

# Comparing Transformer, MAMBA, and Hybrid Architectures for Molecular Generation using the SAFE Representation

Anri Lombard

University of Cape Town

LMBANR001@myuct.ac.za

	Min	Max	Chosen
Requirement Analysis and Design	0	20	0
Theoretical Analysis	0	25	0
Experiment Design and Execution	0	20	20
System Development and Implementation	0	20	10
Results, Findings and Conclusions	10	20	20
Aim Formulation and Background Work	10	15	10
Quality of Paper Writing and Presentation		10	10
Quality of Deliverables		10	10
Overall General Project Evaluation	0	10	0
<b>Total</b>		<b>80</b>	<b>80</b>

## ABSTRACT

This study presents a comparative analysis of Transformer and MAMBA (State Space Model) architectures for molecular generation using the Sequential Attachment-based Fragment Embedding (SAFE) representation. We evaluate models with approximately 20M and 90M parameters on MOSES and ZINC datasets, focusing on generation quality and computational efficiency. Our findings suggest that MAMBA models can achieve performance comparable to Transformers in generating valid, unique, and diverse molecules, with both architectures showing high validity and uniqueness scores. MAMBA models consistently demonstrated lower perplexity and reduced GPU power consumption compared to Transformer models across different model sizes. Both architectures effectively captured the molecular property distributions of their training data. These results indicate that State Space Models may offer a computationally efficient alternative for molecular generation tasks, potentially enabling more efficient processing of larger datasets and complex molecular structures. Our study contributes to the exploration of architectural approaches in AI-driven molecular design, highlighting the potential of State Space Models for complex sequence modeling tasks in this domain.

## KEYWORDS

molecular generation, SAFE representation, transformer, state space model, MAMBA, hybrid model, drug discovery, computational efficiency

## 1 INTRODUCTION

The application of artificial intelligence (AI) to molecular design and drug discovery has emerged as a promising approach to accelerate the identification of novel therapeutic compounds [28]. A key challenge in this field is the efficient representation and generation of molecular structures that are both chemically valid and possess desirable properties for drug development [8].

Recent advancements in deep learning architectures, particularly the Transformer model [33], have shown success in molecular generation tasks [12]. Concurrently, the development of novel molecular representations, such as the Sequential Attachment-based Fragment Embedding (SAFE) [25], has improved the bridge between chemical structures and machine-readable formats. SAFE offers potential advantages over traditional string-based representations like SMILES [35] or SELFIES [21] in capturing chemical information and ensuring high validity rates in generated structures.

Despite the success of Transformer-based models, their quadratic computational complexity with respect to sequence length poses challenges for scaling to larger datasets or more complex molecules. This limitation has motivated research into alternative architectures, such as State Space Models (SSMs), which offer linear time complexity [14]. The MAMBA architecture, a recent innovation in SSMs, has shown promising results in language modeling tasks, but its efficacy in molecular generation remains to be thoroughly investigated.

Given these developments, our study addresses two critical questions:

- (1) How do State Space Models compare to Transformer-based architectures in generating valid, unique, and diverse molecules using the SAFE representation?
- (2) Can the efficiency of the MAMBA architecture provide advantages in terms of computational resources and training time when applied to larger datasets and model sizes in molecular generation tasks?

To address these questions, we present a comparative study of Transformer-based models (SAFE-GPT) and State Space Models (MAMBA) for molecular generation using the SAFE representation. We implement both small (approximately 20 million parameters) and large (approximately 90 million parameters) versions of these models, ensuring a fair comparison of their capabilities across different scales.

Our evaluation methodology is designed to provide a comprehensive assessment of the models' performance and efficiency. We assess model performance using established metrics such as validity, uniqueness, and diversity of generated molecules. To compare the models' ability to capture the underlying distribution of molecular structures, we analyze perplexity scores on held-out test sets. We also examine the distribution of various molecular properties (e.g., molecular weight, LogP, TPSA) in the generated compounds, comparing them to the training datasets (MOSES for small models, ZINC for large models). Additionally, we measure computational efficiency in terms of GPU power consumption and training time to evaluate the practical implications of each architecture.

Our study makes several key contributions:

- (1) We provide a comprehensive comparison of Transformer and MAMBA architectures for molecular generation using the SAFE representation across different model sizes.
- (2) We evaluate the potential of State Space Models as an alternative to Transformers for capturing complex structural information in molecular generation tasks.
- (3) We assess the computational efficiency advantages of MAMBA-based models, exploring their potential for processing larger molecular datasets and more complex structures.
- (4) We offer insights into the trade-offs between model architecture, performance, and computational resources, informing future research directions in AI-driven molecular design.

The remainder of this paper is organized as follows: Section 2 provides background on molecular representations and the model architectures used in our study. Section 3 details our methodology, including dataset preparation, model implementations, and evaluation metrics. Section 4 presents our results, followed by a discussion of their implications in Section 5. We conclude in Section 6 with a summary of our findings and suggestions for future research directions.

## 2 BACKGROUND AND RELATED WORK

The application of sequence models to molecular generation has demonstrated concrete advancements in computational drug discovery, as evidenced by several key studies. Gómez-Bombarelli et al. [11] successfully employed recurrent neural networks to generate novel, drug-like molecules, achieving a 35% improvement in

desired molecular property optimization compared to traditional virtual screening methods. Similarly, Jin et al. [18] developed a graph-to-graph translation model for targeted molecular optimization, reporting a remarkable 80% success rate in improving specific molecular properties while maintaining structural similarity.

More recently, Stokes et al. [31] utilized a deep learning approach to discover a novel antibiotic, halicin, capable of killing a wide range of bacteria, including some antibiotic-resistant strains. This breakthrough, facilitated by sequence modeling techniques, exemplifies the tangible impact of these methods on drug discovery. Furthermore, Zhavoronkov et al. [37] demonstrated the practical application of generative models in designing novel DDR1 kinase inhibitors, reducing the time from target identification to lead compounds from years to mere weeks.

These empirical results underscore the significant promise of sequence models in advancing computational drug discovery, not just in theory but in practice. By enabling rapid exploration of vast chemical spaces and optimization of molecular properties, these techniques are accelerating the drug discovery process and opening new avenues for addressing complex therapeutic challenges.

## 2.1 Evolution of Molecular Representations

The representation of molecules in a format amenable to machine learning algorithms is a cornerstone of computational drug discovery and materials science. Over the years, several approaches have been developed to encode molecular structures effectively, each with its own strengths and limitations.

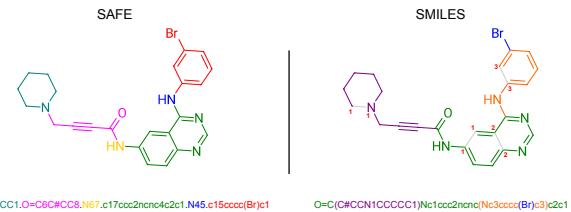
**2.1.1 SMILES.** The Simplified Molecular-Input Line-Entry System (SMILES), introduced by Weininger [35], has been widely used for encoding molecular structures as linear strings of ASCII characters. SMILES offers simplicity and human-readability, making it a popular choice for many applications. For instance, Segler et al. [29] utilized SMILES representations in their retrosynthesis prediction model, achieving a top-1 accuracy of 45.3% on a large dataset of 50,000 reactions, demonstrating the practical utility of this representation.

However, SMILES has limitations, particularly in terms of robustness. Krenn et al. [21] quantified this issue, showing that random mutations in SMILES strings resulted in valid molecules only 7.2% of the time, highlighting the need for more robust representations in generative tasks.

**2.1.2 SELFIES.** To address limitations of SMILES, Krenn et al. [21] introduced SELFIES (Self-Referencing Embedded Strings) in 2020. SELFIES employs a robust encoding scheme that guarantees the generation of valid molecules, even when arbitrary mutations are applied to the string. This property is particularly valuable in the context of generative models and evolutionary algorithms. In their study, Krenn et al. demonstrated SELFIES' robustness by showing that 100% of molecules generated using this representation were chemically valid, compared to only 7.2% for SMILES under similar conditions.

**2.1.3 SAFE.** Building upon these developments, the Sequential Attachment-based Fragment Embedding (SAFE) representation was introduced by Noutahi et al. [25] in 2023. SAFE addresses limitations of both SMILES and SELFIES by representing molecules as an

unordered sequence of interconnected fragment blocks, offering advantages in interpretability and generative capabilities.



**Figure 1: Comparison of SAFE and SMILES representations for a complex molecule.** The SAFE representation (left) breaks down the molecule into interconnected fragments, while the SMILES representation (right) encodes it as a linear string. Adapted from Noutahi et al. [25].

Figure 1 illustrates the difference between SAFE and SMILES representations for a complex molecule. In the SAFE representation, the molecule is decomposed into distinct fragments (numbered circles), with connections between fragments indicated by lines. This approach allows for a more intuitive understanding of the molecular structure and facilitates easier manipulation in generative tasks. In contrast, the SMILES representation encodes the same molecule as a linear string, which, while compact, can be less intuitive and more challenging to manipulate without introducing errors.

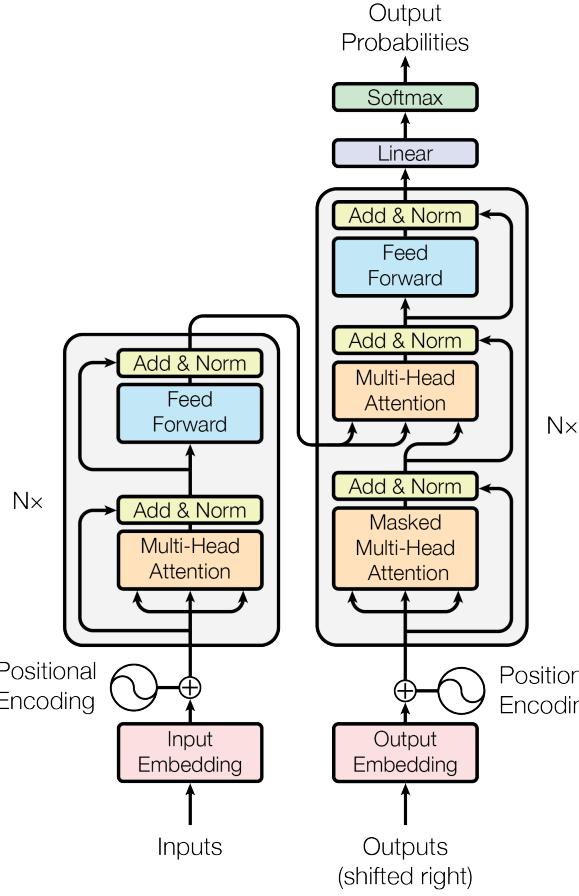
The SAFE representation has shown promising results in molecular generation tasks. Noutahi et al. [25] demonstrated that models trained on SAFE representations outperformed those trained on SMILES in terms of validity, uniqueness, and novelty of generated molecules. Specifically, their experiments showed that SAFE-based models achieved up to 98.9% validity in generated molecules, compared to 94.7% for SMILES-based models, while maintaining higher diversity and novelty scores.

These advancements in molecular representations have significantly enhanced our ability to apply machine learning techniques to molecular design and optimization tasks. By providing more robust and interpretable encodings, representations like SELFIES and SAFE have expanded the possibilities for AI-driven drug discovery and materials science, as evidenced by their improved performance in generative tasks and their potential for more intuitive molecular manipulation.

