Comparative Performance of State Space and Transformer Models for Molecule Generation

Anri Lombard
University of Cape Town
Cape Town, South Africa
LMBANR001@myuct.ac.za

Abstract

Generative models for molecule generation have emerged as a promising approach to accelerate the discovery of novel compounds with desirable properties. This review provides a comprehensive overview of the current state-of-the-art in Transformer-based generative models and introduces State Space Models (SSMs) as a potential alternative. We discuss the impact of molecular string representations, such as SMILES, SELFIES, and SAFE, on the performance and interpretability of generative models. Furthermore, we highlight the strengths and limitations of Transformers and SSMs in capturing molecular structures and properties. We also compare evaluation metrics for assessing the quality and practicality of machine-generated molecules. By critically analyzing the current landscape and identifying promising research directions, this review aims to contribute to the advancement of molecule generation and accelerate the development of novel therapeutic agents and materials.

1 Introduction and Motivation

The discovery of novel molecules with desired properties is a fundamental challenge in drug discovery, materials science, and chemical engineering. Traditional methods, such as high-throughput screening and iterative synthesis, are time-consuming and resource-intensive, limiting their ability to efficiently explore the vast chemical space [12, 20]. Machine learning-based generative models have emerged as a promising approach to accelerate the discovery and design of novel molecules with targeted properties, potentially revolutionizing the way we develop new therapeutic agents and functional materials [31, 32].

Generative models learn the underlying distribution of molecules in a given chemical space and generate novel compounds that exhibit desired properties [31]. The development of these models has been facilitated by the availability of large-scale molecular datasets [14, 19] and advances in molecular representation learning [15, 26]. Among the various architectures employed for molecule generation, Transformers [34] have gained significant attention due to their ability to capture long-range dependencies and generate diverse molecular structures. Transformer-based models, such as MoLeR [27] and LigGPT [1], have demonstrated impressive performance in generating novel molecules with optimized properties.

However, Transformers face challenges in handling long sequences and ensuring the stability and synthesizability of generated molecules [21]. The quadratic complexity of self-attention limits their scalability to very long sequences, hindering the generation of complex molecules with many atoms and bonds. Additionally, the interpretability and controllability of Transformer-based models are crucial for their adoption in real-world applications, as domain experts need to understand and guide the generation process [20].

Recent advancements in State Space Models (SSMs) [16, 17] have shown promise in efficiently modeling long sequences and capturing complex temporal dependencies, making them a potential alternative to Transformers for molecular generation tasks. SSMs offer linear time complexity with respect to sequence length and have demonstrated strong performance in various sequence modeling tasks [16]. However, their application to molecule generation remains largely unexplored, and their ability to capture the complex spatial and geometric constraints of molecular structures requires further investigation.

In this review, we aim to provide a comprehensive overview of the current state-of-the-art in generative models for molecule generation, focusing on Transformer-based models and their applications. We will discuss the importance of molecular string representations, such as SMILES [35], SELFIES [23], and the recently proposed SAFE [28], and their impact on the performance and interpretability of generative models. Furthermore, we will introduce SSMs and their variants, discussing their potential advantages and limitations in the context of molecular generation.

The main objectives of this review are:

1. To critically analyze the strengths and limitations of Transformer-based models in generating diverse and novel molecules with desired properties.
2. To explore the potential of SSMs as an alternative to Transformers, investigating their ability to handle long sequences and capture complex molecular substructures.
3. To assess the impact of molecular string representations on the performance and interpretability of generative models, focusing on SMILES, SELFIES, and SAFE.

4. To identify promising research directions and propose strategies for advancing the field of molecule generation, addressing the limitations of existing models and enhancing their practical applicability.

The significance of this research lies in its potential to guide the development of more effective and efficient generative models for molecular design. By conducting a fair and rigorous comparison of Transformer-based models and SSMs, and investigating the impact of molecular string representations, we aim to provide valuable insights into the strengths and limitations of each approach. The findings of this review can inform the design of novel architectures and techniques that address the challenges of existing models, ultimately accelerating the discovery of innovative therapeutic agents and functional materials.

To achieve these objectives, we propose training Transformer-based models and SSMs from scratch on the SAFE-GPT dataset [28], a large-scale collection of over 1 billion molecules, and evaluating their performance on various molecular generation tasks. By leveraging this comprehensive dataset and conducting a systematic comparison, we aim to provide a solid foundation for future research in this domain.

The remainder of this review is organized as follows: Section 2 provides a background on molecular string representations and sequence models, including Transformers and SSMs. Section 3 discusses the datasets used for training and evaluating generative models. Section 4 presents a comparative analysis of Transformers and SSMs in molecule generation, highlighting their strengths and limitations. Finally, Section 5 concludes the review and outlines future research directions.

2 Background

2.1 String Representations for Molecules

Molecular string representations play a crucial role in the development of generative models for molecular design. These representations encode the structural information of molecules as linear sequences of characters, enabling the application of sequence-based machine learning models. The choice of string representation can significantly impact the performance and capabilities of generative models.

