sc2Flow: Mitigating Mean Prediction Bias in Single-Cell Perturbation with Dual-Stage Flow Matching

Hanwen Lyu[1] , **Jiawei Luo**[1]
[1]Hunan University
{hanwenlv, luojiawei}@hnu.edu.cn

Abstract

Predicting single-cell gene responses to chemical perturbations is vital for personalized therapy, yet existing deep learning models face significant hurdles. Standard regression-based approaches suffer from "mean prediction bias", failing to capture cellular heterogeneity, while current Flow Matching (FM) methods struggle with the extreme sparsity of single-cell data and the dimensionality mismatches inherent in transformer-based generation. To address these challenges, we introduce sc2Flow, a dual-stage framework that unifies discrete and continuous flow matching. sc2Flow first predicts the binary mask of expressed genes and subsequently models their quantitative levels using a scalable Transformer. This decoupling effectively resolves dimensionality conflicts and eliminates parameter redundancy. Extensive benchmarks on Sci-Plex3 and other datasets demonstrate that sc2Flow significantly outperforms state-of-the-art baselines on distribution matching, successfully mitigating mean bias to preserve critical biological heterogeneity.

1 Introduction

High-throughput screening is a critical technology for advancing personalized cancer therapy [Carnero, 2006], enabling the identification of effective drugs and key genes through the analysis of post-perturbation gene expression profiles [Szalai and Veres, 2023]. However, testing all possible combinations of approved drugs would require astronomical resources and time [Menden *et al.*, 2019]. Although numerous deep learning–based approaches have been proposed to predict unobserved perturbation combinations [Aliper *et al.*, 2016], significant challenges remain.

Single-cell drug response exhibits a high degree of heterogeneity. As illustrated in Fig. 1a and Fig. 1b, when a minimal fraction of a cellular subtype (samples in the red box) responds to drug perturbation, other subtypes of the same cell population remain unresponsive. Traditional deep learning approaches for perturbation prediction, such as Variational Autoencoders (VAEs [Kingma *et al.*, 2013]), typically rely on regression objectives like Mean Squared Error (MSE). From a statistical perspective, minimizing MSE converges to the conditional expectation of the target distribution [Bishop and Nasrabadi, 2006]. Thus, these models tend to predict the average of the two modes (response/non-response), resulting in universal non-response for all samples or biologically non-existent intermediate states. At the gene level, the same issue persists: models tend to predict zero or underestimate its expression value.

To overcome this, Flow Matching (FM) has emerged as a powerful generative paradigm [Lipman *et al.*, 2022]. This approach aims to reconstruct the full target distribution rather than merely its expectation. Even when a gene is unexpressed in the majority of cells, Flow Matching can learn to "push" the appropriate probability mass toward high-expression states for specific subpopulations.

However, existing Flow Matching methods (e.g., CellFlow [Klein *et al.*, 2025]) predominantly rely on MLP architectures, which require fixed-dimensional inputs and outputs, forcing the model to process the entire genome. However, in most samples, fewer than 5% of genes are expressed (Fig. 1d left), even after HVGs selection (Fig. 1d right). This leads to substantial parameter redundancy, as a vast number of weights are wasted computing invalid "zero" values. Moreover, their parameter count grows linearly with gene set size. In contrast, Transformer [Vaswani *et al.*, 2017] is inherently suited for processing variable-length sequences. Its number of parameters is independent of the input sequence length, offering superior scalability. Although the attention mechanism scales quadratically, the input length is very short if only expressed genes are processed.

Nevertheless, implementing Transformer-based FM for perturbation prediction presents a unique technical challenge: FM typically requires the source and target spaces to share consistent dimensionality. However, gene set before and after perturbation are different. This creates a dimensionality mismatch, preventing the direct application of Control-to-Perturbed flow generation.

To tackle these challenges, we introduce sc2Flow, a dual-stage flow matching framework designed to predict single-cell drug perturbations. Our method decomposes the problem into two distinct phases: **(1) Discrete Value Prediction**: The model takes the discretized expression state of control cells (binary indicators of gene expression) and perturbation embeddings as input to predict the post-perturbation gene expression mask using discrete flow matching [Gat *et al.*, 2024].

