



Article

ICVAE: Interpretable Conditional Variational Autoencoder for De Novo Molecular Design

Xiaqiong Fan ¹, Senlin Fang ², Zhengyan Li ^{2,3}, Hongchao Ji ², Minghan Yue ¹, Jiamin Li ¹ and Xiaozhen Ren ^{1,*}

¹ School of Artificial Intelligence and Big Data, Henan University of Technology, Zhengzhou 450001, China; fxq@haut.edu.cn (X.F.); 231210100521@stu.haut.edu.cn (M.Y.); jiaminli@stu.haut.edu.cn (J.L.)

² Agricultural Genomics Institute at Shenzhen, Chinese Academy of Agricultural Sciences, Shenzhen 518120, China; d22092100360@cityu.mo (S.F.); liz913925@gmail.com (Z.L.); jihongchao@caas.cn (H.J.)

³ State Key Laboratory of Crop Stress Adaptation and Improvement, School of Life Sciences, Henan University, Kaifeng 475001, China

* Correspondence: rxz235@haut.edu.cn

Abstract: Recent studies have demonstrated that machine learning-based generative models can create novel molecules with desirable properties. Among them, Conditional Variational Autoencoder (CVAE) is a powerful approach to generate molecules with desired physiochemical and pharmacological properties. However, the CVAE's latent space is still a black-box, making it difficult to understand the relationship between the latent space and molecular properties. To address this issue, we propose the Interpretable Conditional Variational Autoencoder (ICVAE), which introduces a modified loss function that correlates the latent value with molecular properties. ICVAE established a linear mapping between latent variables and molecular properties. This linearity is not only crucial for improving interpretability, by assigning clear semantic meaning to latent dimensions, but also provides a practical advantage. It enables direct manipulation of molecular attributes through simple coordinate shifts in latent space, rather than relying on opaque, black-box optimization algorithms. Our experimental results show that the ICVAE can linearly relate one or multiple molecular properties with the latent value and generate molecules with precise properties by controlling the latent values. The ICVAE's interpretability allows us to gain insight into the molecular generation process, making it a promising approach in drug discovery and material design.

Keywords: molecular generation; variational autoencoder; drug discovery



Academic Editors: Pufeng Du,
Bing Niu and Suren Rao Sooranna

Received: 18 March 2025

Revised: 15 April 2025

Accepted: 21 April 2025

Published: 23 April 2025

Citation: Fan, X.; Fang, S.; Li, Z.; Ji, H.; Yue, M.; Li, J.; Ren, X. ICVAE: Interpretable Conditional Variational Autoencoder for De Novo Molecular Design. *Int. J. Mol. Sci.* **2025**, *26*, 3980. <https://doi.org/10.3390/ijms26093980>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

From ancient times to the present, human beings have been suffering from diverse diseases, and they continue to find corresponding medicines that can treat diseases. Until the end of the nineteenth century, the screening of new drugs was achieved by a trial-and-error method. However, this method requires immense manpower and financial resources and suffers from low efficiency and high uncertainty. In the past few years, the integration of machine learning predictions in the development of bioactive compounds has become an essential aspect of contemporary drug discovery efforts [1,2]. The utilization of deep learning techniques for de novo drug design, such as molecular generation, has become increasingly popular due to their potential to accelerate the discovery and optimization of novel ligands with desirable therapeutic properties [3,4].

A plethora of generative models based on machine learning have been developed and customized for the specific goal of *de novo* design. Some promising outcomes have unequivocally showcased the potential of these approaches. For example, the deep generative model called generative tensorial reinforcement learning (GENTRL) was used for optimizing synthetic feasibility, novelty, and biological activity, which enabled the discovery and synthesis of a potent inhibitor of discoidin domain receptor 1 (DDR1) within 21 days [5]. Another model named MproI-GEN was used for designing potential inhibitors against SARS-CoV-2 Mpro, which provided a list of candidates which passed quantum chemical screening [6].

Sequence-based recurrent neural networks (RNNs) [7–15], variational autoencoders (VAEs) [16–24], reinforcement learning (RL) [25–30], and generative adversarial networks (GANs) [31–35] are among the many machine learning frameworks and models currently available for molecular generation. The core idea of the VAE revolves around using an encoder to understand the intrinsic molecular distribution and a decoder to generate new molecules sampled from this distribution. Within the VAE framework, introducing randomness into the molecular distribution contributes to enriching the variety of generated molecules. The encoder of the VAE converts one-hot encoded SMILES vectors or molecular graph into latent vectors. These latent vectors are subsequently exposed to controlled noise to generate original SMILES or molecular graph, facilitated by the VAE's decoder. VAEs possess a distinct attribute in their capability to acquire a continuous and organized latent space. This latent space can be manipulated to create molecules possessing specific properties by combining it with optimization algorithms, such as Gaussian process [19] or Bayesian optimization [36].

One potential issue is that the latent space may be highly entangled, meaning that each dimension of the latent space cannot be interpreted independently of the others. This makes it challenging to understand how changes in a specific dimension of the latent space will affect the generated molecule. In the realm of interpretable machine learning (ML) for molecular design, Leguy et al. proposed the EvoMol approach [37], which utilizes an evolutionary algorithm to systematically construct molecular graphs in a sequential manner. This process of building can be effectively traced, allowing for the generation of chemically meaningful visualizations. Such an interpretable methodology holds promise for refining the exploration process in molecular design. Krishnan et al. [38] introduced a machine learning strategy aimed at autonomously designing protein kinase inhibitors. Their method combines a Variational Autoencoder (VAE) with a novel cluster-based perturbation technique, enabling a systematic exploration of the chemical latent space. Notably, this approach promotes the grouping of molecules with comparable structures in the latent space. Moreover, interpolating between molecules in this space enables the seamless transformation of molecular structures and associated properties. Noutahi et al. introduced LaPool (Laplacian Pooling) [39], a hierarchical graph pooling technique that is data-driven and interpretable. This inventive method efficiently merges node features and graph structures to improve molecular representation. By utilizing the graph Laplacian, LaPool identifies crucial graph substructures and promotes the grouping of nodes from the same substructure within the condensed graph.

Together, the previously mentioned approaches are aligned in their pursuit of enhancing the interpretability of molecular graph generation. These methodologies are focused on utilizing machine learning techniques to craft molecules exhibiting similar structures or substructures, while also providing a transparent avenue for refining the generative process. However, there is still a gap in property-driven latent space interpretability. Importantly, the primary objective is not to render the generative process geared towards similar

structures or substructures more interpretable. Rather, its emphasis lies in enhancing the interpretability of the generative process with respect to similar properties.

In this work, we present a novel approach to achieve interpretability in the latent space of CVAE. Our method, ICVAE, establishes a linear relationship between the conditional input and latent value (Figure 1). We evaluate various conditional inputs, including individual and pairwise property combinations. ICVAE enable the interpretation of generated molecule properties in the latent space. It makes it possible for the researchers to understand the meaning behind the latent space, e.g., the kind or the value of molecular properties. This capability supports intuitive, interactive molecular design, particularly in multi-objective settings where multiple properties must be balanced. We believe this property-centric latent space is a significant step toward bridging the gap between deep generative models and real-world applications in drug discovery.

