LiGen: Active Lipid Generation via a Molecular Language Model

Ying Zhan[1], Xiuqi Tang[2], Yan Zhang[1], Xiao Tan[1], Dian Shen[1], Zhou Yu[3], Beilun Wang[1]*

[1]School of Computer Science and Engineering, Southeast University
[2]School of Computer Science and Engineering, Xidian University
[3]Department of Computer Science, Columbia University
Email: y_zhan@seu.edu.cn

Abstract

Lipid nanoparticles (LNPs) can deliver cargos to both tumor and immune cells, playing a crucial role in biomedicine. Traditional approaches rely on experimental screening and expert knowledge, which can be costly and time-consuming. Recent methods based on language models have accelerated this process using deep learning. Although these methods can retrieve molecules for fusion or rank candidates from existing libraries, they are still limited by the scope of known formulations. In this work, we propose LiGen to generate lipid molecules efficiently and actively, facilitating the discovery of high-performing LNP formulations. We first train a lipid-specific molecular language model, LiCore, to learn hidden representations of lipid molecules. We then explore the learned latent space to generate improved candidate formulations. This process is guided by a trained predictor, which evaluates delivery efficiency and provides directional signals. In reconstruction task, LiCore achieves near-perfect reconstruction performance output with a low invalid ratio on both the LNP-Virtual900k and LNP-Exp12k datasets. The predictor consistently improves ranking-oriented metrics across multiple cell lines, with our method outperforming the best baselines by an average of 4.1%, 10.8%, and 8.1% in Top-50, Top-10, and Top-5 identification accuracy, respectively. Guided by predictor, LiGen generates novel lipid candidates that achieve a 30.7% relative improvement over baseline methods in predicted delivery efficiency, with some candidates exceeding 50% improvement.

1 Introduction

Lipid nanoparticles (LNPs) enable targeted delivery of siRNA and mRNA to tumor cells and immune cells, facilitating gene silencing and immunotherapy in cancer models (Cheng et al., 2025).

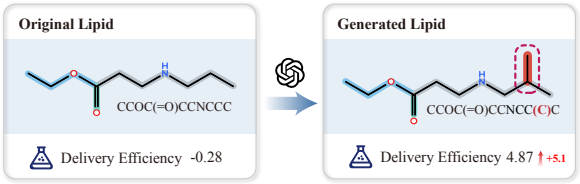

Figure 1: Task Overview. Given an original lipid structure, we aim to develop a model that generates structurally modified lipids with improved delivery efficiency.

Despite their clinical success, the process of designing effective LNP formulations still depends heavily on experimental screening and expert judgment. The preparation typically involves several components in varying proportions. Moreover, the same formulation can perform differently in distinct cell lines. Conventional experimental methods for LNP formulation screening are often limited to tens of formulations per day. This makes it expensive to empirically explore thousands of candidate compositions across diverse conditions (Lam et al., 2025; Li et al., 2025).

While many classical approaches adopt sequence-level molecular language models to improve purely experimental exploration (Goh et al., 2017; Paul et al., 2018), recent years have seen growing interest in language-model-based pipelines for molecular design. Two strands dominate: encoder–decoder architectures (Wang et al., 2023) that retrieve the existing molecules, and predictors (Witten et al., 2025; Cui et al., 2025) that act as oracles to rank large virtual libraries. Their appeal — beyond scaling and richer token-level representations — lies in accelerating formulation decisions to reduce wet-lab cost.

However, there is one big catch: both strands remain constrained by the existing formulations. Encoder–decoder generators depend on retrieving high-quality existing molecules for fusion. Oracle-based screening is limited to searching in prede-

*Corresponding author

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 8674–8686

fined virtual libraries. They both struggle to actively explore novel structures for generating LNP candidates.

This motivates a critical question: can we pair an encoder–decoder backbone with a predictor to drive formulation generation, without constraints from existing formulations?

The answer to this question is indeed not foreseeable: on one hand, current encoder–decoder language models are largely general-purpose. They struggle to focus on the fine lipid-specific distinctions needed for formulation. On the other hand, a predictor must reason over multi-component formulations and continuous ratios. It must also adapt to multiple cell lines with different distributions.

In this work, we provide a unified method named **LiGen** that enables active **Li**pid **Gen**eration by a pretrained lipid molecule language model, **LiCore**. We pretrain LiCore on a large set of ionizable cationic lipids molecules, capturing rich structural signals relevant to formulation behavior. Based on it, we design a predictor that serves as an oracle which balances differences across cell lines and improves cross-cell-line prediction. We implement a generator that uses direction signals to iteratively design improved candidate formulations in the learned representation space. Together, we provide an accurate and generative approach to LNP design.

Our main contributions are as follows:

- We train a lipid-specific molecular language model to learn representations tailored to ionizable lipid structures. We collect a large lipid-focused dataset containing over 900,000 ionizable lipids to support the training.
- We introduce a predictor-guided generation framework that explores the LNP design space in the learned latent representation. The predictor provides direction signals to guide candidate generation across multiple cell lines and formulation components, enabling targeted exploration with limited experimental data.
- We validate the proposed framework on multiple LNP design tasks. The language model achieves near-perfect reconstruction accuracy with an extremely low invalid rate. The predictor consistently enhances ranking performance, with average gains of approximately 10.8% and 8.1% for Top-10 and Top-5 identification accuracy, respectively. In generation experiments, LiGen produces lipid candidates that outperform existing optimization baselines by 30.7% on average.

2 Related Works

2.1 Encoder–Decoder Based Generation

Encoder–decoder based language models have become a central paradigm in generative molecular design (Merz Jr et al., 2020; Sahu et al., 2025; Owoyemi and Medzhidov, 2023). These models learn latent representations of molecules that can be decoded into valid chemical structures, supporting tasks such as property-guided synthesis, fragment assembly, and latent space optimization (Blaschke et al., 2018; Abeer et al., 2024). RetMol (Wang et al., 2023), which incorporates exemplar retrieval to guide generation toward target properties. LIMO (Eckmann et al., 2022) combines variational autoencoder latent embeddings with gradient-based property guidance to accelerate the discovery of high-affinity compounds (Ochiai et al., 2023). Most of these methods have been developed and evaluated on small molecules or general chemical libraries, focusing on exploring chemical space efficiently and controllably (Kong et al., 2022).