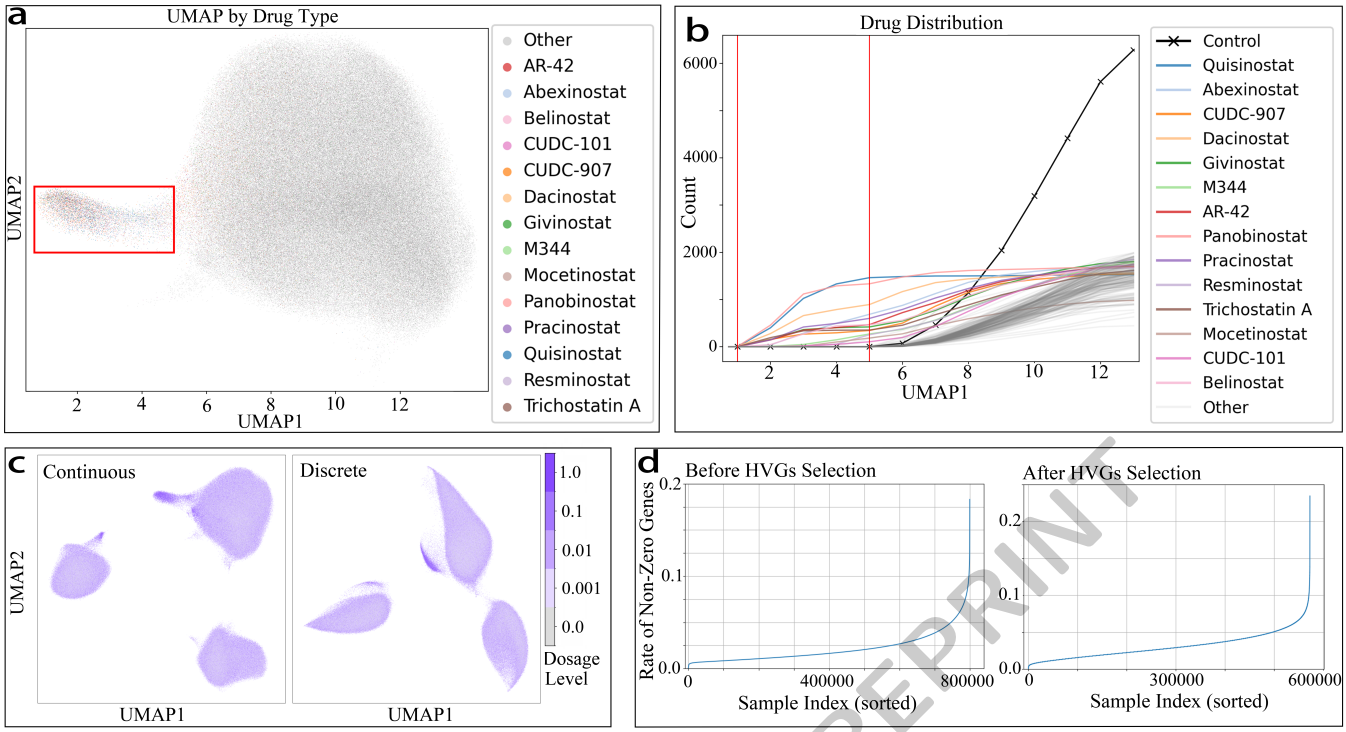

Figure 1: Data Distribution Analysis. This figure demonstrates the distribution of single-cell perturbation data using the Sci-Plex3 dataset as an example. (a) UMAP visualization of MCF7 gene profiles. Only a small fraction of samples respond to specific drugs (in the red box), forming a cluster distinct from the control group. (b) Sample counts for each drug along the distribution axis. The region between the two red lines contains no control samples, indicating that these drugs are effective only on specific cellular subtypes. (c) Comparison of UMAP embeddings derived from continuous expression values versus discrete (binary) data. This demonstrates that even with the loss of continuous signals, the discrete representation preserves significant perturbation information, as drugs alter not only expression levels but also the set of expressed genes. (d) The proportion of expressed genes within the total gene pool and the Highly Variable Genes (HVGs) subset.

To investigate potential biological information loss caused by disregarding continuous expression values, we compared UMAP visualizations of discretized (binary) data against the original continuous data. The results demonstrate that the discrete representation preserves significant perturbation feature (Fig. 1c). Given the enormous input dimensionality in this stage, we employ a multi-layer residual network based on MLPs for prediction to reduce computational costs. **(2) Continuous Value Prediction:** A Transformer model utilizes tokens from these filtered genes to predict their quantitative expression level. Since the first stage has already predicted the target post-perturbation genes, we can construct a vector field that maps Gaussian noise to the true expression values for this gene subset, enabling training via flow matching while avoiding the dimensionality mismatch problem.

Extensive experiments on Sci-Plex3 [Srivatsan *et al.*, 2020] and two other single-cell datasets [McFarland *et al.*, 2020; Zhao *et al.*, 2021b] demonstrate that sc2Flow significantly outperforms fully connected baselines (chemCPA [Hetzel *et al.*, 2022], biolord [Piran *et al.*, 2024]) and single-stage flow matching models CellFlow [Klein *et al.*, 2025]. Our contributions can be summarized as follows:

- We present sc2Flow, a Transformer-based framework that unifies discrete and continuous flow matching for precise drug perturbation prediction.
- sc2Flow is the first method that effectively mitigates mean prediction bias while maintaining both scalability and high parameter efficiency.
- Extensive benchmarking across multiple datasets confirms that sc2Flow not only predicts expression values accurately but also significantly outperforms existing state-of-the-art methods in distribution matching.

2 Related Work

2.1 Encoder-Decoder based Prediction

Previous methods primarily rely on encoder-decoder frameworks. Early approaches used VAEs to encode gene expression before and after perturbation into a latent space to predict responses on novel cell types (e.g., scGen [Lotfollahi *et al.*, 2019], CPA [Lotfollahi *et al.*, 2023]). Later works incorporated drug structure encodings to generalize to novel compounds (e.g., PRnet [Qi *et al.*, 2024], ChemCPA [Hetzel *et al.*, 2022], Biolord [Piran *et al.*, 2024]). Although these methods can achieve high scores on numerical prediction metrics (such as R^2), due to limitations in sequencing technology, researchers cannot obtain paired samples before and after perturbation. Consequently, they can only evaluate the difference between the average values of all predicted and target samples under a specific perturbation condition,

thereby concealing the aforementioned mean prediction bias problem. sc2Flow employs flow matching methods to mitigate this issue.

2.2 Flow Matching

Flow matching aims to learn a velocity field defined by Ordinary Differential Equations (ODEs) to construct a deterministic mapping between two distributions [Lipman *et al.*, 2022]. By optimizing the probability flow field of the perturbation response rather than relying on traditional Mean Squared Error (MSE) objectives, this approach explicitly models the continuous trajectory of cell state transitions, thereby mitigating mean prediction bias. Recently, many flow matching-based models for drug perturbation prediction (such as CellFlow) have emerged. Since flow matching requires consistent input and output dimensions, even when using Transformer architectures, these single-stage models must include expression values for all genes. In contrast, while sc2Flow faces the same constraint, it avoids interference from non-expressed genes because the expressed genes have already been filtered out during the discrete prediction stage.

3 Methods

This section outlines the architecture and computational methodology of sc2Flow. As illustrated in Fig. 2, the framework decomposes the drug perturbation prediction task into two independent components: **(1) Discrete Flow Matching (DFM)** for predicting the binary gene expression states (i.e., identifying which genes are expressed), and **(2) Continuous Flow Matching (CFM)** for predicting the precise expression values of these genes.

