SUPPLEMENTARY INFORMATION

Benchmarking Study of Deep Generative Models for Inverse Polymer Design

Tianle Yue,[1] Lei Tao,[2] Vikas Varshney,[3] Ying Li[1*]

[1]Department of Mechanical Engineering, University of Wisconsin-Madison, Madison, WI 53706, United States

[2]Department of Mechanical Engineering, University of Connecticut, Storrs, CT 06269, United States

[3]Materials and Manufacturing Directorate, Air Force Research Laboratory, Wright-Patterson Air Force Base, OH 45433, United States

*Corresponding author: yli2562@wisc.edu

Table S1. Count of each atom type present in real polymer dataset based on PolyInfo.

Atom	Count
C	325910
O	50008
N	20415
F	9063
S	3399
Cl	754
P	432
Si	984
Br	360
Sn	13
Na	66
As	4
Ca	2
Pb	2
Te	3
Ge	12
K	8
B	12
Se	29
I	29
Co	2
Zn	2
Cd	2
Li	8

Table S2. Count of each atom type present in hypothetical polyimide dataset based on PubChem.

Atom	Count

C	21198924
N	2681288
F	229489
O	5348624
S	206982

Table S3. Count of each atom type present in hypothetical polyimide dataset based on GDB-13.

Atom	Count
C	29742733
N	3013512
F	284368
O	6083206
S	105153
B	3233
Cl	60317
P	9347
Si	52218
I	12136
Br	16823
Ar	251
Sn	280
As	1408
Se	258
Al	544
In	3
Hg	3

Table S4. Source codes for studied generative models.

Model	GitHub repository
AAE	https://github.com/molecularsets/moses/tree/dd7ed6ab38e23afd3ef5371d67939a1760bd8599/moses/aae
VAE	https://github.com/molecularsets/moses/tree/dd7ed6ab38e23afd3ef5371d67939a1760bd8599/moses/vae
CharRNN	https://github.com/molecularsets/moses/tree/dd7ed6ab38e23afd3ef5371d67939a1760bd8599/moses/char_rnn
ORGAN	https://github.com/molecularsets/moses/tree/dd7ed6ab38e23afd3ef5371d67939a1760bd8599/moses/organ
REINVENT	https://github.com/undeadpixel/reinvent-randomized/tree/2eeca2d73e197943bc7f704022d30eee14c49cb6
GraphINVENT	https://github.com/MolecularAI/GraphINVENT/tree/6ef587ddb983f0c853dc8bc7b418f43cb69420c9

Table S5. Source codes for studied generative models with reinforcement learning.

Model	GitHub repository
CharRNN	https://github.com/aspuru-guzik-group/Tartarus/tree/b112b811616bee01fb3348e867b7406e4e6a62f4/models_jupyter
GraphINVENT	https://github.com/olsson-group/RL-GraphINVENT/tree/d4629a3c411c793e1ed1682592d5bf67937564a1
REINVENT	https://github.com/MolecularAI/Reinvent/tree/99b8f28c2a76196017eabf23118195ae546f5714

Table S6. Scores for each metric of six generative models on the real homopolymer dataset from PolyInfo.

	Train	AAE	CharRNN	GraphINVENT	ORGAN	REINVENT	VAE
f_v	1	0.273	0.866	0.738	0.005	0.601	0.498
f_{10k}	1	0.867	0.696	0.581	0.879	0.833	0.820
FCD	0.738	4.931	1.232	6.493	49.039	2.266	3.148
SNN	0.629	0.538	0.621	0.467	0.153	0.609	0.596
IntDiv	0.866	0.856	0.868	0.886	0.573	0.860	0.857

Table S7. Scores for each metric of six generative models on the hypothetical polyimide dataset synthesized based on GDB-13.

	Train	AAE	CharRNN	GraphINVENT	ORGAN	REINVENT	VAE
f_v	1	0.703	0.675	0.191	0.331	0.962	0.658
f_{10k}	1	1	0.999	0.907	0.707	0.999	0.999
FCD	0.067	1.008	0.681	7.030	39.164	0.033	0.697
SNN	0.598	0.554	0.544	0.330	0.357	0.574	0.548
IntDiv	0.843	0.852	0.855	0.880	0.508	0.855	0.855

Table S8. Scores for each metric of six generative models on the hypothetical polyimide dataset synthesized based on PubChem.

