
TECHNICAL REPORT: IMPROVING THE PROPERTIES OF MOLECULES GENERATED BY LIMO

Vineet Thumuluri

Computer Science and Engineering
University of California San Diego
vthumuluri@ucsd.edu

Peter Eckmann

Computer Science and Engineering
University of California San Diego
peckmann@ucsd.edu

Michael K. Gilson

Skaggs School of Pharmacy and Pharmaceutical Sciences
University of California San Diego
mgilson@health.ucsd.edu

Rose Yu

Computer Science and Engineering
University of California San Diego
roseyu@ucsd.edu

ABSTRACT

This technical report investigates variants of the Latent Inceptionism on Molecules (LIMO) framework to improve the properties of generated molecules. We conduct ablative studies of molecular representation, decoder model, and surrogate model training scheme. The experiments suggest that an autogressive Transformer decoder with GroupSELFIES achieves the best average properties for the random generation task.

1 LIMO FRAMEWORK

LIMO (Eckmann et al. (2022)) is a molecular generation technique that improves a given set of properties by mapping molecules to a latent space and uses inexpensive property surrogates to turn a discrete space optimization problem into a continuous one. There are multiple components to this framework which are described below.

1.1 GENERATIVE MODEL

Given a dataset of molecular strings, a generative model is first trained to mimic the distribution of SELFIES Krenn et al. (2020) tokens, i.e. smallest units of a molecule string. In this case, a Variational AutoEncoder (VAE) Kingma & Welling (2022), is trained to map molecules (X) to a latent space (Z) using an encoder (E) i.e. $E : X \rightarrow Z$ and decoder (D) that learns an inverse mapping i.e. $D : Z \rightarrow X$, such that it is easy to reconstruct X from Z using the decoder (D) i.e. $D(E(X)) \approx X$ (Reconstruction), and the latent space is close to a unit normal distribution i.e. $D_{KL}(z||\mathcal{N}(0, I)) \approx 0$ (Regularization).

This defines a generative model $p(x) = \int p(x|z)p(z)$. The VAE parameters are obtained by minimizing the ELBO loss Kingma & Welling (2022). In this work, we use a variant that weights the regularization term to avoid a problem during optimization where the KL term vanishes Higgins et al. (2017).

$$L_{\text{BETA}}(\phi, \beta) = -\mathbb{E}_{\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x} | \mathbf{z}) + \beta D_{\text{KL}}(q_{\phi}(\mathbf{z} | \mathbf{x}) || p_{\theta}(\mathbf{z}))$$

β value of 0.1 was used in LIMO (Eckmann et al. (2022)).

1.2 SURROGATE MODEL

LIMO uses a property predictor model to map the molecular representation to property values Eckmann et al. (2022). The molecular representation can either be a sample from the latent space or the decoder output. The property predictor acts as a fully differentiable and faster surrogate model

to the oracle property predictor tools that are often computationally expensive. The properties are generally scalar values and thus the parameters of the neural network are learned by minimizing the mean-squared error between the predictions and oracle property values.

1.3 REVERSE OPTIMIZATION

The key idea of LIMO is the latent inceptionism technique to reverse optimize the latent space and generate molecules with desired properties. Specifically, a large number of random points are sampled according to the unit normal distribution in the latent space. These serve as the starting points for the search. Molecules with the optimized properties are generated by local optimization in the latent space with the gradient approximated using the trained surrogate model. The overall objective being optimized is a weighted sum of one or more property values. All of the minima are then decoded to molecules and the top molecules, measured using the oracle property value predictor, are chosen.

2 ATTEMPTED MODIFICATIONS TO LIMO

We conduct an extensive ablation study of the LIMO framework, investigating different variations of LIMO components. These include changes to the tokenization of the molecular strings, the decoding model from the latent space to the molecule, and changes to how the property predictor is trained.