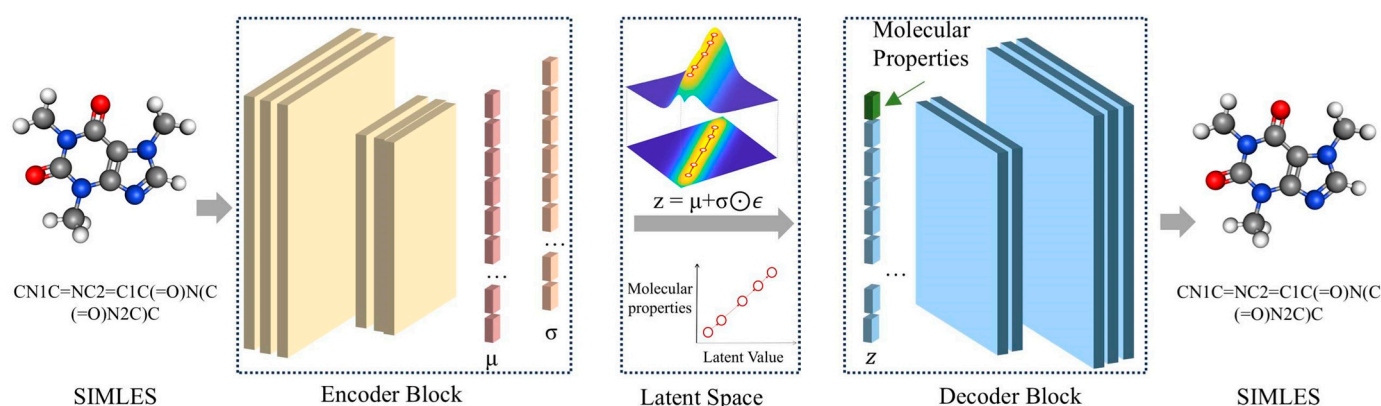


Figure 1. Overall architecture of ICVAE. The ICVAE consists of an encoder block, a latent space, and a decoder block. Initially, the one-hot SMILES vector and molecular properties are fed into the encoder block to derive the latent space representation of the input data. Subsequently, the decoder block generates the one-hot SMILES vector from the latent space. The key difference between the ICVAE and traditional CVAE is that the latent values of ICVAE are linearly dependent with the conditional inputs (molecular properties).

2. Results

The following section outlines three experiments that were conducted to demonstrate the interpretability of ICVAE. Experiment 1 involved a comparison of the latent space of ICVAE with VAE and CVAE, demonstrating the efficacy of ICVAE in rendering the latent space explicable. Experiment 2 further showcased the versatility of ICVAE, with the model capable of mapping multiple molecular properties to the latent space. Finally, experiment 3 highlighted the ability of researchers to randomly select a point in the ICVAE latent space to generate molecules with specific target properties.

2.1. Workflow Summary

The ICVAE workflow consists of three interconnected phases: (1) Data Preprocessing: SMILES strings are standardized, padded with start/end tokens (maximum length = 120), tokenized into atomic symbols, and converted into one-hot vectors. Simultaneously, molecular properties are normalized into conditional labels aligned with the latent dimensions. (2) Model Training: The encoder—built with convolutional and dense layers—maps SMILES-condition pairs into a latent space where each dimension is explicitly associated with a target property. A modified loss function enforces a linear correlation between latent values (z) and property labels (c) by penalizing deviations from the relation $z = \tau c + \epsilon$, where τ scales the latent range and ϵ represents stochasticity. Unlike standard CVAEs, conditional inputs are incorporated at both the encoding and decoding stages to reinforce

interpretability. (3) Controlled Generation: Users specify target property values, which are directly mapped to predefined latent coordinates. The decoder then generates molecules whose latent vectors correspond to these coordinates, ensuring the desired property fidelity.

2.2. The Interpretability of ICVAE

Figure 2 presents a comparison of the latent space constructed by ICVAE, VAE, and CVAE trained on the same dataset. Two-dimensional axes were extracted from the latent vectors to show their relationship with the HBA values. The soft label of HBA served as the conditional input for both CVAE and ICVAE. For the VAE, a multivariate right-skewed normal distribution of latent points is obtained, while the distribution of CVAE more closely resembled a multivariate normal distribution. That is because the CVAE can correspond the conditional input (HBA values) with the distributions of latent points. Therefore, ten HBA scaled values correspond to ten norm distributions and the distributions are independent of each other. This approach allows for the generation of molecules with a target property, with CVAE separating the latent space according to molecular properties to enable controllable property generation. However, the interpretability of the latent space is sacrificed, with the meaning behind each latent point remaining unknown. In other words, the relationship between the molecular properties and the latent point is unknown. In contrast, ICVAE establishes a linear relationship between the property and latent point, resulting in ten distributions in the latent space that are linearly related to ten scaled HBA values. As a result, the latent point of each HBA scale value has its own distribution, allowing for the linear interpretation of molecular properties.

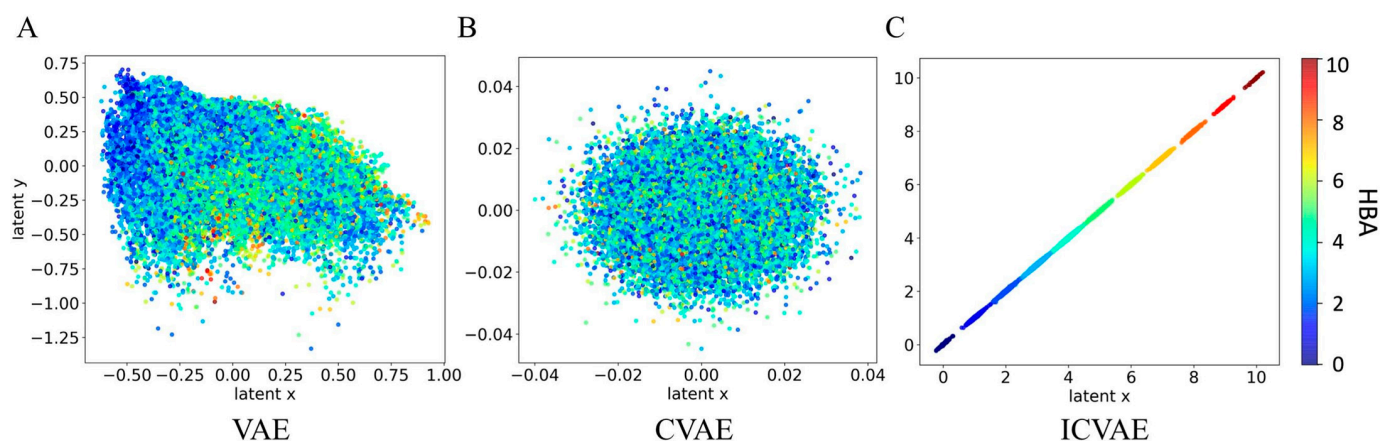


Figure 2. Comparison among the two-dimensional latent space of different methods in terms of their interpretability with respect to molecular HBA (hydrogen bond acceptor) property. (A) Two-dimensional latent spaces of VAE. (B) Two-dimensional latent spaces of CVAE. (C) Two-dimensional latent spaces of ICVAE. The horizontal and vertical axes correspond to the two dimensions of the latent vector: x represents the first dimension, while y represents the second dimension. The colormap depicts the relationship between the latent value and HBA value, where red represents higher HBA values and blue represents lower values. The VAE latent space shows some interpretability, but its distribution is non-standard and varies during training. The CVAE has a standard normal distribution, but the latent space lacks interpretability. In contrast, the ICVAE latent space exhibits a good linear relationship with the HBA value, making it more interpretable.