While encoder–decoder frameworks demonstrate strong performance for small molecule generation, their direct application to lipid nanoparticle (LNP) design is challenging. LNPs are multi-component systems with complex combinatorial compositions, where interactions among lipid types and ratios influence delivery efficacy. The latent spaces learned by small molecule models do not directly capture these multi-component relationships (Schiff et al., 2020), although the representation learning and controllable generation principles inspire approaches for designing novel lipid formulations.

2.2 Oracle Based Screening

Another line of work uses language models to predict properties of candidate molecules or formulations and then filters or ranks them based on predicted efficacy (Tran and Ekenna, 2023). In the context of LNPs, transformer-based predictors and multi-component feature models (Kumar and Ardekani, 2025), such as COMET (Chan et al., 2025), encode the composition and structural fea-

tures of lipids to predict delivery effectiveness. Other approaches employ machine learning models trained on experimental LNP datasets to rank existing formulations or identify promising candidates (Xu et al., 2024; Witten et al., 2025). These methods have been effective in guiding experimental screening and improving candidate selection without requiring full combinatorial searches.

A key limitation of prediction and screening-based methods is that they do not actively generate new candidates. They rely on existing libraries or previously synthesized molecules, scoring or fusing them to identify improved formulations. While they provide useful guidance for formulation design and have informed subsequent generative approaches, they remain constrained by the chemical space represented in their training data (Wei et al., 2023). This motivates the development of models capable of exploring the LNP design space more broadly, combining representation learning, generation, and predictive guidance.

3 Methodology

Our overall objective is to provide a method LiGen for formulation design that can generate improved formulations guided by predicted delivery performance. The language model encodes formulation candidates into a latent space. It iteratively updates these embeddings using a learned predictive model. It then decodes the embeddings back into valid candidates. This process enables efficient exploration of the formulation space.

LiGen consists of three key components that correspond to this process. The molecular language model LiCore (Section 3.1) is trained to encode formulations into latent embeddings and decode them back, ensuring accurate representation. The predictor (Section 3.2) predicts how well the candidate will perform in terms of delivery efficiency. The generator (Section 3.3) iteratively updates the latent embedding along the direction signal and decodes it, proposing improved formulation candidates based on predicted performance.

3.1 Lipid-aware Molecular Language Model Pretraining

We first train a language model that captures transferable structural representations of ionizable lipid molecules for downstream prediction and generation tasks.

We adopt an encoder-decoder molecular language model LiCore to learn transferable representations of ionizable lipid molecules. Given a SMILES sequence s, the model is trained to reconstruct the original sequence through an autoencoding objective:

$$h(s) = \mathrm{Dec}(\mathrm{Enc}(s)) = \mathrm{Dec}(z) = \hat{s}, \quad (1)$$

where

$$z = \mathrm{Enc}(s) \in \mathbb{R}^d, \quad (2)$$

which is the latent embedding of the ionizable lipid. $\hat{s}$ denotes the reconstructed SMILES sequence predicted by the model. The training objective aims to minimize the discrepancy between the predicted sequence $\hat{s}$ and the original sequence s.

During pretraining, each SMILES sequence is first converted into a non-canonical form to introduce structural diversity and serve as a form of data augmentation, which has been shown to improve the robustness of molecular representation learning. Then we use token-level span masking to modify the input sequence (Irwin et al., 2022). The bidirectional encoder processes the modified sequence, while the autoregressive decoder is trained to reconstruct the original SMILES.

To train the model, we minimize the difference between the predicted sequence $\hat{s}$ and the original s using a sequence-level cross-entropy loss:

$$\mathcal{L}_{\mathrm{h}}(\theta, s) = -\sum_{l=1}^{L} \log p_\theta(s_l \mid s_{<l}), \quad (3)$$

where L denotes the length of the sequence, $p_\theta(s_l \mid s_{<l})$ is the probability of the l-th token predicted by the decoder.

Inspired by recent large language models that incorporate high-quality data at later stages of pretraining, we further refine LiCore. We train LiCore-FT on a smaller and task-specific set of ionizable lipid molecules. This stage follows the same training objective and architecture as the base model, differing only in the data distribution. It allows the learned representations to better align with downstream LNP design.

3.2 Predictor for Multi-component LNP Formulations

We introduce a predictor, denoted as g, to estimate the delivery efficiency of LNP formulations under specific cellular contexts. Formally, the prediction target $y \in \mathbb{R}$ is a real-valued scalar, where larger

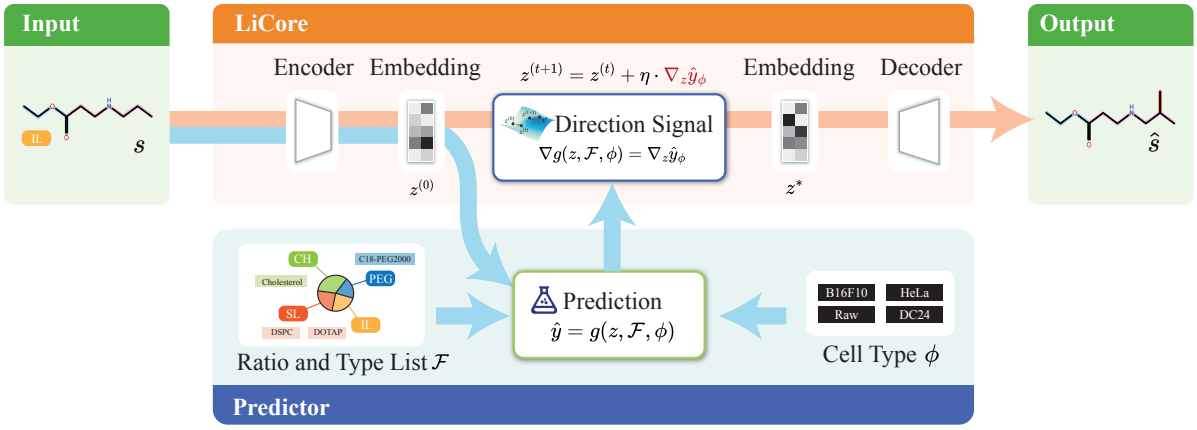

Figure 2: Overview of the LiGen pipeline for lipid generation. Given an input lipid, LiCore encodes the molecule into a latent embedding and iteratively updates it using a direction signal derived from a property predictor. The predictor estimates delivery efficiency conditioned on formulation features and cell type. The optimized embedding is then decoded to generate a new lipid with enhanced delivery efficiency.

values indicate higher delivery efficiency. In practice, y is obtained from standardized experimental assays and serves as the supervision signal for training the predictor.