2.1.1 Importance of String Representations. Molecules are composed of atoms connected by chemical bonds, forming intricate graphs. However, most machine learning models, particularly sequence-based models like Transformers, require input data to be in a linear format. String representations bridge this gap by converting the graphical structure of molecules into a sequence of characters that can be processed by these models.

2.1.2 SMILES Representation. The most widely adopted string representation is the Simplified Molecular-Input Line-Entry System (SMILES) [35]. SMILES strings encode the molecular structure using a combination of characters and rules for connectivity. For example, the molecule in Figure 1 is represented by the SMILES string "O=C(C#CCN1CCCCC1)Nc1ccc2ncnc(Nc3cccc(Br)c3)c2c1". While SMILES is compact and human-readable, it has several limitations. SMILES strings are not robust to minor changes in the molecular structure and may produce invalid or unintended molecules when modified. Furthermore, SMILES does not provide a consistent representation of molecular substructures, making it challenging to perform tasks like scaffold decoration or fragment linking.

2.1.3 SELFIES Representation. Self-Referencing Embedded Strings (SELFIES) [23] addresses some limitations of SMILES by introducing a robust representation that guarantees the validity of generated molecules. SELFIES achieves this by using a set of derivation rules that enforce chemical constraints. However, SELFIES strings can be less human-readable and may not capture the inherent structure of molecules as effectively as other representations.

2.2 SAFE Representation

A recent development in molecular string representations is the Sequential Attachment-based Fragment Embedding (SAFE) [28]. SAFE represents molecules as an unordered sequence of interconnected fragment blocks while maintaining compatibility with SMILES parsers. As shown in Figure 1, SAFE strings provide a more interpretable and consistent representation of molecular substructures compared to SMILES. This enables SAFE to excel in various molecular design tasks, as highlighted in Table 1.

The SAFE algorithm converts a SMILES string to a SAFE representation through the following steps:

1. Extract all unique ring digits from the SMILES string.
2. Fragment the molecule on specified bonds (e.g., using the BRICS algorithm).
3. Sort the fragments by size in descending order.
4. Concatenate the SMILES strings of the fragments, separating each fragment with a dot character (".").
5. Extract the attachment points from the concatenated string.
6. Replace the attachment points with new ring digits, starting from the next available digit after the maximum ring digit found in step 1.

Let's consider an example molecule with the SMILES string "c1cc2ccc1OC2". Here's how the SAFE algorithm converts this SMILES string to a SAFE representation:

1. The unique ring digits in the SMILES string are 1 and 2.
2. The molecule is fragmented using the BRICS algorithm, resulting in two fragments: "c1cc([ ])ccc1" and "O2C".
3. The fragments are sorted by size: "c1cc([ ])ccc1" and "O2C".

Task	SAFE	SMILES	Deep/Gen SMILES	SELFIES	Group SELFIES	InChi	GRAPHS
De novo design	✓	✓	✓	✓	✓	?	✓
Linker design	✓	?	×	×	?	×	?
Motif extension	✓	?	×	?	?	×	✓
Scaffold decoration	✓	?	×	×	?	×	✓
Scaffold morphing	✓	×	×	×	?	×	?
Super structure	✓	×	×	×	?	×	✓

Table 1. Pure generative capabilities of various molecular representations. Adapted from [28].

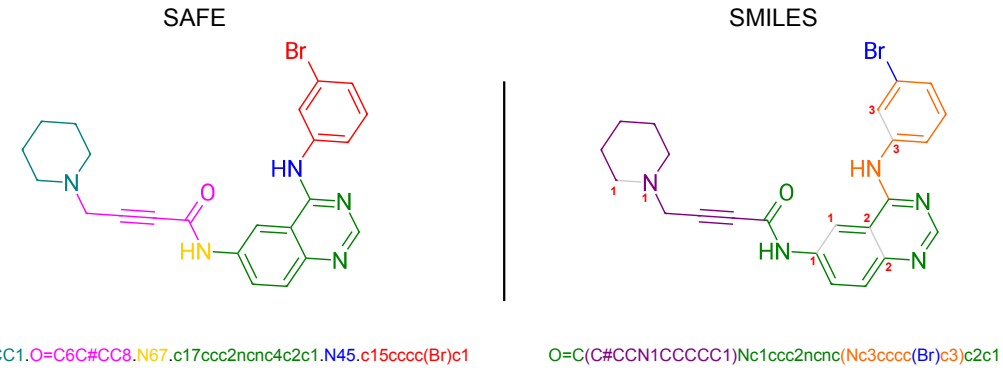

Figure 1. Example of a molecule represented as a SAFE string and a SMILES string. The colored fragments and their corresponding placement in each string demonstrate how the ordering of the fragments in the SAFE representation is more easily readable and interpretable than the comparable SMILES string. Adapted from [28].

4. The SMILES strings of the fragments are concatenated with a dot character: "c1cc([ ])ccc1.O2C".
5. The attachment points are extracted: "c1cc(1)ccc1.O2C".
6. The attachment points are replaced with the next available ring digit (3): "c1cc(3)ccc1.O2C3".

