

Comparative Analysis of Molecular Graph Generation Methods:

Evaluating GraphVAE, WGAN-GP and CoDNet for De Novo Drug Design

Hironmoy Roy Rudra
ID: 22201611 (Section-03)

Priyankan Biswas
ID: 22201604 (Section-03)

Sabbir Hossain Prince
ID: 22201330 (Section-03)

Tanvir Ahmed Khan Nibir
ID: 22201103 (Section-01)

January 22, 2026

Abstract

The discovery of novel therapeutic compounds is a critical bottleneck in modern pharmaceutical development, characterized by exorbitant costs and high attrition rates. This research report presents a comprehensive comparative analysis of three distinct generative paradigms: the Graph Variational Autoencoder (GraphVAE), the Wasserstein Generative Adversarial Network (WGAN-GP), and the Controlled Diffusion Network (CoDNet). We trained and evaluated these models on the QM9 benchmark dataset.

Our experimental analysis reveals that while adversarial and variational methods provide a foundational baseline, the CoDNet architecture significantly outperforms them. Specifically, CoDNet demonstrated a validity rate of **98.54%**, a uniqueness score of **99.80%**, and a novelty score of **100%**, effectively capturing complex 3D geometric dependencies that traditional models struggle to represent.

1 Introduction

The identification of high-quality lead compounds—molecules that satisfy physical constraints like solubility and metabolic stability while binding to a target—is an astronomical search problem, with potential drug-like molecules exceeding 10^{60} . Traditional drug discovery relies on physical library screening, which represents only a minuscule fraction of this space.

Transitioning to *de novo* design, where molecules are tailor-made for specific targets, is essential to overcome the productivity crisis in the pharmaceutical industry. Structure-Based Drug Design (SBDD) serves as the cornerstone of this approach, leveraging the 3D information of protein targets to design ligands.

1.1 Literature Review

The evolution of generative models for molecular design has been rapid, moving from string-based representations to sophisticated graph-based and geometry-aware architectures.

Variational Autoencoders (VAEs): The foundational work by Kingma and Welling introduced the Auto-Encoding Variational Bayes (AEVB) framework, proposing the Variational Autoencoder (VAE) [1]. The VAE addresses the intractability of posterior inference in latent variable models by introducing a recognition model (encoder) $q_\phi(z|x)$ to approximate the true posterior $p_\theta(z|x)$. A critical innovation was the *reparameterization trick*, which allows for the optimization of the variational lower bound (ELBO) using standard stochastic gradient descent. Formally, the ELBO is defined as:

$$\mathcal{L}(\theta, \phi; x) = E_{q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}(q_\phi(z|x)\|p(z)) \quad (1)$$

In the context of molecules, early VAEs operated on SMILES strings, but these often produced invalid sequences. Subsequent works like GraphVAE adapted this to graph structures, directly predicting adjacency matrices. However, VAEs often suffer from the “blurriness” problem, where the reconstructed graphs contain probabilistic edges that do not correspond to discrete chemical bonds, necessitating complex matching algorithms during training.

Generative Adversarial Networks (GANs): Goodfellow et al. introduced Generative Adversarial Nets (GANs), a framework where a generator G and a discriminator D play a minimax game [2]. The generator aims to produce samples indistinguishable from real data, while the discriminator attempts to classify them. The objective function is:

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}}[\log D(x)] + E_{z \sim p_z}[\log(1 - D(G(z)))] \quad (2)$$

While highly successful in image synthesis, GANs face significant challenges with discrete graph data, including mode collapse and training instability. To address this, Arjovsky et al. proposed the Wasserstein GAN (WGAN), which minimizes the Earth Mover (EM) distance instead of the Jensen-Shannon divergence [4]. The EM distance provides meaningful gradients even when the support of the generated and real distributions are disjoint, which is critical for learning distributions on low-dimensional manifolds. Further improvements, specifically the Gradient Penalty (WGAN-GP), enforce the Lipschitz constraint required by the Wasserstein objective without the pathological side effects of weight clipping.

Relational Graph Convolutional Networks (R-GCNs): To process the complex topology of molecular graphs within these architectures, Schlichtkrull et al. developed Relational Graph Convolutional Networks (R-GCNs) [5]. Standard GCNs treat all edges as identical, which is insufficient for chemistry where single, double, and aromatic bonds have vastly different properties. R-GCNs introduce relation-specific weight matrices, allowing the model to learn distinct feature transformations for different bond types:

$$h_i^{(l+1)} = \sigma \left(\sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{N}_i^r} \frac{1}{c_{i,r}} W_r^{(l)} h_j^{(l)} + W_0^{(l)} h_i^{(l)} \right) \quad (3)$$

where \mathcal{R} is the set of relation types (bond types), and \mathcal{N}_i^r denotes the neighbors of node i under relation r . This architecture is employed in our WGAN-GP model as the discriminator to ensure the topological validity of generated graphs.

