CaLMFlow: Volterra Flow Matching using Causal Language Models

Sizhuang He*, Daniel Levine*, Ivan Vrkic, Marco Bressana, David Zhang, Syed Rizvi, Yangtian Zhang, Emanuele Zappala† and David van Dijk†

Yale University

Introduction

We introduce **CaLMFlow (Causal Language Models for Flow Matching)**, a novel framework that casts flow matching as a Volterra integral equation (VIE), leveraging the power of Causal Language Models (CLMs) for **continuous data generation**.

Theoretical Background

Flow Matching

Flow matching is a *simulation-free* framework that models data *flows* $\phi(x,t)$ with an **ordinary differential equation (ODE)**.

$$\frac{d\phi}{dt} = v(\phi, t), \qquad \phi(x,0) = x, \tag{1}$$

Limitations of ODE-based methods

Many ODE systems suffer from **stiffness**.

- This usually happens in systems with *rapid changes or long-range dependencies*.
- Such systems are *numerically unstable and computationally expensive* to solve.

Volterra Integral Equations

- Volterra Integral Equations (VIEs) generalize ODEs, avoiding stiffness.
- We can reformulate the flow matching problem using VIEs

$$z(t) = f(z(t), t) + \int_0^t G(z(s), t, s)ds \tag{2}$$

Solving VIEs with Causal Language Models

- We derive a discretized VIE through discretization of the time domain:

$$\hat{z}^{i+1} = f(z^i, t_{i+1}) + \sum_{j=0}^{i} \Delta t_{i+1} G(z_j, t_{i+1}, t_j), \tag{3}$$

- CLMs act as an iterative integral equation solver of the discretized VIE.
- Training CLMs is also *simulation-free*

Key Contributions

- We formulate **flow matching as VIEs that are solved by CLMs**, enhancing stability and generation quality of ODE-based flow matching.
- Our framework natively enable **controllable generation of flows using natural language prompts**.
- We developed **continuous space tokens via variational decoding**, extending language modeling techniques to continuous domains.
- We implemented **integration over different domains via spatiotemporal and trajectory tokenization**.

CaLMFlow can model high dimensional distributions where ODE-based methods fail

	Gaussian → 2 Gaussians	Gaussian → 8 Gaussians	Gaussian → 2 Moons
		Data Dimension = 100	
CFM	5.483 ± 0.569	4.846 ± 0.054	5.061 ± 0.103
CFM-OT	5.494 ± 0.517	4.795 ± 0.031	5.013 ± 0.058
CFM-SB	5.504 ± 0.446	4.914 ± 0.038	5.294 ± 0.042
CaLMFlow	**3.137±1.028**	**2.317±0.226**	**2.944±0.195**
		Data Dimension = 1000	
CFM	25.064 ± 1.291	23.294 ± 0.166	23.428 ± 0.187
CFM-OT	25.131 ± 1.209	23.116 ± 0.118	23.339 ± 0.133
CFM-SB	25.053 ± 1.558	23.211 ± 0.078	23.805 ± 0.132
CaLMFlow	**11.027±3.853**	**8.272±0.272**	**13.423±0.258**

Table 1. 2-Wasserstein distance comparison (lower values are better) of CaLMFlow and CFM variants across different distribution pairs and dimensions. As the dimensionality increases, **CaLMFlow consistently outperforms CFM variants**, particularly in high-dimensional settings where we expect traditional ODE-based methods, such as CFM, struggle.