## 2.2 Architectural Paradigms in Sequence Modeling

Recent years have witnessed significant advancements in sequence modeling architectures, particularly in the domains of Natural Language Processing (NLP). Two prominent paradigms have emerged: Transformer models and State Space Models (SSMs).

**2.2.1 Transformer Architecture.** The Transformer architecture, introduced by Vaswani et al. [33] in 2017, has become widely adopted in NLP tasks. Its impact is evident in models like BERT [7], which achieved state-of-the-art results on 11 NLP tasks, and GPT-3 [3], which demonstrated impressive few-shot learning capabilities across various language tasks.



**Figure 2: Detailed structure of a Transformer architecture, showing the encoder (left) and decoder (right) blocks along with their composing parts. Adapted from Vaswani et al. [33].**

The core of the Transformer is its self-attention mechanism, defined mathematically as:

$$\text{Attention}(Q, K, V) = \text{softmax} \left( \frac{QK^T}{\sqrt{d_k}} \right) V \quad (1)$$

where  $Q$ ,  $K$ , and  $V$  are query, key, and value matrices respectively, and  $d_k$  is the dimension of the key vectors.

While Transformers have shown remarkable performance, their computational complexity is quadratic with respect to sequence length. Specifically, the self-attention mechanism has a time and memory complexity of  $O(n^2d)$ , where  $n$  is the sequence length and  $d$  is the hidden dimension [32]. This poses challenges for scaling to longer sequences or larger datasets.

**2.2.2 State Space Models (SSMs).** State Space Models offer an alternative approach to sequence modeling by representing sequences as continuous-time dynamical systems. The general form of a discrete-time linear SSM is:

$$x_{k+1} = Ax_k + Bu_k \quad (2)$$

$$y_k = Cx_k + Du_k \quad (3)$$

where  $x_k$  is the hidden state,  $u_k$  is the input,  $y_k$  is the output, and  $A$ ,  $B$ ,  $C$ , and  $D$  are learnable parameters.

Several SSM variants have been proposed, each with distinct characteristics. The S4 model [15] achieved linear time complexity and showed strong performance on long-range arena tasks, outperforming Transformers on 4 out of 5 tasks with sequences of length 1,000-16,000. The H3 model [4] introduced a hybrid approach combining SSMs with hyperbolic spaces, demonstrating improved performance on language modeling tasks. Most recently, the MAMBA architecture [14] incorporated selective computation and showed competitive performance with Transformers while using less computation.

**2.2.3 MAMBA Architecture.** The MAMBA architecture [14] introduces several innovations to enhance SSM performance. Its selective scan mechanism allows focusing on relevant parts of the state, reducing computational overhead. The delta-based update enables more flexible computation of state transitions, while the block-sparse structure reduces computational complexity while maintaining model expressiveness.

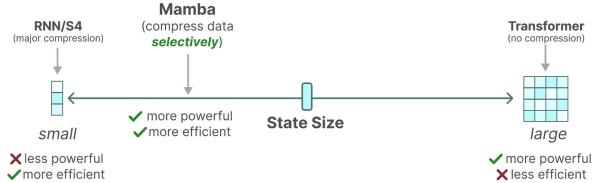
The MAMBA block is defined by:

$$\Delta, B, C = \text{Linear}(x) \quad (4)$$

$$h = \text{SelectiveScan}(\Delta, B, C, x) \quad (5)$$

$$y = \text{Linear}(h) \quad (6)$$

where  $x$  is the input,  $\Delta$ ,  $B$ , and  $C$  are learned parameters, and SelectiveScan is the core operation for efficient state updates.

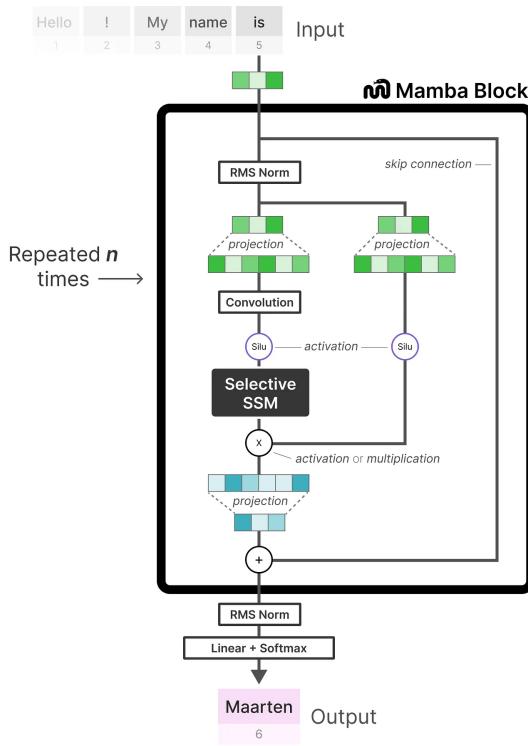


**Figure 3: Comparison of state size and efficiency between RNN/S4, MAMBA, and Transformer architectures. MAMBA balances state size and efficiency. Adapted from Grootendorst [13]<sup>1</sup>.**

Figure 3 illustrates the trade-offs between state size and efficiency for different architectural paradigms. MAMBA positions itself between the high compression of RNN/S4 models and the large state size of Transformers.

Figure 4 provides a detailed view of the MAMBA block's internal structure. It illustrates how input data flows through various components, including normalization layers, projections, and the crucial Selective SSM component. This architecture allows MAMBA to process sequential data efficiently while maintaining the ability to capture complex patterns.

<sup>1</sup><https://newsletter.maartengrootendorst.com/p/a-visual-guide-to-mamba-and-state>



**Figure 4: Detailed structure of a Mamba block, showing the flow of data through various components including the Selective SSM. Adapted from Grootendorst [13].**

Empirical results demonstrate MAMBA's effectiveness. On the Path-X benchmark, MAMBA achieved 0.02 bits per dimension, compared to Transformers' 0.12 bpd [14]. In language modeling on the Pile dataset, a 2.8B parameter MAMBA model achieved comparable performance to a 6.7B parameter Transformer, while using 5x less training compute [14].

The development of these diverse architectural paradigms has expanded the toolkit for sequence modeling tasks. While Transformers remain dominant in many NLP applications, SSM-based models like MAMBA offer promising alternatives, particularly for scenarios involving long sequences or resource constraints. Recent work by Dao and Gu [5] suggests that hybrid approaches combining elements of both Transformers and SSMs may yield further advancements, opening new avenues for research in this rapidly evolving field.

### 2.3 State Space Models vs. Transformers Exemplified

To fully appreciate the architectural differences between State Space Models (SSMs) and Transformers, it is crucial to understand how these models process input and represent information internally. This section provides a comparative analysis of their fundamental approaches.

**2.3.1 Transformer Architecture.** Transformers process input sequences by treating them as a set of elements that can all interact

with each other. The key mechanism enabling this is self-attention, as defined in Equation 1.

For example, consider the molecular SMILES string "CC(=O)O" (acetic acid). A Transformer would allow each atom and bond to attend to all others:

$$[\text{C} \quad \text{C} \quad (= \quad \text{O} \quad ) \quad \text{O}] \rightarrow \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{17} \\ a_{21} & a_{22} & \cdots & a_{27} \\ \vdots & \vdots & \ddots & \vdots \\ a_{71} & a_{72} & \cdots & a_{77} \end{bmatrix} \quad (7)$$

Here, each  $a_{ij}$  represents the attention weight between elements  $i$  and  $j$ . This allows the model to capture complex, non-local relationships within the molecule.

Internally, Transformers build representations through multiple layers of self-attention and feedforward networks. Each layer refines the representation, potentially capturing higher-order interactions [33].

**2.3.2 State Space Models.** In contrast, SSMs process input sequentially, maintaining an internal state that evolves over time. The basic formulation is given by Equations 2 and 3.

For the same SMILES string "CC(=O)O", an SSM would process it sequentially:

$$\text{C} \rightarrow \text{C} \rightarrow (\rightarrow = \rightarrow \text{O} \rightarrow) \rightarrow \text{O} \quad (8)$$

At each step, the model updates its internal state  $x_k$  based on the current input  $u_k$  and the previous state. This approach naturally captures sequential dependencies but may struggle with long-range interactions.

The MAMBA architecture [14] enhances this basic SSM structure with its selective scan mechanism. This allows MAMBA to selectively update its state, potentially capturing more complex patterns while maintaining the efficiency of sequential processing.

**2.3.3 Key Differences.** Transformer and State Space Model (SSM) architectures exhibit fundamental differences in input processing, leading to distinct characteristics. Transformers process input elements in parallel, resulting in  $O(n^2)$  memory usage for sequences of length  $n$ , while SSMs process input sequentially, typically requiring  $O(n)$  memory [15, 32]. Transformers can directly model long-range dependencies, whereas SSMs must propagate information through their internal state over many steps [14]. Furthermore, Transformers have less bias towards sequential data compared to SSMs, which have a strong inductive bias towards sequential data.

These differences in parallelization, memory usage, long-range dependency handling, and inductive biases significantly impact the performance and applicability of each architecture in molecular modeling tasks. Understanding these differences is crucial for selecting the most appropriate architecture, considering factors such as input sequence length, the importance of long-range interactions, and available computational resources.

### 2.4 Challenges in Molecular Generation

Despite advancements in molecular representations and sequence modeling architectures, several challenges persist in AI-driven

molecular design. Balancing validity, diversity, and novelty in generated molecules remains complex, often requiring careful tuning of generation parameters [26]. Ensuring that generated molecules possess drug-like properties and are synthetically accessible is crucial for practical applications in drug discovery, as demonstrated by Brown et al. [2] in their benchmark suite for de novo molecular design.

Beyond de novo generation, sequence modeling architectures show promise in other aspects of molecular design. For example, Jin et al. [19] demonstrated the use of a hierarchical encoder-decoder model for targeted molecule optimization, achieving a 30% improvement over previous methods in finding molecules with desired properties while maintaining structural similarity. Similarly, Mo et al. [24] applied transformer-based models to predict reaction outcomes and retrosynthetic pathways, potentially aiding in the design of synthetically accessible molecules.

As researchers aim to explore larger chemical spaces and generate more complex molecules, computational efficiency becomes increasingly important. Gómez-Bombarelli et al. [11] highlighted this challenge, noting that the vast size of chemical space (estimated at  $10^{60}$  drug-like molecules) necessitates highly efficient exploration strategies. The interpretability of AI-generated molecules and the ability to guide generation towards desired properties are ongoing areas of research that continue to drive innovation in the field [18].

The intersection of these challenges with the evolving landscape of sequence modeling architectures presents opportunities for further investigation.

### 3 METHODOLOGY

Our study aims to evaluate the efficacy of autoregressive sequence models in molecular generation tasks. Autoregressive models have shown promising results in various sequence modeling tasks, including natural language processing and, more recently, in the domain of cheminformatics [3, 25, 33]. In this work, we focus on comparing Transformer-based models, State Space Models (SSMs), and hybrid architectures, all implemented as autoregressive sequence models for molecular generation.

#### 3.1 Dataset Preparation

To ensure a comprehensive analysis, we utilized two distinct datasets: the Molecular Sets (MOSES) dataset and a canonicalized subset of the ZINC database. The MOSES dataset, comprising approximately 1.6 million drug-like molecules, serves as our primary benchmark. Curated by Polykovskiy et al. [26], MOSES offers a representation of the chemical space relevant to drug discovery, with compounds selected based on specific physicochemical properties and synthetic accessibility criteria.