The resulting SAFE representation, "c1cc(3)ccc1.O2C3", clearly separates the fragments and uses virtual connections (ring digits) to indicate their attachment points. This representation facilitates tasks like fragment-based molecular design, as the fragments are easily identifiable and their connectivity is explicitly defined.

By representing molecules as a sequence of interconnected fragments, SAFE enables more efficient and interpretable molecular design compared to traditional SMILES strings.

2.2.1 Comparison of Generative Capabilities.

Table 1 compares the generative capabilities of SAFE with other molecular representations across different tasks. These tasks include:

- **De novo design**: Generating entirely new molecules from scratch.
- **Linker design**: Generating molecular fragments that connect two or more existing fragments.
- **Motif extension**: Extending a given molecular motif by adding new atoms or fragments.
- **Scaffold decoration**: Adding functional groups or substituents to a predefined molecular scaffold.
- **Scaffold morphing**: Modifying the core scaffold of a molecule while preserving its key structural features.
- **Super structure generation**: Generating new molecules that contain a specified substructure.

As evident from Table 1, SAFE outperforms other string representations in most of these tasks. The unordered fragment-based representation of SAFE allows generative models to efficiently perform complex molecular design tasks that are challenging with traditional SMILES-based approaches. This highlights the potential of SAFE as a powerful representation for driving innovation in de novo molecular design and drug discovery.

2.3 Overview of Sequence Models

Sequence models have evolved to capture dependencies and patterns within sequential data. Early models, such as Recurrent Neural Networks (RNNs), introduced the concept of hidden states to capture information from previous time steps [11]. However, RNNs suffered from the vanishing gradient problem, limiting their ability to capture long-range dependencies [2]. Long Short-Term Memory (LSTM) [18] and Gated Recurrent Units (GRU) [7] addressed this limitation by introducing gating mechanisms to control the flow of information, allowing for better preservation of long-term dependencies.

2.3.1 RNN Mathematics and Intuition.

The hidden state $\mathbf{h}_t$ of an RNN at time step t is computed based on the current input $\mathbf{x}_t$ and the previous hidden state $\mathbf{h}_{t-1}$:

$$\mathbf{h}_t = \tanh(\mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{W}_{xh}\mathbf{x}_t), \tag{1}$$

where $\mathbf{W}_{hh}$ and $\mathbf{W}_{xh}$ are weight matrices learned during training, and tanh is the hyperbolic tangent activation function. The hidden state acts as a summary of the information seen so far in the sequence, allowing the model to capture dependencies between the current input and previous inputs.

However, the vanishing gradient problem arises when the gradient signal becomes increasingly small as it propagates back through time, making it difficult for the model to learn long-range dependencies. This limitation motivated the development of more advanced architectures like LSTMs and GRUs, which introduce gating mechanisms to selectively retain or forget information from previous time steps.

2.3.2 Transformers as Sequence Models.

Transformers [34] have revolutionized sequence modeling by relying solely on attention mechanisms to capture dependencies between input tokens. This allows for parallel processing of input sequences, making Transformers computationally efficient and scalable.

The core of the Transformer architecture is the self-attention mechanism. Self-attention allows each token in the input sequence to attend to all other tokens, enabling the model to capture both short-range and long-range dependencies. Given an input sequence $\mathbf{X} \in \mathbb{R}^{n \times d}$, where n is the sequence length and d is the embedding dimension, the self-attention mechanism computes three matrices: the query matrix $\mathbf{Q}$, the key matrix $\mathbf{K}$, and the value matrix $\mathbf{V}$, as follows:

$$\mathbf{Q} = \mathbf{X}\mathbf{W}^Q, \quad \mathbf{K} = \mathbf{X}\mathbf{W}^K, \quad \mathbf{V} = \mathbf{X}\mathbf{W}^V, \tag{2}$$

where $\mathbf{W}^Q$, $\mathbf{W}^K$, and $\mathbf{W}^V$ are learned projection matrices. The attention weights are then computed using the scaled dot-product between the query and key matrices:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}, \tag{3}$$

where d_k is the dimensionality of the key vectors. The resulting attention output captures the weighted sum of the value vectors, allowing the model to focus on the most relevant information for each token.

Intuitively, the self-attention mechanism allows each token to query the importance of other tokens in the sequence, with the attention weights determining the strength of the relationships between tokens. This enables Transformers to capture complex dependencies and generate contextually relevant outputs.

To preserve the order of the input sequence, Transformers introduce positional encoding, which adds a unique vector to each token based on its position. The positional encoding allows the model to distinguish between tokens at different positions and capture positional information.

2.3.3 State Space Models (SSMs) as Sequence Models.

State Space Models (SSMs) have emerged as a promising alternative to Transformers for efficient sequence modeling. SSMs are based on the principles of control theory and offer a mathematically grounded approach to modeling dynamical systems [22]. The key idea behind SSMs is to represent the system's dynamics using a set of latent states that evolve over time according to a set of transition matrices.