CaLMFlow Framework

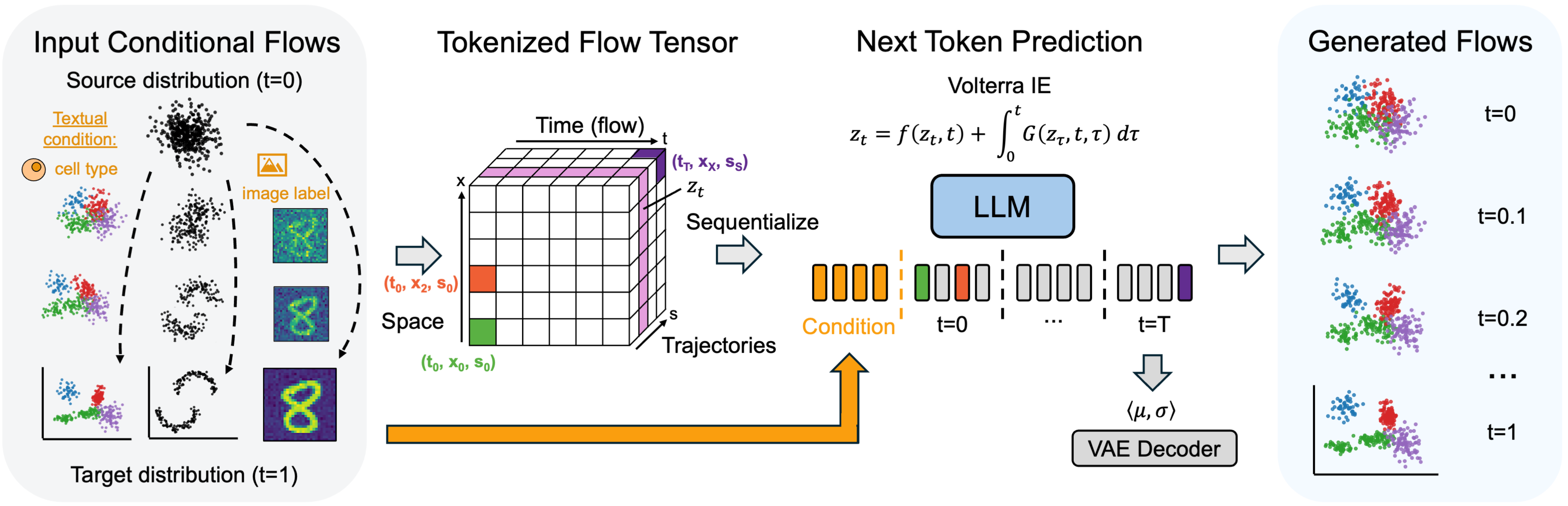


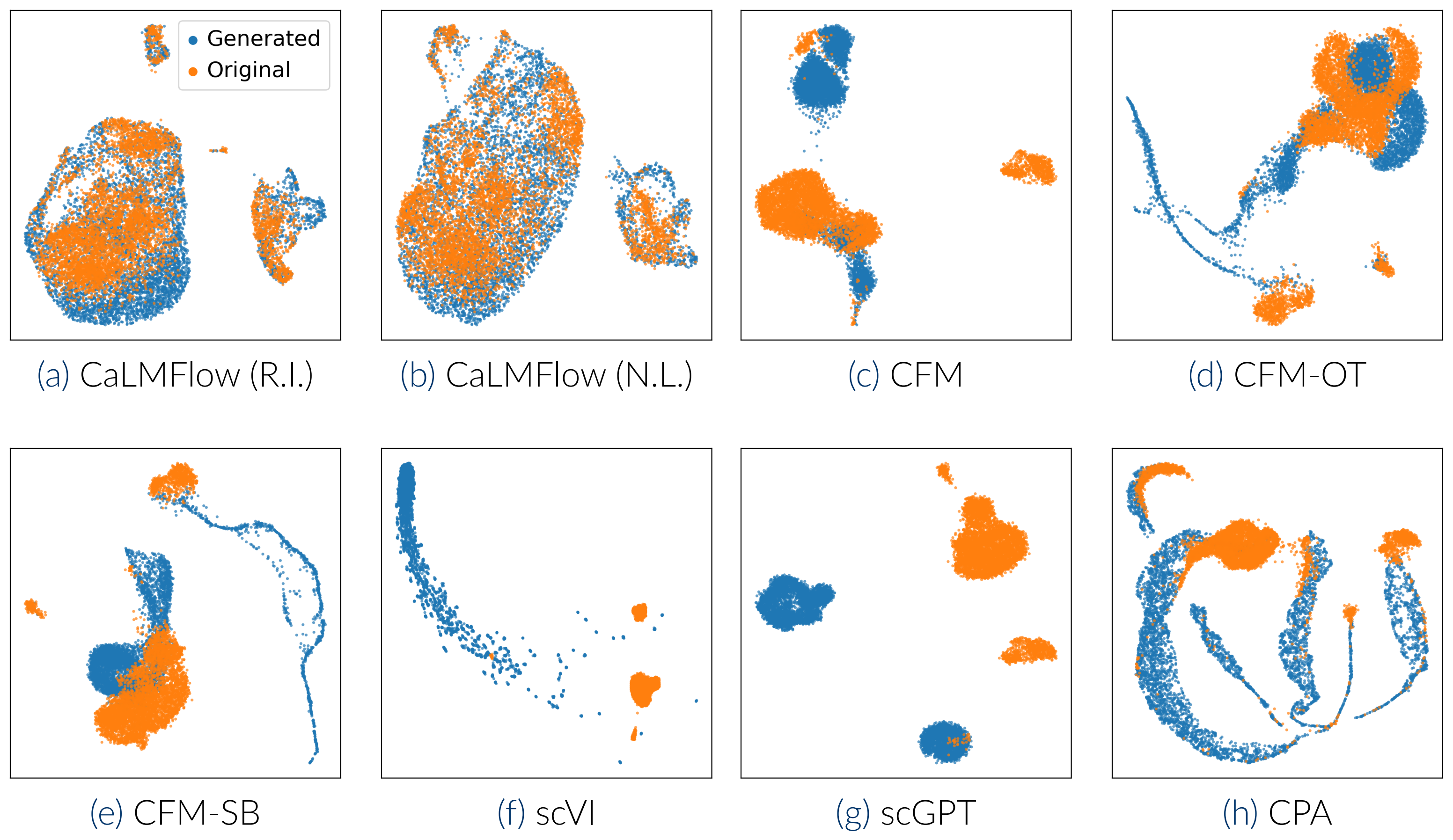
Figure 1. Overview of the CaLMFlow framework. CaLMFlow takes as input textual conditions and flows and generates the next time point for the flows. The textual condition is tokenized and embedded using the LLM tokenizer and embedding layer, while the conditional flows are transformed into spatial-temporal tokens using a learned projection. If multiple conditional flows are input simultaneously, the tokens are ordered by flow, space, and then time. The LLM applies causal language modeling and generates the next time point for each flow.

CaLMFlow accurately generates complex real world single-cell data distributions

Method ↓ Metric →	MMD(↓)	2-Wasserstein(↓)	Leiden KLD(↓)	adMMD(↓)
CFM	0.0763 ± 0.0275	0.0158 ± 0.0043	0.0330 ± 0.0027e-2	9.3568e-4 ± 0.7058e-4