	Train	AAE	CharRNN	GraphINVENT	ORGAN	REINVENT	VAE
f_v	1	0.925	0.959	0.144	0.002	0.977	0.901
f_{10k}	1	0.999	1	0.857	0.167	1	1
FCD	26.892	0.362	0.027	8.557	56.636	0.032	0.067
SNN	0.291	0.597	0.596	0.319	0.423	0.603	0.598
IntDiv	0.830	0.848	0.843	0.886	0.645	0.843	0.843

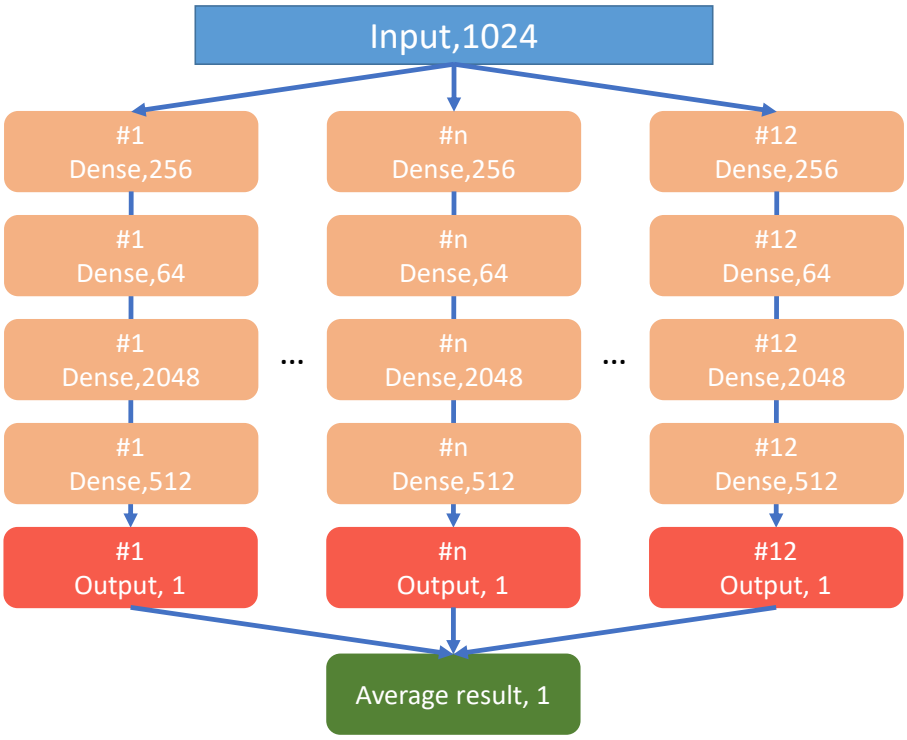

Figure S1. The architecture of the FNN model. The network's hyperparameters were optimized through grid search.

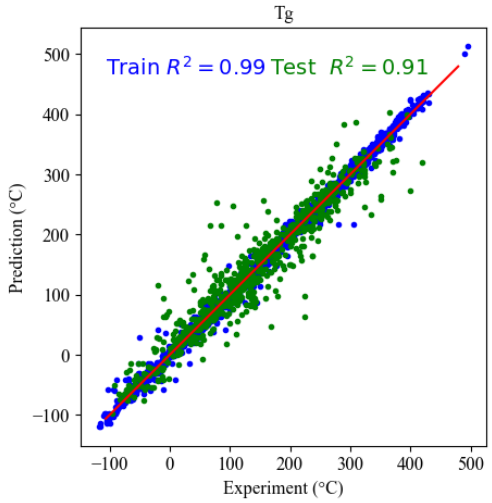

Figure S2. Parity plot of FNN models for T_g.