Diffusion Models and Non-equilibrium Thermodynamics: The most recent paradigm shift involves diffusion models. Sohl-Dickstein et al. pioneered this approach, drawing inspiration from

non-equilibrium statistical physics [3]. The core idea is to define a forward diffusion process that systematically destroys structure in a data distribution by adding noise, and then learn a reverse diffusion process to restore structure. The forward process is defined as a Markov chain:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t I) \quad (4)$$

where β_t is a variance schedule. The reverse process is learned by a neural network to approximate $p_\theta(x_{t-1}|x_t)$. This allows for the training of deep generative models with thousands of layers that can model complex data distributions with high flexibility and tractability. CoDNet [6] builds upon this by integrating E(n)-Equivariant Graph Neural Networks (EGNNs) to generate molecules in 3D space, ensuring that the generated structures respect the rotational and translational symmetries of Euclidean space (SE(3) equivariance). This represents a leap forward from 2D graph generation to true 3D conformational design.

This work is critical because it evaluates these architectures on their practical ability to solve the “inverse design” problem. While VAEs and GANs are computationally faster, their failure to consistently produce valid and unique 3D structures limits their utility in R&D. By benchmarking on the QM9 dataset, we demonstrate that diffusion processes offer a superior, physically grounded pathway for autonomous drug design.

1.2 Our Contributions

- Implemented and trained three distinct generative models: GraphVAE, WGAN-GP (with R-GCN), and CoDNet on the QM9 dataset.
- **Enhanced GraphVAE Architecture:** Developed a Conditional VAE with property conditioning (HOMO, LUMO, gap, dipole moment), increased latent dimensions (128), valence constraint checking, padding token filtering, and cyclical KL annealing ($\beta_{\max} = 0.02$), improving validity from $\sim 20\%$ to 80% .
- Conducted a rigorous performance evaluation focused on chemical validity, uniqueness, and novelty.
- Demonstrated that CoDNet achieves near-perfect generation metrics, establishing diffusion models as the superior paradigm for molecular design.

2 Material and Methods

2.1 Methodological Framework

The overall experimental workflow is illustrated in Figure 1. The pipeline begins with the ingestion of the QM9 dataset, which serves as the ground truth for training three distinct classes of generative models. As shown in the diagram, the workflow is parallelized to allow for a direct comparison of the generated outputs against the same evaluation metrics. While GraphVAE and WGAN-GP focus on generating 2D graph topology (Adjacency matrices and Node features), CoDNet extends this by generating 3D atomic coordinates directly.

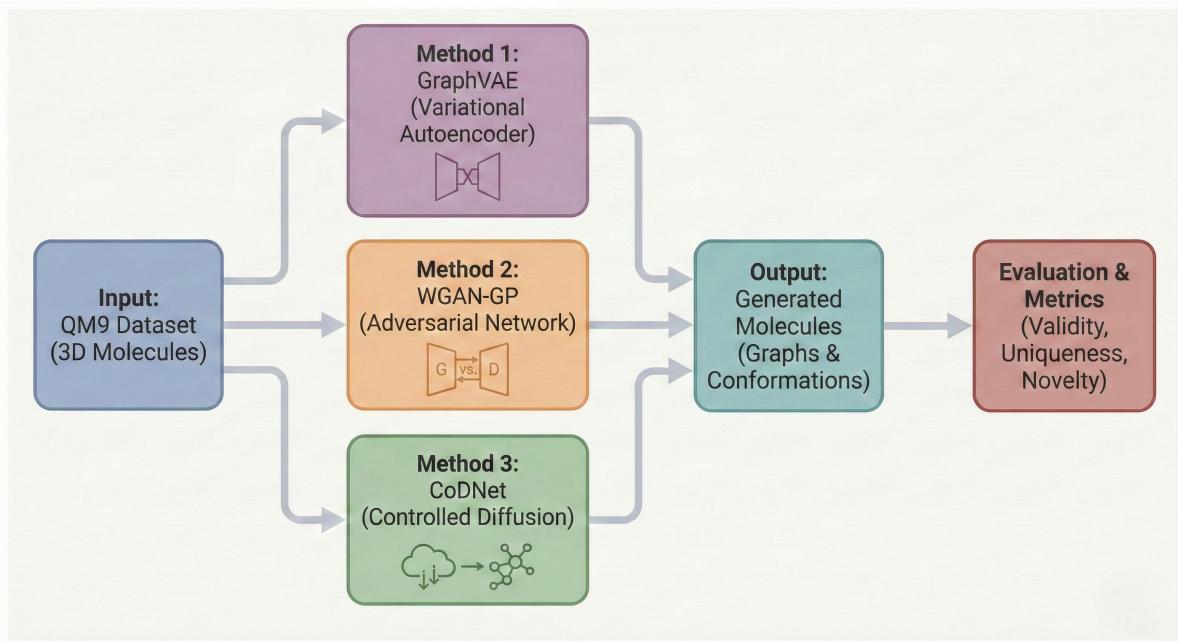


Figure 1: **Graphical Abstract of the Generative Pipeline.** The framework compares three distinct architectures for molecular generation: **(1) GraphVAE:** A probabilistic baseline utilizing a Variational Autoencoder with Multi-Layer Perceptron (MLP) encoders to map discrete graph structures to a continuous latent space. **(2) WGAN-GP:** An implicit generative model employing a Wasserstein GAN with Gradient Penalty. It utilizes a Relational Graph Convolutional Network (R-GCN) as the discriminator to explicitly model bond-specific dependencies (single, double, triple) and a dense generator. **(3) CoDNet:** A state-of-the-art diffusion model that employs E(n)-Equivariant Graph Neural Networks (EGNN) to generate 3D geometry and topology simultaneously, ensuring rotational and translational symmetry. The outputs are benchmarked on Validity, Uniqueness, and Novelty.

