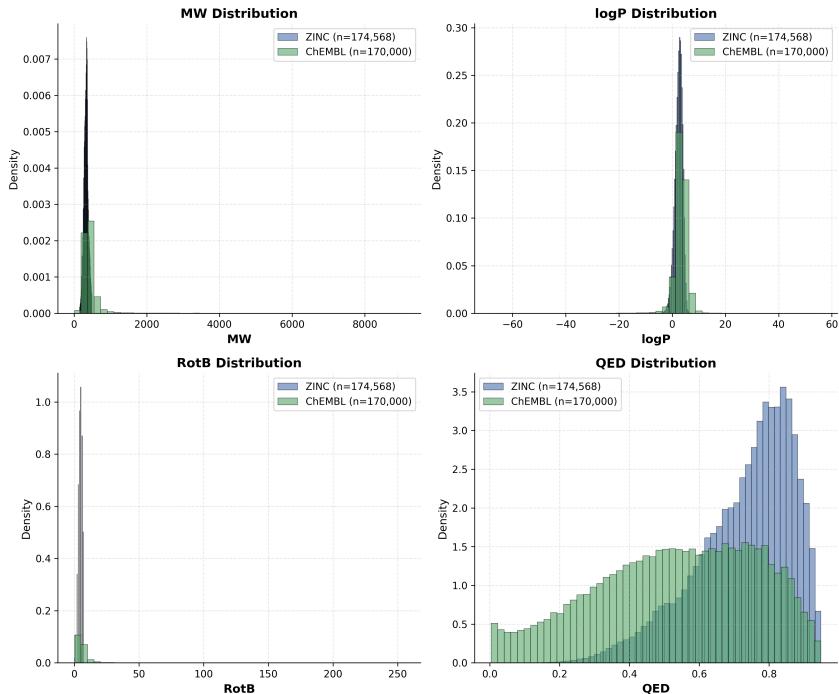


Figure 1: Property distribution comparison between **ZINC** and **ChEMBL**: MW, logP, RotB, and QED.

One of the concerns about comparing these two models could be from the fact that they are trained on different datasets. If the distribution of trained molecules is totally different, it would be hard to set a proper range of chemical properties for fair comparison. Therefore, a subset of molecules, around 170K, is randomly sampled from both datasets for visualization to see if there are huge differences. As shown in Fig. 1, we can see that for molecular weight (MW), LogP, and number of rotatable bonds (RotB), both datasets have highly similar distributions except for QED. Therefore, we only use properties that have similar distributions as the constraint elements. This difference in QED is expected, since ZINC is a drug-like filtered dataset while ChEMBL contains more diverse bioactive scaffolds, leading to broader QED variation.

3.3 Constraints Design

We evaluate controlled molecular generation using two types of constraints. The first, **range-based constraints**, derives property bounds from the percentile distribution of the combined sampled ZINC and ChEMBL data analyzed in Fig. 1. We define three difficulty levels: loose (5th–95th percentile: MW 233–577, logP –0.07–5.73, RotB 2–10), tight (25th–75th percentile: MW 304–419, logP 1.88–4.00, RotB 3–6), and ultra-tight (40th–60th percentile: MW 336–372, logP 2.58–3.36, RotB 4–5). These data-driven ranges provide a fair basis for comparing SMC-guided controlled generation for GPT2-ZINC-87M with SmileyLlama under a unified set of acceptable ranges. The second paradigm, **gradual constraints**, uses upper-bound-only specifications designed to match SmileyLlama’s instruction-following training format. We similarly define three levels; at each level, one additional constraint is introduced as shown in Table 2. The purpose of these two designs is to examine how well reward-based control and pure prompting can satisfy the specific constraints as the difficulty escalates.

| Constraint Type | Level | MW | logP | RotB |
|--------------------------|----------------------|------------|------------|----------|
| Range-Based (Percentile) | Loose (5–95%) | 233–577 | –0.07–5.73 | 2–10 |
| | Tight (25–75%) | 304–419 | 1.88–4.00 | 3–6 |
| | Ultra-tight (40–60%) | 336–372 | 2.58–3.36 | 4–5 |
| Gradual (Upper-Bound) | Loosen | ≤ 500 | – | – |
| | Tight | ≤ 400 | ≤ 4 | – |
| | Ultra-tight | ≤ 350 | ≤ 3.5 | ≤ 8 |

Table 2: Specification of range-based percentile constraints and gradual upper-bound constraints.