This decoupling addresses the zero-inflation and high dimensionality of single-cell data, which make direct vector prediction computationally intensive and difficult to optimize. By filtering genes via DFM, the Transformer-based CFM can focus on a compact sequence of relevant genes. We employ Uni-Mol2 for 3D molecular encoding and incorporate dosage via explicit embeddings. At inference, DFM identifies active genes to guide CFM in generating the final perturbed profile.

3.1 Problem Definition

Traditionally, drug perturbation prediction is formulated as a multivariate regression task. In this work, we reframe it as a conditional generative modeling problem using the Flow Matching framework. Let $\mathcal{X} \in \mathbb{R}^N$ denote the gene expression space for N genes. We define a source distribution p_0 representing the control group profiles $\mathbf{x}_0$, and a target distribution p_1 representing the perturbed group profiles $\mathbf{x}_1$. The generation is conditioned on the perturbation context $\mathbf{c} = \{\mathbf{c}_{\text{drug}}, \mathbf{c}_{\text{dose}}\}$. The goal is to learn a time-dependent vector field $v_t(\mathbf{x}, \mathbf{c})$ that pushes samples from p_0 to p_1 over the time interval $t \in [0, 1]$.

3.2 Conditional Encoding

Both DFM and CFM components share a unified conditional encoding scheme to process gene identities and drug perturbations.

Gene and Drug Embeddings

For gene identification, we assign a learnable embedding vector to each gene. Let $\mathbf{E}_{\text{gene}} \in \mathbb{R}^{N \times d}$ be the embedding matrix where the i-th row corresponds to the i-th gene. The process to obtain gene identity embeddings $\mathbf{h}_{\text{gene}} \in \mathbb{R}^{N \times d}$ for sample $\mathbf{x} \in \mathbb{R}^N$ is defined as:

$$\mathbf{h}_{\text{gene}} = \mathbf{E}_{\text{gene}}(\mathbf{x}) \tag{1}$$

For drug representation, the SMILES string of the drug is processed by the frozen Uni-Mol2 model. The resulting representation is projected to the model dimension d via a linear layer:

$$\mathbf{h}_{\text{drug}} = \text{Linear}_{\text{drug}}(\text{Uni-Mol2}(\text{SMILES})) \in \mathbb{R}^d \tag{2}$$

Dosage Encoding

To incorporate dosage information, we normalize the scalar dosage values to the range $[0, 1]$. This normalized scalar s_{dose} scales a learnable dosage embedding vector $\mathbf{e}_{\text{dose}} \in \mathbb{R}^d$:

$$\mathbf{h}_{\text{dose}} = s_{\text{dose}} \cdot \mathbf{e}_{\text{dose}} \tag{3}$$

This multiplicative interaction ensures that the dosage magnitude directly modulates the embedding intensity.

3.3 Discrete Flow Matching (DFM)

The DFM module models the transition of gene states from the control group to the perturbed group. We treat the expression state as a binary variable (0 for unexpressed, 1 for expressed).

Conditional Probability Path

Let $\mathbf{x} \in \{0, 1\}^N$ represent the binary expression vector. To apply Flow Matching on discrete data, we operate on the probability simplex. We map the discrete states to one-hot distributions. Let $\mathbf{p}_0 = \delta(\mathbf{x}_0)$ and $\mathbf{p}_1 = \delta(\mathbf{x}_1)$ be the probability mass functions of the source and target states, respectively. We define a linear conditional probability path $\mathbf{p}_t(\mathbf{x})$ at time t as:

$$\mathbf{p}_t = (1 - t)\delta(\mathbf{x}_0) + t\delta(\mathbf{x}_1) \tag{4}$$

where $\mathbf{p}_t \in \mathbb{R}^{N \times 2}$ represents the probability of each gene being expressed or not at time t. A noisy sample $\mathbf{x}_t$ can be drawn from this intermediate distribution:

$$\mathbf{x}_t \sim \text{Categorical}(\mathbf{p}_t) \tag{5}$$

Velocity Prediction via ResNet

The core of DFM is to predict the flow velocity, which drives the probability mass from $t = 0$ to $t = 1$.

We employ a Residual Network (ResNet) composed of $N_d = 12$ MLP Residual blocks. First, the model sets two learnable embeddings for expressed and unexpressed genes. Let $\mathbf{h}_t \in \mathbb{R}^{N \times d}$ be the embedding of $\mathbf{x}_t$. Then, the input to the network is the concatenation of $\mathbf{h}_t$, gene embeddings, condition embeddings, and the time step embedding t_{emb}:

$$\mathbf{H}_{\text{in}} = \text{Concat}(\mathbf{h}_t, \mathbf{h}_{\text{drug}}, \mathbf{h}_{\text{dose}}, t_{\text{emb}}) \tag{6}$$

The network predicts the logits $\hat{\mathbf{p}}_1$ for the target state $\mathbf{x}_1$:

$$\hat{\mathbf{p}}_1 = \text{Softmax}(\text{ResNet}(\mathbf{H}_{\text{in}})) \tag{7}$$

Although the model predicts the final state distribution $\hat{\mathbf{p}}_1$, this formulation implicitly defines the vector field necessary to reach the target.