An LNP formulation is characterized by three elements: the latent ionizable lipid (IL) embedding z as defined in Eq.2, the formulation features $\mathcal{F} = \{r_0, (c_1, r_1), (c_2, r_2), \ldots, (c_k, r_k)\}$, and the target cell line ϕ. Here, each c_j is a categorical variable indicating the type of a non-ionizable component. Each r_j is the corresponding molar fraction with $\sum_j r_j = 1$. r_0 is the ratio of ionizable lipid. ϕ is a discrete variable specifying the specific cell type.

Given a formulation $(z, \mathcal{F})$ and cell line ϕ, the model predicts the delivery efficiency:

$$\hat{y} = g(z, \mathcal{F}, \phi). \tag{4}$$

It first maps each component in $\mathcal{C}$ through learnable embeddings and projections, then fuses the resulting features along with the component ratios in $\mathcal{R}$ to capture interactions between lipid types and their relative fractions. The fused representation is combined with a cell-specific scaling factor to account for systematic differences across cell lines:

$$\hat{y}_\phi = \hat{y} \cdot \exp(\alpha_\phi), \tag{5}$$

where α_ϕ is a learnable scalar associated with cell line ϕ.

Let θ_g denote all learnable parameters of g, including component embeddings, projection layers, fusion network weights, and cell-specific scaling factors. The predictor is trained by minimizing the mean squared error between predicted and measured delivery efficiencies:

$$\theta_g^* = \arg\min_{\theta_g} \frac{1}{N} \sum_{i=1}^{N} \left(\hat{y}_{\phi_i}^{(i)} - y^{(i)}\right)^2. \tag{6}$$

3.3 Generator for Directed LNP Design

The generator explores the latent space of ionizable lipid embeddings to propose candidate LNP formulations.

It leverages the direction signal from the trained predictor g, using the scaled predicted delivery efficiency $\hat{y}_\phi$ defined in Eq. (5). The direction signal with respect to the IL embedding is computed as

$$\nabla_z g(z, \mathcal{F}, \phi) = \nabla_z \hat{y}_\phi = \frac{\partial \hat{y}_\phi}{\partial z}. \tag{7}$$

Let $z^{(t)}$ denote the IL embedding at iteration t. The IL embedding is iteratively updated along this gradient to increase the predicted delivery efficiency:

$$z^{(t+1)} = z^{(t)} + \eta \cdot \nabla_z \hat{y}_\phi, \tag{8}$$

where η is a step size that controls the extent of each latent-space update.

The updated embedding is decoded back to a chemically valid SMILES sequence using the pretrained decoder (Eq. (1)):

$$\hat{s}^{(t+1)} = \mathrm{Dec}\left(z^{(t+1)}\right), \tag{9}$$

where $\hat{s}^{(t+1)}$ is the generated candidate sequence.

This iterative procedure continues until either the scaled predicted delivery efficiency exceeds a desired threshold or a maximum number of iterations $T_{\max}$ is reached.

Overall, the goal of our framework is captured by

$$f(s, \mathcal{F}, \phi, \eta) = \text{Dec}\Big(z + \eta \cdot \nabla_z g(z, \mathcal{F}, \phi)\Big), \quad (10)$$

which formalizes the process of iteratively updating the latent embedding of a candidate formulation along the gradient of the predictor and decoding it to propose improved candidates.

4 Experiments

4.1 Settings

Datasets We pretrained LiCore on a large-scale molecular corpus containing 900,351 ionizable lipid structures, which we name LNP-Virtual900k. This dataset was generated by computationally applying publicly reported synthetic reactions to commercially available substrates, following established in silico reaction enumeration protocols. It provides broad coverage of chemically feasible lipid structures while respecting practical synthesis constraints.

We further fine-tuned the model as LiCore-FT using a downstream dataset of 12,467 ionizable lipid molecules, which we refer to as LNP-Exp12k. We collect these molecules from publicly available datasets reported in prior studies (Witten et al., 2025; Xu et al., 2024). This fine-tuning stage adapts the pretrained representation to downstream prediction and generation tasks.

For downstream predictor training, we reorganize several publicly available lipid formulation datasets from prior studies. We refine formulation annotations based on the original publications and train model on data from five distinct cell lines: HeLa, DC24, B16F10, Raw, and A549 (Xu et al., 2024; Chan et al., 2025; Witten et al., 2025). Ground-truth delivery efficiency values are obtained from wet-lab experimental assays reported in the original studies, typically measured via mRNA-mediated luciferase expression. These measurements reflect complex biological processes, including cellular uptake and endosomal escape. As a result, learned predictors are commonly used as surrogate evaluators in prior work. Detailed dataset statistics and processing procedures appear in the Appendix B.1 .

Baselines We evaluate our approach against several baseline methods. We use the ChemFormer (Irwin et al., 2022) to benchmark pretraining performance. For prediction, we compare our predictor with existing lipid nanoparticle predictors, including LiON (Witten et al., 2025), AGILE (Xu et al., 2024), and COMET (Chan et al., 2025). For optimization, we compare against RetMol (Wang et al., 2023), LiON (Witten et al., 2025), and AGILE (Xu et al., 2024). Further details on the baselines appear in Appendix B.2.

Evaluation Metrics For the pretrained language model, we assess the quality of the learned representations using a reconstruction task on a held-out test set. Specifically, we quantify the similarity between an input molecule and the molecule reconstructed by decoding its latent embedding (as defined in Eq. (1)). We compute similarity using two widely used fingerprint-based metrics: MACCS structural keys and topological fingerprints (FP) (Rogers and Hahn, 2010). The MACCS fingerprint (Durant et al., 2002) captures predefined substructure features, while the topological fingerprint encodes molecular connectivity patterns based on hashed paths through the molecular graph. The similarity scores are averaged over the evaluation set. In addition, we report the fraction of generated molecules that are chemically invalid.