CFM-OT	0.0893 ± 0.0193	0.0149 ± 0.0012	0.0324 ± 0.0039e-2	9.1720e-4 ± 0.4719e-4
CFM-SB	0.0998 ± 0.0050	0.0151 ± 0.0024	0.0338 ± 0.0045e-2	9.5234e-4 ± 0.3037e-4
DDPM	0.0709 ± 0.0010	0.0348 ± 0.0068	0.0364 ± 0.0101e-2	3.8040e-4 ± 0.1516e-4
scVI	0.1326 ± 0.0230	0.0349 ± 0.0020	0.0360 ± 0.0096e-2	11.1673e-4 ± 0.4967e-4
scGPT	0.3118 ± 0.0063	0.4716 ± 0.0741	–	18.1949e-4 ± 0.0531e-4
CaLMFlow (1 traj.)	0.0060 ± 0.0002	0.0100 ± 0.0006	**0.0311 ± 0.0045e-2**	2.4795e-4 ± 0.0460e-4
CaLMFlow (5 traj.)	**0.0031 ± 0.0001**	**0.0087 ± 0.0006**	0.0331 ± 0.0158e-2	**1.8039e-4 ± 0.0239e-4**

Table 2. Distributional metrics between model generated data and ground truth data. **Our default CaLMFlow outperforms all methods across all metrics**, demonstrating CaLMFlow's ability to model the data distribution. **Further improvement is seen with CaLMFlow (5 traj.)**, showing the benefit of multi-trajectory tokenization.