2.1 MOLECULAR REPRESENTATION

To represent molecules, LIMO used SELFIES Krenn et al. (2020), a molecular string representation that ensures the tokens can be combined in any way to always form valid molecules. The molecular strings can be decomposed into tokens in many ways. GroupSELFIES Cheng et al. (2022) builds on top of the chemical validity guarantees of SELFIES by enabling group tokens, thereby creating additional flexibility to the representation. It extends SELFIES where the tokens can be larger fragments and depend on the dataset and the fragmentation technique.

In this work, we use the default fragmentation method from Cheng et al. (2022) with different datasets:

- (1) GS-Paper: GroupSELFIES tokens from the original paper Cheng et al. (2022).
- (2) GS-USPTO: GroupSELFIES tokens extracted from USPTO reactions Lowe (2017).
- (3) GS-Zinc: GroupSELFIES tokens extracted from ZINC250K molecules Akhmetshin et al. (2021).

Table 1 shows the total number of tokens, the average and maximum sequence length for each of the molecular string representations for the training dataset (ZINC 250K) considered in this study.

Table 1: Statistics of different molecular string representation schemes when tokenizing the training dataset (ZINC250K).

	Tokenizer			
	SELFIES	GroupSELFIES		
		ZINC250K	USPTO	Paper
Total tokens	108	304	242	248
Max length	72	75	77	93
Avg. length	37.43	37.07	36.61	29.86

2.2 DECODER MODEL

The decoder in LIMO maps from latent space to a molecular string representation. Such mapping can be approximated in multiple ways. Here we discuss two broad strategies i.e. Non-autoregressive (NAR) and Autoregressive (AR) decoder models. In this work, both of these are implemented using the transformer architecture Vaswani et al. (2023).

2.2.1 NON-AUTOREGRESSIVE MODELS

LIMO Eckmann et al. (2022) uses an MLP decoder by modelling the joint distribution of the target tokens (y) as conditionally independent i.e. $P(y) = \prod_{i=1}^n P(y_i | Z)$. While this has the advantage of inference being parallelizable, it suffers from the multi-modality problem Gu et al. (2018). There have been multiple works that try to mitigate this issue through iterative refinement such as CMLM Yang et al. (2021), which uses masked language modeling and multi-step decoding. At inference time, first, the input is fully masked and in each subsequent iteration, the least likely tokens are masked and the procedure is repeated. A follow-up work CMLMC Huang et al. (2022), aims to improve the training objective of CMLM by addressing the mismatch at inference by including a token denoising loss in addition to the masking loss. Another advantage of NAR models is the ability to do conditional generation by constraining on a given scaffold.

2.2.2 AUTOREGRESSIVE MODELS

Autoregressive models, such as Transformers, are slower since only one token is decoded at each step and the cost scales at least linearly with the length of the target sequence length. However, the full joint distribution can be modeled i.e. $P(y) = \prod_{i=1}^n P(y_i | y_1, \dots, y_{i-1}, Z)$, and thus does not suffer from the multi-modality problem. AR models thus have been shown to have superior task performance Ren et al. (2020). In contrast to NAR models, conditional generation is non-trivial, thus losing the ability to do scaffold-constrained generation.

2.3 SURROGATE MODEL TRAINING

The property predictor surrogate model training depends on the VAE latent space i.e. the input to the surrogate predictor used in LIMO is a decoded representation of the point in the latent space, there are two approaches to optimizing the whole system.

The first is to perform the training sequentially, in which case, the VAE is trained to convergence using the molecular string dataset and then its parameters are frozen. This ensures a fixed latent space. Random molecules are generated and their property values are computed to create the training dataset for the neural network-based surrogate.

An alternative strategy is to jointly train both the VAE and the surrogate. In this case, the property values of the training molecules need to be computed. An advantage of this approach is that the latent space can be informed of the desired properties explicitly, however, the generative model optimization becomes harder as there are more training objectives to balance (empirically we find the losses to be higher) and a lot more oracle compute is required. In this study we chose to use a surrogate that directly predicts using the latent space to simplify the optimization.