2.3. The Latent Space of ICVAE with Single Property

We present a comprehensive analysis of the interpretability of the ICVAE, which enables a linear correlation between the latent space and molecular properties. Figure 3 illustrates this relationship between the latent space and six additional molecular properties, namely MW, logP, SAS, QED, HBA, HBD, and TPSA, with each property corresponding

to two-dimensional axes in the latent vector. The latent space of MW shows the strong, continuous linear relationship between the MW value and the latent space, indicating that the learned latent points can represent the MW latent value. By unraveling the neural network's "black box", the researchers can understand the representation of latent points learned by the network for specific molecular properties. The latent space of the other properties also shows a similar phenomenon.

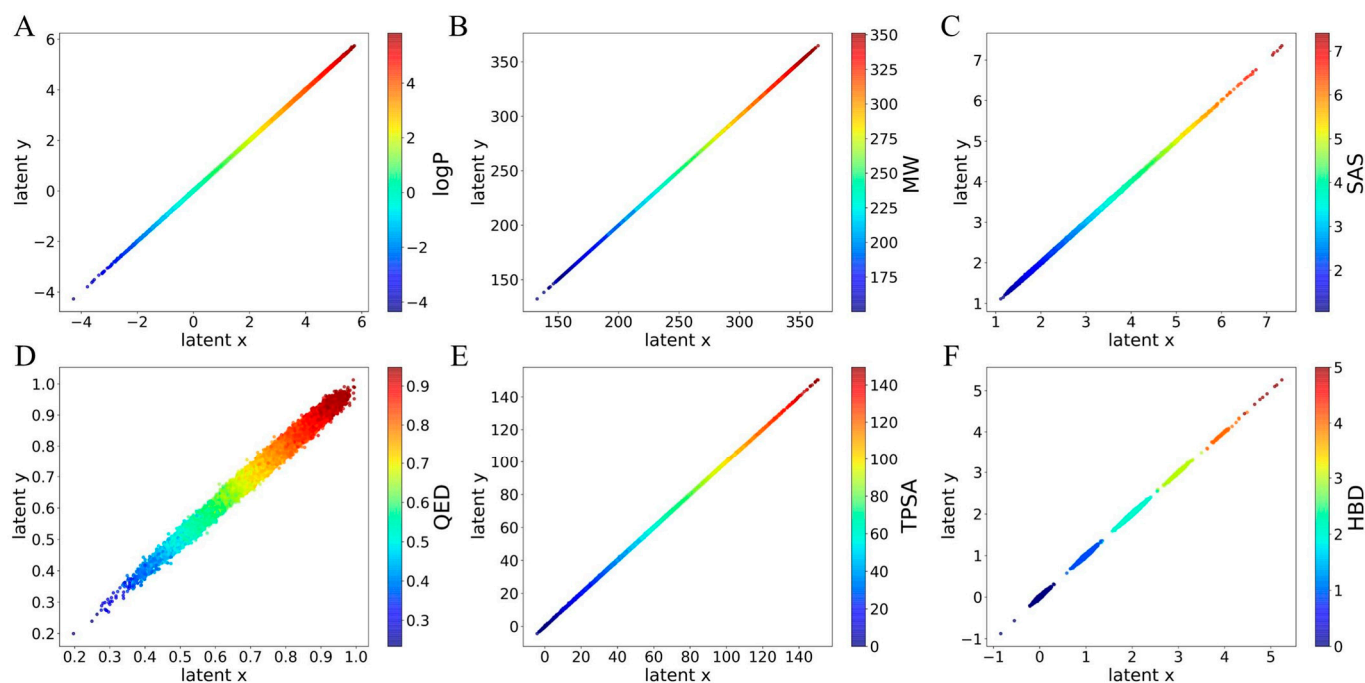


Figure 3. The relationship between latent space of ICVAE with single molecular property. (A) The relationship between latent space of ICVAE with logP. (B) The relationship between latent space of ICVAE with MW. (C) The relationship between latent space of ICVAE with SAS. (D) The relationship between latent space of ICVAE with QED. (E) The relationship between latent space of ICVAE with TPSA. (F) The relationship between latent space of ICVAE with HBD. The horizontal and vertical axes correspond to the two dimensions of the latent vector: x represents the first dimension, while y represents the second dimension. The value of the color bar represents each molecular property value.

Below the MW range of 150 of latent space, the latent points are discontinuous because there are very few molecules in the training dataset with a MW below 150. The latent points cannot maintain good continuity near the boundaries (minima and maxima) of molecular properties, which is attributable to epistemic rather than aleatoric uncertainty in our ICVAE. In other words, the reason for discontinuous latent space is the insufficient number of training samples with molecular features close to the boundary, and not the inherent recognition error of ICVAE. Additionally, the latent space of HBA and HBD is discontinuous and contains discrete line segments because the HBA and HBD values are discrete. In the latent space of HBD, the ICVAE erroneously converges to latent values less than zero at very few points. This is incorrect because the HBD is always greater than or equal to zero.

2.4. The Latent Space of ICVAE with Multiple Properties

Furthermore, the ICVAE not only allows for the correlation of a single molecular property with the latent value, but also enables the correlation of two or more properties with the latent value. Each property corresponds to one dimension of the latent space. For example, Figure 4 shows the results of correspond latent space with two molecular

properties, one at a time. The first plot is the result corresponding the HBA and HBD with the latent space. In order to give the different properties the same scale of latent space, we normalized the latent value of HBA and HBD to the range from 0.0 to 1.0. As HBA and HBD only have integer values, the two dimensions of latent space exhibit several discrete latent distributions. In the second plot, we correlated logP and HBA with the latent space, where the dimension of logP is continuous, while the dimension of HBA is discrete, which is reasonable because logP is a continuous property value. The third plot shows that the dimensions of MW and TPSA exhibit continuous value distributions in both latent dimensions, as both properties have continuous values.