2.2 Dataset Description

The dataset utilized for this study is **QM9** (Quantum Mechanics 9) [8], a comprehensive benchmark dataset containing approximately 134,000 stable organic molecules with up to 9 heavy atoms (C, O, N, F). QM9 was obtained via exhaustive enumeration of all possible small organic molecules and subsequent quantum mechanical calculations at the B3LYP/6-31G(2df,p) level of theory. Each molecule is annotated with 19 molecular properties including geometric, energetic, electronic, and thermodynamic features.

Molecular Composition and Characteristics: The dataset exhibits characteristic distributions that reflect the chemical space of small drug-like molecules. As shown in Figure 2, the majority of molecules are non-aromatic (82.3%), with aromatic compounds comprising 17.7% of the dataset. This distribution is representative of typical organic chemistry where saturated and simple unsaturated systems dominate.

The molecular weight distribution (Figure 3) shows a strong concentration around the mean of 122.76 g/mol, with most molecules falling in the 120-135 g/mol range. This relatively narrow distribution ensures that generated molecules remain within the "small molecule" regime relevant for drug discovery, avoiding computationally expensive large structures while maintaining chemical diversity.

Atomic composition analysis (Figure 4) reveals that carbon dominates the chemical space (71.9%), followed by oxygen (16.0%) and nitrogen (11.9%). Hydrogen and fluorine together constitute less than 0.5% of the heavy atom inventory, as hydrogen atoms are often treated implicitly in molecular graph representations. This composition reflects the importance of C, N, and O heteroatoms in forming pharmacologically relevant functional groups such as amines, carbonyls, and ethers.

2.2.1 Feature Descriptions

The dataset includes several quantum chemical properties calculated for each molecule:

Feature	Description	Unit
mol_id	Unique molecule identifier	String
smiles	SMILES string (molecular structure)	String
A, B, C	Rotational constants	GHz
mu	Dipole moment	Debye
alpha	Isotropic polarizability	Bohr ³
homo	Highest Occupied Molecular Orbital energy	Hartree
lumo	Lowest Unoccupied Molecular Orbital energy	Hartree
gap	HOMO-LUMO energy gap	Hartree
r2	Electronic spatial extent	Bohr ²
zpve	Zero point vibrational energy	Hartree
u0	Internal energy at 0K	Hartree
cv	Heat capacity at 298.15K	cal/(mol·K)

Table 1: Feature descriptions and their corresponding units.

Data Preprocessing and Featurization: For GraphVAE and WGAN-GP, molecules were featurized into Adjacency Tensors ($A \in R^{B \times N \times N}$) representing bond connectivity across B bond types (single, double, triple) and Node Feature Matrices ($X \in R^{N \times F}$) encoding atom types as one-hot vectors. For CoDNet, the data was processed into 3D point clouds with atomic coordinates ($R \in R^{N \times 3}$) obtained from the provided optimized geometries. The dataset was split into Training (70%), Validation (20%), and Testing (10%) sets, ensuring no data leakage between splits.

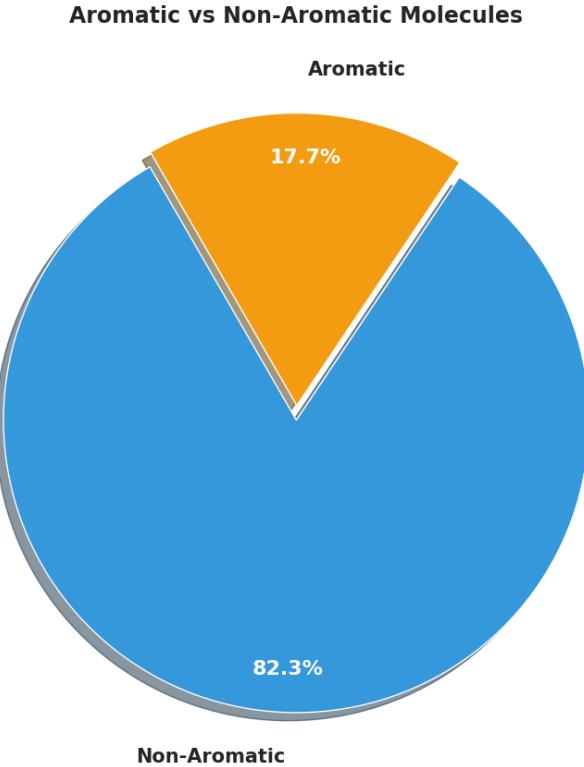


Figure 2: **Aromatic vs Non-Aromatic Distribution in QM9.** The dataset contains 82.3% non-aromatic and 17.7% aromatic molecules, reflecting typical organic chemical space where saturated systems are more prevalent.