3.4 Reward (Potential) Design

To guide SMC toward molecules that satisfy the desired property ranges, we use a **constraint-based molecular potential** that scores each completed SMILES according to how well its RDKit-computed properties match the target constraints. For each property (MW, logP, RotB), we apply a soft distance-based penalty that rewards values near the center of the target interval and smoothly decays toward the boundaries. Additional mild bonuses (e.g., ring count) encourage reasonable, drug-like structures. Compared to the molecular generation formulation used in [9]—which relied primarily on a simple validity potential that checks whether a completion corresponds to a chemically valid molecule—our approach, based on the idea from previous work, extends the potential to incorporate **continuous, property-aware scoring**. The final molecular reward is combined with a SMILES-validity constraint in a GenLM Potential class, yielding a smooth critic of the form

$$\Phi(x) \propto \exp\left(\lambda \sum_k \phi_k(x)\right),$$

where each $\phi_k(x)$ represents the alignment of a single molecular property to its target range. This richer potential replaces hard threshold checks and enables SMC to favor molecules that are both valid and well aligned with the specified property constraints, while allowing properties to be added or removed easily without modifying or retraining the underlying language model.

3.5 Evaluation Metrics

Three key quantitative metrics are reported to assess the quality and controllability of molecule generation.

- **Validity (%)** — the percentage of generated SMILES strings that correspond to chemically

valid molecules as determined by RDKit. This metric reflects the model’s ability to produce syntactically and chemically feasible structures.

- **Adherence (%)** — the percentage of valid molecules whose physicochemical properties (e.g., molecular weight, logP, rotatable bonds) fall within the predefined constraint ranges for the current generation condition. This measures how effectively the model satisfies the desired property constraints.
- **Diversity** — the average pairwise Tanimoto distance between valid molecules, computed using RDKit fingerprints. Higher values indicate a more structurally diverse set of generated molecules, with less redundancy among samples.

3.6 Prompts

The prompts used for both models are different due to their nature. For GPT2-ZINC-87M, the prompt is the prefix of a SMILES string. We choose 20 different prefixes, shown in Table 3, as the start of 20 particles for providing a good start in generating valid molecules.

| Name | Prefix (SMILES) | Chemical Intent |
|-------------------------|-----------------|------------------------------------------------|
| aromatic_core | C1=CC= | Aromatic ring growth; encourages ring closure. |
| amino_aliphatic | CCN | Early N introduction for amines. |
| polar_ether | CO _C | Adds O; biases toward ethers/alcohols. |
| carbonyl_anchor | CC(=O) | Seeds esters / amides. |
| heterocycle_friendly | C1CN | N-heterocycle formation (piperazine-like). |
| sulfur_phosphorus | CCS | Thioether / sulfur motifs. |
| extended_polar | CCOCCN | Drug-like logP (2–4); polar fragments. |
| fallback_generic | CCC | Neutral aliphatic baseline. |
| aliphatic_ring | C1CCCCC1 | Cyclohexane seed; saturated rings. |
| aromatic_ring | c1ccccc1 | Benzene core. |
| hetero_aromatic | n1ccccc1 | Pyridine; hetero-aromatic bias. |
| saturated_n_heterocycle | N1CCCCC1 | Piperidine scaffold. |
| morpholine | O1CCNCC1 | Morpholine O/N heterocycle. |
| amide_starter | NC(=O) | Amide initiation. |
| sulfonamide_starter | NS(=O)(=O) | Sulfonamide seed. |
| urea | NC(=O)N | Urea/guanidine starter. |
| ether_builder | CO _C | Simple ether builder. |
| nitrile | C#N | Nitrile handle. |
| aryl_chloride | Clc1ccccc1 | Aryl chloride substitution. |
| phosphonate | P(=O)(O)O | Phosphonate moiety. |

Table 3: Compact list of prefix prompts for GPT2-ZINC-87M.

As for SmileyLlama-8B, we use the official prompt template from the model, which contains three sections:

Instruction: You love and excel at generating SMILES strings of drug-like molecules.

Input: Output a SMILES string for a molecule with: MW 300–400, logP 2–4, and ≤ 7 rotatable bonds.

Response:

3.7 SMC Configuration

We use a standardized SMC setup across all experiments. Each run uses 20 particles, a sampling temperature of 1.0, and an ESS resampling threshold of $0.3N$. Generation is performed with nucleus sampling ($\text{top-}p = 0.9$ and $\text{top-}k = 30$), and sequences are truncated at a maximum of 128 tokens.