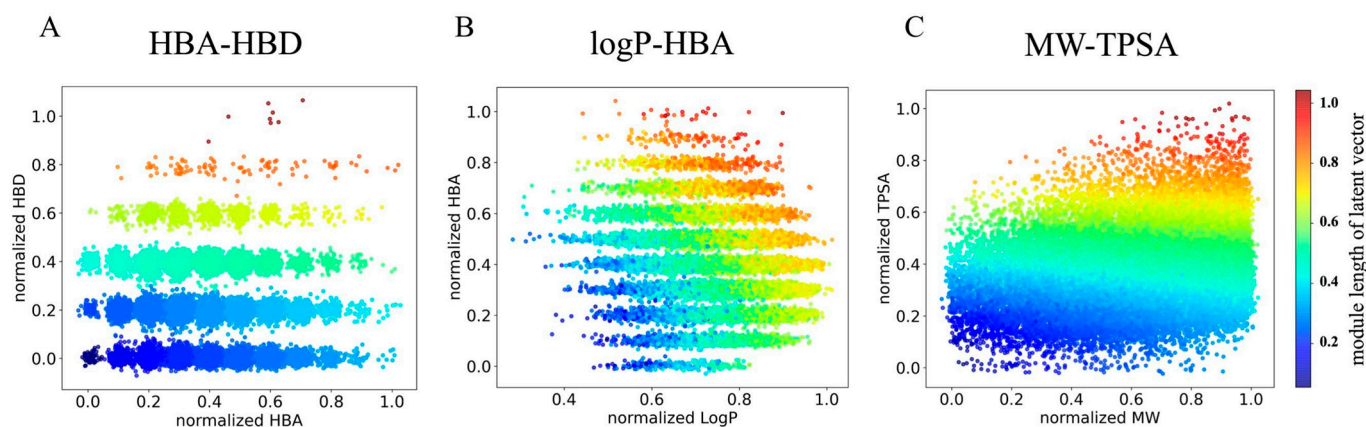


Figure 4. The relationship between latent space of ICVAE with multiple properties. The horizontal and vertical axes represent the two dimensions of latent x and y which are the first and second dimensions of the latent vector, respectively. (A) HBA-HBD. (B) logP-HBA. (C) MW-TPSA. To make two dimensions have same range, we calculate the normalized value of each property. The value of color bar represents the module length of latent vector, which is calculated by $\sqrt{x^2 + y^2}$.

2.5. The Performances of Molecular Generation

The performances of the ICVAE were evaluated and compared with two baseline models, VAE and CVAE, using MOSES evaluation metrics [40] to assess the quality of the generative models. The MOSES metrics include Fraction of valid (Valid), Similarity to a nearest neighbor (SNN), Fragment similarity (Frag), Novelty and Internal diversity (IntDiv), and Fraction of unique (Unique). Valid is defined using RDKit's molecular structure parser that checks atoms' valency and consistency of bonds in aromatic rings. The assessment involved sampling 25,000 molecules from each model under randomly chosen molecular property values. SNN is an average Tanimoto similarity between fingerprints of a molecule from the generated set and its nearest neighbor molecule in the reference dataset. Novelty is the fraction of generated molecules that pass filters applied during dataset construction. IntDiv assesses the chemical diversity within the generated set of molecules. Unique@1k quantifies ratios of unique molecules that are present within the top-k generated molecules. The results, as presented in Table 1, showed that the CVAE model achieved a notable improvement in validity, novelty, and IntDiv at the expense of reduced precision of SNN and Frag, compared to VAE. This is believed to be due to the condition input of CVAE, which limits its optimization for molecules under given conditions or molecular properties. Consequently, the CVAE model could focus more on finding molecules with enhanced validity, novelty and unique@1k, albeit possibly deviating from the distribution distance (SNN, Frag) of the test dataset.

In contrast, the ICVAE model was further optimized for finding molecules with specific molecular properties by correlating the conditions with the latent variable. As such, the ICVAE model could identify more valid and novel molecules compared to the CVAE

model, as it made the conditions stronger. In essence, the ICVAE model could efficiently locate more molecules that match the given molecular properties while still achieving high validity and novelty.

Table 1. Comparison of ICVAE and baseline models in terms of MOSES evaluation metrics of the molecules [#].

Model	Unique@1k [↑] ^a	Valid [↑]	SNN [↑]	Frag [↑]	Novelty [↑]	IntDiv [↑]
VAE	1.000	0.977	0.626	0.999	0.695	0.856
CVAE	0.975	0.971	0.315	0.925	0.985	0.832
ICVAE	1.000	0.979	0.339	0.843	1.000	0.869

[#] The ZINC test set was used as a reference set for the (MOSES framework). ^a [↑] Represents the higher the better. Bold text indicates the best results.

2.6. Molecular Generation with Target Properties

The primary advantage of the ICVAE model is its ability to generate SMILES strings of molecules with desirable properties by controlling the latent value, which is equivalent to the molecular property value. To evaluate the model's performance, we trained it on the ZINC dataset and tested its ability to generate molecules with single molecular property by changing the latent value of the single property. We varied the latent value of molecular weight (MW) from 210 to 490 at 20 intervals, logP from 0.2 to 5.8 at 0.4 intervals, QED from 0.05 to 0.61 at 0.04 intervals, SAS from 1.2 to 6.8 at 0.4 intervals, TPSA from 18 to 74 at 4 intervals, HBA from 0 to 10 at 1 interval, and HBD from 0 to 5 at 1 interval. This experiment allowed us to test the model's ability to accurately control and generate molecules with a range of desirable properties.

After generating SMILES from the ICVAE model at equal sampling size for each conditional latent value, the validity of the generated SMILES was calculated using RDKit. Only the absolute error of generated properties and the target properties less than a boundary value were selected for each condition. As shown in Figure 5, a large number of valid SMILES (red) were generated by the ICVAE model for single-property condition input (blue), demonstrating the model's ability to precisely control the single molecular property by changing the latent value of the single property.

This experimental process brings confidence to the practical application of the ICVAE method. For example, using ICVAE, medicinal chemists can generate molecules by regulating the molecular property values, analyze the generated molecular structures and evaluate their potential value in practical applications, thereby interactively generating and optimizing the molecular property values.

Upon examining Figure 5, we observed that generating molecular properties beyond a certain boundary value was challenging in some cases. Specifically, we found it difficult to generate molecules with MW values exceeding 400 and SAS values greater than 12. This may be due to a lack of training samples in these value intervals. Additionally, we observed that generating molecules with desirable HBA and HBD values was more difficult than generating other properties. This is a known issue with other conditional neural network models for molecular generation, such as cRNN [23].

To further validate the generation ability of the ICVAE model, the model with two different properties (MW and TPSA) as input condition is also trained and tested by changing the latent value of two properties. Specifically, for the two-properties-generation experiment, the latent value of MW is varied from 170 to 350 at 10 intervals, and the latent value of TPSA is varied from 10 to 90 at 10 intervals. Please note the MW and TPSA are varied simultaneously at each condition step. Figure 6 shows the molecular properties of valid SMILES generated by the ICVAE model input with the two-property condition

(MW and TPSA), demonstrating the model's ability to precisely control the generation of molecules with two molecular properties simultaneously. Upon examining Figure 6, we observed that it is difficult to generate molecules with TPSA values greater than 70 and MW values below 190. Except this range, other conditions generate enough molecules with target properties.