CaLMFlow accurately performs conditional single cell generation using natural language prompts

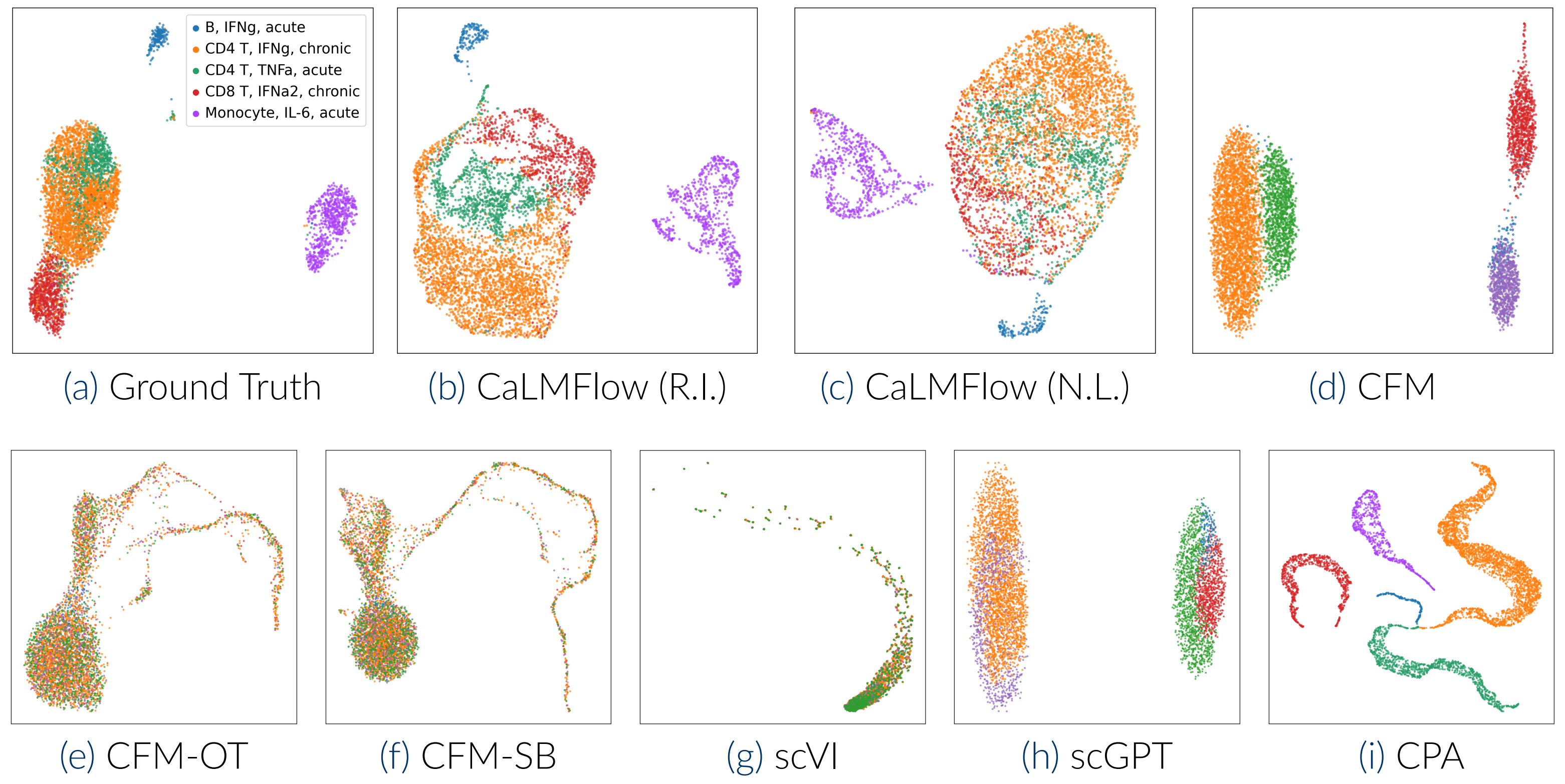


(a) CaLMFlow (R.I.) (b) CaLMFlow (N.L.) (c) CFM (d) CFM-OT

(e) CFM-SB (f) scVI (g) scGPT (h) CPA

Figure 2. Comparison of conditional generation quality across different models for single-cell perturbation data. **CaLMFlow exhibits strong overlap between generated data (blue) and the ground-truth distribution (orange)**, highlighting its superior capability to model data with unseen combinatorial perturbations. For CaLMFlow, R.I. refers to randomly initialized CLM, and N.L. refers to natural language pretrained CLM.