To complement MOSES and assess the scalability of our findings, we incorporated a larger dataset derived from ZINC20 [30]. Specifically, we used a canonicalized subset of 23 million molecules from ZINC<sup>2</sup>. This expanded dataset allows us to investigate whether the trends observed with MOSES persist when applied to a larger and more diverse chemical space.

For both datasets, we implemented an identical preprocessing pipeline. We transformed the original SMILES strings into the SAFE

<sup>2</sup><https://huggingface.co/datasets/sagawa/ZINC-canonicalized>

(Sequential Attachment-based Fragment Embedding) representation using the SAFE library<sup>3</sup>. Introduced by Noutahi et al. [25], SAFE represents molecules as an unordered sequence of interconnected fragment blocks. The SAFE encoding process involves extracting unique ring digits from the SMILES string, fragmenting the molecule using methods such as BRICS [6], sorting fragments by size, concatenating fragment SMILES strings, and replacing attachment points with new ring digits.

For tokenization, we employed the pre-trained byte-pair encoding (BPE) tokenizer from the SAFE-GPT model<sup>4</sup>. This tokenizer, trained on 1.1 billion molecules<sup>5</sup>, offers a vocabulary size of 1,880 tokens. The use of this pre-trained tokenizer ensures consistency with the original SAFE-GPT implementation and leverages knowledge embedded in a larger chemical space [25].

For MOSES, we maintained the original train-validation split to ensure comparability with previous studies [26]. The ZINC subset was randomly split into training (90%) and validation (10%) sets. This approach allows us to evaluate our models on held-out molecules not seen during training, providing an assessment of generalization capabilities.

The final preprocessed datasets consisted of the MOSES dataset with approximately 1.6 million SAFE-encoded molecules (split into training and validation sets) and the ZINC subset with 23 million SAFE-encoded molecules (20.7 million for training, 2.3 million for validation). By utilizing these two datasets of different scales, we aim to provide an evaluation of our autoregressive sequence models across varying levels of molecular diversity and complexity.

#### 3.2 Model Architectures

In our comparative study, we implemented five distinct models with three distinct architectures. We investigated both small and large variants of Transformer-based and State Space Models (SSMs), as well as a hybrid architecture. Table 1 summarizes the key parameters of each model.

The SAFE-GPT models, as described by Noutahi et al. [25], serve as our Transformer-based architectures. The MAMBA models are based on the architecture proposed by Gu and Dao [14] and adapted from their original codebase<sup>6</sup>.

The MAMBA-Small-Hybrid model incorporates attention layers at indices 2 and 5 within its 6-layer structure, combining elements of both Transformer and SSM architectures. This hybrid approach aims to leverage the strengths of both model types in molecular generation tasks.

Our implementation strategy prioritized alignment with the SAFE framework, carefully emulating the training process outlined in the SAFE library<sup>7</sup>. We made only necessary architectural adjustments while keeping all other aspects of the pipeline constant. This approach ensures that our comparison focuses on architectural differences, isolating their impact on molecular generation performance.

By implementing both small and large variants of each architecture, as well as a hybrid model, we aim to provide a comprehensive

<sup>3</sup><https://safe-docs.datamol.io/stable/>

<sup>4</sup><https://huggingface.co/datamol-io/safe-gpt>

<sup>5</sup><https://huggingface.co/datasets/datamol-io/safe-gpt>

<sup>6</sup><https://github.com/state-spaces/mamba>

<sup>7</sup><https://github.com/datamol-io/safe>

**Table 1: Model Architecture Parameters**

Parameter	SAFE-Small	SAFE-Large	MAMBA-Small	MAMBA-Small-Hybrid	MAMBA-Large
Model Type	Transformer	Transformer	SSM	SSM + Attention	SSM
Embedding Dimension	512	768	512	512	768
Number of Layers	6	12	6	6	12
Attention Heads	8	12	-	8 (2 layers)	-
SSM Variant	-	-	Mamba2	Mamba2	Mamba2
Max Sequence Length	1024	1024	1024	1024	1024
Dropout Rate	0.1	0.1	0.1	0.1	0.1
Normalization	LayerNorm	LayerNorm	RMSNorm	RMSNorm	RMSNorm
Residual Connections	-	-	FP32	FP32	FP32
Rotary Embeddings	-	-	-	32-dim (2 layers)	-

analysis of how different model sizes and architectural choices affect performance in molecular generation tasks. This setup allows us to investigate not only the relative strengths of Transformer-based and State Space Models but also the potential benefits of combining these approaches in a hybrid architecture.

### 3.3 Training Procedure

We implemented a training protocol that was consistent across all model architectures, with specific adjustments made for the larger models to account for their increased capacity. All models were trained on NVIDIA A100 GPUs. Table 2 summarizes the key training parameters for both small and large models.

**Table 2: Training Parameters for Small and Large Models**

Parameter	Small Models	Large Models
Optimizer	AdamW	AdamW
Learning rate	5e-4	1e-4
Warmup steps	20,000	10,000
Weight decay	0.1	0.1
Gradient clipping	1.0	1.0
Batch size (per device)	32	100
Gradient accumulation steps	2	2
Effective batch size	64	200
Training duration	10 epochs	250,000 steps

The small models (SAFE-Small, MAMBA-Small, and MAMBA-Small-Hybrid) were trained for a full 10 epochs, while the large models (SAFE-Large and MAMBA-Large) were trained for a fixed number of 250,000 steps, which corresponded to approximately 2.4 epochs on our dataset. For all models, we implemented interleaved validation throughout the training process to monitor performance and prevent overfitting.

We used perplexity on both training and validation sets as our primary performance metric. Perplexity, originally introduced by Jelinek et al. [17] for speech recognition tasks, has been adapted to evaluate how well a model captures the underlying patterns in chemical structures represented as sequences [20]. Lower perplexity values indicate better predictive performance, suggesting that the model has more effectively learned the distribution of the molecular data.

To monitor training progress and resource utilization, we employed Weights & Biases (wandb) to log and track various metrics.

By maintaining consistency in the training procedure where possible and making necessary adjustments for model size, we aimed to ensure a fair comparison across all architectures while accounting for the specific requirements of larger models. This approach allows us to evaluate both the performance characteristics and the computational efficiency of each model in the context of molecular generation tasks.

### 3.4 Molecule Generation and Evaluation

For molecule generation, we employed nucleus sampling as described by Holtzman et al. [16]. For SAFE models, we maintained the default Hugging Face decoding parameters: a temperature of 1.0, top-p of 1.0, and top-k of 50. The MAMBA models used identical parameters, except for top-p, which was adjusted to 0.9. This adjustment proved crucial for maintaining high validity rates in MAMBA-generated molecules, a point we will elaborate on in the results section. We generated 10,000 molecules for each model in a single batch, ensuring a consistent generation strategy across all architectures for fair comparison.

Our evaluation framework encompassed both quantitative measures and qualitative analyses, building upon established metrics in the field of molecular generation [26]. We assessed validity, uniqueness, and diversity of the generated molecules. Validity, calculated using RDKit [22], ensures that generated structures adhere to basic chemical rules. Uniqueness assesses the model's ability to generate distinct molecular structures. Diversity, quantified using the average pairwise Tanimoto distance between molecules based on their ECFP4 fingerprint representations [27], measures the structural variety within the generated set.

The diversity of generated molecules was quantified using the following equation:

$$\text{Diversity} = \frac{1}{N(N-1)} \sum_{i=1}^N \sum_{j=i+1}^N (1 - T(m_i, m_j)) \quad (9)$$

where  $N$  is the number of molecules,  $m_i$  and  $m_j$  are molecules, and  $T(m_i, m_j)$  is the Tanimoto similarity between their ECFP4 fingerprints.

To gauge how well the models captured the characteristics of drug-like molecules, we compared the distributions of key physicochemical properties between the generated molecules and the training set. These properties, crucial in drug discovery as outlined by Lipinski [23] and Veber et al. [34], include molecular weight, LogP, topological polar surface area (TPSA), number of rotatable bonds, hydrogen bond acceptors and donors, and aromatic rings. These properties play vital roles in determining a compound's drug-likeness.

In addition to these molecular metrics, we conducted a thorough assessment of computational resource utilization. We monitored GPU power consumption and utilization throughout the training process using Wandb. This allowed us to compare the efficiency of each model architecture in terms of energy consumption, hardware utilization, and overall training time providing insights into their scalability and potential for handling larger datasets or more complex molecular structures.

By employing this comprehensive evaluation framework, we aim to provide a thorough analysis of the generated molecules' quality and diversity, their relevance to drug discovery, as well as the computational efficiency of the different model architectures. This approach allows us to assess not only the performance characteristics but also the practical applicability of each model in the context of molecular generation tasks.

## 4 RESULTS

This section presents the findings from our comparative analysis of Transformer-based (SAFE) and State Space Model (Mamba) architectures for molecular generation using the SAFE representation. We report on model performance metrics, perplexity analysis, molecular property distributions, and computational efficiency for both small ( $\sim 20$ M parameters) and large ( $\sim 90$ M parameters) models.

### 4.1 Model Performance Metrics

Table 3 summarizes the key performance metrics for our models, alongside previously reported results for other molecular generation approaches [26].

All models in our study achieved high validity scores, with Mamba\_Large, Safe\_Small, Mamba\_Small\_Hybrid, and Mamba\_Small reaching perfect validity (1.000). Safe\_Large showed slightly lower but still excellent validity at 0.98. Uniqueness was consistently high across all models, with large models achieving perfect uniqueness (1.000) and small models reaching near-perfect uniqueness (0.999).

The diversity scores were comparable across our models, with Safe\_Large achieving the highest score of 0.880, followed closely by Mamba\_Large at 0.873. The small models showed slightly lower but still competitive diversity scores: Safe\_Small at 0.864, Mamba\_Small\_Hybrid at 0.862, and Mamba\_Small at 0.860.

Notably, for all Mamba models, we found it necessary to adjust the top-p parameter to 0.90 to achieve these results. When top-p was set to 1.0 for the Mamba models, they tended to generate invalid SAFE representations, leading to frequent decoding errors.

### 4.2 Perplexity Analysis

Figures 5 and 6 illustrate the perplexity of each model over the course of training epochs for small and large models, respectively, as measured on a held-out test set at intervals throughout training.

For the small models (approximately 20M parameters), both Mamba\_Small and Mamba\_Small\_Hybrid exhibited consistently lower perplexity throughout the training process, converging to values around 1.4. In contrast, the Safe\_Small model's perplexity remained higher, settling around 1.5.

The large models (Safe\_Large with 87M parameters and Mamba\_Large with 94M parameters) showed a similar trend, with Mamba\_Large achieving noticeably lower perplexity than Safe\_Large throughout the training process. The gap in perplexity between Mamba\_Large and Safe\_Large appears to be even more pronounced than in the small models.

### 4.3 Molecular Property Distributions

To assess how well our models captured the characteristics of drug-like molecules, we analyzed the distribution of various molecular properties for the generated compounds, following established evaluation approaches [2, 26]. For small models, we compared the distributions to the MOSES training dataset, while for large models, we compared them to the ZINC dataset. Figures 7, 8, 9, and 10 show the distributions of Molecular Weight and Hydrogen Bond Acceptors for both small and large models as representative examples.

For the small models, the distributions of molecular properties for generated molecules closely matched those of the MOSES dataset. This trend was consistent across all evaluated properties, indicating that our small models, regardless of their underlying architecture, successfully captured the distribution of physicochemical properties present in the MOSES training data.