In the context of sequence modeling, SSMs aim to learn a compact representation of the input sequence in a continuous-time state space. The state space representation allows SSMs to capture long-range dependencies efficiently, as the model can access information from any point in the past through the latent states. The state space equations for an SSM can be written as follows:

$$\begin{aligned} \mathbf{z}(t) &= \mathbf{A}\mathbf{z}(t-1) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{z}(t) + \mathbf{D}\mathbf{u}(t), \end{aligned} \tag{4}$$

where $\mathbf{z}(t) \in \mathbb{R}^d$ is the latent state at time t, $\mathbf{u}(t)$ is the input at time t, $\mathbf{y}(t)$ is the output at time t, and $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ are the transition matrices.

Intuitively, the latent state $\mathbf{z}(t)$ captures the relevant information from the past inputs, while the transition matrix $\mathbf{A}$ determines how this information is updated over time. The input matrix $\mathbf{B}$ determines how the current input affects the latent state, and the output matrix $\mathbf{C}$ maps the latent state to the output. By learning appropriate transition matrices, SSMs can model complex temporal dependencies and generate meaningful outputs.

The advantages of SSMs include their ability to handle long sequences efficiently and their linear time complexity with respect to the sequence length. However, the linear dynamics of SSMs may not be expressive enough to capture highly nonlinear and complex patterns in the data, and the interpretability of the learned latent states and transition matrices can be challenging.

Various extensions and variants of SSMs, such as the Structured State Space (S4) model [17], the Mamba architecture [16], and the Griffin architecture [9], have been proposed to address these limitations and enhance the expressiveness and adaptability of SSMs.

2.4 Advancements in SSM Architectures

Recent advancements in SSM architectures have further enhanced their expressiveness and efficiency. The Mamba architecture [16] introduces several key innovations to improve the performance of SSMs. Mamba incorporates an input-dependent gating mechanism that allows the model to adaptively modulate the SSM dynamics based on the current

input. The gating mechanism is defined as follows:

$$\begin{aligned} \mathbf{g}(t) &= \sigma(\mathbf{W}_g \mathbf{u}(t) + \mathbf{b}_g), \\ \mathbf{A}(t) &= \mathbf{g}(t) \odot \mathbf{A} + (\mathbf{1} - \mathbf{g}(t)) \odot \mathbf{I}, \end{aligned} \tag{5}$$

where $\mathbf{g}(t)$ is the gating vector, $\sigma(\cdot)$ is the sigmoid function, $\mathbf{W}_g$ and $\mathbf{b}_g$ are learned parameters, and $\odot$ denotes element-wise multiplication. This gating mechanism enables Mamba to selectively update or retain information from the past, enhancing its ability to capture complex temporal patterns.

Another notable feature of Mamba is the use of parallel scan operations for efficient computation. By replacing the traditional convolutional computation in SSMs with a parallel scan, Mamba achieves significant speedups and memory savings, making it scalable to large-scale sequence modeling tasks.

The Griffin architecture [9] takes a different approach by combining SSM layers with local attention. Griffin introduces the Real-Gated Linear Recurrent Unit (RG-LRU), an improved gating mechanism that allows the model to flexibly retain or discard past information based on the current input. The RG-LRU is defined as follows:

$$\begin{aligned} \mathbf{r}(t) &= \sigma(\mathbf{W}_r \mathbf{u}(t) + \mathbf{b}_r), \\ \mathbf{z}(t) &= \mathbf{r}(t) \odot \mathbf{A}\mathbf{z}(t-1) + (\mathbf{1} - \mathbf{r}(t)) \odot \mathbf{u}(t), \end{aligned} \tag{6}$$

where $\mathbf{r}(t)$ is the gating vector, $\mathbf{W}_r$ and $\mathbf{b}_r$ are learned parameters. Additionally, Griffin employs a hybrid architecture that alternates RG-LRU layers with local sliding window attention layers, enabling the model to capture both local and global context.

The hybrid approach in Griffin draws inspiration from the Transformer-XL model [8], which introduces a segment-level recurrence mechanism to capture long-range dependencies. By combining the strengths of SSMs and local attention, Griffin aims to achieve the best of both worlds: the efficiency of SSMs in modeling long sequences and the expressiveness of attention in capturing local patterns.

These advancements in SSM architectures have led to significant performance improvements on various sequence modeling tasks. Mamba and Griffin have demonstrated state-of-the-art performance on benchmarks such as language modeling, time-series forecasting, and speech recognition, often outperforming larger Transformer models while being more computationally efficient.

2.4.1 Application of Transformers in Molecule Generation. Transformers have been successfully applied to the task of molecule generation, leveraging their ability to learn complex patterns and generate diverse molecular structures. One of the key challenges in applying Transformers to molecule generation is representing molecules as sequences. Various string-based representations, such as SMILES [35], SELFIES [23], and SAFE [28], have been used to encode molecular structures as sequences of tokens.

Early works, such as the Molecular Transformer [33], demonstrated the effectiveness of Transformers in predicting chemical reactions. Subsequent works, such as MoLeR [27] and LigGPT [1], adapted Transformer-based models specifically for molecule generation. These models showcased the ability to generate diverse and novel molecules, outperforming traditional variational autoencoder (VAE) based approaches.