3 EXPERIMENTS

3.1 SETUP

All methods are trained on the ZINC-250K dataset. The parameters are optimized using the Adam optimizer Kingma & Ba (2017), using a cosine-annealing learning rate schedule for one cycle Loshchilov & Hutter (2017) for a total of 100K iterations. To prevent KL from vanishing across models with minimal tuning, a cyclic beta schedule was used while training the VAE Fu et al. (2019). We follow the LIMO paper and use the following properties to evaluate the quality of generated optimized molecules.

- **SA**: Synthetic-accessibility, computed with SAScorer Ertl & Schuffenhauer (2009). A score between 1 (easy to make) and 10 (very difficult to make).
- **QED**: Quantitative Estimate of Drug-likeness, estimated by RDKit Landrum. A value between 0 (non-drug-like) and 1 (drug-like).
- **BA**: Binding Affinity to the human estrogen receptor (PDB 1ERR), computed with AutoDock-GPU Santos-Martins et al. (2021). Outputs a prediction of the ΔG in kcal/mol, with lower values indicating stronger binding. We used 0 when the value could not be computed.

Table 2: Property comparison of all generated molecules from different model variants.

Model Variants		Count with valid AutoDock BA	BA ↓	SA ↓	QED ↑
LIMO	Default-Retrained	9813	-4.67	4.46	0.57
	GS-Paper	8729	-5.44	2.89	0.62
	GS-USPTO	9947	-5.56	4.46	0.61
	GS-Zinc	9868	-5.57	4.63	0.59
AR	Default	9194	-5.03	4.93	0.56
	Joint-Z	9229	-5.57	3.56	0.76
AR-JointZ	GS-Paper	9768	-5.26	2.91	0.81
	GS-USPTO	9939	-5.58	3.28	0.73
	GS-Zinc	9967	-5.74	3.39	0.80
CMLMC		8758	-5.11	3.65	0.30
ZINC250K (Training dataset)		242591	-5.35	3.05	0.73

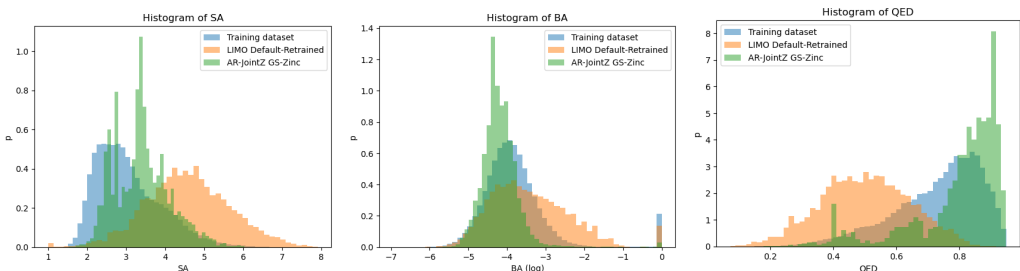


Figure 1: Distributions of property values for optimized molecules generated by LIMO, AR-JointZ GS-Zinc, and the training dataset.

3.2 PROPERTY OPTIMIZATION RESULTS

We apply reverse optimization to generate a set of random molecules from the trained generative model. For each experiment, the property optimization step was applied with 10,000 starting points and all of the resulting minima are considered.

We consider the following model variants (1) LIMO: The original LIMO Eckmann et al. (2022). (2) AR: LIMO with decoder replaced with an autoregressive Transformer model. (3) AR-JointZ: AR with surrogate trained simultaneously with VAE. (4) CMLMC: Non-Autoregressive decoder model from Huang et al. (2022) For LIMO, AR and AR-JointZ, we further consider different molecular string representation schemes (see Section 2.1 for more details on the various molecular representations).

The results in Tables 2 show the number of binding ligands and the average property scores of those ligands for the human estrogen receptor (PDB 1ERR) target by considering 10,000 generated molecules each. We also report in Appendix Table 3, 4, 5 the results for molecules with the top 100 BA, the molecules with the top 100 SA, and the molecules with the top 100 QED respectively. The AR VAE decoder with the GroupSELFIES variants has the molecules with the best generated SA, QED (tokens provided by Cheng et al. (2022)), and BA (tokens extracted from ZINC250K).