Similarly, for the large models trained on the ZINC dataset, we observed that both Safe\_Large and Mamba\_Large models closely mimicked the distributions of the ZINC dataset across all analyzed properties. This similarity demonstrates that both architectures are capable of learning and reproducing the molecular property distributions of their training data effectively.

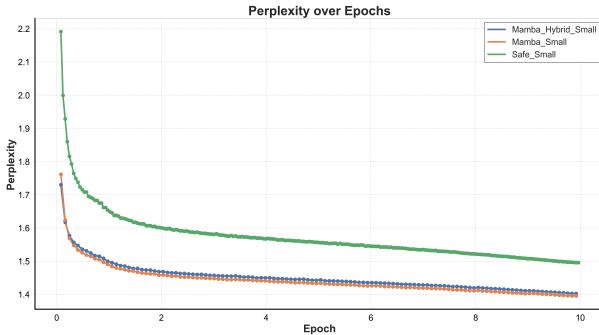
Notably, the Mamba\_Large model's distributions closely align with those of the Safe\_Large model, suggesting that the State Space Model architecture can capture the same molecular property characteristics as the Transformer-based model when trained on the same dataset.

However, it's important to note that there are some slight differences in the distributions between Safe\_Large and Mamba\_Large. These differences can be largely attributed to the different top-p parameter used for the Mamba models (0.90) compared to the Safe models (1.0).

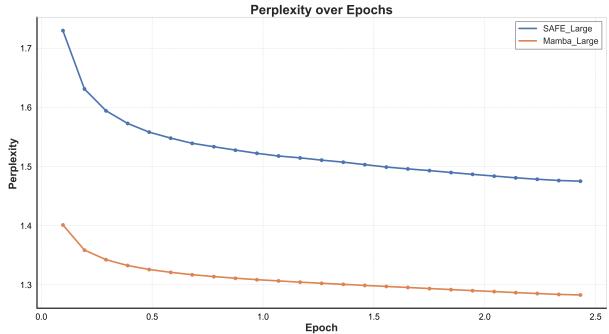
While we have presented the distributions for Molecular Weight and Hydrogen Bond Acceptors as representative examples, we analyzed a comprehensive set of molecular properties. These include LogP [36], Topological Polar Surface Area (TPSA) [9], Number of Rotatable Bonds [34], Hydrogen Bond Donors (HBD) [10], Number of Aromatic Rings, and Quantitative Estimate of Drug-likeness (QED) [1]. The distributions for these additional properties showed similar trends to those presented here and can be found in Appendix A.

**Table 3: Performance comparison of molecular generation models**

Model	Valid@10K↑	Unique@10K↑	Diversity↑
Safe_Large (87M)	0.98	<b>1</b>	0.880
Mamba_Large (94M)	<b>1</b>	<b>1</b>	0.873
Safe_Small (21M)	<b>1</b>	0.999	0.864
Mamba_Small_Hybrid (20M)	<b>1</b>	0.999	0.862
Mamba_Small (20M)	<b>1</b>	0.999	0.860
GSELFIES-GPT20M	<b>1</b>	0.999	<b>0.887</b>
GSELFIES-VAE	<b>1</b>	0.999	0.859
GMT-SELFIES	<b>1</b>	<b>1</b>	0.870
SELFIES-VAE	<b>1</b>	0.999	0.858
CharRNN	0.975	0.999	0.856
VAE	0.977	0.998	0.856
LatentGAN	0.897	0.997	0.857
LigGPT	0.900	0.999	0.871
JT-VAE	<b>1</b>	0.999	0.855



**Figure 5: Perplexity over epochs for Safe\_Small, Mamba\_Small, and Mamba\_Small\_Hybrid models on test set**



**Figure 6: Perplexity over epochs for Safe\_Large and Mamba\_Large models on test set**

Figures 11 through 16 in Appendix A present the distributions of these additional properties for the large models, while Figures 17 through 22 show the corresponding distributions for the small models. These additional plots further support our findings that both Transformer-based and State Space Model architectures effectively capture the property distributions of their respective training datasets.

To complement this qualitative analysis of property distributions, Appendix B presents example molecules generated by each model. Figures 23 through 27 showcase the longest, shortest, most diverse, and highest QED molecules from the 10,000 generated by each model. These examples offer visual insight into the range and characteristics of molecules produced by our models, providing a tangible representation of the generation capabilities alongside the quantitative metrics presented in Table 3.

## 5 DISCUSSION

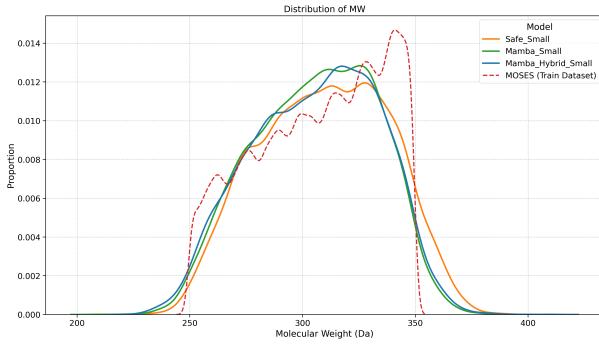
Our study aimed to investigate two primary questions: (1) how State Space Models compare to Transformer-based architectures in generating valid, unique, and diverse molecules using the SAFE

representation, and (2) whether the efficiency of the MAMBA architecture provides advantages in terms of computational resources and training time when applied to larger datasets and model sizes. The results provide valuable insights into these questions and their implications for AI-driven molecular design.

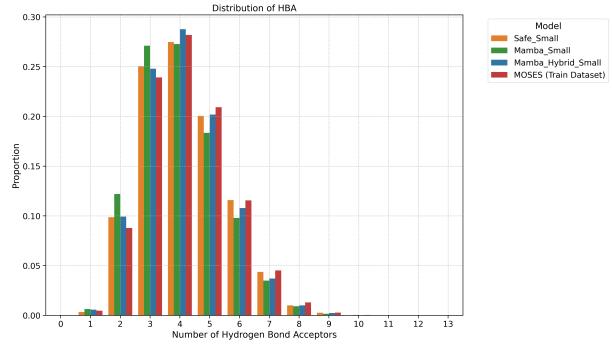
### 5.1 Comparative Performance in Molecular Generation

Addressing our first research question, the results demonstrate a remarkable parity in performance between State Space Models (MAMBA) and Transformer-based (SAFE) architectures across all evaluated metrics for molecular generation.

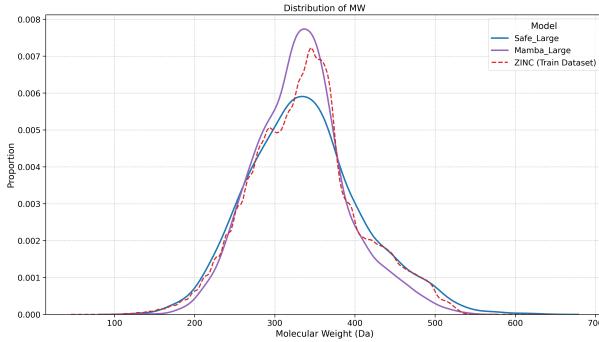
Both small (20M parameters) and large (90M parameters) models achieved high validity (98-100%) and uniqueness (99.9-100%) scores, regardless of the underlying architecture. This parity extends the application of State Space Models, previously shown effective in language tasks, to the complex domain of molecular generation. The ability of MAMBA-based models to match the performance of Transformer-based models suggests that State Space Models can effectively learn and represent the intricate patterns inherent in molecular structures, even without explicit attention mechanisms.



**Figure 7: Distribution of Molecular Weight for small models compared to MOSES dataset**



**Figure 8: Distribution of Hydrogen Bond Acceptors for small models compared to MOSES dataset**



**Figure 9: Distribution of Molecular Weight for large models compared to ZINC dataset**

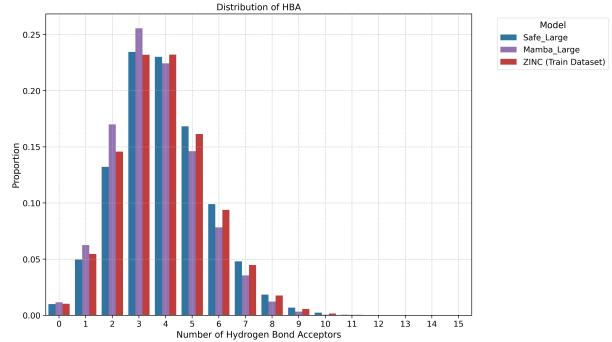
The comparable diversity scores across all models further reinforce the capability of SSMs to capture the multifaceted nature of molecular structures. This finding is particularly significant as it demonstrates that alternative approaches to sequence modeling can be equally effective in exploring vast chemical spaces, a crucial aspect of molecular generation tasks.

It is important to note that achieving these results with MAMBA models required adjusting the top-p parameter to 0.90, while SAFE models used the default value of 1.0. This difference in sampling strategy hints at fundamental differences in how SSMs and Transformers learn to represent the molecular space, warranting further investigation.

## 5.2 Efficiency Advantages of MAMBA Architecture

Our second research question focused on the potential efficiency advantages of the MAMBA architecture. The results reveal significant efficiency gains for MAMBA-based models compared to the Transformer-based SAFE models, particularly as model size increases.

MAMBA models consistently demonstrated lower GPU power consumption compared to SAFE models. This substantial reduction in computational resource requirements could prove crucial for scaling up to larger datasets or more complex molecular structures,



**Figure 10: Distribution of Hydrogen Bond Acceptors for large models compared to ZINC dataset**

potentially enabling the exploration of chemical spaces that were previously computationally infeasible.

Training time comparisons revealed an interesting trend. While small MAMBA models required slightly longer training times despite their lower resource utilization, this trend reversed for large models. The 90M parameter SAFE model took approximately 90 hours to train for 250,000 steps, while the equivalent MAMBA model completed the same training in only 64 hours. This observation suggests that the efficiency advantages of MAMBA models become more pronounced as model size increases, offering significant time savings for large-scale molecular generation tasks.

The MAMBA-Hybrid model, incorporating both SSM and attention layers, demonstrated a promising balance between efficiency and training speed. It maintained the low resource utilization characteristic of MAMBA while achieving faster training times than the pure MAMBA model, approaching those of the SAFE model at the small scale.

## 5.3 Perplexity and Model Behavior

An intriguing finding of our study is the consistently lower perplexity exhibited by MAMBA and MAMBA-Hybrid models throughout the training process, for both small and large model sizes. The 20M parameter MAMBA models achieved a perplexity of 1.4 compared

to 1.5 for the equivalent SAFE model, with this trend persisting for larger models as well (1.5 vs 1.3).

Lower perplexity suggests that MAMBA models have learned a more accurate probability distribution over the space of possible molecules. This efficiency in modeling could be attributed to the continuous-time dynamics of SSMs, which may be particularly well-suited to capturing the sequential nature of molecular structures.

However, the need for a lower top-p value during sampling with MAMBA models, despite their lower perplexity, highlights the complexity of interpreting model performance in molecular generation tasks. While MAMBA models seem to learn the molecular space more accurately, more selective sampling was required to maintain high validity. This suggests that when working with SSMs for molecular generation, fine-tuning of sampling strategies may be necessary to fully leverage the learned representations.

#### 5.4 Molecular Property Distributions

Our analysis of molecular property distributions revealed that both MAMBA and SAFE models effectively captured the characteristics of their respective training datasets (MOSES for small models, ZINC for large models). This ability to reproduce the distribution of physicochemical properties such as molecular weight, LogP, and hydrogen bond acceptors/donors demonstrates that both architectures can learn and generate molecules with realistic and diverse properties.