LigGPT, in particular, introduced a masked self-attention mechanism to capture long-range dependencies between SMILES tokens. This allowed the model to generate molecules with desired properties by conditioning on target values of various physicochemical descriptors. LigGPT also demonstrated the ability to perform scaffold-based generation, where the model optimizes molecular properties while maintaining a specific scaffold structure.

The SAFE-GPT model [28] further advanced the field by leveraging the SAFE representation, which encodes molecules as an unordered sequence of fragment blocks. By training on a large dataset of SAFE strings, SAFE-GPT showcased strong performance in fragment-constrained generation tasks and goal-directed optimization of molecular properties.

Despite the promising results, Transformer-based models for molecule generation still face challenges, such as handling long SMILES sequences and ensuring the stability and synthesizability of generated molecules. The quadratic complexity of self-attention limits the scalability of Transformers to very long sequences, which can hinder the generation of complex molecules with many atoms and bonds. Efforts to address these limitations include the development of sparse attention mechanisms [6] and the use of hierarchical representations [21].

2.4.2 Potential of SSMs in Molecule Generation. The success of SSMs in modeling long sequences and capturing complex temporal dependencies makes them a promising approach for molecule generation. The linear time complexity of SSMs allows them to scale to large molecules with many atoms and bonds without the quadratic overhead of self-attention in Transformers. Moreover, the continuous-time dynamics of SSMs can potentially capture the temporal aspects of molecular formation and evolution, such as the order and timing of bond formation and breaking events in chemical reactions.

The gating mechanisms introduced in SSM architectures like Mamba and Griffin can further enhance their expressiveness in modeling molecular sequences by allowing the model to adaptively modulate the SSM dynamics based on the current input. This adaptive behavior can be particularly useful in generating molecules with specific functional groups or substructures. However, the ability of SSMs to effectively capture the complex spatial and geometric constraints present in molecular structures remains to be explored, and techniques

for interpreting and controlling the generation process in SSMs need to be developed.

2.5 Evaluation Metrics for Generated Molecules

Evaluating the quality and practicality of machine-generated molecules is a critical aspect of developing generative models for molecular design. Several metrics have been proposed to assess the performance of these models, each focusing on different aspects of the generated molecules.

2.5.1 Validity, Uniqueness, and Diversity. The most fundamental metrics for evaluating generative models are validity, uniqueness, and diversity. Validity measures the percentage of chemically valid structures according to a molecular parsing tool such as RDKit [24]. Given a set of generated molecules G, the validity is defined as:

$$\text{Validity}(G) = \frac{|m \in G : \text{is\_valid}(m)|}{|G|} \times 100 \tag{7}$$

where is_valid$(\cdot)$ is a function that determines the chemical validity of a molecule using a molecular parsing tool.

Uniqueness refers to the fraction of non-duplicate molecules within a set of generated compounds. It is calculated as:

$$\text{Uniqueness}(G) = \frac{|\text{unique}(G)|}{|G|} \times 100 \tag{8}$$

where unique$(\cdot)$ returns the set of unique molecules in G.

Diversity, often quantified using the average pairwise Tanimoto distance between molecules based on their fingerprint representations (e.g., ECFP4) [30], assesses the chemical space coverage of the generated set. Given a set of generated molecules G and a fingerprint function $f(\cdot)$, the diversity is defined as:

$$\text{Diversity}(G) = \frac{2}{|G|(|G|-1)} \sum_{i=1}^{|G|} \sum_{j=i+1}^{|G|} \text{Tanimoto}(f(m_i), f(m_j)), \tag{9}$$

where $m_i, m_j \in G$ and Tanimoto$(\cdot,\cdot)$ is the Tanimoto similarity between two fingerprint representations.

These metrics provide a basic assessment of the generative models' performance and are widely reported in the literature. However, they do not capture the models' ability to generate molecules with specific desired properties, such as druglikeness or synthetic accessibility.

2.5.2 Quantitative Estimate of Druglikeness (QED). The Quantitative Estimate of Druglikeness (QED) [3] is a novel metric that quantifies the druglikeness of a molecule by integrating eight physicochemical properties into a single desirability score. These properties include molecular weight (MW), octanol-water partition coefficient (ALOGP), number of hydrogen bond donors (HBD), number of hydrogen bond acceptors (HBA), molecular polar surface area (PSA), number of rotatable bonds (ROTB), number of aromatic rings (AROM), and the presence of structural alerts (ALERTS).

The desirability functions for each property p_i are derived empirically by fitting asymmetric double sigmoidal functions to the distribution of these properties in a set of approved oral drugs:

$$d(p_i) = \frac{1}{1+\exp[-a_i(p_i-b_i)]} \times \frac{1}{1+\exp[-c_i(p_i-d_i)]}, \tag{10}$$

where a_i, b_i, c_i, and d_i are the fitted parameters for property p_i.