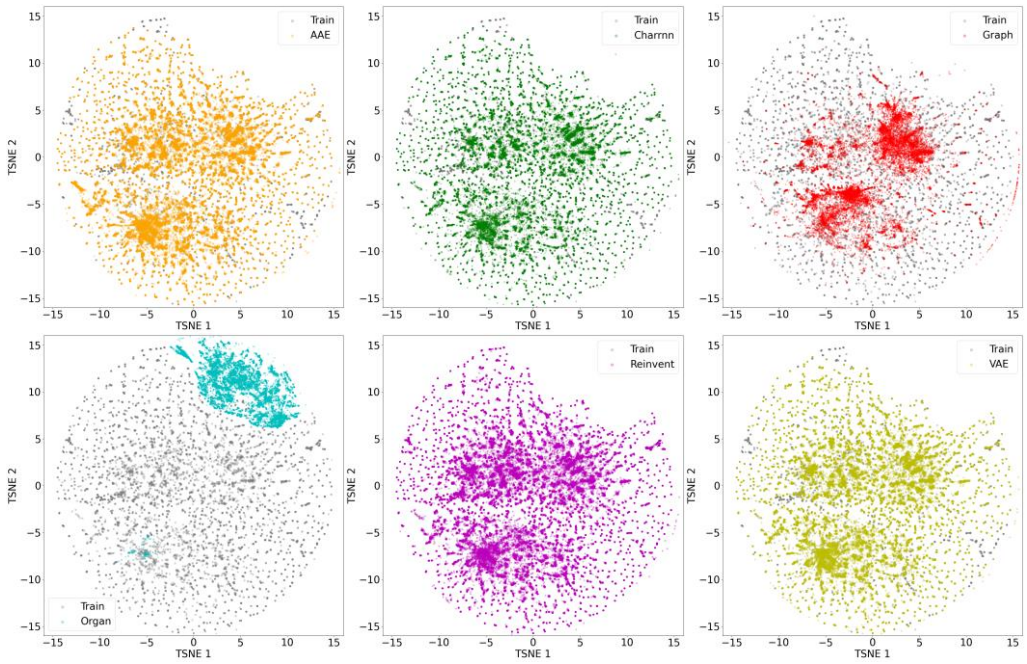

Figure S3. The individual t-SNE results for each model trained on the hypothetical polyimide dataset generated based on GDB-13.

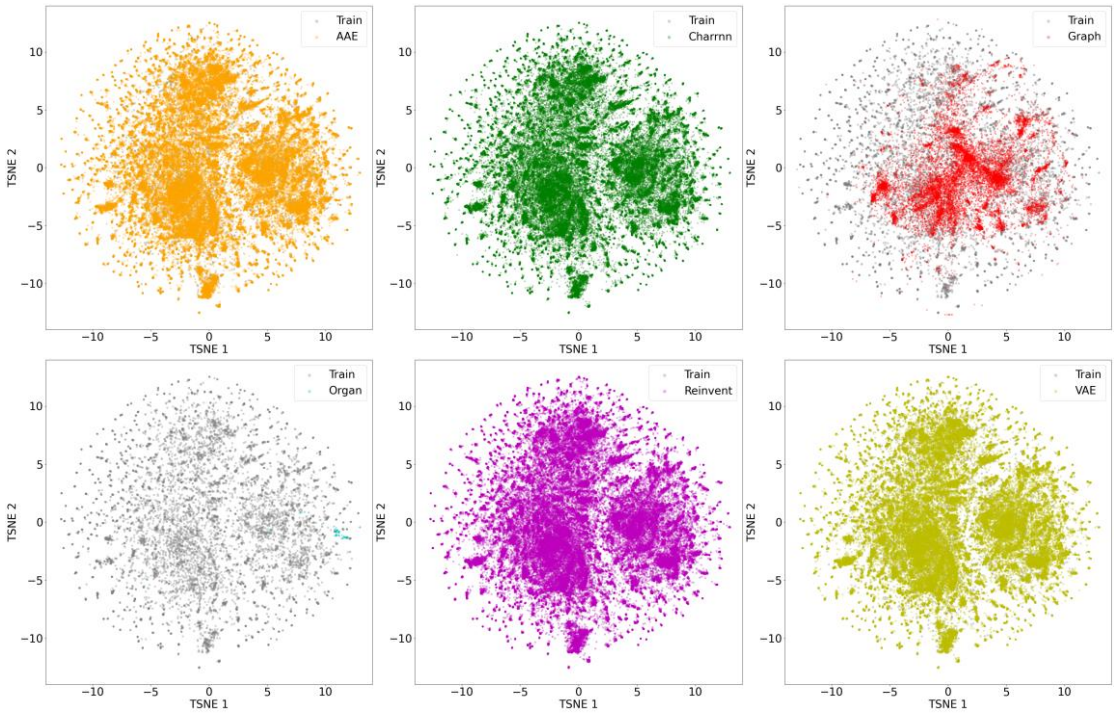

Figure S4. The individual t-SNE results for each model trained on the hypothetical polyimide dataset generated based on PubChem.