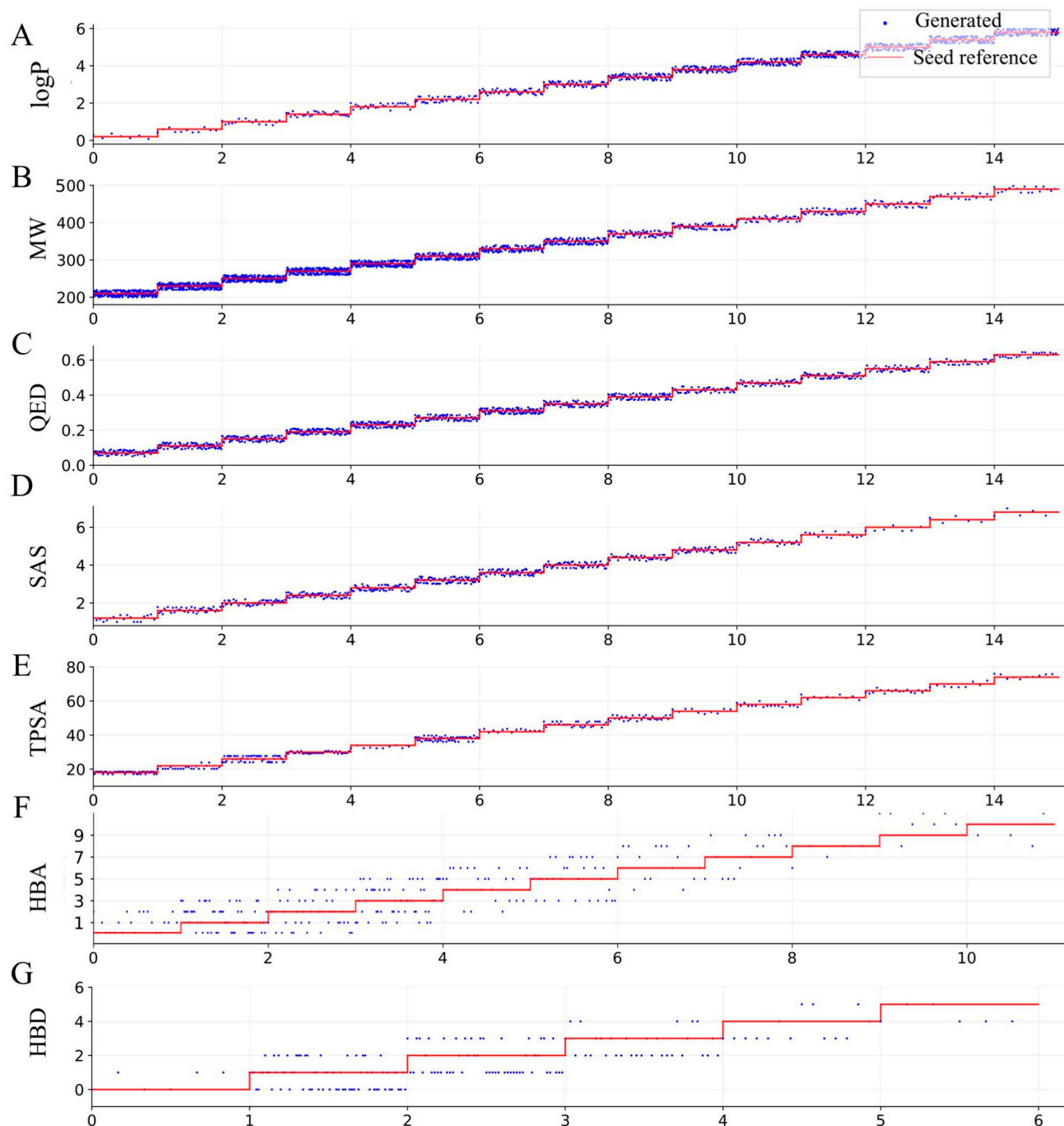


Figure 5. Molecular properties of valid SMILES generated by the ICVAE model with single-property condition input. The plot shows the seven calculated molecular properties, including (A) logP, (B) MW, (C) QED, (D) SAS, (E) TPSA, (F) HBA, and (G) HBD, of the generated molecules (blue) with given conditional latent values (red). The sampling size is equal for each conditional step.

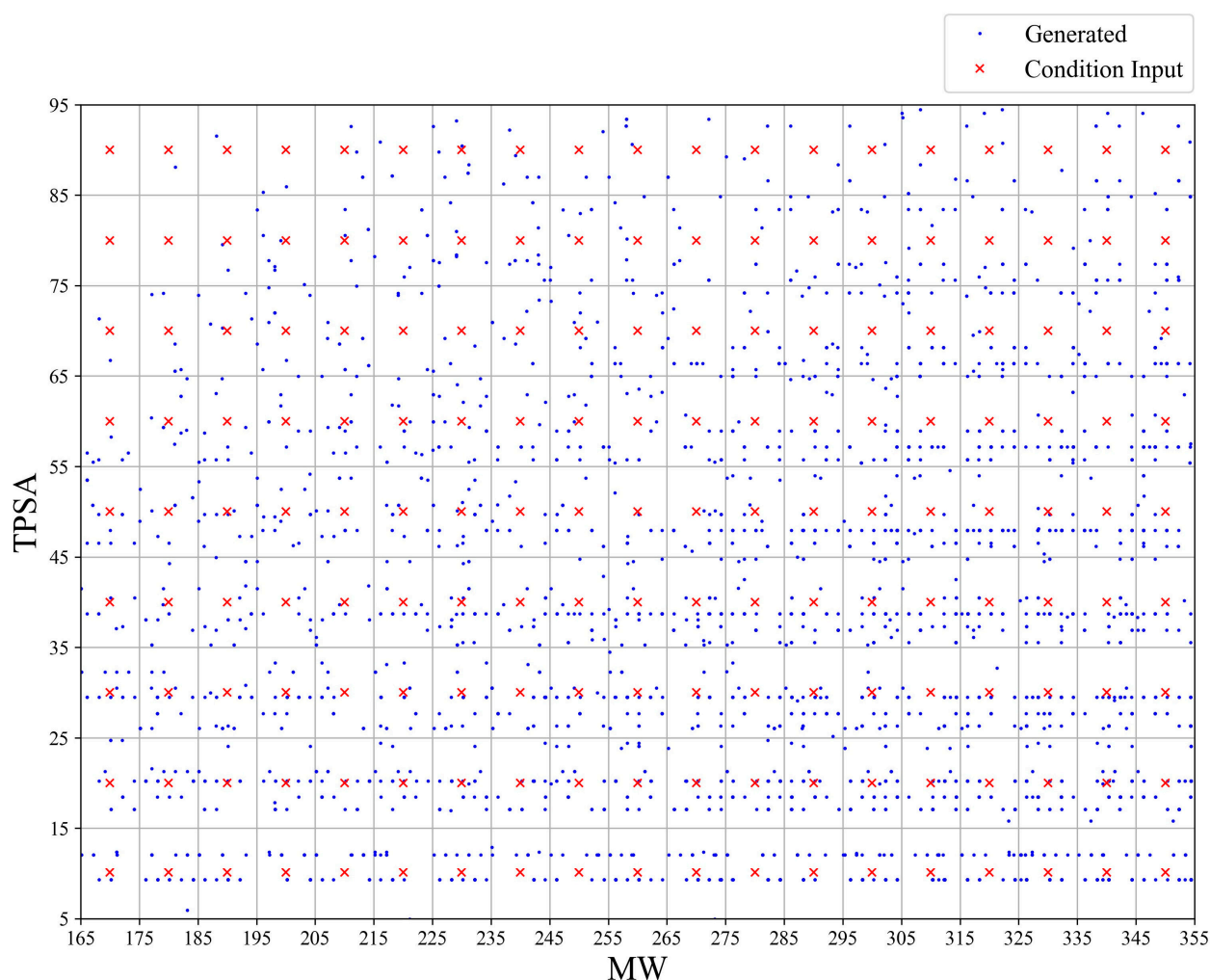


Figure 6. Molecular properties of valid SMILES generated by the ICVAE model with two-property condition as input. The plot shows the MW and TPSA value of the generated molecules (blue) with given conditional input (red). The x axis represents the MW value, and the y axis represents the TPSA value. The sampling size is equal for each conditional step.