For the predictor, we use delivery efficiency as the target label, which quantifies the effectiveness of a lipid nanoparticle formulation in delivering its payload to the target cells. We evaluate the oracle using standard regression metrics, including mean squared error (MSE), mean absolute error (MAE), and Pearson correlation coefficient (Pearson, 1895). In addition, we specifically assess the oracle's ability to correctly identify the top-performing samples. We report the fraction of samples in the top 50%, 10%, and 5% of true delivery efficiency that are correctly classified by the model. This focus is motivated by the inherent imbalance in real-world data, where most samples cluster around average delivery efficiency. Accurate identification of high-efficiency formulations is critical, as these represent the most relevant and actionable candidates for experimental validation.

For the generator, we select representative input molecules as case studies and compare the delivery efficiency scores of newly generated molecules produced by different methods. We evaluate generation-based approaches by reporting the predicted delivery efficiency of the generated candidates. For oracle-based screening methods that cannot generate new molecules, we report their evaluation scores on the original inputs to illustrate

Table 1: Reconstruction performance on test datasets. LiCore achieves substantially improved reconstruction accuracy compared to the baseline, and fine-tuning on experimental data further enhances reconstruction fidelity while reducing the fraction of invalid molecules.

Test Dataset	Model	MACCS (%)	FP (%)	Invalid Ratio (%)
LNP-Virtual900k	Chemformer	79.10	71.09	2.46
	LiCore	**99.99**	**99.99**	**0.08**
	LiCore-FT	**99.99**	**99.99**	0.11
LNP-Exp12k	Chemformer	75.87	56.45	18.72
	LiCore	94.51	92.65	0.49
	LiCore-FT	**99.99**	**100.00**	**0.00**

their inability to explore beyond existing candidates

4.2 Results and Analysis

Performance of pretraining LiCore We first evaluate the reconstruction performance of Chemformer on the two test datasets, as summarized in Table 1. Chemformer achieves relatively modest results. MACCS and fingerprint similarities are below 80% on both LNP-Virtual900k and LNP-Exp12k. The fraction of invalid sequences is notable, especially on the experimental LNP-Exp12k dataset, where nearly 19% of the outputs are invalid. We analyze this and suggest that a model pretrained on general molecular structures cannot fully capture the structural patterns of lipid molecules. This limitation leads to lower reconstruction accuracy.

In contrast, LiCore, pretrained on the large LNP-Virtual900k dataset, shows a clear improvement over Chemformer. On the virtual dataset, MACCS similarity increases by 20.9% and fingerprint similarity by 28.9%, while the invalid ratio decreases by 2.35%. Even for the unseen LNP-Exp12k dataset, reconstruction improves. MACCS similarity rises by 18.6% and fingerprint similarity by 36.2%. The invalid fraction drops by 18.2%. We believe this improvement comes from the model learning lipid-specific patterns during pretraining. This training allows it to encode and decode lipid molecules more accurately.

Fine-tuning LiCore on the experimental LNP-Exp12k data further increases reconstruction performance. On the LNP-Exp12k dataset, MACCS similarity increases by 5.5% and fingerprint similarity by 7.4% compared to LiCore pretrained, while the invalid ratio decreases by 0.5%. Reconstruction on the virtual dataset show consistent performance. This shows that fine-tuning preserves the knowledge from pretraining while adapting the model to high-quality experimental molecules. Figure 3 summarizes the reconstruction behavior of different pre-training strategies across a range of molecular weights. On both the LNP-Virtual900k and LNP-Exp12k datasets, all models exhibit high MACCS similarity for lower-molecular-weight compounds, indicating reliable reconstruction in simpler chemical regimes. As molecular weight increases, Chemformer exhibits increasingly unstable reconstruction behavior. This instability is reflected in a clear decline in reconstruction similarity between the original and reconstructed molecules, accompanied by a rapid

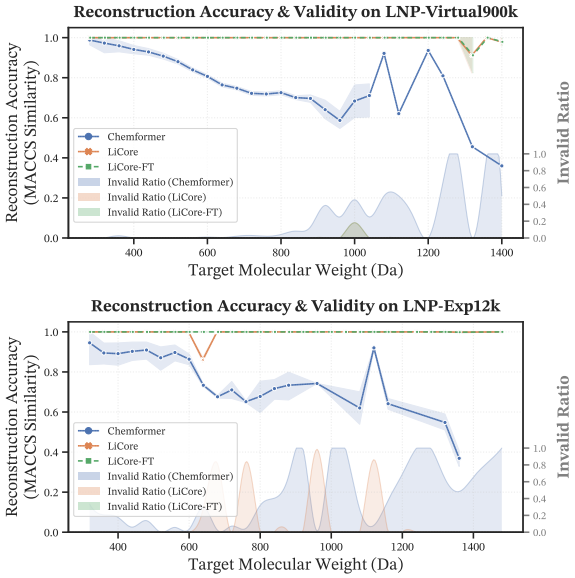

Figure 3: Comparison of reconstruction quality under different pre-training strategies on two datasets. In each subplot, the upper line plots reflect the trend of reconstruction accuracy, measured by MACCS similarity between the original and reconstructed molecules. The shaded distributions in the lower area illustrate the proportion of invalid molecules generated by each model at the corresponding molecular weight range.

Table 2: Performance of the forward predictor across different datasets.