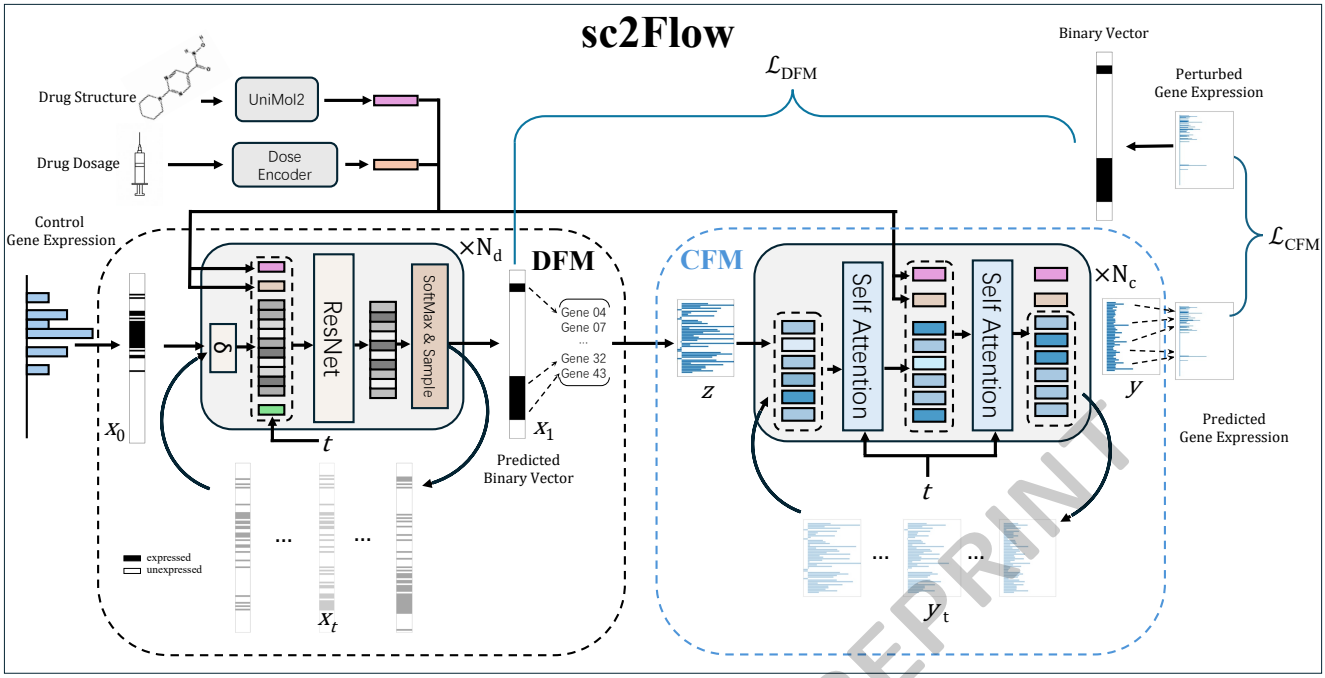

Figure 2: Overview of the sc2Flow Framework. The model consists of two decoupled stages based on Conditional Flow Matching. (Left Box) Discrete Flow Matching: This stage predicts the binary expression state (expressed/not expressed) of the whole gene set. It constructs a probability path from the control state ($\mathbf{x}_0$) to the perturbed state ($\mathbf{x}_1$) and utilizes a ResNet-based velocity field approximator to predict the final probability mass. (Right Box) Continuous Flow Matching (CFM): Focusing only on the genes predicted to be expressed, this stage predicts their specific continuous expression values. It employs a Transformer-based architecture with a single-stream mechanism to fuse gene tokens with drug and dosage embeddings, mapping the initial noise distribution $\mathbf{z}$ to the target expression profile $\mathbf{y}$.

Training Objective

Since the target $\mathbf{x}_1$ is binary, the model is trained using the binary cross-entropy loss averaged over all genes:

$$\mathcal{L}_{\text{DFM}} = -\frac{1}{N}\sum_{i=1}^{N}\left[x_{1,i}\log(\hat{p}_{1,i}) + (1-x_{1,i})\log(1-\hat{p}_{1,i})\right] \tag{8}$$

Inference Process

During inference, we solve the ODE defined by the predicted vector field. The update rule for the probability $\mathbf{p}_t$ moving from t to $t+h$ is derived from the linear path assumption. The velocity field u_t required to reach $\hat{\mathbf{p}}_1$ from $\mathbf{p}_t$ is:

$$u_t(\mathbf{x}_t) = \frac{\hat{\mathbf{p}}_1 - \mathbf{p}_t}{1-t} \tag{9}$$

Applying an Euler step:

$$\mathbf{p}_{t+h} = \mathbf{p}_t + h \cdot u_t(\mathbf{x}_t) = \mathbf{p}_t + h\frac{\hat{\mathbf{p}}_1 - \mathbf{p}_t}{1-t} \tag{10}$$

Finally, the discrete vector is sampled: $\mathbf{x}_{t+h} = \text{Categorical}(\mathbf{p}_{t+h})$. During inference, we generate the values by numerically integrating the ODE from $t=0$ (control) to $t=1$ (perturbation) using the predicted vector field u_t.

3.4 Continuous Flow Matching (CFM)

The CFM module predicts the precise expression values. To enhance computational efficiency, the CFM module restricts its input exclusively to the subset of expressed genes. During training, we select this subset based on ground truth labels to avoid being affected by the overall performance of the DFM, thereby ensuring training stability. Conversely, during inference, the subset is dynamically determined by the binary states predicted by the DFM stage.

Sequence Construction

Let M be the number of expressed genes in $\mathbf{x}_1$. We construct a sequence of tokens $\mathbf{Y} \in \mathbb{R}^{M\times d}$ from $\mathbf{x}_1$. Each token combines the gene's identity embedding and its continuous expression value. The conditional flow for the continuous values is defined as:

$$\mathbf{y}_t = (1-t)\mathbf{z} + t\mathbf{y}_1 \tag{11}$$

where $\mathbf{z} \sim \mathcal{N}(0, I)$ is the initial Gaussian noise and $\mathbf{y}_1$ is the target expression.