We also include the corresponding property values of the training dataset (ZINC250K) as a reference. Note that the property values of generated molecules are even better than those of the training dataset. This demonstrates the great potential of using LIMO for de novo drug discovery.

Figure 1 compares the distributions of property values for optimized molecules generated by LIMO, AR-JointZ GS-Zinc, and the training dataset. We can see that AR-JointZ with GS-ZINC representation improves the original LIMO model across all metrics. The model even outperforms the training dataset in terms of BA and QED values.

3.3 LATENT SPACE ANALYSIS

While Table 2 shows that the trained models are able to match the property distribution in the training dataset approximately, Table 3, 4, 5 in Appendix show that the optimized molecules, however, fall short of molecules with the best property values in the training dataset. This difference in the best property values is due to the surrogate model-based optimization in the latent space.

We first investigate the organization of the latent space and identify features that would indicate the ease of optimization in this space. Figure 2 shows the property value distribution in the learned latent space of LIMO. In the case of sequentially training the VAE and then the surrogate, the latent space receives no explicit feedback from the property values. Hence, the property values are not smoothly distributed, i.e. molecules decoded from the same neighborhood of the latent space can have large variations in their property values.

Since the surrogate is a simple approximation of the true latent space to property mapping, we measure the mean-squared error and the Dirichlet energy (Equation 1) to quantify the difference in surface smoothness, as well as the local correlation as measured using the Pearson correlation coefficient to quantify the correctness of gradient directions in both the surfaces.

$$\lambda_y = \frac{1}{N} y^T L y \quad (1)$$

where y is the property value, $L = D - A$, D is the degree matrix, A is the KNN adjacency matrix, and N is the number of samples considered for computing the Dirichlet energy. The property value measured in our experiments is the weighted sum of SA, QED, and BA as used in Eckmann et al. (2022). The training molecules are encoded into the latent space and 5 nearest neighbors are found using KNN which are then used to compute the Dirichlet energy.

The mean-squared error was found to decrease when training the surrogate jointly with the VAE (0.634) vs when training them separately (0.986), indicating that joint training leads to a better fit.

Additionally, the Dirichlet energies are lower when jointly training (33.31 vs 94.04), indicating that optimizing for properties using the jointly trained surrogate leads to smoother changes in property values i.e. molecules decoded from the same neighborhood of the latent space tend to have more similar property values.

Lastly, the local correlation as measured using the Pearson correlation coefficient decreases when jointly training (0.48 vs 0.76), indicating that local gradient-based optimization using the jointly trained surrogate is less accurate.

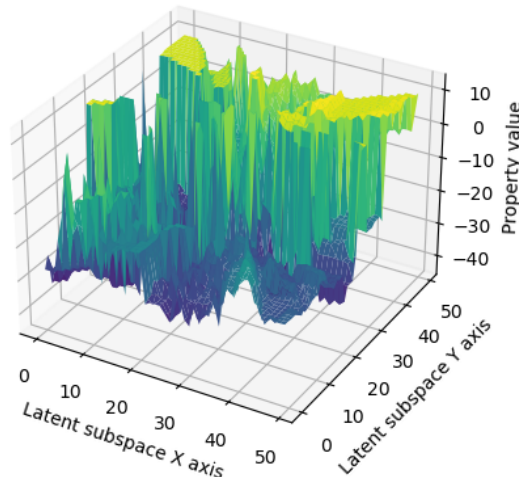


Figure 2: Objective plotted along a random subspace of the latent space. Latent space points were decoded to molecules by uniformly sampling a region constructed using two orthogonal random directions. The plot shows that objective values vary greatly even for nearby latent points.

4 FUTURE WORK

4.1 LATENT SPACE ORGANIZATION

While jointly training the VAE and the surrogate improves the generated molecules by explicit feedback making the latent space interpolable with respect to properties, further improvements to the latent space organization via topological constraints Moor et al. (2021); Keller & Welling (2022) is a promising direction.