Cell Type	Method	MSE ↓	MAE ↓	r ↑	Top-50 (%) ↑	Top-10 (%) ↑	Top-5 (%) ↑
Raw	AGILE	2.92	1.34	0.39	54.17	25.00	33.33
	LiON	2.79	1.30	0.45	64.17	25.00	33.33
	Ours	**2.79**	**1.22**	**0.65**	**74.58**	**45.83**	**41.67**
A549	AGILE	1.05	0.83	0.45	63.33	19.44	16.67
	LiON	0.70	0.66	0.72	75.14	29.73	21.05
	Ours	**0.47**	**0.53**	**0.81**	**78.24**	**47.06**	**35.29**
HeLa	AGILE	0.62	0.63	0.32	57.52	36.36	18.18
	LiON	**0.52**	0.57	0.46	65.49	**52.17**	33.33
	Ours	0.70	**0.62**	**0.51**	**65.83**	45.83	**50.00**
B16F10	AGILE	0.03	0.13	0.40	65.38	27.42	9.68
	LiON	0.02	0.11	0.67	**75.00**	44.44	**40.62**
	COMET	0.07	0.21	0.42	60.58	27.42	19.35
	Ours	**0.02**	**0.10**	**0.75**	72.56	**57.81**	37.50
DC24	AGILE	0.04	0.17	0.44	69.23	22.58	9.68
	LiON	0.03	0.15	0.62	70.83	34.92	28.12
	COMET	0.04	0.17	0.71	76.28	40.32	29.03
	Ours	**0.02**	**0.10**	**0.84**	**85.30**	**49.21**	**50.0**

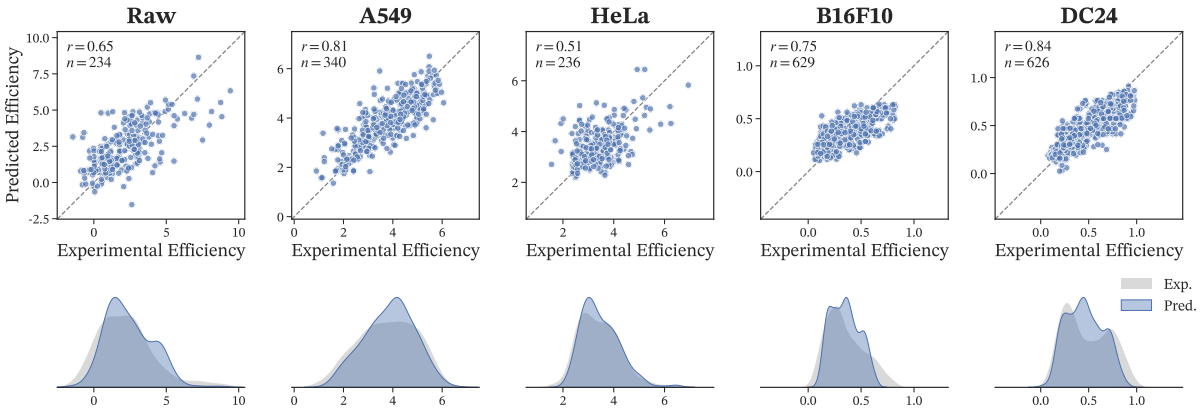

Figure 4: Comparison between experimental and predicted efficiencies across five cell lines. Scatter plots illustrate the agreement between predicted and experimental values, while distribution plots summarize the overall consistency between predictions and measurements.

increase in the proportion of invalid outputs.

Notably, for molecules with molecular weights exceeding 1000 Da, Chemformer frequently produces invalid molecules exclusively, resulting in invalid ratios approaching 100%. In contrast, both LiCore and LiCore-FT demonstrate substantially more stable reconstruction performance across the molecular weight spectrum, maintaining reconstruction similarities close to unity. While these two models achieve comparable reconstruction accuracy, LiCore-FT shows a marked improvement in molecular validity relative to LiCore, an effect that is particularly pronounced on the LNP-Exp12k dataset. These results indicate that the proposed fine-tuning strategy effectively enhances molecular validity without compromising reconstruction accuracy, especially in chemically complex regimes.

Performance of Forward Predictor We evaluate the performance of the prediction oracle across multiple cell lines, as summarized in Table 2. Overall, our model reduces prediction errors compared to baseline methods. For example, on the Raw cell line, mean absolute error decreases by about 0.08, while Pearson correlation improves by 0.14. Simi-

Table 3: Comparison of generated molecules across different methods for the same original lipids. Some methods do not actively generate molecules; for these, we only report the predicted values. Molecules generated by LiGen generally achieve higher predicted scores and exhibit expected molecular lengths.

Method	SMILES	Score
(Ori)	C1(C(=O)OCC)N=CC(CCCCCCC)(CCCCCCC)N1(CCN(CC)CC)	2.82
LiON	-	2.52
AGILE	-	1.92
RetMol	CC1(C(=O)OCC)N=C(C(=O)OCC)N=C(CCCCCCCCCCCC)N1CCOCC	3.06
LiGen	C1(C(=O)OCC)N=CC(CCCCCCC)(CCCCCCC)N1CCCCCCCC/C=C\CCCCCCCC	**5.47**
(Ori)	N(CS(=O)(=O)c3ccc(C)cc3)C(=O)C(CCCCCCCCC)(CCCCCCCCC)NCCN(CC)CC	3.23
LiON	-	3.12
AGILE	-	3.00
RetMol	CCN(C(=O)OC(C)(C)C)C/C=C/C(=O)OC/C=C/CCCCCCCCC	3.30
LiGen	N(CS(=O)(=O)c3ccc(C)cc3)C(=O)C(CCCCCCCCCC)(CCCCCCCCCC)NCCN(CC)CC	**5.19**
(Ori)	N(C(C)(C)C)C(=O)C(CCCCCCCCC)(CCCCCCCCC)NCCN(CC)CC	2.71
LiON	-	2.79
AGILE	-	2.49
RetMol	CN(CCCCCCCCCCCCCCCCCNC(=O)C(=O)NNC(C)(C)C)CC	3.14
LiGen	N(C(C)(C)C)C(=O)C(CCCCCCCCCC)(CCCCCCCCCC)NCCN(CC)CC	**3.91**

larly, for A549, our model lowers MSE by roughly 0.23 and raises Pearson r by 0.09 compared to the best baseline. These improvements suggest that the model can better capture the relationship between molecular features and delivery efficiency.

We also observe that our model identifies high-efficiency molecules more reliably. The predictor improves the fraction of correctly identified molecules across ranking thresholds, with average gains of approximately 4.1%, 10.8%, and 8.1% for the Top-50, Top-10, and Top-5 fractions, respectively. We analyze this and attribute it to the model learning more precise patterns in the latent space, allowing it to distinguish the most promising formulations even when the dataset is imbalanced and most samples are near average efficiency.