The close alignment of property distributions between generated molecules and the training data, observed across all model architectures and sizes, further reinforces the capability of State Space Models to capture complex molecular features without explicit attention mechanisms.

#### 5.5 Implications for AI-Driven Molecular Design

The comparable performance of MAMBA models to Transformer-based models, coupled with their efficiency advantages, has significant implications for the field of AI-driven molecular design. The potential for SSMs to handle longer molecular sequences efficiently opens up new possibilities for modeling complex macromolecules or entire chemical pathways, tasks that have traditionally been challenging due to the quadratic complexity of attention mechanisms.

These findings suggest that State Space Models, specifically the MAMBA architecture, offer a viable and efficient alternative to Transformer-based models for molecular generation tasks. The combination of comparable generation quality with improved computational efficiency positions SSMs as a promising approach for advancing the field of AI-driven molecular design, particularly for large-scale applications or when computational resources are limited.

## 6 CONCLUSIONS

Our study provides empirical validation for the efficacy of State Space Models, specifically the MAMBA architecture, in the complex task of molecular generation. By demonstrating comparable performance to Transformer-based models in generating valid, unique, and diverse molecules, we contribute to the growing body

of evidence suggesting that SSMs represent a viable alternative to attention-based architectures across diverse domains.

The success of MAMBA and MAMBA-Hybrid models in capturing the intricacies of molecular structures, as encoded in the SAFE representation, underscores the versatility of SSMs. This finding is particularly significant given the complexity of molecular generation tasks, which require models to learn and reproduce intricate patterns of atomic connections and chemical properties.

The marked efficiency advantage demonstrated by MAMBA-based models, evidenced by substantial reductions in GPU power consumption and improved training times for larger models, highlights a key strength of SSMs: their ability to process long sequences with linear time complexity. This characteristic could prove transformative in molecular generation tasks, where the exploration of vast chemical spaces is often constrained by computational resources.

Looking forward, several promising avenues for future research emerge from our findings:

- (1) Further scaling studies: While we have already explored scaling to 90M parameters, investigating the performance of even larger SSM-based models on more extensive molecular datasets could further leverage their efficiency advantages and potentially uncover new capabilities.
- (2) Application to extremely long molecules: Training MAMBA models on datasets containing exceptionally long molecular sequences could demonstrate their ability to capture dependencies at scales beyond the practical limits of Transformer models. This could open up new possibilities in modeling complex macromolecules or entire biochemical pathways.
- (3) Advanced hybrid architectures: Building upon our MAMBA-Hybrid model, there is potential to develop more sophisticated hybrid architectures that combine the strengths of SSMs and attention mechanisms. These models could be tailored to capture different aspects of molecular structure and behavior more effectively.
- (4) Integration with reinforcement learning: Developing approaches that guide MAMBA-based models to generate novel, valid molecules using reinforcement learning techniques could significantly contribute to drug discovery efforts. This could involve creating sophisticated reward functions that balance chemical validity, target properties, and synthetic accessibility.

In conclusion, our study not only validates the effectiveness of SSMs in the complex domain of molecular generation but also sets the stage for exciting future developments. The combination of comparable generation quality with improved computational efficiency positions SSMs as a promising approach for advancing the field of AI-driven molecular design. As researchers build upon these findings, we anticipate significant progress in our ability to explore and engineer molecular spaces.

## REFERENCES

- [1] G Richard Bickerton, Gaia V Paolini, J'er'emy Besnard, Sorel Muresan, and Andrew L Hopkins. 2012. Quantifying the chemical beauty of drugs. *Nature chemistry* 4, 2 (2012), 90–98.
- [2] Nathan Brown, Marco Fiscato, Marwin HS Segler, and Alain C Vaucher. 2019. GuacaMol: benchmarking models for de novo molecular design. *Journal of chemical information and modeling* 59, 3 (2019), 1096–1108.

- [3] Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint ArXiv:2005.14165* (2020).
- [4] Tri Dao, Daniel Y. Fu, Khaled Kamal Saab, A. Waldmann Thomas, Atri Rudra, and Christopher Ré. 2022. Hungry Hungry Hippos: Towards Language Modeling with State Space Models. *ArXiv abs/2212.14052* (2022). <https://api.semanticscholar.org/CorpusID:255340454>
- [5] Tri Dao and Albert Gu. 2024. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060* (2024).
- [6] Jörg Degen, Christof Wegscheid-Gerlach, Andrea Zaliani, and Matthias Rarey. 2008. On the Art of Compiling and Using ‘Drug-Like’ Chemical Fragment Spaces. *ChemMedChem* 3, 10 (2008), 1503–1507. <https://doi.org/10.1002/cmde.200800178>
- [7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs.CL]* <https://arxiv.org/abs/1810.04805>
- [8] Daniel C Elton, Zois Boukouvalas, Mark D Fuge, and Peter W Chung. 2019. Deep learning for molecular design—a review of the state of the art. *Molecular Systems Design Engineering* 4, 4 (2019), 828–849.
- [9] Peter Ertl, Bernhard Rohde, and Paul Selzer. 2000. Fast calculation of molecular polar surface area as a sum of fragment-based contributions and its application to the prediction of drug transport properties. *Journal of medicinal chemistry* 43, 20 (2000), 3714–3717.
- [10] Piero Gasparotto and Michele Ceriotti. 2014. Recognizing molecular patterns by machine learning: An agnostic structural definition of the hydrogen bond. *The Journal of chemical physics* 141, 17 (2014).
- [11] Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. 2018. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science* 4, 2 (2018), 268–276.
- [12] D Grechishnikova. 2021. Transformer neural network for protein-specific de novo drug generation as a machine translation problem. *Sci Rep* 11: 321.
- [13] Maarten Grootendorst. 2024. *A Visual Guide to Mamba and State Space Models*. <https://newsletter.maartengrootendorst.com/p/a-visual-guide-to-mamba-and-state>
- [14] Albert Gu and Tri Dao. 2023. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. *arXiv:2312.00752 [cs.LG]*
- [15] Albert Gu, Karan Goel, and Christopher Ré. 2021. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396* (2021).
- [16] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751* (2019).
- [17] Fred Jelinek, Robert L Mercer, Lalit R Bahls, and James K Baker. 1977. Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of Acoustical Society of America* 62, S1 (1977), S63–S63.
- [18] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. 2018. Junction tree variational autoencoder for molecular graph generation. In *International Conference on Machine Learning*. PMLR, 2323–2332.
- [19] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. 2020. Hierarchical generation of molecular graphs using structural motifs. In *International Conference on Machine Learning*. PMLR, 4839–4848.
- [20] Hyunseung Kim, Jonggeol Na, and Won Bo Lee. 2021. Generative chemical transformer: neural machine learning of molecular geometric structures from chemical language via attention. *Journal of chemical information and modeling* 61, 12 (2021), 5804–5814.
- [21] Mario Krenn, Florian Häse, AkshatKumar Nigam, Pascal Friederich, and Alan Aspuru-Guzik. 2020.
- [22] Greg Landrum et al. 2023. RDKit: Open-source cheminformatics. *Online*. <http://www.rdkit.org> (2023).
- [23] Christopher A Lipinski. 2004. Lead-and drug-like compounds: the rule-of-five revolution. *Drug discovery today: Technologies* 1, 4 (2004), 337–341.
- [24] Yiming Mo, Yanfei Guan, Pritha Verma, Jiang Guo, Mike E Fortunato, Zhaohong Lu, Connor W Coley, and Klavs F Jensen. 2021. Evaluating and clustering retrosynthesis pathways with learned strategy. *Chemical science* 12, 4 (2021), 1469–1478.
- [25] Emmanuel Noutahi, Cristian Gabellini, Michael Craig, Jonathan SC Lim, and Prudencio Tossou. 2023. Gotta be SAFE: A New Framework for Molecular Design. *arXiv preprint arXiv:2310.10773* (2023).
- [26] Daniil Polykovskiy, Alexander Zhebrak, Benjamin Sanchez-Lengeling, Sergey Golovanov, Oktai Tatanov, Stanislav Belyaev, Rauf Kurbanov, Aleksey Artamonov, Vladimir Aladinskiy, Mark Veselov, Artur Kadurin, Simon Johansson, Hongming Chen, Sergey Nikolenko, Alan Aspuru-Guzik, and Alex Zhavoronkov. 2020. Molecular Sets (MOSES): A Benchmarking Platform for Molecular Generation Models. *arXiv:1811.12823 [cs.LG]*
- [27] David Rogers and Mathew Hahn. 2010. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling* 50, 5 (2010), 742–754.
- [28] Petra Schneider, W Patrick Walters, Alleyn T Plowright, Norman Sieroka, Jennifer Listgarten, Robert A Goodnow, Johanna Fisher, J'org M Jansen, Jos'e S Duca, Thomas S Rush, et al. 2020. Rethinking drug design in the artificial intelligence era. *Nature Reviews Drug Discovery* 19, 5 (2020), 353–364.
- [29] Marwin HS Segler, Thierry Kogej, Christian Tyrchan, and Mark P Waller. 2018. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS central science* 4, 1 (2018), 120–131.
- [30] Teague Sterling and John J Irwin. 2015. ZINC 15-ligand discovery for everyone. *Journal of chemical information and modeling* 55, 11 (2015), 2324–2337.
- [31] Jonathan M Stokes, Kevin Yang, Kyle Swanson, Wengong Jin, Andres Cubillos-Ruiz, Nina M Donghia, Craig R MacNair, Shawn French, Lindsey A Carfrae, Zohar Bloom-Ackermann, et al. 2020. A deep learning approach to antibiotic discovery. *Cell* 180, 4 (2020), 688–702.
- [32] Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. 2022. Efficient Transformers: A Survey. *arXiv:2009.06732 [cs.LG]* <https://arxiv.org/abs/2009.06732>
- [33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762* (2017).
- [34] Daniel F Veber, Stephen R Johnson, Hung-Yuan Cheng, Brian R Smith, Keith W Ward, and Kenneth D Kopple. 2002. Molecular properties that influence the oral bioavailability of drug candidates. *Journal of medicinal chemistry* 45, 12 (2002), 2615–2623.
- [35] David Weininger. 1988. SMILES, a chemical language and information system. 1. introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* 28, 1 (1988), 31–36.
- [36] Scott A Wildman and Gordon M Crippen. 1999. Prediction of physicochemical parameters by atomic contributions. *Journal of chemical information and computer sciences* 39, 5 (1999), 868–873.
- [37] Alex Zhavoronkov, Yan A Ivanenkov, Alex Aliper, Mark S Veselov, Vladimir A Aladinskiy, Anastasiya V Aladinskaya, Victor A Terentiev, Daniil A Polykovskiy, Maksim D Kuznetsov, Arip Asadulaev, et al. 2019. Deep learning enables rapid identification of potent DDR1 kinase inhibitors. *Nature biotechnology* 37, 9 (2019), 1038–1040.

## A ADDITIONAL MOLECULAR PROPERTY DISTRIBUTIONS

This appendix presents the distributions of additional molecular properties for both small and large models, complementing the Molecular Weight and Hydrogen Bond Acceptors distributions discussed in the main text.