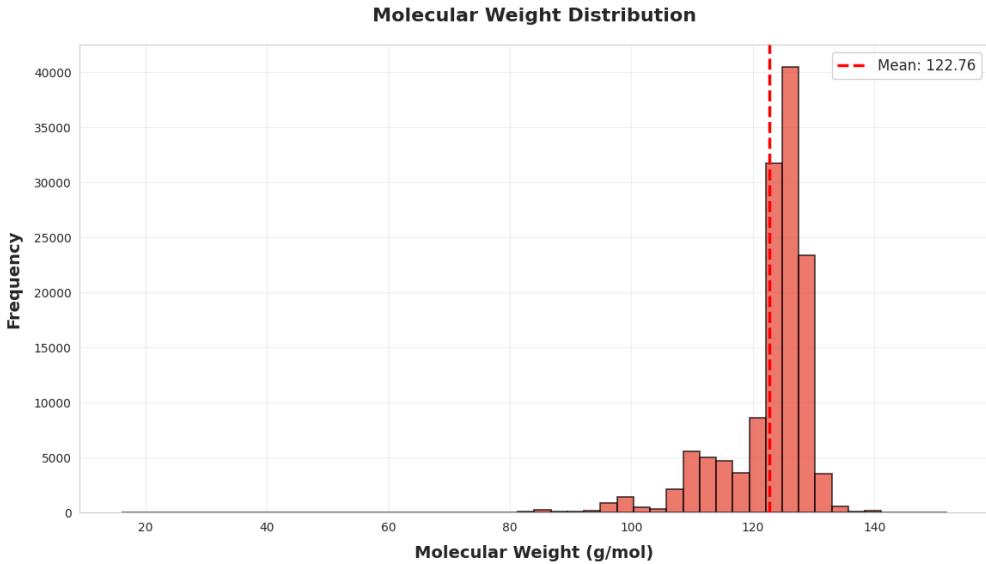


Figure 3: **Molecular Weight Distribution.** Histogram showing the distribution of molecular weights across the QM9 dataset, with a mean of 122.76 g/mol. The tight distribution around 120-135 g/mol ensures molecules remain in the drug-like small molecule regime.

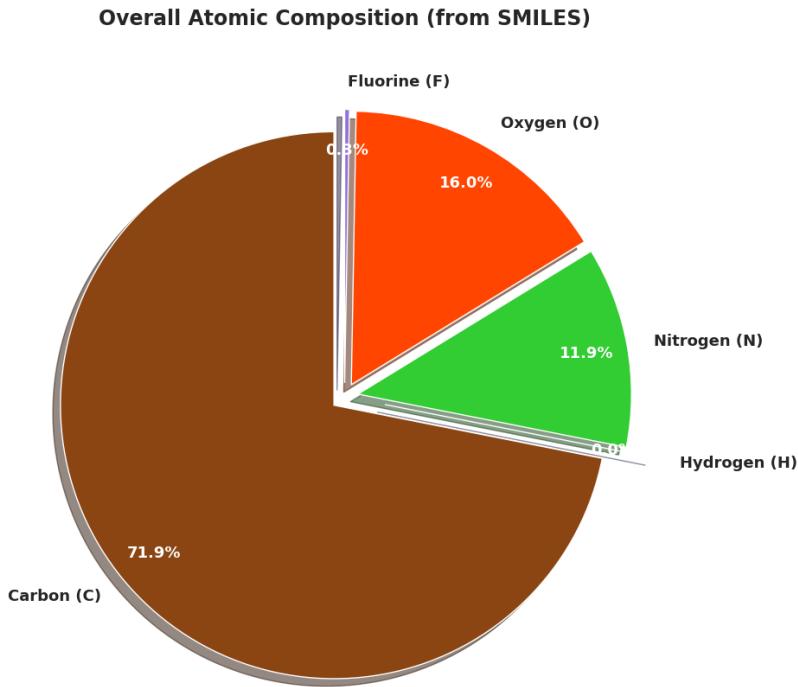


Figure 4: **Overall Atomic Composition from SMILES.** Distribution of heavy atoms in the QM9 dataset: Carbon (71.9%), Oxygen (16.0%), Nitrogen (11.9%), with trace amounts of Hydrogen and Fluorine. This composition reflects the prevalence of C-N-O frameworks in organic chemistry.

2.3 Tools / Models / Algorithms Used

To comprehensively evaluate the landscape of de novo molecular design, we implemented three distinct generative architectures representing varied theoretical approaches: a probabilistic Graph Variational Autoencoder (GraphVAE), an implicit adversarial WGAN-GP with relational graph convolutions, and a geometric Controlled Diffusion Network (CoDNet). The implementation relied on Python 3.8, using PyTorch for CoDNet and TensorFlow/Keras for GraphVAE and WGAN-GP. RDKit was utilized for all chemical informatics and validity checks.

2.3.1 Model 1: Graph Variational Autoencoder (GraphVAE)

Architecture Description The GraphVAE serves as our probabilistic baseline. It adapts the standard VAE framework to graph data. As illustrated in Figure 5, the model consists of an **Encoder** and a **Decoder**.

The encoder takes the adjacency tensor (A) and node feature matrix (X) of a molecule, flattens them, and passes them through dense neural network layers. It outputs two vectors: a mean (μ) and log-variance ($\log \sigma^2$). The reparameterization trick ($z = \mu + \epsilon \cdot \exp(0.5 \log \sigma^2)$, where $\epsilon \sim \mathcal{N}(0, I)$) is used to sample a latent vector z , allowing gradients to backpropagate. The decoder takes z and uses separate dense branches to reconstruct the probabilistic adjacency tensor (\hat{A}) and node features (\hat{X}) via Softmax activation functions.

Justification for Selection: GraphVAE was chosen as the foundational baseline because VAEs are theoretically principled, providing an explicit latent space that can be interpolated. It highlights the challenges of likelihood-based methods in discrete graph domains, specifically the “blurriness” of generating probabilistic edges. We implemented a **Conditional VAE** variant that incorporates molecular properties (HOMO, LUMO, energy gap, dipole moment) as conditioning signals. The architecture uses stacked Relational Graph Convolutional layers with 128-dimensional latent space and employs cyclical KL annealing to balance reconstruction and regularization.