Figure 4 shows the relationship between experimentally measured and model-predicted efficiencies across five different cell lines. For all cell types, the predicted values exhibit clear positive correlations with experimental measurements, indicating that the model captures the overall trends across diverse biological contexts. Although the strength of correlation varies among cell lines, the predicted distributions largely overlap with the experimental ones, suggesting good agreement at the population level. These results demonstrate that the proposed model achieves robust and consistent predictive performance across multiple cellular systems.

The improvements are consistent across different cell types, indicating robust predictive performance. The overall trend shows reduced error and higher correlation. This suggests that, for structured biochemical design tasks such as LNP formulation, aligning pretraining data with domain-specific structural patterns is critical for enabling effective representation learning and downstream optimization. This insight highlights the importance of domain-specific pretraining in low-resource, high-complexity scientific applications.

Performance of Optimization As summarized in Table 3, we compare (1) generative methods and (2) oracle-based screening methods, which are included to highlight the limitations of non-generative approaches. Methods that rely solely on predictive screening, such as LiON and AGILE, can only evaluate existing molecules and show limited improvement over the original compounds. Their selection remains confined to the provided chemical space and does not actively explore new molecular variations.

For active generation methods, RetMol tends to produce molecules resembling conventional small molecules due to its generic modeling approach, which limits its suitability for LNP design. In con-

trast, LiGen generates lipid candidates that consistently achieve higher predicted scores while maintaining appropriate molecular lengths. Across all evaluated samples, LiGen improves predicted values by an average of 30.7% compared to RetMol and achieves improvements exceeding 50% on several representative examples, as illustrated in Table 3. These results highlight the superior ability of LiGen to explore the chemical space and generate molecules that are better aligned with the objectives of LNP design.

5 Conclusion

We present a method, LiGen, for active lipid molecule generation to guide the design of improved LNP formulations. It leverages a lipid-specific language model, LiCore, and a trained predictor to guide latent-space exploration. LiGen generates novel lipid candidates efficiently. This approach shows the potential of combining generative modeling with predictive guidance.

Limitations

While LiGen demonstrates promising results in LNP design, several limitations remain. The optimization relies on surrogate predictions rather than experimental validation, which may lead to discrepancies between predicted and actual performance, particularly for rare or out-of-distribution candidates. Appendix C discusses how future studies might explore active learning strategies to address this limitation. Moreover, we have not yet incorporated additional constraints during the generation process. In principle, LiGen allows exploration of adding more design considerations during generation to generate improved candidates.

References

ANM Nafiz Abeer, Nathan M Urban, M Ryan Weil, Francis J Alexander, and Byung-Jun Yoon. 2024. Multi-objective latent space optimization of generative molecular design models. *Patterns*, 5(10).

Camilla Hald Albertsen, Jayesh A Kulkarni, Dominik Witzigmann, Marianne Lind, Karsten Petersson, and Jens B Simonsen. 2022. The role of lipid components in lipid nanoparticles for vaccines and gene therapy. *Advanced drug delivery reviews*, 188:114416.

Thomas Blaschke, Marcus Olivecrona, Ola Engkvist, Jürgen Bajorath, and Hongming Chen. 2018. Application of generative autoencoder in de novo molecular design. *Molecular informatics*, 37(1-2):1700123.

Laura Catenacci, Rachele Rossi, Francesca Sechi, Daniela Buonocore, Milena Sorrenti, Sara Perteghella, Marco Peviani, and Maria Cristina Bonferoni. 2024. Effect of lipid nanoparticle physico-chemical properties and composition on their interaction with the immune system. *Pharmaceutics*, 16(12):1521.

Alvin Chan, Ameya R. Kirtane, Qing Rui Qu, Xisha Huang, Jonathan Woo, Deepak A. Subramanian, Rajib Dey, Rika Semalty, Joshua D. Bernstock, Taksim Ahmed, Rowan Honeywell, Charles Hanhurst, Isaac Diaz Becdach, Leah S. Prizant, Ashley K. Brown, Hao Song, Justin Law Cobb, Louis B. DeRidder, Bruna Santos, and 5 others. 2025. Designing lipid nanoparticles using a transformer-based neural network. *Nature Nanotechnology*, 20(10):1491–1501.

Zhe Cheng, Huichao Huang, Meilong Yin, and Huaizheng Liu. 2025. Applications of liposomes and lipid nanoparticles in cancer therapy: current advances and prospects. *Experimental Hematology & Oncology*, 14(1):11.

Haotian Cui, Yue Xu, Kuan Pang, Gen Li, Fanglin Gong, Bo Wang, and Bowen Li. 2025. Lumi-lab: a foundation model-driven autonomous platform enabling

discovery of new ionizable lipid designs for mrna delivery. *bioRxiv*.

Joseph L Durant, Burton A Leland, Douglas R Henry, and James G Nourse. 2002. Reoptimization of mdl keys for use in drug discovery. *Journal of chemical information and computer sciences*, 42(6):1273–1280.

Peter Eckmann, Kunyang Sun, Bo Zhao, Mudong Feng, Michael K Gilson, and Rose Yu. 2022. Limo: Latent inceptionism for targeted molecule generation. *Proceedings of machine learning research*, 162:5777.

Garrett B Goh, Nathan O Hodas, Charles Siegel, and Abhinav Vishnu. 2017. Smiles2vec: An interpretable general-purpose deep neural network for predicting chemical properties. *arXiv preprint arXiv:1712.02034*.

Ross Irwin, Spyridon Dimitriadis, Jiazhen He, and Esben Jannik Bjerrum. 2022. Chemformer: a pre-trained transformer for computational chemistry. *Machine Learning: Science and Technology*, 3(1):015022.

Weiya Kong, Yuejuan Hu, Jiao Zhang, and Qiaoyin Tan. 2022. Application of smiles-based molecular generative model in new drug design. *Frontiers in Pharmacology*, 13:1046524.

Gaurav Kumar and Arezoo M Ardekani. 2025. Machine-learning framework to predict the performance of lipid nanoparticles for nucleic acid delivery. *ACS Applied Bio Materials*, 8(5):3717–3727.