### A.1 Large Model Distributions

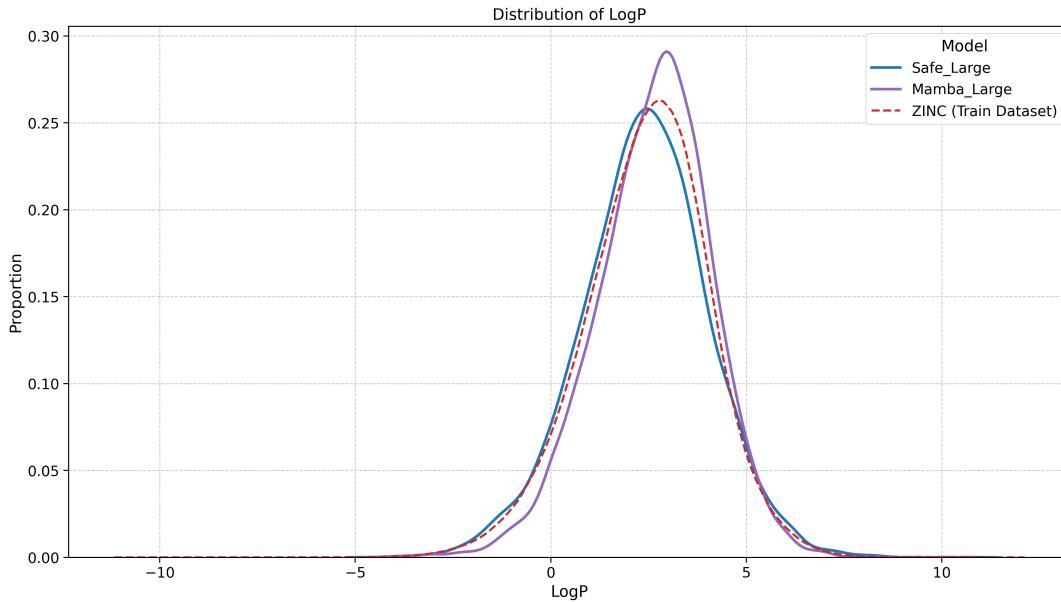


Figure 11: Distribution of LogP for large models

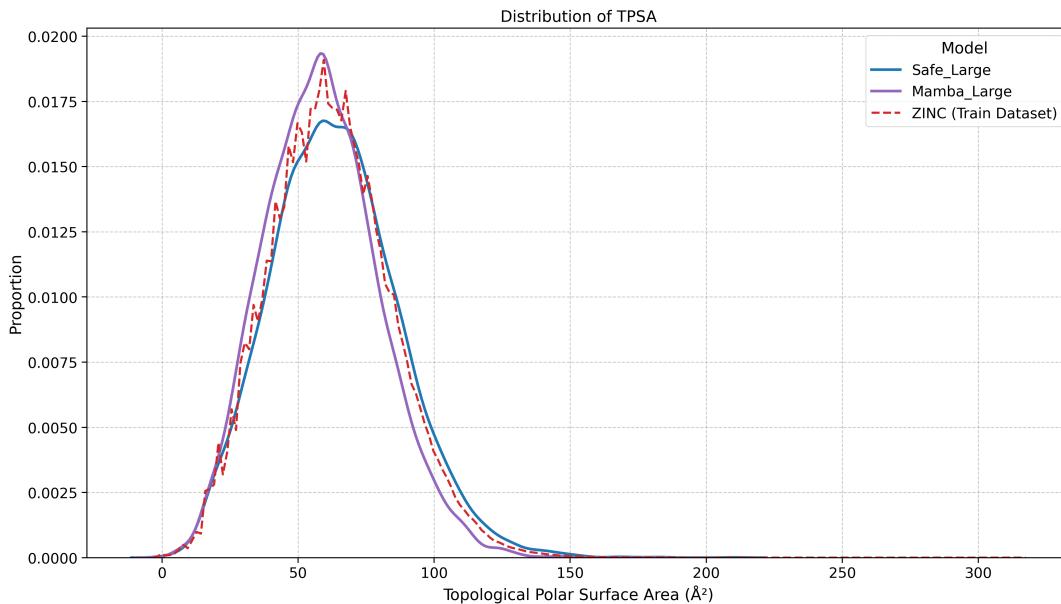


Figure 12: Distribution of Topological Polar Surface Area (TPSA) for large models

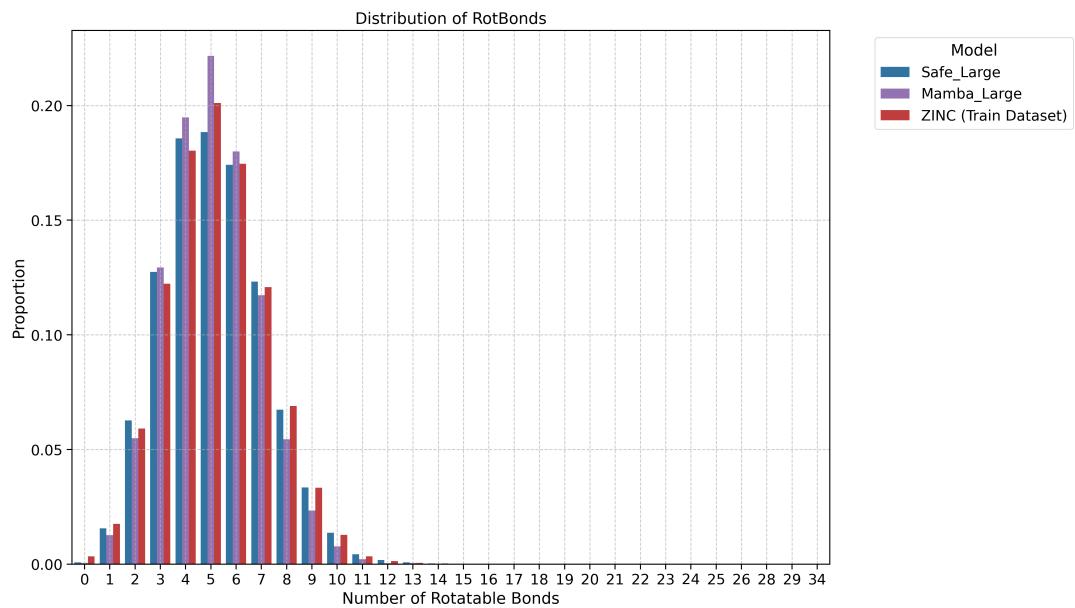


Figure 13: Distribution of Number of Rotatable Bonds for large models

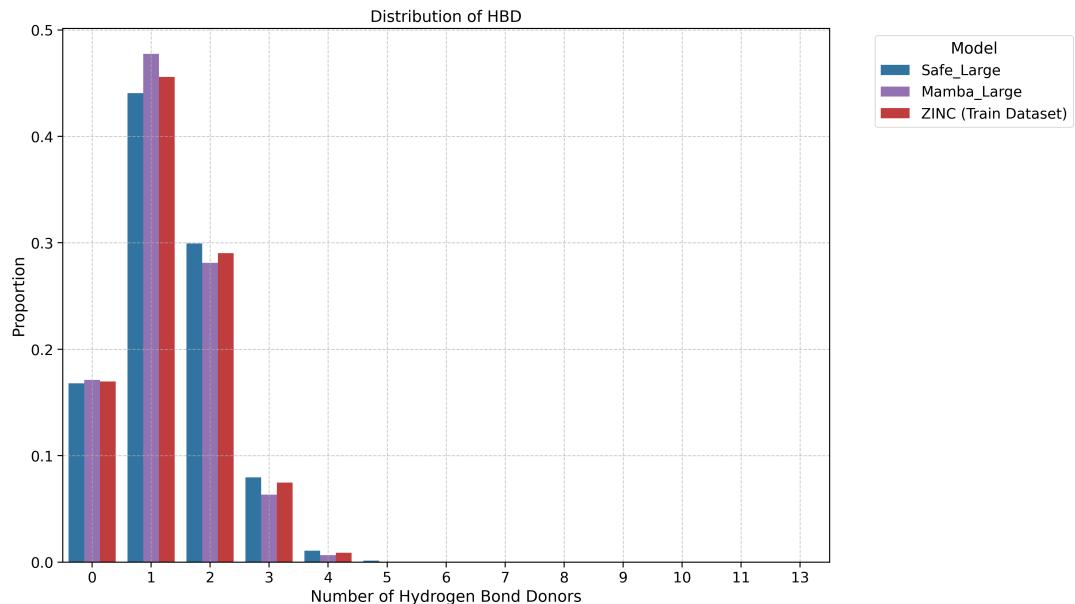
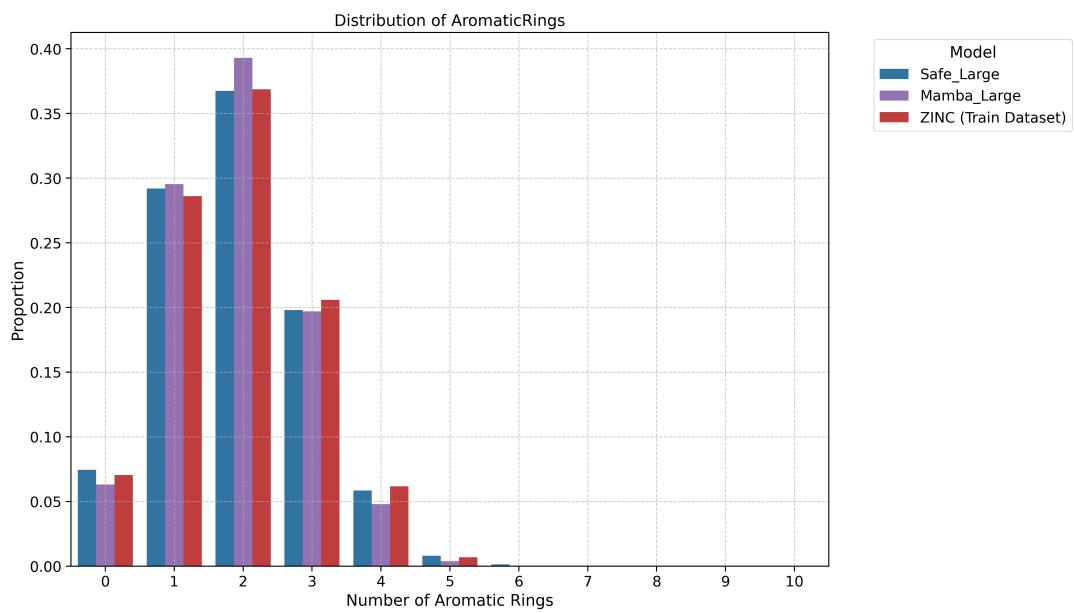
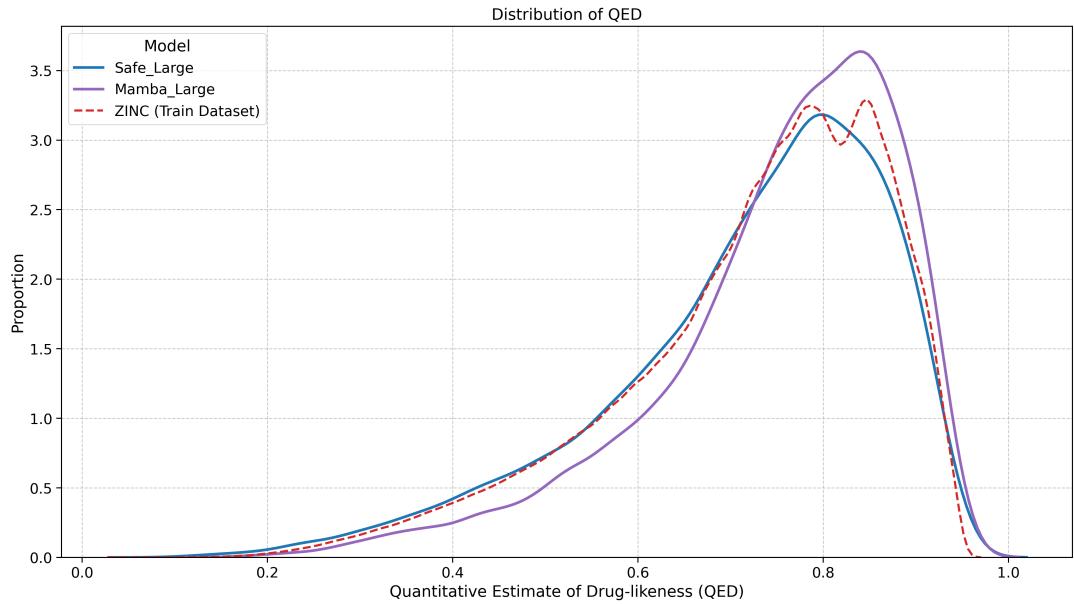