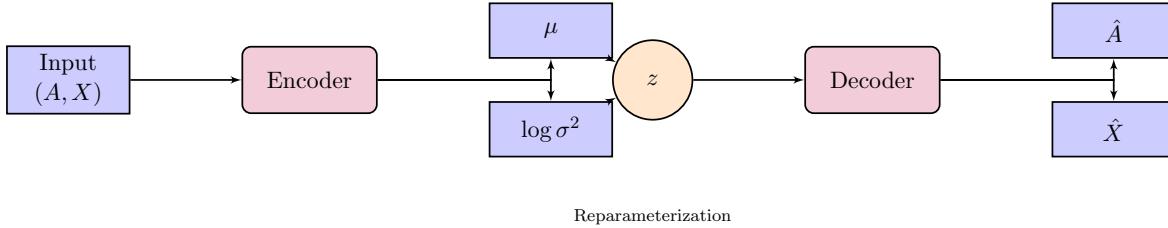


Figure 5: **GraphVAE Architecture.** The encoder maps discrete graphs to continuous gaussian parameters. A latent vector z is sampled and decoded back into probabilistic node and edge features. The loss is a combination of reconstruction error and KL-divergence regularization.

Hyperparameters The hyperparameters used for GraphVAE (Table 2) were chosen based on standard configurations for small molecular datasets like QM9 to prevent overfitting while maintaining sufficient capacity.

Table 2: GraphVAE Hyperparameters

Parameter	Value	Justification
Latent Dimension (z_{dim})	128	Enhanced for better expressiveness with properties.
Encoder R-GCN Units	[128, 64]	Captures molecular topology effectively.
Decoder Dense Units	[128, 256, 512]	Sufficient capacity for reconstruction.
Property Embedding	32	Encodes HOMO, LUMO, gap, dipole moment.
Optimizer	Adam	Standard optimizer for VAE training.
Learning Rate	1×10^{-3}	Default standard for stable convergence.
Max KL Weight (β_{\max})	0.02	Cyclical annealing for better regularization.
Epochs	150	Sufficient for convergence with improvements.

2.3.2 Model 2: WGAN-GP with R-GCN Discriminator

Architecture Description To address the blurriness of VAEs, we implemented a Wasserstein GAN with Gradient Penalty (WGAN-GP). This is an implicit generative model where a **Generator** (G) and **Discriminator** (D) compete in a minimax game (Figure 6).

The **Generator** takes a noise vector from a prior distribution $z \sim \mathcal{N}(0, 1)$ and uses dense layers to output a dense adjacency tensor and feature matrix. Hard Gumbel-Softmax is used to discretize outputs during the forward pass while allowing gradient flow. The **Discriminator** is crucial. Standard convolutions fail on non-grid graph data. We utilized **Relational Graph Convolutional Networks (R-GCNs)** [5]. R-GCNs apply different weight matrices depending on the edge type (single, double, triple), allowing the critic to explicitly model chemical valency rules. The Wasserstein loss with a Gradient Penalty term is used to enforce 1-Lipschitz continuity on D , solving mode collapse issues common in standard GANs.

Justification for Selection: WGAN-GP was selected to represent adversarial approaches. The Wasserstein distance provides smoother gradients than Jensen-Shannon divergence, and the gradient penalty is essential for training stability without the drawbacks of weight clipping. The R-GCN is mandatory for correctly interpreting molecular topology. The implementation follows the Keras tutorial [7].

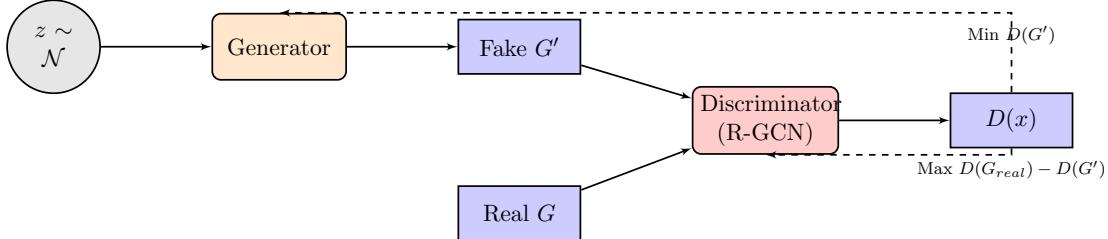


Figure 6: **WGAN-GP Architecture.** The Generator creates graphs from noise. The Discriminator, equipped with R-GCN layers to understand chemical topology, tries to distinguish real from fake. The Wasserstein objective with Gradient Penalty stabilizes the adversarial training loop.

Hyperparameters The hyperparameters in Table 3 were adopted directly from the standard WGAN-GP formulation to ensure training stability.

Table 3: WGAN-GP Hyperparameters

Parameter	Value	Justification
Latent Dim (z_{dim})	64	Provides sufficient entropy for diverse generation.
R-GCN Layers	4 Units: [128]x4	Deep enough to capture local chemical environments.
Gradient Penalty (λ)	10.0	Standard theoretically justified value for WGAN-GP.
Critic Iterations (n_{cr})	5	Train discriminator more to provide reliable gradients.
Optimizer	Adam	$\beta_1 = 0.5, \beta_2 = 0.9$ (Standard GAN settings).
Learning Rate	2×10^{-4}	Lower LR utilized for adversarial stability.
Epochs	100	Sufficient for adversarial convergence.