Transformer with Single-Stream Fusion

We employ a Transformer architecture to capture the interactions between genes and the perturbation conditions. To efficiently fuse the multimodal inputs (gene tokens, drug, dose), we adopt a Single-Stream architecture proposed in [Cai *et al.*,

2025]. We also use Sinusoidal Positional Encoding $\text{SinCos}(\cdot)$ [Su *et al.*, 2024] to map the scalar expression value $y \in \mathbb{R}$ to a vector:

$$\mathbf{y}_{\text{emb}} = \mathbf{E}_{\text{gene}}(\mathbf{y}_t) + \text{SinCos}(\mathbf{y}_t) \tag{12}$$

Then, this sequence is processed by $N_c = 5$ layers of Transformer blocks. Each block consists of Multi-Head Self-Attention (MSA) and a Feed-Forward Network (FFN), with Adaptive Layer Norm (AdaLN) conditioning on the time t. Additionally, to stabilize training directly in the original data space, we adopt the method of [Li and He, 2025], which directly predicts the final sample $\mathbf{y}_1$:

$$\begin{aligned} \mathbf{y}'_{\text{emb}} &= \text{MSA}(\text{AdaLN}(\mathbf{y}_{\text{emb}}, t_{\text{emb}})) \\ \hat{\mathbf{y}}_1 &= \text{FFN}(\text{AdaLN}([\mathbf{y}'_{\text{emb}}, \mathbf{h}_{\text{drug}}, \mathbf{h}_{\text{dose}}], t_{\text{emb}})) \end{aligned} \tag{13}$$

This allows every gene token to directly attend to the drug and dosage information within the self-attention mechanism.

Prediction and Loss

Similar to the DFM stage, the CFM model is trained to predict the clean target data $\mathbf{y}_1$ from the noisy input $\mathbf{y}_t$. The output corresponding to the gene tokens is projected back to the scalar expression space to obtain $\hat{\mathbf{y}}_1$. The training objective is the Mean Squared Error (MSE):

$$\mathcal{L}_{\text{CFM}} = \|\hat{\mathbf{y}}_1 - \mathbf{y}_1\|^2 \tag{14}$$

During inference, we generate the values by numerically integrating the ODE from $t = 0$ (noise) to $t = 1$ (prediction) using the predicted vector field $v_t = (\hat{\mathbf{y}}_1 - \mathbf{y}_t)/(1 - t)$.

4 Experiment

4.1 Setup

Baseline Methods

We trained all models using their official open-source implementations with optimal hyperparameters. Detailed training configurations are provided in the Supplementary Materials. The baseline methods are listed as follows:

- **chemCPA** [Hetzel *et al.*, 2022] is an adversarial learning method based on VAE. It demonstrates state-of-the-art performance on single-cell drug perturbation response benchmarks for single-drug treatments [Wei *et al.*, 2025].
- **biolord** [Piran *et al.*, 2024] is based on VAE and enables counterfactual predictions by disentangling the biological state of cells from external perturbations. It has since become a widely adopted state-of-the-art baseline in the field.
- **CellFlow** [Klein *et al.*, 2025] is a recent drug perturbation response generation model based on Optimal Transport Conditional Flow Matching, which predicts velocity fields via MLPs.
- **Unpredicted** is a reference method. It makes no predictions and directly outputs the control group, serving to illustrate the inherent difficulty of predicting perturbation effects in the dataset.

Metrics

For the evaluation of generated molecules, we consider the following metrics.

- $\mathbf{R^2}$ **(Coefficient of Determination)** Measures the proportion of variance in the true gene expression data that is predictable from the generated data. It quantifies the overall fidelity of the predictions, with a higher R^2 indicating better agreement.
- **Energy Distance (ED)** This parameter-free metric provides a robust, objective measure of global distribution similarity. By capturing all statistical moments without hyperparameter tuning, it ensures the generated population faithfully mirrors the shape and spread of real data.
- **Maximum Mean Discrepancy (MMD)** A kernel-based metric (using Gaussian RBF) that is particularly advantageous for high-dimensional data, allowing it to capture fine-grained, local discrepancies and subtle structural differences in complex single-cell phenotypes.

Dataset

We conduct experiments on the following single-cell drug perturbation datasets.

- **Sci-Plex3** [Srivatsan *et al.*, 2020] comprises approximately 572,853 samples, involving 3 cell lines and 188 drug perturbations. Following the practice of prior work, we allocate the 9 drugs with the most significant perturbation effects as the test set.
- **McFarland** [McFarland *et al.*, 2020] contains 127,973 samples across 172 cell types treated with 11 drugs. We create partitions based on drugs. Specifically, we generate 5 different data splits, each with 2 to 3 drugs held out in the test set.
- **Zhao** [Zhao *et al.*, 2021a] includes roughly 155,980 samples, recording the perturbation effects of 4 drugs on Glioma. We adopt a leave-one-drug-out strategy, resulting in 4 data splits where each drug serves as the test set once.

Gene Selection

All models were trained on an identical gene panel to ensure a fair comparison.

- **HVGs:** Since all baseline methods predict only a preselected set of 2000 HVGs, for a fair comparison, our method (sc2Flow) formally operates on the same full gene set of size 2000 after normalization using $\log(1 + \mathbf{x})$.
- **Differentially Expressed Genes (DEGs):** Since most genes remain unchanged post-perturbation, we focus evaluation on the top 50 DEGs per drug (identified by Seurat). This provides a more rigorous assessment of the model's predictive power for biologically relevant responses.