3. Discussion

This study investigates the effectiveness of an interpretable molecule generation method based on Conditional Variational Autoencoders. The proposed method correlates the molecular properties with the latent values of molecular generation, making the molecular properties interpretable in the latent space. Our findings demonstrate that this method not only enables the interpretability of a single molecular property but also multiple molecular properties simultaneously. Moreover, the proposed method is shown to be highly effective and precise in generating molecular SMILES with corresponding molecular properties.

The ICVAE offers a notable benefit by enabling the encoding of high-dimensional molecular features (such as molecular properties) into latent vectors. This paves the way for efficient multi-objective optimization by constraining the values of these latent vectors. Building upon the foundation of CVAE, ICVAE enhances the interpretability of the latent vector by establishing a correlation between molecular properties and the latent vector itself. This advancement allows for the visualization and elucidation of the multi-objective optimization process.

Limitations and Future Work

While the ICVAE successfully enhances interpretability within the latent space of CVAE, it encounters challenges in generating a substantial portion of molecules when the training dataset is limited in size. Looking ahead, our focus will involve investigating the fusion of few-shot learning methodologies with ICVAE. This integration aims to amplify the generation capability of ICVAE, particularly in scenarios where the availability of training samples is restricted. This exploration holds the potential to bolster the model's capacity to generate diverse and meaningful molecular structures even with sparse training data.

4. Materials and Methods

4.1. Dataset

The dataset used in this paper is the publicly available ZINC dataset [41]. Instead of using the total dataset to feed into our ICVAE, we randomly selected 500,000 molecules. This is because increasing the used molecules did not increase the performance of our model. A total of 70% of the selected molecules were used for training, and the rest were used for testing. The five molecular properties MW, logP, HBD, HBA and TPSA were calculated by RDKit (Version: 2024.9.4).

4.2. Data Processing

Our ICVAE is an extension of the CVAE model, which takes in two input data. The first input data are a one-hot encoding vector of the molecular Simplified Molecular Input Line Entry System (SMILES). This representation provides a simple way to describe the chemical structure of molecules that can be easily processed by computers. To indicate the end of the SMILES string, we added an 'E' at each end of the string. Subsequently, the SMILES representations were transformed into one-hot encoding vectors, each with a standard size of 33×120 . Here, 33 denotes the encoding notation of the SMILES, and 120 represents the maximum length of the SMILES strings.

The second input data to our ICVAE model is molecular properties, which are conditional labels that the model uses to control the properties of the generated molecules. The raw labels are normalized into the conditional labels ranging from 0 to 500 for each molecular property. The wide range can make sure the ICVAE learns the distance between every single property value. After that, the conditional labels are not only fed with the input SMILES one-hot encoding vectors, but also with the latent vectors to reconstruct the SMILES one-hot encoding vectors. The way of feeding the conditional vector both to the input and latent vector is the same as the CVAE. By appending the conditional vector at both the encoding and decoding stages, the CVAE is ensuring that the generated data adhere to the desired condition from the very beginning (during encoding) and throughout the generation process (during decoding). This helps in producing data samples that are not only realistic but also satisfy specific requirements.

Note that including the conditional vector in both the input and latent vector is crucial for producing molecules with desired properties for CVAE. Building upon the foundation of CVAE, ICVAE establishes a link between the latent vector's value and the conditional input. This connection enhances the interpretability and control of the generative process by modifying the latent value. Although the latent space of ICVAE can retain its linearity even without the added conditional vector, the capability to generate molecules with target properties would be forfeited.

4.3. Algorithm of ICVAE

The architecture of our ICVAE model is illustrated in Figure 1, which closely resembles the CVAE. It consists of an encoder block, a latent space, and a decoder block.

The encoder block comprises three convolutional layers and three dense layers. Initially, the one-hot SMILES vector and molecular properties (conditional labels) are fed into the three convolutional layers which contains 16, 32, and 64 filters, respectively, with the kernel size of 11×3 and stride of 2. The first dense layers are added after the three convolutional layers to convert the high-dimensional outputs from the convolutional layers into the 292-dimensional hidden vectors. Then, the hidden vectors pass through the second and the third layers to obtain two K -dimensional latent vectors. The two K -dimensional latent vectors are the mean vector μ and standard deviation vector σ , respectively. The reparameterization trick is used to create the K -dimensional latent vector z :

$$z = \mu + \sigma \odot \epsilon \quad (1)$$

where \odot is the Hadamard product, and ϵ is the random vector sampling from a normal distribution. In this study, we set K as 128 because the generation performance of ICVAE is greatly improved when using K from 32 or 64 to 128 and not significantly improved when using K from 128 to 256 (Figure 7). Subsequently, the decoder block symmetrical to the encoder block generates the one-hot SMILES vector from the latent vector z , taking into account the given molecular properties.

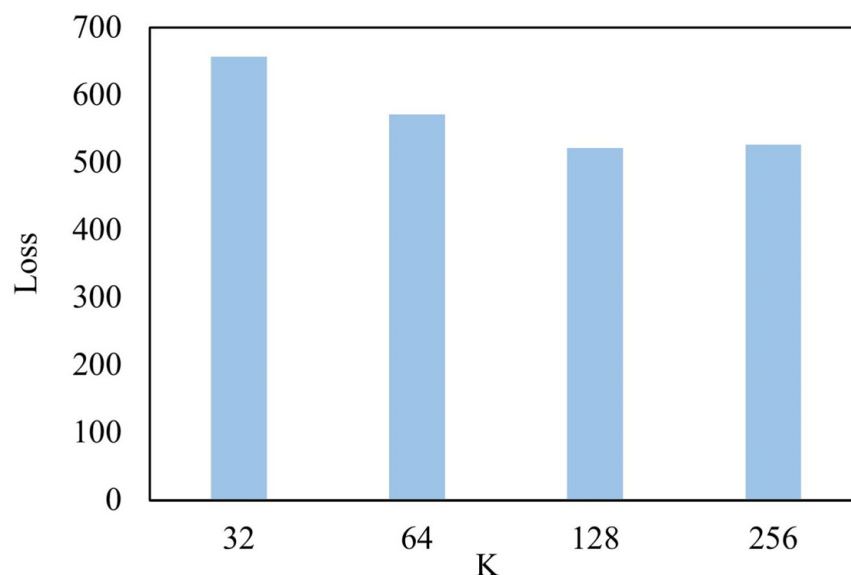


Figure 7. The loss value of test dataset under different K values.