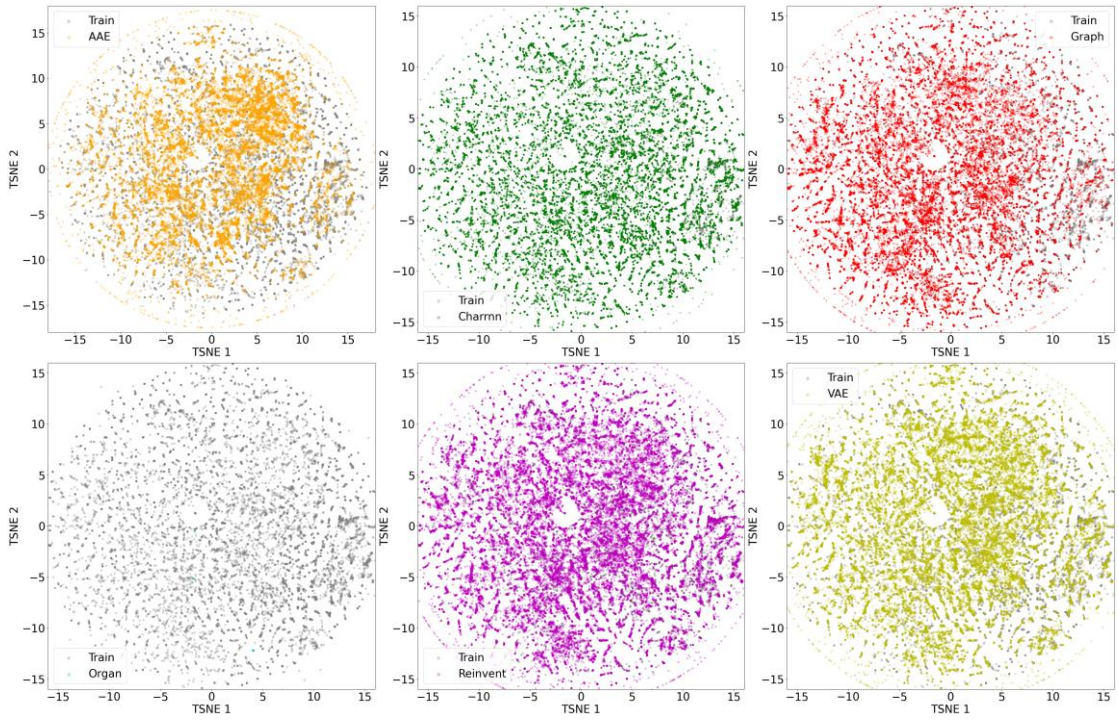

Figure S5. The individual t-SNE results for each model trained on the real polymers dataset collected from PolyInfo.

Table S9. Scores for each metric of five generative models on the real homopolymer dataset from PolyInfo with BigSMILES.

	Train	AAE	CharRNN	ORGAN	REINVENT	VAE
f_v	1	0.366	0.768	0.582	0.082	0.476
f_{10k}	1	0.895	0.682	0.515	0.752	0.875
FCD	0.738	3.948	1.254	13.303	12.363	2.313
SNN	0.629	0.518	0.603	0.563	0.431	0.574
IntDiv	0.866	0.857	0.873	0.857	0.857	0.860

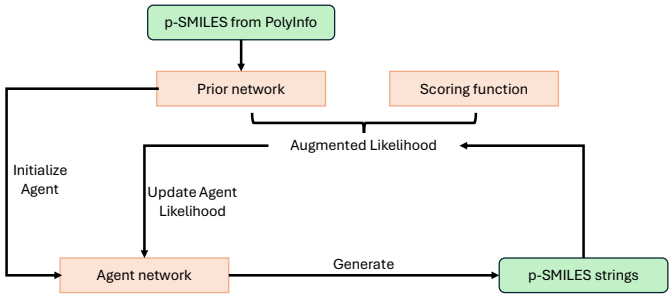

Figure S6. The Agent. Illustration of how the model is constructed. Starting with a Prior network trained on the PolyInfo dataset, the Agent is trained using the augmented likelihood of the generated p-SMILES strings. This training process gradually shifts the probability distribution from the original prior distribution toward one influenced by the desirability of the generated structures. The approach is inspired by Jaques et al.,[1] employing a policy-based reinforcement learning method that introduces a novel cost function to reduce reliance on handwritten rules and mitigate the issue of generating overly simplistic structures.[2]