4.2 Results and Analysis

Robustness Across Dosage and Datasets

We evaluated sc2Flow against state-of-the-art baselines across three datasets (Sci-Plex3, McFarland, and Zhao). The

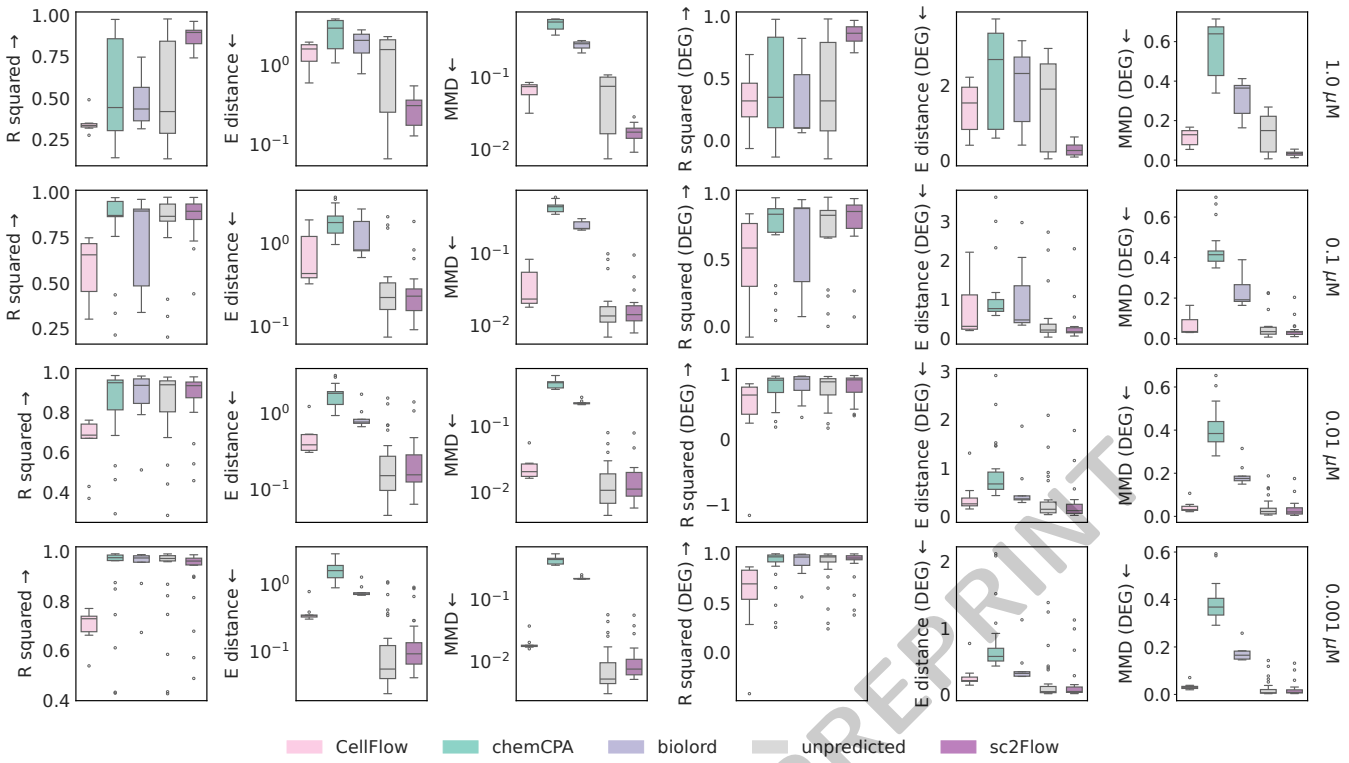

Figure 3: Box Plots of Performance Results on Sci-Plex3. The figure shows the distribution of results across six metrics for different methods under four dosage conditions. An upward arrow indicates that a higher value of the metric corresponds to better model performance, while a downward arrow indicates the opposite.

quantitative results (Table 1 and Table 2) and visual distributions (Fig. 3) demonstrate that sc2Flow establishes a new standard for perturbation prediction.

Superiority in Distributional Alignment

The most significant advantage of sc2Flow lies in its ability to model the high-dimensional data distribution rather than merely predicting conditional means. As shown in Table 1 and Table 2, sc2Flow achieves substantial improvements in ED and MMD across all datasets. On the McFarland dataset, sc2Flow reduces the MMD by 89.5% compared to biolord (0.639 → 0.067) and 85% compared to chemCPA. Similarly, the Energy Distance is reduced by over 68% compared to the strongest baseline. On the Zhao dataset, the performance gap is even more pronounced, with sc2Flow achieving an MMD of 0.012, an order of magnitude lower than chemCPA (0.139) and biolord (0.633). These metrics confirm that while VAE-based methods (chemCPA, biolord) often suffer from or averaging effects—resulting in generated cells that cluster around the mean—sc2Flow's flow matching paradigm successfully captures the heterogeneity of the perturbed cell populations.

Precision in Gene Expression Recovery

Beyond distribution matching, sc2Flow excels in recovering precise gene expression levels, measured by R^2. In the McFarland dataset (Table 1), our model achieves an R^2 of 0.901, comparable to the state-of-the-art biolord (0.907) and significantly outperforming chemCPA (0.785). However, in the more challenging Zhao dataset (Table 2), sc2Flow demonstrates superior robustness, achieving the highest R^2 of 0.816, whereas baseline performance degrades significantly (chemCPA: 0.700). Single-stage flow matching models like CellFlow compress high-dimensional gene space into low-dimensional latent vectors (e.g., 100 principal components)

Metric	chemCPA	biolord	**sc2Flow** (Ours)
R^2 ↑	0.785 ± 0.060*	**0.907 ± 0.026**	0.901 ± 0.023
ED ↓	5.304 ± 0.986*	4.136 ± 0.449*	**1.304 ± 0.329**
MMD ↓	0.449 ± 0.069*	0.639 ± 0.040*	**0.067 ± 0.016**
R^2 (DEGs) ↑	0.798 ± 0.067	0.856 ± 0.061	**0.857 ± 0.056**
ED (DEGs) ↓	2.256 ± 0.439*	1.979 ± 0.324*	**0.858 ± 0.268**
MMD (DEGs) ↓	0.495 ± 0.055*	0.583 ± 0.030*	**0.075 ± 0.019**

Table 1: Performance comparison on McFarland dataset. * indicates t-test $p < 0.01$. Bold values indicate the best performance.