The primary distinguishing feature of the ICVAE model is its interpretable latent space. Specifically, the latent value is linearly dependent on the molecular properties. To highlight the rationale behind the ICVAE, we compared its objective functions with those of the VAE and CVAE models. The objective of the VAE is to maximize the following function:

$$E_{q(z|x)}[\log p(x|z)] - D_{KL}(q(z|x)||p(z)) \quad (2)$$

However, the properties of molecular generation cannot be readily controlled since the latent distributions vary with different training processes in the VAE. In contrast, the CVAE incorporates an additional condition input that can guide the network to generate molecules with desirable properties. The objective of the CVAE is to maximize the following function:

$$E_{q(z|x,c)}[\log p(x|z,c)] - D_{KL}(q(z|x,c)||p(z|c)) \quad (3)$$

where $p(x|c)$ is the probability of generating x given the conditional input c . The CVAE provides researchers with the ability to control the molecular properties by providing input conditions c . However, the latent space in the CVAE is inexplicable and implicit, making it difficult for researchers to associate specific points in the latent space with particular molecular properties. To overcome this limitation, the proposed ICVAE correlates the input conditions c with the latent z . In other words, ICVAE optimizes the latent values associated with individual molecular property by aligning them with distinct independent Gaussian distribution, each with its own specific mean. The objective of ICVAE is to maximize the following function:

$$E_{q(z-\tau c|x,c)}[\log p(x|z-\tau c,c)] - D_{KL}(q(z-\tau c|x,c)||p(z-\tau c|c)) \quad (4)$$

where τ is the factor controlling the ratio of condition c to latent z . In the experiment, we found the best τ is to expand the range of latent conditions to $[0, 500]$. For example, the τ of training the ICVAE with the HBA condition input is 50.

Author Contributions: Methodology, H.J. and X.R.; validation, J.L., M.Y. and Z.L.; resources, S.F.; writing—original draft preparation, X.F. and S.F.; writing—review and editing, X.F.; visualization, M.Y. and Z.L.; funding acquisition, X.F., X.R. and H.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Key Research and Development Program of China, grant number 2023YFA0915800; the Natural science project of Science and Technology Department of Henan Province, grant number 232102210005 and 252102211029; High-level Talents Fund of Henan University of Technology, grant number 2022BS075.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used to train all models are randomly selected from the ZINC dataset, which are available in <https://zenodo.org/record/8278179> (accessed on 1 March 2025). The Python (Version: 3.6.13) code and the trained neural networks used in this work are available under MIT licenses at <https://github.com/Xiaqiong-Fan/ICVAE> (accessed on 1 March 2025).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Fleming, N. How artificial intelligence is changing drug discovery. *Nature* **2018**, *577*, S55–S57. [CrossRef] [PubMed]
2. Schneider, P.; Walters, W.P.; Plowright, A.T.; Sieroka, N.; Listgarten, J.; Goodnow, R.A., Jr.; Fisher, J.; Jansen, J.M.; Duca, J.S.; Rush, T.S. Rethinking drug design in the artificial intelligence era. *Nat. Rev. Drug Discov.* **2020**, *19*, 353–364. [CrossRef]
3. Zeng, X.; Wang, F.; Luo, Y.; Kang, S.-g.; Tang, J.; Lightstone, F.C.; Fang, E.F.; Cornell, W.; Nussinov, R.; Cheng, F. Deep generative molecular design reshapes drug discovery. *Cell Rep. Med.* **2022**, *3*, 100794. [CrossRef] [PubMed]
4. Tong, X.; Liu, X.; Tan, X.; Li, X.; Jiang, J.; Xiong, Z.; Xu, T.; Jiang, H.; Qiao, N.; Zheng, M. Generative models for de novo drug design. *J. Med. Chem.* **2021**, *64*, 14011–14027. [CrossRef]
5. Zhavoronkov, A.; Ivanenkov, Y.A.; Aliper, A.; Veselov, M.S.; Aladinskiy, V.A.; Aladinskaya, A.V.; Terentiev, V.A.; Polykovskiy, D.A.; Kuznetsov, M.D.; Asadulaev, A. Deep learning enables rapid identification of potent DDR1 kinase inhibitors. *Nat. Biotechnol.* **2019**, *37*, 1038–1040. [CrossRef] [PubMed]
6. Li, S.; Wang, L.; Meng, J.; Zhao, Q.; Zhang, L.; Liu, H. De Novo design of potential inhibitors against SARS-CoV-2 Mpro. *Comput. Biol. Med.* **2022**, *147*, 105728. [CrossRef]
7. Gupta, A.; Müller, A.T.; Huisman, B.J.; Fuchs, J.A.; Schneider, P.; Schneider, G. Generative recurrent networks for de novo drug design. *Mol. Inform.* **2018**, *37*, 1700111. [CrossRef]
8. Pogány, P.; Arad, N.; Genway, S.; Pickett, S.D. De novo molecule design by translating from reduced graphs to SMILES. *J. Chem. Inf. Model.* **2018**, *59*, 1136–1146. [CrossRef]
9. Yang, X.; Zhang, J.; Yoshizoe, K.; Terayama, K.; Tsuda, K. ChemTS: An efficient python library for de novo molecular generation. *Sci. Technol. Adv. Mater.* **2017**, *18*, 972–976. [CrossRef]