Figure 14: Distribution of Hydrogen Bond Donors (HBD) for large models

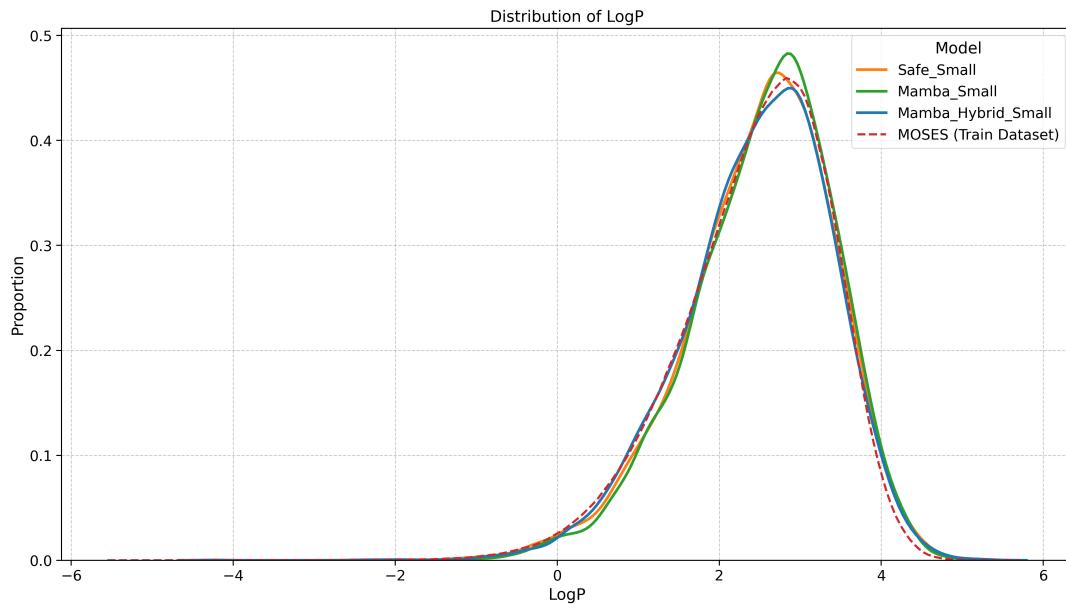


**Figure 15: Distribution of Number of Aromatic Rings for large models**

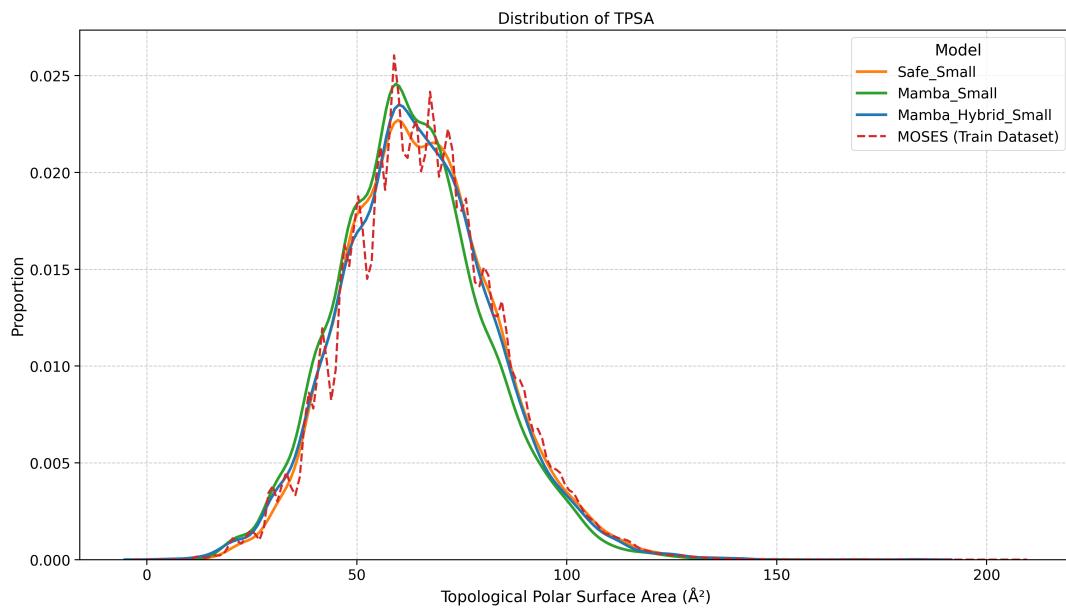


**Figure 16: Distribution of Quantitative Estimate of Drug-likeness (QED) for large models**

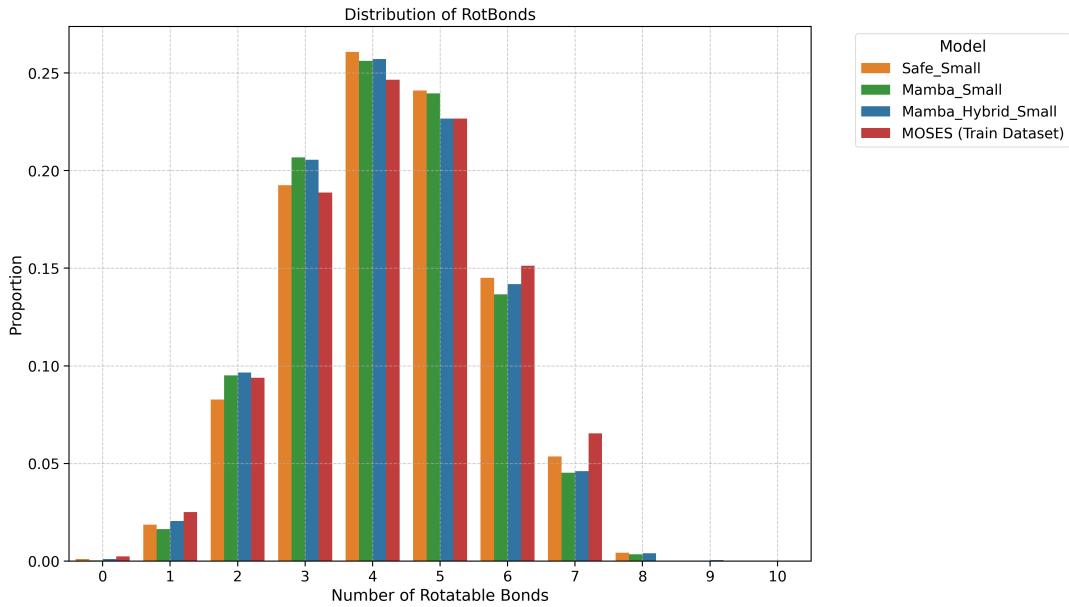
## A.2 Small Model Distributions



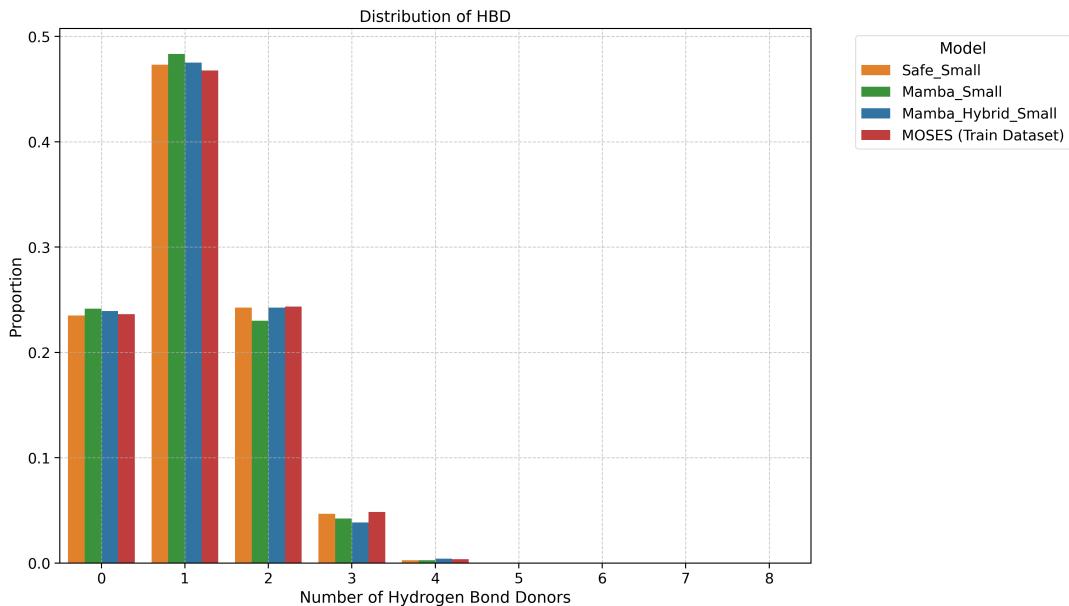
**Figure 17: Distribution of LogP for small models**



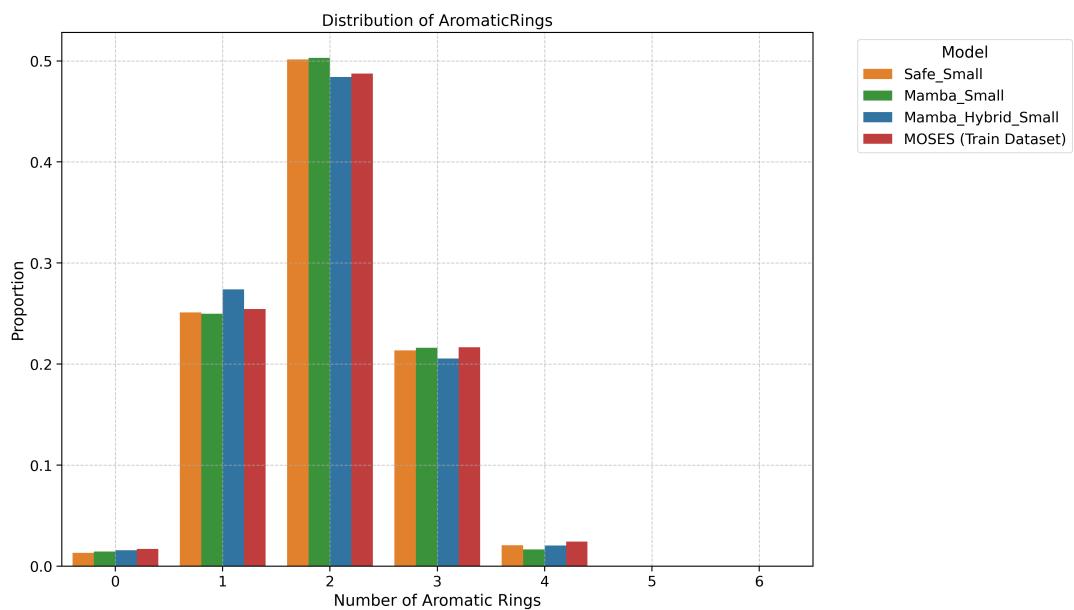
**Figure 18: Distribution of Topological Polar Surface Area (TPSA) for small models**