4.2 SEMI-AUTOREGRESSIVE MODELLING

There is a tradeoff between controllable generation and sequence modeling ability by changing the non-autoregressive decoder to an autoregressive one. Huang et al. (2022) is an attempt at merging the best of both worlds by having an iterative non-autoregressive decoder. Other such approaches include semi-autoregressive models such as those used in Han et al. (2023) which generate blocks of text autoregressively, and order-agnostic autoregressive models such as those used in Alamdari et al. (2023) which can generate text in an arbitrary order i.e. not just left to right.

4.3 FURTHER ANALYSIS OF GROUPSELFIES TOKENS

The experiments done in this work show that the performance of the GroupSELFIES tokens depends greatly on the extracted tokens. While there is no obvious difference between the tokens extracted from different datasets, further study is needed to determine the cause of the apparent difference in performance.

Pursuing these directions should improve the properties of molecules generated by LIMO, as well as maintain the ability to constrain the molecular scaffold.

REFERENCES

- Tagir Akhmetshin, Arkadii I. Lin, Daniyar Mazitov, Evgenii Ziaikin, Timur Madzhidov, and Alexandre Varnek. ZINC 250K data sets. 12 2021. doi: 10.6084/m9.figshare.17122427.v1. URL https://figshare.com/articles/dataset/ZINC_250K_data_sets/17122427.
- Sarah Alamdari, Nitya Thakkar, Rianne van den Berg, Alex X. Lu, Nicolo Fusi, Ava P. Amini, and Kevin K. Yang. Protein generation with evolutionary diffusion: sequence is all you need. September 2023. doi: 10.1101/2023.09.11.556673. URL <http://dx.doi.org/10.1101/2023.09.11.556673>.
- Austin Cheng, Andy Cai, Santiago Miret, Gustavo Malkomes, Mariano Phielipp, and Alán Aspuru-Guzik. Group selfies: A robust fragment-based molecular string representation. 2022. doi: 10.48550/ARXIV.2211.13322. URL <https://arxiv.org/abs/2211.13322>.
- Peter Eckmann, Kunyang Sun, Bo Zhao, Mudong Feng, Michael K Gilson, and Rose Yu. Limo: Latent inceptionism for targeted molecule generation. 2022.
- Peter Ertl and Ansgar Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of Cheminformatics*, 1(1), June 2009. ISSN 1758-2946. doi: 10.1186/1758-2946-1-8. URL <http://dx.doi.org/10.1186/1758-2946-1-8>.
- Hao Fu, Chunyuan Li, Xiaodong Liu, Jianfeng Gao, Asli Celikyilmaz, and Lawrence Carin. Cyclical annealing schedule: A simple approach to mitigating kl vanishing, 2019.
- Jiatao Gu, James Bradbury, Caiming Xiong, Victor O. K. Li, and Richard Socher. Non-autoregressive neural machine translation, 2018.
- Xiaochuang Han, Sachin Kumar, and Yulia Tsvetkov. Ssd-lm: Semi-autoregressive simplex-based diffusion language model for text generation and modular control, 2023.
- Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017. URL <https://openreview.net/forum?id=Sy2fzU9gl>.
- Xiao Shi Huang, Felipe Perez, and Maksims Volkovs. Improving non-autoregressive translation models without distillation. In *International Conference on Learning Representations*, 2022. URL <https://openreview.net/forum?id=I2Hw58KHp80>.
- T. Anderson Keller and Max Welling. Topographic vaes learn equivariant capsules, 2022.