Metric	chemCPA	biolord	**sc2Flow** (Ours)
R^2 ↑	0.700 ± 0.156	0.775 ± 0.066	**0.816 ± 0.124**
ED ↓	1.261 ± 0.182*	2.396 ± 0.192*	**0.273 ± 0.174**
MMD ↓	0.139 ± 0.011*	0.633 ± 0.017*	**0.012 ± 0.008**
R^2 (DEGs) ↑	0.337 ± 0.486	0.431 ± 0.214	**0.520 ± 0.438**
ED (DEGs) ↓	0.796 ± 0.033*	1.387 ± 0.068*	**0.211 ± 0.142**
MMD (DEGs) ↓	0.207 ± 0.015*	0.547 ± 0.004*	**0.018 ± 0.012**

Table 2: Performance comparison on Zhao dataset. * indicates t-test $p < 0.01$. Bold values indicate the best performance.

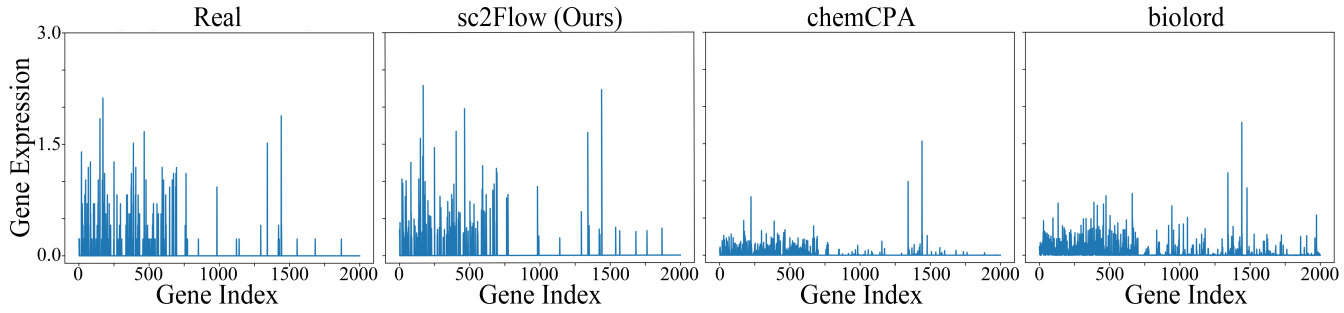

Figure 4: Visualization of Distribution Matching Performance. We compare a real sample (A549 cells treated with 0.001 μM Alvespimycin) against the best-performing predictions (lowest MSE) from all models. The samples from left to right are derived from real data, sc2Flow, chemCPA, and Biolord, respectively.

to manage computational cost. This compression inherently causes information loss, leading to larger reconstruction errors (Fig. 3, R^2 panel). In contrast, sc2Flow's first stage (DFM) explicitly filters non-expressed genes, allowing the second stage (CFM) to operate on the full raw dimensionality of the active genes without information loss.

A critical requirement for drug discovery is the ability to predict changes in DEGs, which represent the biologically relevant response to treatment. When evaluating specifically on DEGs, sc2Flow maintains high R^2 on DEGs, whereas baselines often show performance drops when zoomed in on DEGs (Table 1). By explicitly modeling the "zero-inflation" via the Discrete Flow Matching stage, sc2Flow avoids the common pitfall of predicting small, non-zero noise values for unexpressed genes. This ensures that the model focuses its generative capacity on genes that are actually biologically active, resulting in generated profiles that are biologically indistinguishable from real data (Fig. 3).

We observed distinct behavioral differences between model architectures across different perturbation intensities. As shown in (Fig. 3), traditional models (chemCPA, biolord) tend to default to the "unpredicted" (control) state when facing high-dose perturbations, essentially failing to capture the strong drug effect. This demonstrates that traditional methods are affected by the mean prediction bias problem.

In contrast, sc2Flow exhibits active predictive capabilities at high doses. By learning the continuous velocity field from control to perturbed states, our model effectively "pushes" the cell representation along the correct trajectory. This is evidenced by the consistent superiority in distribution metrics (ED and MMD) across the comprehensive 9-split cross-validation on the experimental datasets, where sc2Flow outperformed competitors in nearly all scenarios.

4.3 Visual Comparison of Generated Samples

To visualize the superior distribution matching capabilities of sc2Flow, we analyzed the response of A549 cells to Alvespimycin (0.001 μM). Fig. 4 displays the ground truth sample alongside the predicted samples from each model that achieved the minimum Mean Squared Error (MSE) relative to the real data.

Results show that sc2Flow best aligns with the ground truth, capturing the high sparsity and sharp peaks characteristic of real gene expression. Unlike baselines that tend toward mean-averaging, sc2Flow's two-stage architecture first identifies expressed genes and then generates precise expression levels. This success is driven by the Flow Matching objective, which learns deterministic trajectories to the target distribution rather than approximating a static mean. Consequently, sc2Flow produces high-fidelity peaks that faithfully reflect biological signals, overcoming the "mean prediction bias" of traditional methods.

5 Conclusion

In this work, we propose sc2Flow, a Transformer-based Two-Stage Flow Matching framework for efficient single-cell drug perturbation prediction. Extensive experiments demonstrate that sc2Flow significantly surpasses state-of-the-art autoencoder and single-stage baselines, particularly in preserving the distributional heterogeneity of cell responses. The reasons for sc2Flow's strong performance include the decoupling of discrete and continuous prediction which effectively handles data sparsity, the explicit exclusion of non-expressed genes to isolate the Transformer from noise, and the use of flow matching to capture complex distributional shifts rather than simple average effects. Moreover, sc2Flow establishes a new efficient paradigm for in silico drug screening under resource-constrained settings.

6 Data and Code Availability

The code and datasets for this work are available at https://github.com/hanwenlv-cmd/sc2Flow.

Acknowledgments

This work was supported by the National Natural Science Foundation of China [62372165].