Lois Lam, Stephanie Watson, Yogambha Ramaswamy, and Gurvinder Singh. 2025. High-throughput strategies for streamlining lipid nanoparticle development pipeline. *Advanced Science*, 12(42):e11551.

Hao Li, Yayi Zhao, and Chenjie Xu. 2025. Machine learning techniques for lipid nanoparticle formulation. *Nano Convergence*, 12(1):35. Published 2025 Jul 15; review article.

Kenneth M Merz Jr, Gianni De Fabritiis, and Guo-Wei Wei. 2020. Generative models for molecular design.

Toshiki Ochiai, Tensei Inukai, Manato Akiyama, Kairi Furui, Masahito Ohue, Nobuaki Matsumori, Shinsuke Inuki, Motonari Uesugi, Toshiaki Sunazuka, Kazuya Kikuchi, and 1 others. 2023. Variational autoencoder-based chemical latent space for large molecular structures with 3d complexity. *Communications Chemistry*, 6(1):249.

Joshua Owoyemi and Nazim Medzhidov. 2023. Smilesformer: Language model for molecular design.

Arindam Paul, Dipendra Jha, Reda Al-Bahrani, Weikeng Liao, Alok Choudhary, and Ankit Agrawal. 2018. Chemixnet: Mixed dnn architectures for predicting chemical properties using multiple molecular representations. *arXiv preprint arXiv:1811.08283*.

Karl Pearson. 1895. Vii. note on regression and inheritance in the case of two parents. *proceedings of the royal society of London*, 58(347-352):240–242.

David Rogers and Mathew Hahn. 2010. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754.

Harikrishna Sahu, Wei Xiong, Anagha Savit, Shivank S Shukla, and Rampi Ramprasad. 2025. An encoder-decoder foundation chemical language model for generative polymer design. *arXiv preprint arXiv:2510.18860*.

Yair Schiff, Vijil Chenthamarakshan, Karthikeyan Natesan Ramamurthy, and Payel Das. 2020. Characterizing the latent space of molecular deep generative models with persistent homology metrics. *arXiv preprint arXiv:2010.08548*.

Yuan Sui, Xiaowen Hou, Juan Zhang, Xuechuan Hong, Hongbo Wang, Yuling Xiao, and Xiaodong Zeng. 2025. Lipid nanoparticle-mediated targeted mrna delivery and its application in cancer therapy. *Journal of Materials Chemistry B*, 13(33):10085–10117.

Tuan Tran and Chinwe Ekenna. 2023. Molecular descriptors property prediction using transformer-based approach. *International Journal of Molecular Sciences*, 24(15):11948.

Zichao Wang, Weili Nie, Zhuoran Qiao, Chaowei Xiao, Richard Baraniuk, and Anima Anandkumar. 2023. Retrieval-based controllable molecule generation. In *The Eleventh International Conference on Learning Representations*.

Lai Wei, Nihang Fu, Yuqi Song, Qian Wang, and Jianjun Hu. 2023. Probabilistic generative transformer language models for generative design of molecules. *Journal of Cheminformatics*, 15(1):88.

Jacob Witten, Idris Raji, Rajith S. Manan, Emily Beyer, Sandra Bartlett, Yinghua Tang, Mehrnoosh Ebadi, Junying Lei, Dien Nguyen, Favour Oladimeji, Allen Yujie Jiang, Elise MacDonald, Yizong Hu, Haseeb Mughal, Ava Self, Evan Collins, Ziying Yan, John F. Engelhardt, Robert Langer, and Daniel G. Anderson. 2025. Artificial intelligence-guided design of lipid nanoparticles for pulmonary gene therapy. *Nature Biotechnology*, 43(11):1790–1799.

Yue Xu, Shihao Ma, Haotian Cui, Jingan Chen, Shufen Xu, Fanglin Gong, Alex Golubovic, Muye Zhou, Kevin Chang Wang, Andrew Varley, Rick Xing Ze Lu, Bo Wang, and Bowen Li. 2024. Agile platform: a deep learning powered approach to accelerate lnp development for mrna delivery. *Nature Communications*, 15(1):6305.

Lei Yang, Liming Gong, Ping Wang, Xinghui Zhao, Feng Zhao, Zhijie Zhang, Yunfei Li, and Wei Huang. 2022. Recent advances in lipid nanoparticles for delivery of mrna. *Pharmaceutics*, 14(12):2682.

A Background

As shown in Figure A.1, LNP formulations typically comprise multiple components, including ionizable cationic lipids (IL), helper lipids such as phospholipids (HL), sterol lipids (SL), and PEGylated lipids (PEG) (Catenacci et al., 2024; Albertsen et al., 2022). Among these, ionizable lipids are the most structurally diverse and play a dominant role in delivery performance (Yang et al., 2022; Sui et al., 2025).

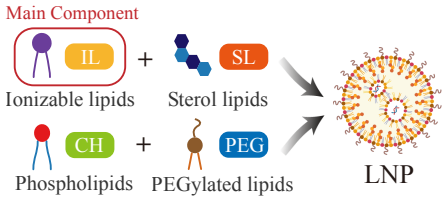

Figure A.1: Four Components of LNP.

B Experimental Details

B.1 Datasets

For downstream predictor training, we reorganize and curate several widely used public datasets proposed in prior work, including **AGILE** (Xu et al., 2024), **COMET** (Chan et al., 2025), and **LiON** (Witten et al., 2025). These datasets were originally reported across multiple studies with heterogeneous experimental settings. We systematically restructure them according to cell-line-specific transfection experiments and refine the associated formulation information based on descriptions in the original publications. In total, five cell lines are considered: HeLa, DC24, B16F10, Raw, and A549. Specifically, HeLa data are obtained from the LiON dataset (LiON-LM_3CR), DC24 and B16F10 data are extracted from the COMET dataset using target-specific attributes (COMET-in_house_lnp_DC24_luc and COMET-in_house_lnp_B16F10_luc), Raw macrophage data are selected from AGILE based on experiment-specific column identifiers (AGILE-expt_Raw), and A549 lung carcinoma data are collected from LiON using screen identifiers (LiON-A549_form_screen). This reorganization enables consistent evaluation across datasets and cell lines while preserving the experimental context defined in the original studies.