The individual desirability scores are then combined into a single QED value using a weighted geometric mean:

$$\text{QED} = \left(\prod_{i=1}^{n} d(p_i)^{w_i} \right)^{\frac{1}{\sum_{i=1}^{n} w_i}}, \tag{11}$$

where w_i is the weight assigned to property p_i, reflecting its relative importance in determining druglikeness.

QED scores range from 0 to 1, with higher values indicating greater druglikeness. Bickerton et al. demonstrated that QED outperforms rule-based methods, such as Lipinski's Rule of Five [25], in distinguishing drugs from non-drugs. Moreover, QED allows compounds to be ranked on a continuous scale of druglikeness, providing a more nuanced assessment than binary classification rules.

One of the key advantages of QED is its ability to identify druglike compounds that may violate traditional rule-based filters. For example, some approved drugs that fail Lipinski's Rule of Five still receive high QED scores, highlighting the metric's capability to capture the continuum of druglikeness. Furthermore, QED can be used to assess the druglikeness of compounds within a specific chemical space, such as a series of analogs or a focused library, enabling the prioritization of the most promising candidates for further optimization.

2.5.3 Synthetic Accessibility. Synthetic accessibility is another important consideration when evaluating generative models. Ertl and Schuffenhauer [13] developed a method to estimate the synthetic accessibility score (SAS) of a molecule based on the complexity of its substructures. The SAS of a molecule m is calculated as:

$$\text{SAS}(m) = \frac{1}{n}\sum_{i=1}^{n} c_i + \frac{1}{n}\sum_{i=1}^{n} r_i + s + t, \tag{12}$$

where n is the number of atoms in the molecule, c_i is the complexity of the i-th atom's substructure, r_i is the number of rings in the substructure, s is the overall complexity of the molecular graph, and t is the number of chiral centers. The complexity values for various substructures are derived from a database of known molecules and their synthetic routes.

SAS ranges from 1 (easy to synthesize) to 10 (difficult to synthesize). Incorporating SAS into the evaluation pipeline can help prioritize generated molecules that are more feasible to synthesize, thus increasing the practical utility of the generative models.

3 Datasets for Molecule Generation

Datasets play a crucial role in training and evaluating generative models for molecule generation. They provide the necessary data for models to learn the underlying distribution of molecules and assess their performance in generating novel compounds. In this section, we discuss the SAFE-GPT training dataset and its importance in creating a fair comparison when training different architectures from scratch.

3.1 SAFE-GPT Training Dataset

The SAFE-GPT training dataset [28] is a vast collection of over 1 billion unlabeled molecules, specifically curated for pre-training generative models in deep generative molecular design. This dataset was constructed by combining molecules from the ZINC [19] and UniChem [5] libraries, resulting in a diverse set of 1.1 billion SMILES strings.

The SAFE-GPT training dataset covers various molecule types, including drug-like compounds, peptides, multi-fragment molecules, polymers, reagents, and non-small molecules. This diversity ensures the broad applicability of generative models trained on this dataset, making them suitable for a wide range of molecular design tasks.

To convert the SMILES strings into the SAFE representation, a combination of BRICS decomposition [10] and Louvain community detection [4] was employed. Molecules that could not undergo successful fragmentation were excluded from the dataset.

3.2 Training Transformer-based Models and SSMs on the SAFE-GPT Dataset

In our research, we propose to train Transformer-based models and State Space Models (SSMs) from scratch on the SAFE-GPT training dataset. By training these models on the same dataset, we aim to conduct a fair and comprehensive comparison of their performance in molecule generation tasks.

The SAFE-GPT training dataset is well-suited for this comparison for several reasons:

1. Size and diversity: With over 1 billion diverse molecules, the dataset provides ample data for training large-scale models like Transformers and SSMs.
2. SAFE representation: The molecules in the dataset are represented using the Sequential Attachment-based Fragment Embedding (SAFE) format, which has been shown to improve the interpretability and controllability of molecule generation [28].
3. Standardized pre-processing: The dataset has undergone standardized pre-processing steps, ensuring that all models are trained on the same input format.
4. Benchmark potential: Given its size and diversity, the dataset has the potential to serve as a benchmark for evaluating and comparing generative models in molecule generation.

Training Transformer-based models and SSMs on the SAFE-GPT training dataset will allow us to assess their relative strengths and weaknesses in capturing long-range dependencies, generating diverse molecules, and learning meaningful representations.

3.3 MOSES Dataset for Comparative Analysis

In addition to the SAFE-GPT training dataset, we will use the Molecular Sets (MOSES) dataset [29] to train smaller versions of the models for comparative analysis. The MOSES dataset contains a curated set of 1.9 million molecules from the ZINC database and serves as a standardized benchmark for evaluating generative models in drug discovery.

By training models on both the SAFE-GPT training dataset and the MOSES dataset, we can assess the impact of dataset size and diversity on model performance. This will allow us to make a fairer comparison between Transformers and SSMs, as we can evaluate their performance on datasets of different scales and characteristics.