10. Segler, M.H.; Kogej, T.; Tyrchan, C.; Waller, M.P. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS Cent. Sci.* **2018**, *4*, 120–131. [\[CrossRef\]](#)
11. Liu, X.; Ye, K.; Van Vlijmen, H.W.; IJzerman, A.P.; Van Westen, G.J. An exploration strategy improves the diversity of de novo ligands using deep reinforcement learning: A case for the adenosine A2A receptor. *J. Cheminform.* **2019**, *11*, 35. [\[CrossRef\]](#)
12. Merk, D.; Friedrich, L.; Grisoni, F.; Schneider, G. De novo design of bioactive small molecules by artificial intelligence. *Mol. Inform.* **2018**, *37*, 1700153. [\[CrossRef\]](#) [\[PubMed\]](#)
13. Grisoni, F.; Moret, M.; Lingwood, R.; Schneider, G. Bidirectional molecule generation with recurrent neural networks. *J. Chem. Inf. Model.* **2020**, *60*, 1175–1183. [\[CrossRef\]](#)
14. Li, C.; Wang, C.; Sun, M.; Zeng, Y.; Yuan, Y.; Gou, Q.; Wang, G.; Guo, Y.; Pu, X. Correlated RNN framework to quickly generate molecules with desired properties for energetic materials in the low data regime. *J. Chem. Inf. Model.* **2022**, *62*, 4873–4887. [\[CrossRef\]](#)
15. Stravs, M.A.; Dührkop, K.; Böcker, S.; Zamboni, N. MSNovelist: De novo structure generation from mass spectra. *Nat. Methods* **2022**, *19*, 865–870. [\[CrossRef\]](#)
16. Kadurin, A.; Nikolenko, S.; Khrabrov, K.; Aliper, A.; Zhavoronkov, A. druGAN: An advanced generative adversarial autoencoder model for de novo generation of new molecules with desired molecular properties in silico. *Mol. Pharm.* **2017**, *14*, 3098–3104. [\[CrossRef\]](#) [\[PubMed\]](#)
17. Kang, S.; Cho, K. Conditional molecular design with deep generative models. *J. Chem. Inf. Model.* **2018**, *59*, 43–52. [\[CrossRef\]](#) [\[PubMed\]](#)
18. Lim, J.; Ryu, S.; Kim, J.W.; Kim, W.Y. Molecular generative model based on conditional variational autoencoder for de novo molecular design. *J. Cheminform.* **2018**, *10*, 1–9. [\[CrossRef\]](#)
19. Gómez-Bombarelli, R.; Wei, J.N.; Duvenaud, D.; Hernández-Lobato, J.M.; Sánchez-Lengeling, B.; Sheberla, D.; Aguilera-Iparraguirre, J.; Hirzel, T.D.; Adams, R.P.; Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Cent. Sci.* **2018**, *4*, 268–276. [\[CrossRef\]](#)
20. Colby, S.M.; Nuñez, J.R.; Hodas, N.O.; Corley, C.D.; Renslow, R.R. Deep learning to generate in silico chemical property libraries and candidate molecules for small molecule identification in complex samples. *Anal. Chem.* **2019**, *92*, 1720–1729. [\[CrossRef\]](#)
21. Joo, S.; Kim, M.S.; Yang, J.; Park, J. Generative model for proposing drug candidates satisfying anticancer properties using a conditional variational autoencoder. *ACS Omega* **2020**, *5*, 18642–18650. [\[CrossRef\]](#)
22. Bjerrum, E.J.; Sattarov, B. Improving chemical autoencoder latent space and molecular de novo generation diversity with heteroencoders. *Biomolecules* **2018**, *8*, 131. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Kotsias, P.-C.; Arús-Pous, J.; Chen, H.; Engkvist, O.; Tyrchan, C.; Bjerrum, E.J. Direct steering of de novo molecular generation with descriptor conditional recurrent neural networks. *Nat. Mach. Intell.* **2020**, *2*, 254–265. [\[CrossRef\]](#)
24. Bagal, V.; Aggarwal, R.; Vinod, P.; Priyakumar, U.D. MolGPT: Molecular generation using a transformer-decoder model. *J. Chem. Inf. Model.* **2021**, *62*, 2064–2076. [\[CrossRef\]](#)
25. Popova, M.; Isayev, O.; Tropsha, A. Deep reinforcement learning for de novo drug design. *Sci. Adv.* **2018**, *4*, eaap7885. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Ståhl, N.; Falkman, G.; Karlsson, A.; Mathiason, G.; Bostrom, J. Deep reinforcement learning for multiparameter optimization in de novo drug design. *J. Chem. Inf. Model.* **2019**, *59*, 3166–3176. [\[CrossRef\]](#)
27. Wang, J.; Hsieh, C.-Y.; Wang, M.; Wang, X.; Wu, Z.; Jiang, D.; Liao, B.; Zhang, X.; Yang, B.; He, Q. Multi-constraint molecular generation based on conditional transformer, knowledge distillation and reinforcement learning. *Nat. Mach. Intell.* **2021**, *3*, 914–922. [\[CrossRef\]](#)
28. Korshunova, M.; Huang, N.; Capuzzi, S.; Radchenko, D.S.; Savych, O.; Moroz, Y.S.; Wells, C.I.; Willson, T.M.; Tropsha, A.; Isayev, O. Generative and reinforcement learning approaches for the automated de novo design of bioactive compounds. *Commun. Chem.* **2022**, *5*, 129. [\[CrossRef\]](#)
29. Olivecrona, M.; Blaschke, T.; Engkvist, O.; Chen, H. Molecular de-novo design through deep reinforcement learning. *J. Cheminform.* **2017**, *9*, 1–14. [\[CrossRef\]](#)
30. Guo, J.; Fialková, V.; Arango, J.D.; Margreitter, C.; Janet, J.P.; Papadopoulos, K.; Engkvist, O.; Patronov, A. Improving de novo molecular design with curriculum learning. *Nat. Mach. Intell.* **2022**, *4*, 555–563. [\[CrossRef\]](#)
31. Wan, C.; Jones, D.T. Improving protein function prediction with synthetic feature samples created by generative adversarial networks. *bioRxiv* **2019**. [\[CrossRef\]](#)
32. Bian, Y.; Wang, J.; Jun, J.J.; Xie, X.-Q. Deep convolutional generative adversarial network (dcGAN) models for screening and design of small molecules targeting cannabinoid receptors. *Mol. Pharm.* **2019**, *16*, 4451–4460. [\[CrossRef\]](#) [\[PubMed\]](#)
33. Prykhodko, O.; Johansson, S.V.; Kotsias, P.-C.; Arús-Pous, J.; Bjerrum, E.J.; Engkvist, O.; Chen, H. A de novo molecular generation method using latent vector based generative adversarial network. *J. Cheminform.* **2019**, *11*, 1–13. [\[CrossRef\]](#) [\[PubMed\]](#)
34. Sousa, T.; Correia, J.; Pereira, V.; Rocha, M. Generative deep learning for targeted compound design. *J. Chem. Inf. Model.* **2021**, *61*, 5343–5361. [\[CrossRef\]](#)

35. De Cao, N.; Kipf, T. MolGAN: An implicit generative model for small molecular graphs. *arXiv* **2018**, arXiv:1805.11973.
36. Griffiths, R.-R.; Hernández-Lobato, J.M. Constrained Bayesian optimization for automatic chemical design using variational autoencoders. *Chem. Sci.* **2020**, *11*, 577–586. [[CrossRef](#)] [[PubMed](#)]
37. Leguy, J.; Cauchy, T.; Glavatskikh, M.; Duval, B.; Da Mota, B. EvoMol: A flexible and interpretable evolutionary algorithm for unbiased de novo molecular generation. *J. Cheminform.* **2020**, *12*, 1–19. [[CrossRef](#)]
38. Krishnan, S.R.; Bung, N.; Bulusu, G.; Roy, A. Accelerating de novo drug design against novel proteins using deep learning. *J. Chem. Inf. Model.* **2021**, *61*, 621–630. [[CrossRef](#)]
39. Noutahi, E.; Beaini, D.; Horwood, J.; Giguère, S.; Tossou, P. Towards interpretable sparse graph representation learning with laplacian pooling. *arXiv* **2019**, arXiv:1905.11577.
40. Polykovskiy, D.; Zhebrak, A.; Sanchez-Lengeling, B.; Golovanov, S.; Tatanov, O.; Belyaev, S.; Kurbanov, R.; Artamonov, A.; Aladinskiy, V.; Veselov, M. Molecular sets (MOSES): A benchmarking platform for molecular generation models. *Front. Pharmacol.* **2020**, *11*, 565644. [[CrossRef](#)]
41. Irwin, J.J.; Sterling, T.; Mysinger, M.M.; Bolstad, E.S.; Coleman, R.G. ZINC: A free tool to discover chemistry for biology. *J. Chem. Inf. Model.* **2012**, *52*, 1757–1768. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.