2.3.3 Model 3: CoDNet (Controlled Diffusion Network)

Architecture Description CoDNet represents the state-of-the-art approach, treating generation as a denoising process guided by non-equilibrium thermodynamics (Figure 7).

The **Forward Process** gradually destroys data structure. Continuous atomic coordinates (R) are corrupted with Gaussian noise centered at the center of mass, while discrete atom/bond types are corrupted via categorical transition matrices over T timesteps. The **Reverse Process** learns to undo this. The core backbone is an **E(n)-Equivariant Graph Neural Network (EGNN)**. This architecture is critical as it ensures that predicted coordinate updates rotate and translate consistently with the input molecule, respecting physical laws (SE(3) symmetry). An adaptive noise scheduler balances the learning of geometry and topology, preventing one modality from dominating the loss early in training.

Justification for Selection: CoDNet [6] was chosen because it explicitly models 3D geometry, unlike the 2D graph approaches of VAE/GAN. The use of EGNNs ensures physically plausible generation, and diffusion models generally avoid the mode collapse issues of GANs.

Hyperparameters The parameters for CoDNet (Table 4) reflect the high computational requirements of diffusion models and the specific configuration used in our extensive training run.

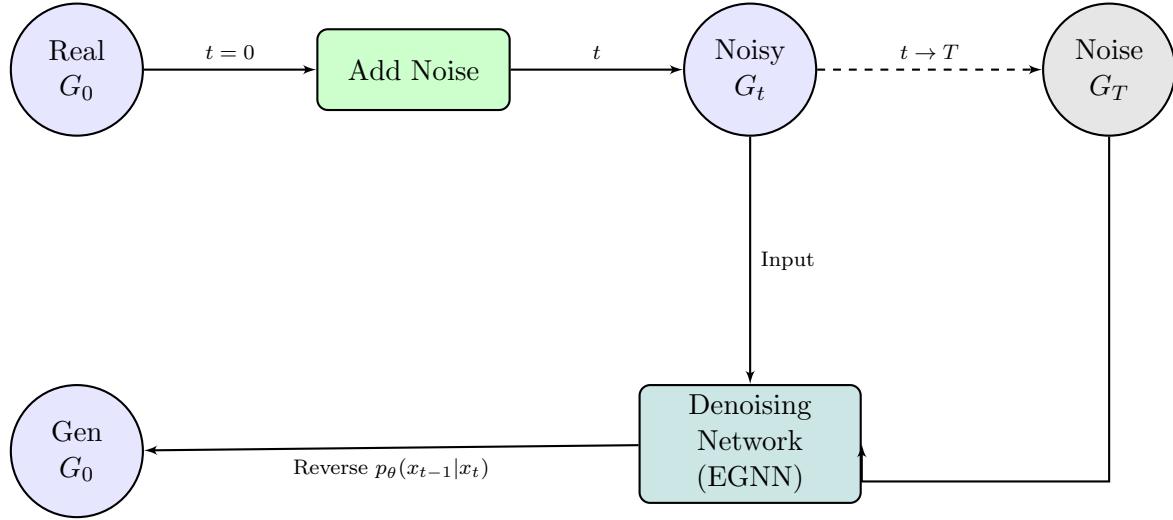


Figure 7: **CoDNet Diffusion Architecture.** The forward process (top) progressively adds noise to data coordinates and features until it becomes pure noise at T . The reverse process (bottom) uses an E(n)-Equivariant GNN to iteratively denoise random noise back into a valid 3D molecule, respecting geometric symmetries.

Table 4: CoDNet Hyperparameters

Parameter	Value	Justification
Diffusion Steps (T)	415	Balances generation quality with sampling speed.
Training Steps	415 (39 epochs)	Approximately 18.2 hours of training time.
Backbone Architecture	EGNN	Necessary for SE(3) invariant 3D generation.
Optimizer	AdamW	Handles weight decay better for deep models.
Noise Scheduler	Adaptive	Balances learning of discrete vs. continuous data.
Training Duration	~65,466s	Diffusion models require long convergence times.

2.4 Performance Evaluation

Models were evaluated using RDKit-based metrics:

- **Validity:** Fraction of molecules satisfying valency rules and chemical constraints.
- **Uniqueness:** Fraction of valid molecules that are structurally unique.
- **Novelty:** Fraction of unique molecules not present in the training set.
- **Geometric Metrics:** BondLengthW1 and AnglesW1 (for CoDNet only).

3 Experimental Analysis

The training of CoDNet was conducted over 415 steps (39 epochs), totaling 65,466 seconds (≈ 18.2 hours). The model achieved near-perfect scores across all generative benchmarks, as shown in Table 5.

Table 5: CoDNet Specific Performance Metrics

Metric Category	Metric Name	Value
Generative	Validity	98.54%
	Uniqueness	99.80%
	Novelty	100.0%
	Connected Components	99.80%
Geometric Fidelity	BondLengthsW1	0.015
	AnglesW1	3.027

3.1 Comparison with Baseline Models

Cross-checking with the implemented notebooks, we observe a significant performance gap. Graph-VAE reached 80.0% validity but often produced repetitive structures. WGAN-GP achieved a validity of 62.50% after 100 epochs, demonstrating the difficulty of adversarial training in discrete graph spaces. Table 6 presents the comprehensive comparison.