4 Results and Discussion

| Model | Constraint Type | Level | Valid (%) | Distinct (%) | Adherence (%) | Diversity | Runtime (s) |
|-----------------|-----------------|-------------|-----------|--------------|---------------|-----------|-------------|
| GPT2-ZINC + SMC | Range-based | Loose | 100 | 100 | 50.8 | 0.9 | 5m 57s |
| | | Tight | 100 | 100 | 19.7 | 0.899 | 5m 6s |
| | | Ultra-tight | 100 | 100 | 2.7 | 0.899 | 6m 42s |
| | Gradual | Loosen | 100 | 100 | 100 | 0.91 | 6m 18s |
| | | Tight | 100 | 100 | 91.4 | 0.91 | 5m 9s |
| | | Ultra-tight | 100 | 100 | 73.65 | 0.91 | 5m 34s |
| SmileyLlama-8B | Range-based | Loose | 100 | 100 | 27.9 | 0.88 | 9m 52s |
| | | Tight | 100 | 100 | 2.3 | 0.889 | 9m 50s |
| | | Ultra-tight | 100 | 100 | 0.5 | 0.893 | 7m 46s |
| | Gradual | Loosen | 100 | 100 | 95.1 | 0.876 | 5m 44s |
| | | Tight | 100 | 100 | 80.4 | 0.875 | 4m 13s |
| | | Ultra-tight | 100 | 100 | 53.3 | 0.871 | 3m 59s |

Table 4: Comparison of GPT2-ZINC and SmileyLlama-8B under both range-based and gradual constraint paradigms. Metrics include validity, diversity, constraint adherence, and runtime.

The results in Table 4 and Figure 2 reveal a clear and consistent trend: **SMC substantially improves controllability for small molecular language models**. First, GPT2-ZINC+SMC achieves **100%** validity across all settings. Second, under *gradual* (upper-bound) constraints, which align with SmileyLlama’s instruction-tuning format, GPT2-ZINC+SMC consistently outperforms the 8B SmileyLlama model. For example, in the ultra-tight setting, GPT2-ZINC+SMC attains **73.65%** adherence compared with SmileyLlama’s **53.3%**, and maintains strong margins at loosen and tight levels as well. These findings highlight a key insight: **a small, chemical-specialized model equipped with constrained decoding can rival or surpass a much larger model relying solely on prompt conditioning**. On the other hand, two-sided *range* constraints (percentile-based) remain extremely difficult for both models. Adherence collapses below 3% with ultra-tight constraint, illustrating that narrow bidirectional property intervals are inherently challenging for autoregressive molecular decoders, even with SMC guidance. Importantly, diversity remains high in all SMC settings ($\sim 0.89\text{--}0.91$), which possibly arises from the use of 20 distinct SMILES prefixes, which offer structural variety before each SMC guided particle starts generation. Finally, SmileyLlama’s performance may be slightly understated because all experiments were run using its 4-bit quantized version due to limited computational resource availability, which is known to exhibit mild degradation compared to full-precision inference.

5 Conclusion and Future Work

This project shows SMC’s potential to turn a pretrained chemical language model into a well-guided de novo molecule generator. When equipped with SMC, GPT2-ZINC-87M reaches **perfect validity, broad variety, and strong compliance across different constraints**, outperforming the bigger 8-billion-parameter SmileyLlama, which only uses prompt guidance alone. The main weakness appears with dual-bound (range-style) conditions: performance drops sharply under narrow intervals, indicating that current potential designs lack enough flexibility for two-way control. Closing this gap might need more sophisticated potential design. Next steps could involve refining SMC through functions with non-linear restrictions, balancing multiple properties at the same time, and enabling core-driven synthesis.