B.2 Baselines

The following provides detailed descriptions of the baseline models considered in our study.

- **Chemformer** (Irwin et al., 2022): A Transformer-based model pretrained on SMILES representations of molecules and applied to both sequence-to-sequence and discriminative cheminformatics tasks. Chemformer benefits from self-supervised pretraining to improve performance and accelerate convergence on downstream tasks such as synthesis and retrosynthesis prediction as well as molecular optimization.

- **LiON** (Witten et al., 2025): An AI-guided optimization framework trained on activity measurements of ionizable lipids. LiON employs directed message-passing neural networks to evaluate and screen large libraries of virtual lipid structures, identifying candidates with enhanced delivery performance.

- **AGILE** (Xu et al., 2024): A platform combining combinatorial chemistry with neural network-based screening. AGILE accelerates the design and evaluation of lipid candidates across diverse cell lines.

- **COMET** (Chan et al., 2025): A transformer-based model encoding multimodal formulation features, including molecular structures and compositional ratios. COMET predicts formulation efficacy from large LNP datasets such as LANCE and outperforms traditional machine learning baselines in identifying high-efficacy formulations.

- **RetMol** (Wang et al., 2023): A generative framework that steers a pretrained generative model using exemplar molecules retrieved from a task-specific database. RetMol fuses retrieved exemplars with input molecules and iteratively refines generations, enabling controllable optimization toward given design criteria across multiple tasks without extensive task-specific fine-tuning.

B.3 Pretraining Details

We pretrained LiCore on a large lipid-focused molecular dataset containing over 900,000 ionizable lipids using a masked reconstruction objective, using one A800 GPU. The model follows a

BART-style encoder-decoder architecture with 8 transformer layers, 16 attention heads, a model dimension of 1024, and a feedforward dimension of 4096. Training was conducted with a batch size of 16 across 4 GPUs for 100 epochs. The learning rate was set to 0.1 with a transformer-based scheduling strategy and 800 warm-up steps.

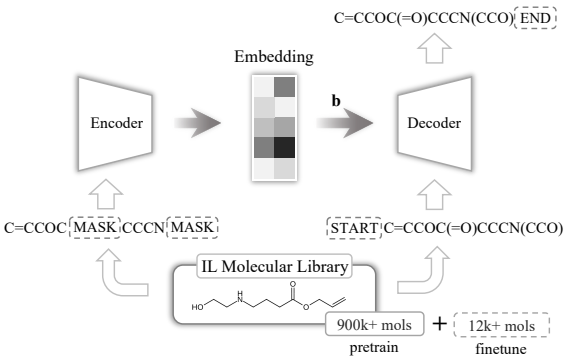

Figure B.1: Schematic of the pretraining procedure for LiCore. The lipid-focused language model is trained on a large molecular corpus using a masked reconstruction objective, learning latent representations that capture structural patterns of ionizable lipids.

Figure B.1 illustrates the pretraining process of LiCore. The model is trained on LNP-Virtual900k. During training, input sequences were partially masked following a span masking scheme with a probability of 0.1. The model learned to reconstruct the original sequences from the masked inputs, thereby capturing structural patterns specific to lipid molecules. During training, validation loss and reconstruction metrics were monitored, and checkpoints were saved periodically.

After pretraining, LiCore was further fine-tuned on LNP-Exp12k to adapt to high-quality experimental distributions, following the same configs.

B.4 Predictor Details

The predictor model adopts a multi-component fusion architecture, where embeddings of the ionic liquid (IL) SMILES are processed via a 1D convolutional layer with 256 output channels, followed by adaptive max pooling and a linear projection. Helper lipids (HL), cholesterol (CH), and polyethylene glycol (PEG) components are represented by categorical embeddings with dimensions 32, 16, and 16 respectively, concatenated with their molar ratios, and projected through separate linear layers. The resulting component features are fused and passed through a two-layer feedforward network to produce the base prediction. To account for dataset-specific value ranges, a *dataset-scale head* is applied, which predicts a nonlinear scaling factor for each dataset, allowing consistent regression across multiple datasets.

The model is trained using a mean squared error loss in the standardized target space. The IL embedding dimension is 1024, and the component projection dimensions are 128 for IL and 64 for other components. The training employed an Adam optimizer with learning rate 1×10^{-3}, batch size 32, and up to 100 epochs. During training, component sequences are padded to the maximum sequence length in the batch, and target standardization is computed per dataset to stabilize optimization. Label encoding for HL, CH, and PEG types includes a padding token to handle variable-length component lists. The predictor is implemented in PyTorch and integrated with Chemformer embeddings, which are obtained from LiCore.

B.5 Generation Details

Initially, IL SMILES strings were encoded using a pre-trained Chemformer encoder, producing token-level embeddings for each molecule. These embeddings were padded or truncated to a consistent length to ensure compatibility with downstream models.

The optimization procedure focused exclusively on the IL embeddings while keeping all other component representations fixed. For each sample, the embedding tensor was set as a learnable parameter, and an Adam optimizer was employed with a learning rate of 5e-3 over 40 steps. At each step, the model predicted the target property from the current IL embedding, and the negative prediction was used as a direction signal to maximize the property. Gradient clipping (max norm 5.0) was applied to stabilize updates.

After each optimization step, the updated embedding was decoded back into a SMILES string using beam search (beam size 3). The decoded SMILES was then re-encoded to obtain a refined embedding, which was used to compute the property prediction again. This decode-and-reencode loop allowed the optimization to explore chemically valid modifications while maintaining the embedding in the model's latent space. The SMILES string that yielded the highest predicted value across all steps was retained as the optimized candidate.

C Future Work

We plan to extend our framework towards an active learning paradigm that tightly integrates experimental feedback. By iteratively incorporating wet-lab results, the model can refine its predictions and guide the generation of formulations with increasingly improved performance. Moreover, we aim to deliberately propose candidates with lower predicted scores but higher structural novelty, enabling broader exploration of the formulation space and mitigating potential biases or gaps in the existing dataset. This approach promises a more comprehensive and data-efficient strategy for LNP design.