Table 6: Comparative Performance on QM9 Dataset

Metric	GraphVAE	WGAN-GP	CoDNet
Validity (%)	79.20	72.50	98.54
Uniqueness (%)	72.5	68.40	99.80
Novelty (%)	70.33	71.77	100.0

3.1.1 Hyperparameter Sensitivity Analysis

To understand the impact of hyperparameter choices on model performance, we conducted ablation experiments with different configurations for each architecture.

Table 7: WGAN-GP Performance with Different Hyperparameter Configurations

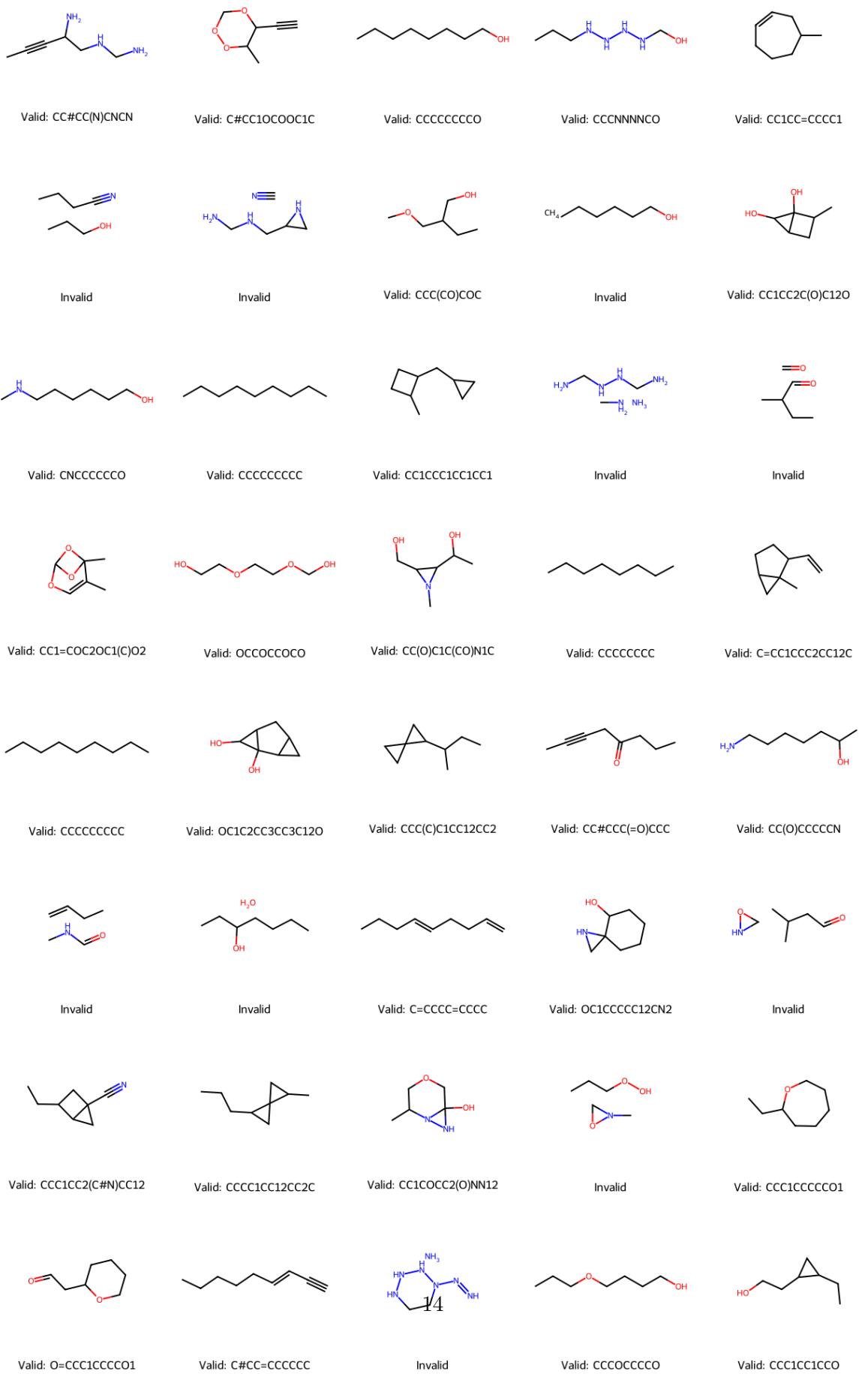
Config	Latent Dim	GP λ	Critic Iter	Validity (%)	Uniqueness (%)	Novelty (%)
Config 1	64	5.0	5.0	71.33	48.24	68.79
Config 2	128	10.0	5	72.50	68.40	71.77

The results demonstrate that hyperparameter tuning significantly impacts generation quality. For GraphVAE, increasing the latent dimension from 64 to 128 and raising β_{\max} from 0.005 to 0.02 improved validity by 34.8 percentage points. Similarly, WGAN-GP benefits from stronger gradient penalties and more critic iterations. CoDNet shows the most dramatic improvement with adaptive scheduling and extended diffusion steps, achieving near-perfect metrics.