Furthermore, training smaller versions of the models on the MOSES dataset will provide insights into the scalability and data efficiency of each architecture. We can observe how the performance of Transformers and SSMs varies when trained on a smaller dataset compared to the larger SAFE-GPT dataset.

The SAFE-GPT model has already been trained on the MOSES dataset for a smaller version (SAFE-GPT-20M) and compared to other generative models. However, to ensure a fair comparison, we will train both Transformer-based models and SSMs from scratch on the MOSES dataset, using the same hyperparameters and training setup. This will allow us to isolate the effects of the model architecture and assess their performance under consistent conditions.

By leveraging both the SAFE-GPT training dataset and the MOSES dataset, we aim to conduct a comprehensive and fair comparison of Transformer-based models and SSMs for molecule generation. This multi-dataset approach will provide valuable insights into the strengths and limitations of each architecture and help identify the most promising approaches for generating diverse and novel molecules with desired properties.

4 Discussion

The background sections of this review have provided a comprehensive overview of the current state-of-the-art in generative models for molecule generation, focusing on Transformer-based models, their applications, and the potential of State Space Models (SSMs) as an alternative approach. The choice of molecular string representation, such as SMILES, SELFIES, and the recently proposed SAFE, has been highlighted as a crucial factor influencing the performance and interpretability of generative models.

SAFE-GPT, a Transformer-based model leveraging the SAFE representation, has demonstrated impressive performance in generating diverse and novel molecules with optimized properties. However, like other Transformer-based models, SAFE-GPT faces challenges in handling long sequences and ensuring the stability and synthesizability of the generated compounds due to the quadratic complexity of self-attention, which limits their scalability to very long sequences.

SSMs have emerged as a promising alternative, offering potential advantages in efficiently handling long sequences and capturing the temporal aspects of molecular formation and evolution. Their linear time complexity makes them well-suited for generating complex molecules with many atoms and bonds. However, the application of SSMs to molecule generation is still in its early stages, and their ability to capture the complex spatial and geometric constraints present in molecular structures requires further investigation.

To conduct a fair and comprehensive comparison between SAFE-GPT and SSMs, we propose training both models from scratch on the SAFE-GPT dataset, which contains over 1 billion diverse molecules. By using consistent training procedures and evaluation metrics, such as validity, uniqueness, diversity, and target property achievement, we aim to obtain a robust assessment of their relative strengths and weaknesses.

However, this comparative study presents several challenges and limitations that need to be addressed. Adapting SSM architectures to handle the SAFE representation effectively may require additional research and development efforts. Furthermore, the interpretability and controllability of SSMs remain important areas for further investigation, as these aspects are critical for their practical application in drug discovery and materials science.

Another challenge lies in the computational complexity and time requirements of training and evaluating large-scale generative models like SAFE-GPT and SSMs. Careful consideration of resource allocation and optimization techniques will be necessary to ensure the feasibility and scalability of the comparative study.

Despite these challenges, the proposed comparative study has the potential to provide valuable insights into the strengths and limitations of different model architectures for molecule generation. By identifying the most promising approaches and guiding the development of more effective and efficient generative models, this study can contribute to the advancement of AI-driven drug discovery and materials science.

5 Conclusion

This literature review has provided a comprehensive overview of the current landscape of generative models for molecule generation, emphasizing the importance of molecular string representations and the potential of SSMs as an alternative to Transformer-based models. The proposed comparative study between SAFE-GPT and promising SSM architectures, using the SAFE representation and consistent training procedures, has the potential to advance our understanding of the capabilities of different model architectures for molecule generation.

The insights gained from this study can guide the development of more effective and efficient generative models, ultimately accelerating the discovery of novel therapeutic agents and functional materials. By identifying the most promising approaches for generating diverse and novel molecules with optimized properties, this research can contribute to the advancement of AI-driven drug discovery and materials science.

However, the comparative study also presents several limitations and challenges that need to be addressed. These include adapting SSM architectures to handle the SAFE representation effectively, enhancing the interpretability and controllability of SSMs, and optimizing computational resources for large-scale model training and evaluation.

Future research should focus on addressing these limitations and challenges to fully realize the potential of generative models in molecule generation. This may involve developing novel techniques to improve the interpretability and controllability of SSMs, exploring innovative architectures that combine the strengths of Transformer-based models and SSMs, and optimizing computational resources for large-scale model training and evaluation.

Furthermore, the broader implications of this research for the field of AI-driven drug discovery and materials science should be emphasized. By pushing the boundaries of molecule generation and identifying the most promising approaches, this study can pave the way for innovative strategies in de novo molecular design, accelerating the discovery of new therapeutic agents and materials with desired properties.

In conclusion, this literature review has highlighted the importance of comparing SAFE-GPT and SSMs for molecule generation, using the SAFE representation and consistent training procedures. While the proposed comparative study presents challenges and limitations, it has the potential to provide valuable insights and guide the development of more effective and efficient generative models. By addressing the identified limitations and challenges through future research, we can unlock the full potential of AI-driven molecule generation and contribute to the advancement of drug discovery and materials science.