-
- Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.
- Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022.
- Mario Krenn, Florian Häse, AkshatKumar Nigam, Pascal Friederich, and Alan Aspuru-Guzik. Self-referencing embedded strings (selfies): A 100 *Machine Learning: Science and Technology*, 1(4):045024, October 2020. ISSN 2632-2153. doi: 10.1088/2632-2153/aba947. URL <http://dx.doi.org/10.1088/2632-2153/aba947>.
- Greg Landrum. Rdkit: Open-source cheminformatics. URL <http://www.rdkit.org>.
- Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts, 2017.
- Daniel Lowe. Chemical reactions from US patents (1976-Sep2016). 6 2017. doi: 10.6084/m9.figshare.5104873.v1. URL https://figshare.com/articles/dataset/Chemical_reactions_from_US_patents_1976-Sep2016_/5104873.
- Michael Moor, Max Horn, Bastian Rieck, and Karsten Borgwardt. Topological autoencoders, 2021.
- Yi Ren, Jinglin Liu, Xu Tan, Zhou Zhao, Sheng Zhao, and Tie-Yan Liu. A study of non-autoregressive model for sequence generation, 2020.
- Diogo Santos-Martins, Leonardo Solis-Vasquez, Andreas F Tillack, Michel F Sanner, Andreas Koch, and Stefano Forli. Accelerating autoencoder4 with gpus and gradient-based local search. *Journal of Chemical Theory and Computation*, 17(2):1060–1073, January 2021. ISSN 1549-9626. doi: 10.1021/acs.jctc.0c01006. URL <http://dx.doi.org/10.1021/acs.jctc.0c01006>.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.
- Ziyi Yang, Yinfei Yang, Daniel Cer, Jax Law, and Eric Darve. Universal sentence representation learning with conditional masked language model, 2021.

A APPENDIX

We report the top 100 molecules in terms of SA, BA and QED in the corresponding tables.

Table 3: Top 100 BA molecules

Model Variants		Count with valid AutoDock BA	BA ↓	SA ↓	QED ↑
LIMO	Default-Retrained	9813	-7.05	4.94	0.58
	GS-Paper	8729	-7.65	3.54	0.49
	GS-USPTO	9947	-7.55	4.75	0.59
	GS-Zinc	9868	-7.46	4.97	0.57
AR	Default	9194	-6.80	5.18	0.67
	Joint-Z	9229	-7.16	4.32	0.78
AR-JointZ	GS-Paper	9768	-6.53	3.05	0.82
	GS-USPTO	9939	-7.11	3.41	0.77
	GS-Zinc	9967	-7.11	3.65	0.84
CMLMC		8758	-7.85	4.04	0.28
Dataset	ZINC250K	242591	-8.22	3.08	0.51

Table 4: Top 100 SA molecules

Model Variants		Count with valid AutoDock BA	BA ↓	SA ↓	QED ↑
LIMO	Default-Retrained	9813	-3.24	2.19	0.52
	GS-Paper	8729	-3.36	1.12	0.53
	GS-USPTO	9947	-3.52	2.27	0.56
	GS-Zinc	9868	-3.87	2.57	0.54
AR	Default	9194	-4.38	2.14	0.68
	Joint-Z	9229	-5.02	1.80	0.85
AR-JointZ	GS-Paper	9768	-5.10	1.72	0.74
	GS-USPTO	9939	-5.19	1.83	0.85
	GS-Zinc	9967	-5.37	2.01	0.85
CMLMC		8758	-4.40	1.74	0.22
Dataset	ZINC250K	242591	-4.74	1.42	0.80

Table 5: Top 100 QED molecules

Model Variants		Count with valid AutoDock BA	$BA \downarrow$	$SA \downarrow$	$QED \uparrow$
LIMO	Default-Retrained	9813	-5.49	4.65	0.83
	GS-Paper	8729	-5.47	3.02	0.91
	GS-USPTO	9947	-5.83	4.51	0.89
	GS-Zinc	9868	-5.93	4.60	0.86
AR	Default	9194	-5.52	3.83	0.91
	Joint-Z	9229	-5.46	2.74	0.94
AR-JointZ	GS-Paper	9768	-5.23	2.55	0.94
	GS-USPTO	9939	-5.54	2.57	0.94
	GS-Zinc	9967	-5.67	2.60	0.94
CMLMC		8758	-5.43	4.59	0.77
Dataset	ZINC250K	242591	-5.38	2.75	0.95