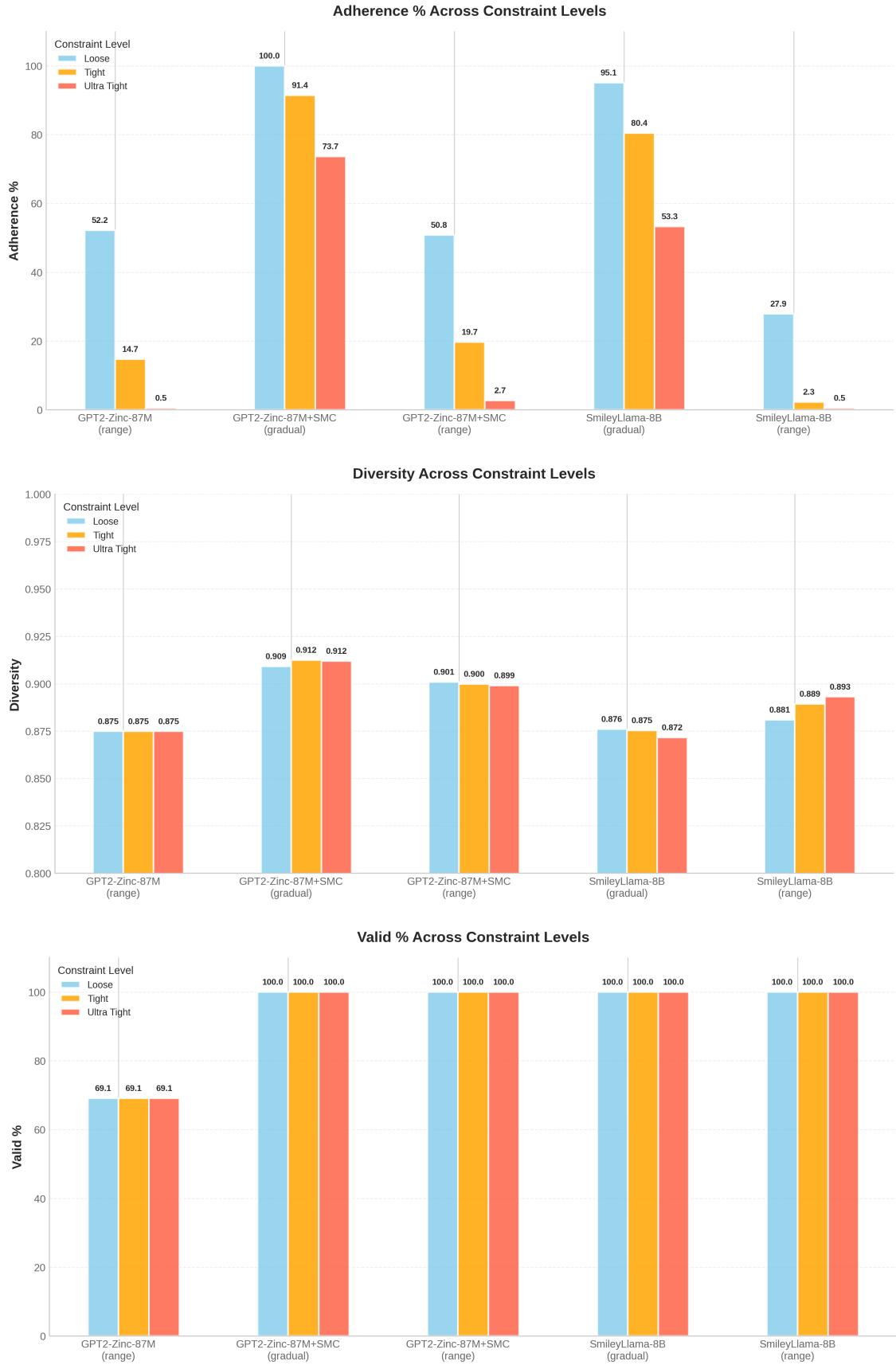


Figure 2: Comparison of validity, diversity, and adherence across constraint levels.

References

- [1] Nathan C. Frey, Ryan Soklaski, Simon Axelrod, Siddharth Samsi, Rafael Gómez-Bombarelli, Connor W. Coley, and Vijay Gadepally. Neural scaling of deep chemical models. *Nature Machine Intelligence*, 5(11):1297–1305, 2023.
- [2] Hannes H. Loeffler, Jiazen He, Alessandro Tibo, Jon Paul Janet, Alexey Voronov, Lewis H. Mervin, and Ola Engkvist. Reinvent 4: Modern ai–driven generative molecule design. *Journal of Cheminformatics*, 16(1):20, 2024.
- [3] Haisong Gong, Qiang Liu, Shu Wu, and Liang Wang. Text-guided molecule generation with diffusion language model, 2024.
- [4] Jerret Ross, Brian Belgodere, Vijil Chenthamarakshan, Inkit Padhi, Youssef Mrueh, and Payel Das. Large-scale chemical language representations capture molecular structure and properties, 2022.
- [5] Joseph M. Cavanagh, Kunyang Sun, Andrew Gritsevskiy, Dorian Bagni, Yingze Wang, Thomas D. Bannister, and Teresa Head-Gordon. Smileyllama: Modifying large language models for directed chemical space exploration, 2025.
- [6] Luca Beurer-Kellner, Marc Fischer, and Martin Vechev. Guiding llms the right way: Fast, non-invasive constrained generation, 2024.
- [7] Yixin Dong, Charlie F. Ruan, Yaxing Cai, Ruihang Lai, Ziyi Xu, Yilong Zhao, and Tianqi Chen. Xgrammar: Flexible and efficient structured generation engine for large language models, 2025.
- [8] Kareem Ahmed, Kai-Wei Chang, and Guy Van den Broeck. Controllable generation via locally constrained resampling, 2024.
- [9] João Loula, Benjamin LeBrun, Li Du, Ben Lipkin, Clemente Pasti, Gabriel Grand, Tianyu Liu, Yahya Emara, Marjorie Freedman, Jason Eisner, Ryan Cotterell, Vikash Mansinghka, Alexander K. Lew, Tim Vieira, and Timothy J. O’Donnell. Syntactic and semantic control of large language models via sequential monte carlo, 2025.
- [10] Benjamin Lipkin, Benjamin LeBrun, Jacob Hoover Vigly, João Loula, David R. MacIver, Li Du, Jason Eisner, Ryan Cotterell, Vikash Mansinghka, Timothy J. O’Donnell, Alexander K. Lew, and Tim Vieira. Fast controlled generation from language models with adaptive weighted rejection sampling, 2025.