References

[Aliper *et al.*, 2016] Alexander Aliper, Sergey Plis, Artem Artemov, Alvaro Ulloa, Polina Mamoshina, and Alex Zhavoronkov. Deep learning applications for predicting pharmacological properties of drugs and drug repurposing using transcriptomic data. *Molecular pharmaceutics*, 13(7):2524–2530, 2016.

[Bishop and Nasrabadi, 2006] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[Cai *et al.*, 2025] Huanqia Cai, Sihan Cao, Ruoyi Du, Peng Gao, Steven Hoi, Zhaohui Hou, Shijie Huang, Dengyang Jiang, Xin Jin, Liangchen Li, et al. Z-image: An efficient image generation foundation model with single-stream diffusion transformer. *arXiv preprint arXiv:2511.22699*, 2025.

[Carnero, 2006] Amancio Carnero. High throughput screening in drug discovery. *Clinical and Translational Oncology*, 8:482–490, 2006.

[Gat *et al.*, 2024] Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky TQ Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching. *Advances in Neural Information Processing Systems*, 37:133345–133385, 2024.

[Hetzel *et al.*, 2022] Leon Hetzel, Simon Boehm, Niki Kilbertus, Stephan Günnemann, Fabian Theis, et al. Predicting cellular responses to novel drug perturbations at a single-cell resolution. *Advances in Neural Information Processing Systems*, 35:26711–26722, 2022.

[Kingma *et al.*, 2013] Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013.

[Klein *et al.*, 2025] Dominik Klein, Jonas Simon Fleck, Daniil Bobrovskiy, Lea Zimmermann, Sören Becker, Alessandro Palma, Leander Dony, Alejandro Tejada-Lapuerta, Guillaume Huguet, Hsiu-Chuan Lin, et al. Cellflow enables generative single-cell phenotype modeling with flow matching. *bioRxiv*, pages 2025–04, 2025.

[Li and He, 2025] Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. *arXiv preprint arXiv:2511.13720*, 2025.

[Lipman *et al.*, 2022] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[Lotfollahi *et al.*, 2019] Mohammad Lotfollahi, F Alexander Wolf, and Fabian J Theis. scgen predicts single-cell perturbation responses. *Nature methods*, 16(8):715–721, 2019.

[Lotfollahi *et al.*, 2023] Mohammad Lotfollahi, Anna KCellOTlimovskaia Susmelj, Carlo De Donno, Leon Hetzel, Yuge Ji, Ignacio L Ibarra, Sanjay R Srivatsan, Mohsen Naghipourfar, Riza M Daza, Beth Martin, et al. Predicting cellular responses to complex perturbations in high-throughput screens. *Molecular systems biology*, 19(6):e11517, 2023.

[McFarland *et al.*, 2020] James M McFarland, Brenton R Paolella, Allison Warren, Kathryn Geiger-Schuller, Tsukasa Shibue, Michael Rothberg, Olena Kuksenko, William N Colgan, Andrew Jones, Emily Chambers, et al. Multiplexed single-cell transcriptional response profiling to define cancer vulnerabilities and therapeutic mechanism of action. *Nature communications*, 11(1):4296, 2020.

[Menden *et al.*, 2019] Michael P Menden, Dennis Wang, Mike J Mason, Bence Szalai, Krishna C Bulusu, Yuanfang Guan, Thomas Yu, Jaewoo Kang, Minji Jeon, Russ Wolfinger, et al. Community assessment to advance computational prediction of cancer drug combinations in a pharmacogenomic screen. *Nature communications*, 10(1):2674, 2019.

[Piran *et al.*, 2024] Zoe Piran, Niv Cohen, Yedid Hoshen, and Mor Nitzan. Disentanglement of single-cell data with biolord. *Nature Biotechnology*, 42(11):1678–1683, 2024.

[Qi *et al.*, 2024] Xiaoning Qi, Lianhe Zhao, Chenyu Tian, Yueyue Li, Zhen-Lin Chen, Peipei Huo, Runsheng Chen, Xiaodong Liu, Baoping Wan, Shengyong Yang, et al. Predicting transcriptional responses to novel chemical perturbations using deep generative model for drug discovery. *Nature Communications*, 15(1):1–19, 2024.

[Srivatsan *et al.*, 2020] Sanjay R Srivatsan, José L McFaline-Figueroa, Vijay Ramani, Lauren Saunders, Junyue Cao, Jonathan Packer, Hannah A Pliner, Dana L Jackson, Riza M Daza, Lena Christiansen, et al. Massively multiplex chemical transcriptomics at single-cell resolution. *Science*, 367(6473):45–51, 2020.

[Su *et al.*, 2024] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[Szalai and Veres, 2023] Bence Szalai and Dániel V Veres. Application of perturbation gene expression profiles in drug discovery—from mechanism of action to quantitative modelling. *Frontiers in Systems Biology*, 3:1126044, 2023.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[Wei *et al.*, 2025] Zhiting Wei, Yiheng Wang, Yicheng Gao, Shuguang Wang, Ping Li, Duanmiao Si, Yuli Gao, Siqi Wu, Danlu Li, Kejing Dong, et al. Benchmarking algorithms for generalizable single-cell perturbation response prediction. *Nature Methods*, pages 1–14, 2025.

[Zhao *et al.*, 2021a] Wenting Zhao, Athanassios Dovas, Eleonora Francesca Spinazzi, Hanna Mendes Levitin, Matei Alexandru Banu, Pavan Upadhyayula, Tejaswi Sudhakar, Tamara Marie, Marc L Otten, Michael B Sisti, et al. Deconvolution of cell type-specific drug responses in human tumor tissue with single-cell rna-seq. *Genome medicine*, 13(1):82, 2021.

[Zhao *et al.*, 2021b] Yucheng Zhao, Guangting Wang, Chuanxin Tang, Chong Luo, Wenjun Zeng, and Zheng-Jun Zha. A battle of network structures: An empirical study of cnn, transformer, and mlp. *arXiv preprint arXiv:2108.13002*, 2021.