References

[1] Viraj Bagal, Rishal Aggarwal, PK Vinod, and U Deva Priyakumar. 2022. Molgpt: molecular generation using a transformer-decoder model. *Journal of Chemical Information and Modeling* 62, 9 (2022), 2064–2076.

[2] Samy Bengio, Yoshua Bengio, and Jocelyn Cloutier. 1994. Use of genetic programming for the search of a new learning rule for neural

networks. In *Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*. IEEE, 324–327.

[3] 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.

[4] Vincent Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast Unfolding of Communities in Large Networks. *Journal of Statistical Mechanics Theory and Experiment* 2008 (04 2008). https://doi.org/10.1088/1742-5468/2008/10/P10008

[5] Jon Chambers, Mark Davies, Anna Gaulton, Anne Hersey, Sameer Velankar, Robert Petryszak, Janna Hastings, Louisa Bellis, Shaun McGlinchey, and John Overington. 2013. UniChem: A unified chemical structure cross-referencing and identifier tracking system. *Journal of cheminformatics* 5 (01 2013), 3. https://doi.org/10.1186/1758-2946-5-3

[6] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. In *arXiv preprint arXiv:1904.10509*.

[7] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[8] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Russ Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2978–2988.

[9] Soham De, Samuel L Smith, Anushan Fernando, Aleksandar Botev, George Cristian-Muraru, Albert Gu, Ruba Haroun, Leonard Berrada, Yutian Chen, Srivatsan Srinivasan, et al. 2024. Griffin: Mixing Gated Linear Recurrences with Local Attention for Efficient Language Models. *arXiv preprint arXiv:2402.19427* (2024).

[10] 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/cmdc.200800178

[11] Jeffrey L Elman. 1990. Finding structure in time. *Cognitive science* 14, 2 (1990), 179–211.

[12] 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.

[13] Peter Ertl and Ansgar Schuffenhauer. 2009. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of Cheminformatics* 1, 1 (2009), 1–11.

[14] Anna Gaulton, Louisa J Bellis, A Patricia Bento, Jon Chambers, Mark Davies, Anne Hersey, Yvonne Light, Shaun McGlinchey, David Michalovich, Bissan Al-Lazikani, et al. 2012. ChEMBL: a large-scale bioactivity database for drug discovery. *Nucleic acids research* 40, D1 (2012), D1100–D1107.

[15] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. 2017. Neural message passing for quantum chemistry. In *International conference on machine learning*. PMLR, 1263–1272.

[16] Albert Gu and Tri Dao. 2023. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. arXiv:2312.00752 [cs.LG]

[17] Albert Gu, Ian Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Re. 2021. Combining recurrent, convolutional, and continuous-time models with linear state-space layers. In *Advances in Neural Information Processing Systems*, Vol. 34. 572–585.

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

[19] John Irwin and Brian Shoichet. 2005. ZINC A Free Database of Commercially Available Compounds for Virtual Screening. *Journal of chemical information and modeling* 45 (04 2005), 177–82. https://doi.org/10.1021/ci049714+

[20] Jos'e Jim'enez-Luna, Francesca Grisoni, and Gisbert Schneider. 2020. Drug design here and now: a review of molecular modeling in drug discovery. *Journal of Chemical Information and Modeling* 60, 4 (2020), 1552–1568.

[21] 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.

[22] Rudolph Emil Kalman. 1960. A new approach to linear filtering and prediction problems. (1960).

[23] Mario Krenn, Florian Häse, AkshatKumar Nigam, Pascal Friederich, and Alan Aspuru-Guzik. 2020.

[24] Greg Landrum et al. 2023. RDKit: Open-source cheminformatics. *Online. http://www. rdkit. org* (2023).

[25] Christopher A Lipinski, Franco Lombardo, Beryl W Dominy, and Paul J Feeney. 1997. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced Drug Delivery Reviews* 23, 1-3 (1997), 3–25.

[26] Shengchao Liu, Mehmet F Demirel, and Yingyu Liang. 2019. N-gram graph: Simple unsupervised representation for graphs, with applications to molecules. *Advances in neural information processing systems* 32 (2019).

[27] Łukasz Ma'ziarka, Agnieszka Pocha, Jan Kaczmarczyk, Krzysztof Rataj, Tomasz Danel, and Michał Warchoł. 2020. Molecule attention transformer. *arXiv preprint arXiv:2002.08264* (2020).

[28] 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).

[29] 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]

[30] David Rogers and Mathew Hahn. 2010. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling* 50, 5 (2010), 742–754.

[31] Benjamin Sanchez-Lengeling and Alán Aspuru-Guzik. 2018. Inverse molecular design using machine learning: Generative models for matter engineering. *Science* 361, 6400 (2018), 360–365.

[32] 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.

[33] Philippe Schwaller, Teodoro Laino, Th'eophile Gaudin, Peter Bolgar, Christopher A Hunter, Costas Bekas, and Alpha A Lee. 2019. Molecular transformer: A model for uncertainty-calibrated chemical reaction prediction. In *ACS central science*, Vol. 5. ACS Publications, 1572–1583.

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762* (2017).

[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.