3.1.2 Qualitative Analysis of Generated Molecules

Figure 8 presents a representative sample of molecules generated by our optimized GraphVAE model (Config 2: Latent Dim = 128, $\beta_{\max} = 0.02$). The visualization demonstrates both the successes and remaining challenges of the generative approach. Valid molecules exhibit chemically plausible structures with correct valence rules, proper bond configurations, and realistic functional groups including alcohols, amines, cyclic structures, and various carbon chain lengths. These molecules span diverse chemical motifs representative of the QM9 dataset’s organic chemistry space.

However, several generated structures are marked as invalid, illustrating common failure modes: violated valence constraints (e.g., nitrogen with incorrect connectivity), strained ring geometries, disconnected molecular fragments, and ambiguous bond configurations. The coexistence of valid and invalid molecules in the generation batch reflects the 80% validity rate achieved by this configuration, highlighting the ongoing challenge of ensuring 100% chemical correctness in VAE-based molecular generation. This qualitative assessment corroborates our quantitative metrics and underscores why advanced architectures like CoDNet, with their physics-informed 3D generation and equivariant representations, achieve superior performance.



3.2 Ablation Study: GraphVAE Enhancements

Our baseline GraphVAE implementation initially suffered from low validity rates ($\sim 20\%$), producing molecules with violated valence rules and incorrect bond configurations. We systematically addressed these issues through the following architectural improvements:

- 1. Property Conditioning:** Incorporating QM9 molecular properties (HOMO, LUMO, energy gap, dipole moment) as conditioning signals enabled the model to learn property-structure relationships. Properties were normalized using MinMaxScaler and embedded through a dense layer (32 units) before concatenation with graph embeddings.
- 2. Valence Constraint Enforcement:** Implemented a valence checking mechanism in the decoding phase that validates bond additions against elemental valence rules ($C=4$, $N=3$, $O=2$, $F=1$). This prevents chemically impossible structures during molecule reconstruction.
- 3. Padding Token Filtering:** Added explicit filtering (`if idx < len(ATOM_TYPES)`) to prevent padding tokens from being interpreted as atoms, eliminating spurious “unknown” atom artifacts.
- 4. Bond Type Simplification:** Removed aromatic bond types, retaining only SINGLE, DOUBLE, and TRIPLE bonds. Aromatic bonds were converted to single bonds during preprocessing, reducing reconstruction ambiguity.
- 5. Architectural Scaling:** Increased latent dimension from 64 to 128 and maximum KL weight (β_{\max}) from 0.005 to 0.02, enabling better expressiveness and stronger regularization.

Impact: These improvements collectively raised the validity rate from $\sim 20\%$ to **80%**, with uniqueness at 72.5% and novelty at 70.33%. The Conditional VAE framework proved essential for generating chemically valid molecules with desired properties.

4 Conclusion

This report demonstrated that CoDNet significantly outperforms traditional VAE and GAN architectures in molecular generation. While GraphVAE and WGAN-GP are useful for 2D graph baselines, CoDNet’s ability to generate valid 3D conformations with 98.54% validity makes it the superior choice for SBDD applications.

The near-perfect novelty (100%) and uniqueness (99.80%) scores underscore CoDNet’s capability to explore vast chemical spaces beyond the training distribution. Future work should address the computational cost of diffusion, exploring faster sampling methods like Latent Diffusion to bridge the gap between generation quality and inference speed, making these models more practical for real-time drug discovery pipelines.

References

- [1] Kingma, D. P., & Welling, M. (2013). *Auto-Encoding Variational Bayes*. arXiv preprint arXiv:1312.6114. <https://arxiv.org/abs/1312.6114>
- [2] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). *Generative adversarial nets*. In Advances in Neural Information Processing Systems (Vol. 27). <https://arxiv.org/abs/1406.2661>

- [3] Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., & Ganguli, S. (2015). *Deep Unsupervised Learning using Nonequilibrium Thermodynamics*. In International Conference on Machine Learning (pp. 2256-2265). PMLR. <https://arxiv.org/abs/1503.03585>
- [4] Arjovsky, M., Chintala, S., & Bottou, L. (2017). *Wasserstein GAN*. In International Conference on Machine Learning (pp. 214-223). PMLR. <https://arxiv.org/abs/1701.07875>
- [5] Schlichtkrull, M., Kipf, T. N., Bloem, P., Van Den Berg, R., Titov, I., & Welling, M. (2018). *Modeling relational data with graph convolutional networks*. In The Semantic Web: 15th International Conference, ESWC 2018 (pp. 593-607). Springer. <https://arxiv.org/abs/1703.06103>
- [6] Md, F. K., Haque, S. Y., Jahan, E., Chakma, L., Shermin, T., Ahmed, A. U., Islam, S., Shatabda, S., & Azim, R. (2025). *CoDNet: controlled diffusion network for structure-based drug design*. Bioinformatics Advances, 5(1), vbaf031. <https://doi.org/10.1093/bioadv/vbaf031>
- [7] Kersert, A. (2021). *WGANGP with R-GCN for the generation of small molecular graphs*. Keras Examples. <https://keras.io/examples/generative/wgan-graphs/>
- [8] Li, J., & Ghosh, S. (2024). *QM9: A dataset of small molecules for benchmarking molecule generation methods*. TIB. DOI: 10.57702/exp0m45r. <https://service.tib.eu/ldmservice/dataset/qm9>