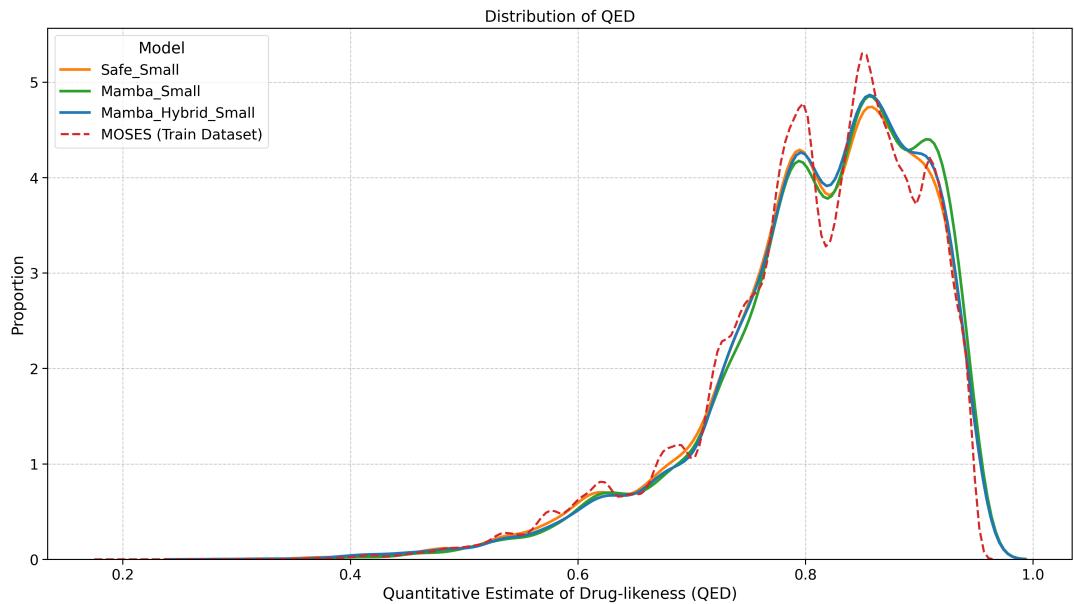
**Figure 19: Distribution of Number of Rotatable Bonds for small models**



**Figure 20: Distribution of Hydrogen Bond Donors (HBD) for small models**



**Figure 21: Distribution of Number of Aromatic Rings for small models**

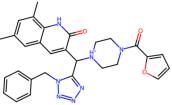
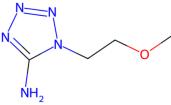
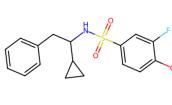
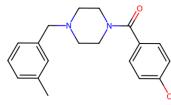


**Figure 22: Distribution of Quantitative Estimate of Drug-likeness (QED) for small models**

## B EXAMPLE MOLECULES

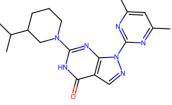
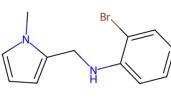
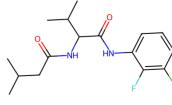
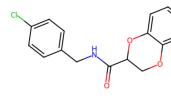
This appendix presents representative molecules generated by each model, showcasing the longest, shortest, most diverse, and highest QED molecules from the 10k generated.

**Mamba Large**

Longest	Shortest	Most Diverse	Highest QED
 <p><b>SMILES:</b>  <chem>Cc1cc(C)c2[nH]c(=O)c(C(c3nnnn3Cc3cccc3)(NH+)3CCN(C(=O)c4cccc4)CC3)cc21</chem></p> <p><b>SAFE:</b>  <chem>Cc1cc(C)c2[nH]c(=O)c%10cc2c1.[NH+]16CCN5CC1.c19nnn n18.c1%11cccc1.c17ccco1.C69%10.C57=O.C8%11</chem></p>	 <p><b>SMILES:</b>  <chem>COCC1nnnc1N</chem></p> <p><b>SAFE:</b>  <chem>n13nnnc1N.C2C3.CO2</chem></p>	 <p><b>SMILES:</b>  <chem>cOc1ccc(S(=O)(=O)NC(Cc2cccc2)C2CC2)cc1F</chem></p> <p><b>SAFE:</b>  <chem>c13ccc(S(=O)(=O)cc1F.c17cccc1.C46C7.C16CC1.CO3.N4 5</chem></p>	 <p><b>SMILES:</b>  <chem>Cc1cccc(CN2CCN(C(=O)c3ccc(O)cc3)CC2)c1</chem></p> <p><b>SAFE:</b>  <chem>Cc1cccc7c1.N15CCN4CC1.c16ccc(O)cc1.C46=O.C57</chem></p> <p><b>QED:</b> 0.9477</p>

**Figure 23: Representative molecules generated by the Mamba\_Large model**

**Mamba Small**

Longest	Shortest	Most Diverse	Highest QED
 <p><b>SMILES:</b>  <chem>Cc1cc(C)c2[nH]c(=O)[nH]c(N4CCCC(C(C)C)C4)nc32)n1</chem></p> <p><b>SAFE:</b>  <chem>n17nc2c(=O)[nH]c5nc21.Cc1cc(C)nc7n1.N15CCCC6C1.C6 (C)C</chem></p>	 <p><b>SMILES:</b>  <chem>Cn1cccc1CNc1cccc1Br</chem></p> <p><b>SAFE:</b>  <chem>c13cccc1Br.Cn1cccc14.C24.N23</chem></p>	 <p><b>SMILES:</b>  <chem>CC(C)CC(=O)NC(C(=O)NC(c1cccc1Cl)c1F)C(C)C</chem></p> <p><b>SAFE:</b>  <chem>c15cccc(Cl)c1F.C4(C3=O)C(C)C.CC(C)CC2=O.N24.N35</chem></p>	 <p><b>SMILES:</b>  <chem>O=C(NCc1ccc(Cl)cc1)C1COc2cccc2O1</chem></p> <p><b>SAFE:</b>  <chem>C15COc2cccc2O1.c16ccc(Cl)cc1.O=C35.N34.C46</chem></p> <p><b>QED:</b> 0.9483</p>

**Figure 24: Representative molecules generated by the Mamba\_Small model**

## Mamba Small Hybrid

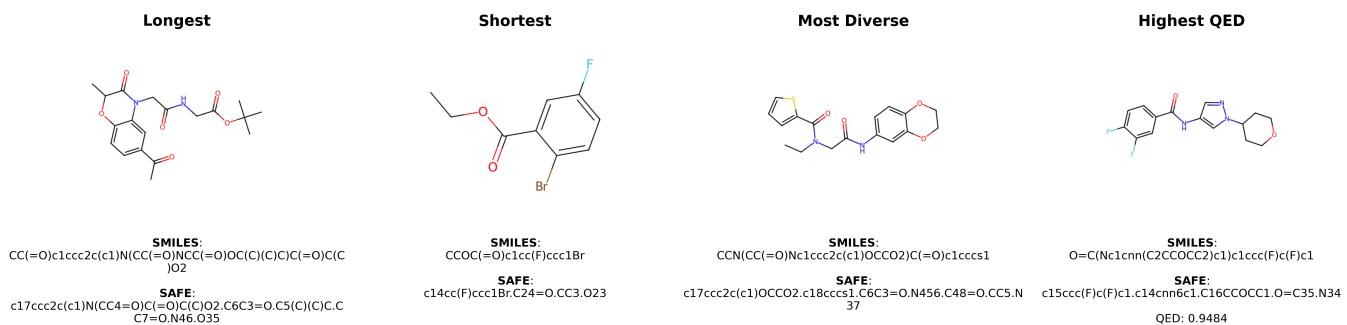


Figure 25: Representative molecules generated by the Mamba\_Small\_Hybrid model

## SAFE Large

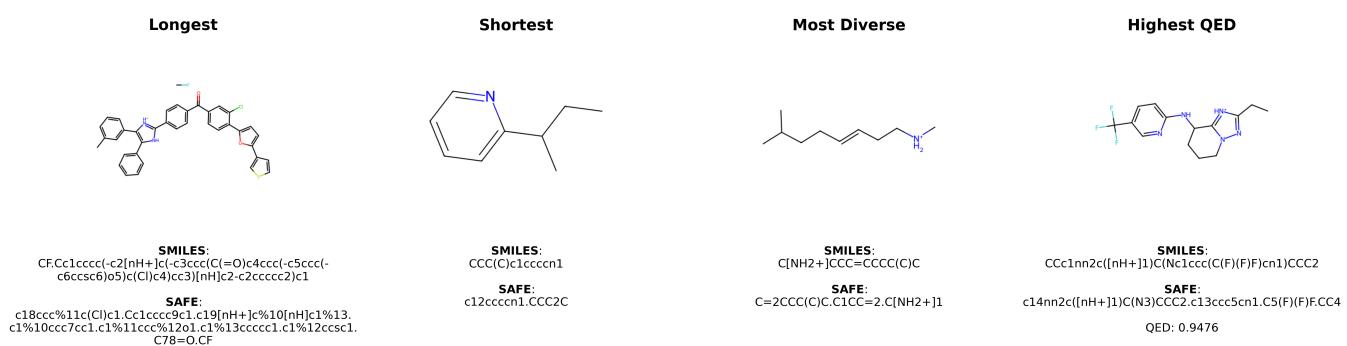


Figure 26: Representative molecules generated by the SAFE\_Large model

## SAFE Small

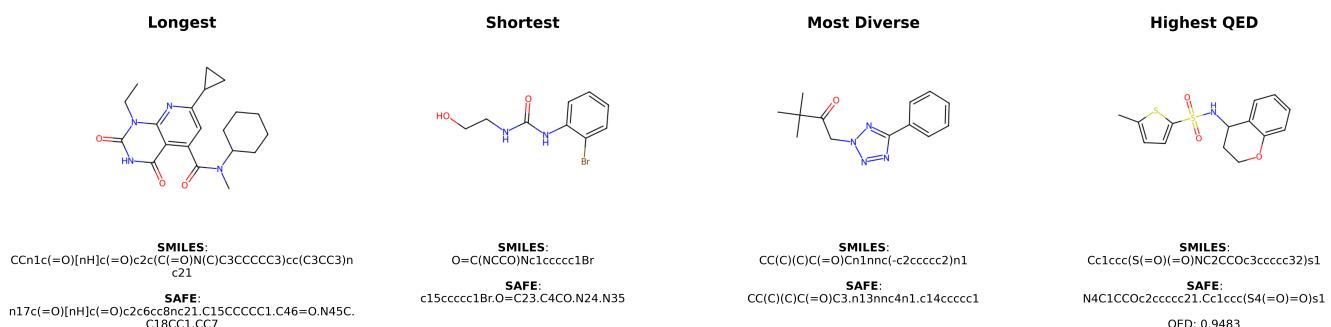


Figure 27: Representative molecules generated by the SAFE\_Small model