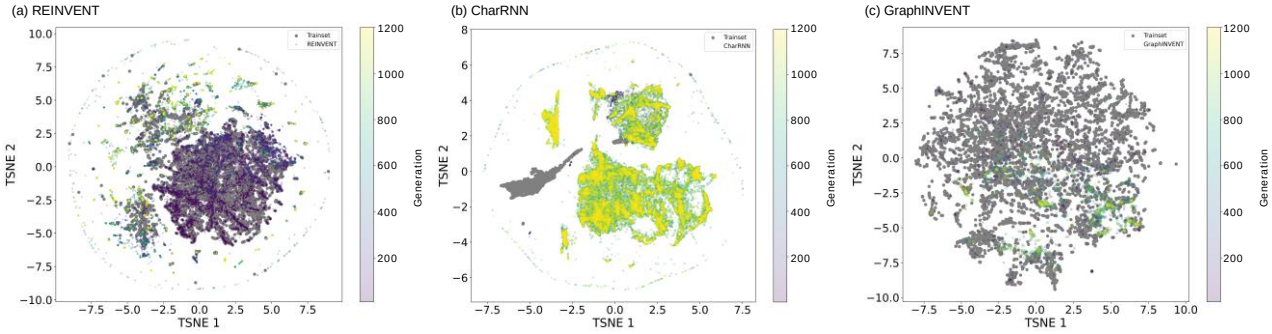

Figure S7. The chemical space covered by the training set is shown in grey, with a color gradient from deep purple to yellow representing the chemical space of polymer structures generated across training epochs from 0 to 1200 for (a) REINVENT, (b) CharRNN, and (c) GraphINVENT.

Details of model parameters

VAE The encoder uses a GRU-based architecture with a hidden dimensionality of 256, a single layer, and a dropout rate of 0.5. The decoder also employs a GRU-based structure, with a hidden dimensionality of 512, three layers, and no dropout. The latent space is represented as a vector of dimensionality 128. Word embeddings are initialized with pre-trained vectors. The training process includes gradient clipping at a value of 50 and employs cosine annealing with restarts for learning rate scheduling. Learning rate is set at 0.0003. The KL divergence weight starts at 0 and linearly increases to 0.05 from a specified epoch. The total number of epochs is 100.

AAE The embedding layer has a size of 32, which is shared by the encoder and decoder. The encoder is configured with a hidden size of 512, a single LSTM layer, and no dropout. The decoder mirrors the encoder with a hidden size of 512, two LSTM layers, and no dropout. The latent space is represented by a vector of size 128, ensuring compact yet meaningful representations. For the discriminator, two fully connected layers with 640 and 256 neurons are used to differentiate between real and generated latent codes. The

training process includes an initial learning rate of 0.001, with periodic learning rate decay every 20 epochs and a decay factor of 0.5. The training runs for 120 epochs. The batch size is set to 512, and discriminator training steps are performed once per autoencoder training step.

Char-RNN The Char-RNN model is configured with 3 LSTM layers and a hidden size of 768. The embedding size matches the vocabulary size, and a dropout rate of 0.2 is applied between LSTM layers. Training parameters include 80 epochs a batch size of 64, an initial learning rate of 0.001, a step size of 10 epochs, and a decay factor of 0.5.

ORGAN The ORGAN model is configured with a default embedding size of 32 for both the generator and discriminator. The generator consists of 2 LSTM layers with a hidden size of 512 and no dropout, while the discriminator uses multiple convolutional layers with varying feature sizes and kernel sizes, also with no dropout. The training process includes 50 epochs of generator pretraining and 50 epochs of discriminator pretraining, followed by 1000 iterations of policy gradient training. The reward weight for the policy gradient is set to 0.7, with 16 rollouts per iteration. The learning rate is 0.0001, and gradient clipping is applied at a value of 5. The batch size is set to 64, with a maximum sequence length of 100, and one worker is used for data loading.

REINVENT The model is trained for 100 epochs with a batch size of 128. The learning rate starts at 1e-4 and can follow an exponential decay mode. The learning rate is updated every epoch by a factor of 0.8. Gradients are clipped at a norm of 1.0.

GraphINVENT The default model architecture is GGNN, which uses 4 layers in the message-passing network and a message size of 100. The number of message-passing steps is set to 3. Training parameters include a batch size of 1000, a learning rate of 0.0001. The training process runs for 100 epochs with sampling every 10 epochs. Additional configurations include the use of canonical RDKit atom ordering, a breadth-first decoding route, and TensorBoard for logging.

REFERENCE

(1) Jaques, N.; Gu, S.; Turner, R. E.; Eck, D. Tuning recurrent neural networks with reinforcement learning. **2017**.
(2) Olivecrona, M.; Blaschke, T.; Engkvist, O.; Chen, H. Molecular de-novo design through deep reinforcement learning. *Journal of cheminformatics* **2017**, *9*, 1-14.