CaLMFlow is able to generalize to unseen single cell combinatorial perturbation conditions

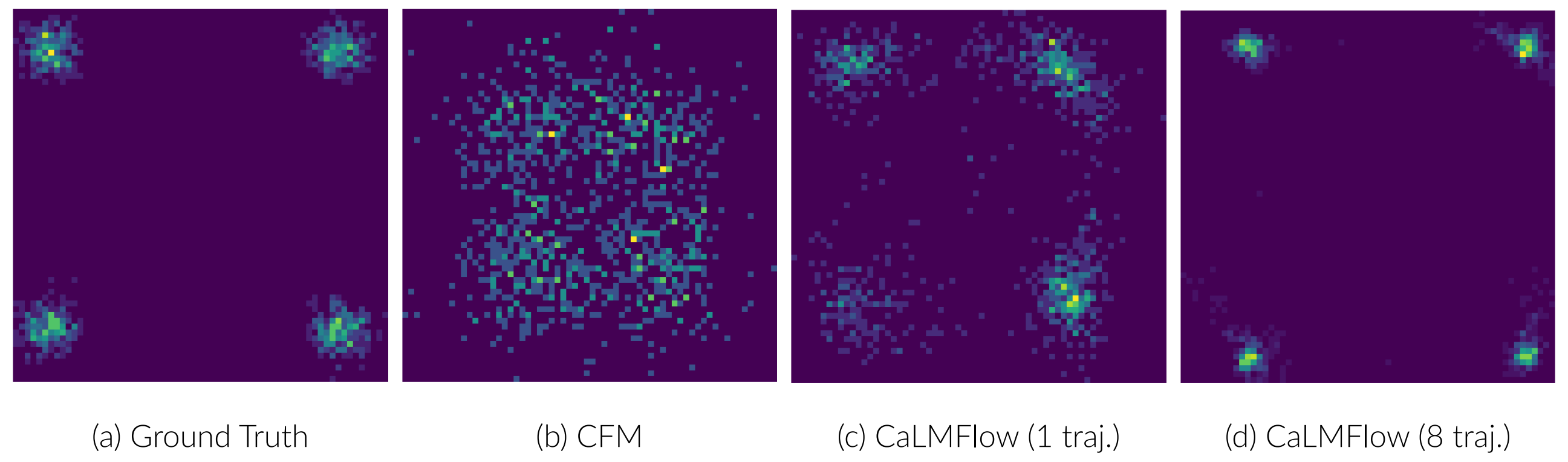


(a) Ground Truth (b) CaLMFlow (R.I.) (c) CaLMFlow (N.L.) (d) CFM

(e) CFM-OT (f) CFM-SB (g) scVI (h) scGPT (i) CPA

Figure 3. Comparison of conditional generation quality across different models for single-cell perturbation data. CaLMFlow generates data that **accurately reflects the ground truth distribution across all combinatorial labels (cell type, perturbation, and chronicity)**, demonstrating its superior ability to understand complex conditions while maintaining a realistic overall data distribution. For CaLMFlow, R.I. refers to randomly initialized CLM, and N.L. refers to natural language pretrained CLM.

CaLMFlow allows concurrent generation of multiple trajectories improving sample quality



(a) Ground Truth (b) CFM (c) CaLMFlow (1 traj.) (d) CaLMFlow (8 traj.)

Figure 4. Heatmaps of the ground truth 4 Gaussians dataset and that generated by CFM and CaLMFlow. CaLMFlow generate distributions closely matching the ground truth, with the 8-traj version distributing the data further **evenly** and **accurately**.

*: Equal contribution, †: Corresponding authors

sizhuang.he@yale.edu, david.vandijk@yale.edu