TRANSFLOW: SOURCE-INITIALIZED FLOW TRANSPORT FOR CROSS-MODALITY TRANSLATION IN SINGLE-CELL MULTI-OMICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Single-cell multi-omics provides complementary molecular measurements, but many cells are profiled in only one modality, motivating cross-modality translation of unmeasured molecular states. Conditional generative approaches often use the observed source modality as context while initializing the target-generation trajectory from noise. For paired data, however, the source and target measurements describe the same cell, making the encoded source a natural candidate for the initial state. We propose TransFlow, a source-initialized flow framework that transports paired RNA and ATAC representations between modality-specific latent spaces. Block-wise variational autoencoders encode the two modalities, and a shared bidirectional interval-conditioned velocity field models transport in both directions. A controlled noise-initialized variant receives the same source information as conditioning but begins from Gaussian noise, separating trajectory initialization from source availability. Multiple-run evaluations on BMMC and PBMC10k show that TransFlow obtains the lowest distributional scores in the main comparison, while feature-level results vary across datasets and directions. A single-run MiniAtlas experiment shows a similar distributional trend, and TransFlow obtains the best reported results in the evaluated BMMC held-out-cell-type setting. These findings support source-initialized transport as a useful formulation for distribution-level alignment, while fine-grained feature recovery remains challenging.

1 INTRODUCTION

Single-cell multi-omics technologies provide complementary views of cellular state through measurements such as gene expression and chromatin accessibility. Because profiling every modality for every cell remains challenging, cross-modality translation offers a way to estimate an unmeasured modality from an observed one. RNA–ATAC translation is particularly difficult because RNA measurements are continuous and overdispersed, whereas ATAC profiles are sparse and approximately binary. Paired measurements nevertheless provide an important structural constraint: the source and target observations correspond to the same cell.

Existing approaches formulate this task through direct prediction, latent-variable modeling, or conditional generation. Diffusion and flow-based methods commonly generate the target from a simple prior while using the source modality as contextual information. This formulation is natural for conditional generation, but it does not use the paired source observation as the starting state of the trajectory. The paired setting therefore raises a specific question: should transport begin from an unrelated noise state or from the encoded source cell?

We study the latter formulation. For a paired observation (x_S, x_T), modality-specific encoders produce latent representations (z_S, z_T), and we define the transport endpoints as $z(0) = z_S$ and $z(1) = z_T$. The source representation thus participates directly in the trajectory rather than serving only as contextual information. In the controlled noise-initialized comparison, the model receives the same source information as conditioning but replaces the initial source latent with Gaussian noise. This design allows us to examine source initialization separately from the availability of source information. Our aim is not to introduce a new optimal-transport solver or probability path, but to investigate source-initialized flow matching for paired RNA–ATAC translation.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Based on this formulation, we propose **TransFlow**. Modality-specific block-wise variational autoencoders map RNA and ATAC profiles into latent representations, and a shared interval-conditioned velocity network models RNA→ATAC and ATAC→RNA transport. The shared network provides a common mechanism for both directions, while modality-specific decoders map transported representations back to their feature spaces. The block-wise representations organize high-dimensional gene and peak features and support block-level cross-modal analysis.

We evaluate TransFlow using population-level distributional metrics, feature-level criteria, and label-transfer scores. In multiple-run experiments on BMMC and PBMC10k, TransFlow obtains the lowest distributional scores in both directions in the main comparison; a single-run MiniAtlas experiment shows the same trend. Feature-level and label-transfer results are more dependent on the dataset and translation direction. The distributional advantage is generally maintained across PCA dimensions, although the W_2 ranking changes for BMMC ATAC→RNA at higher dimensions. Controlled comparisons with noise-initialized flow matching and direct latent regression support the roles of source initialization and continuous transport. In the evaluated BMMC held-out-cell-type setting without cell-type conditioning, TransFlow obtains the best results across the reported metrics. These results suggest that source initialization is most helpful for distribution-level alignment, while fine-grained feature recovery remains less uniform.

Our contributions can be summarized as follows:

- We formulate paired RNA–ATAC translation as continuous transport from the observed source latent state to its paired target state, distinguishing source initialization from noise-initialized conditional generation.
- We develop TransFlow with modality-specific block-wise representations and a shared interval-conditioned velocity network for bidirectional translation.
- We evaluate distributional alignment, feature-level correspondence, and label consistency across multiple datasets, including PCA-dimension sensitivity, a held-out-cell-type setting, and controlled comparisons of transport initialization and continuous flow integration.

2 Related Work

Latent-variable models for multi-omics integration. A large body of work approaches single-cell multi-omics modeling through shared latent representations. Methods such as MultiVI Ashuach et al. (2023), scMM Minoura et al. (2021), and MIDAS He et al. (2024) extend variational autoencoder frameworks to jointly model multiple modalities, enabling integration, imputation, and missing-modality inference. Translation-focused approaches, including BABEL Wu et al. (2021), Polarbear Zhang et al. (2021), and scButterfly Cao et al. (2024), further leverage paired data to learn mappings between RNA and ATAC profiles. These methods have demonstrated strong empirical performance and remain important baselines for cross-modality prediction.

Alternative modeling strategies for cross-modal learning. Beyond latent-variable approaches, several methods explore alternative inductive biases for modeling relationships between modalities. For example, UnitedNet Tang et al. (2023) formulates multimodal prediction as a multi-task learning problem, uniPort Cao et al. (2022) integrates optimal transport with variational modeling, and scMoGNN Wen et al. (2022) uses graph neural networks to capture feature-level dependencies. Matrix factorization approaches such as scMoMaT Zhang et al. (2023) provide another complementary perspective. These works highlight the diversity of modeling strategies for multi-omics data, though they are not typically framed as explicit transport between observed modality distributions.

Diffusion and flow-based conditional generation. Recent work has introduced diffusion and flow-based objectives into single-cell modeling. Methods such as scDiffusion-X Luo et al. (2025) adopt latent diffusion models for conditional generation and translation, while CFGen Palma et al. (2025) applies flow matching with conditional guidance. In these approaches, generation typically proceeds from a simple prior distribution, with the conditioning signal guiding the generation process.

Our work is most closely related to these generative approaches, but differs in formulation. Instead of starting from noise, we consider a setting where paired observations are available and directly model transport between modality distributions using the observed source as the starting point. This

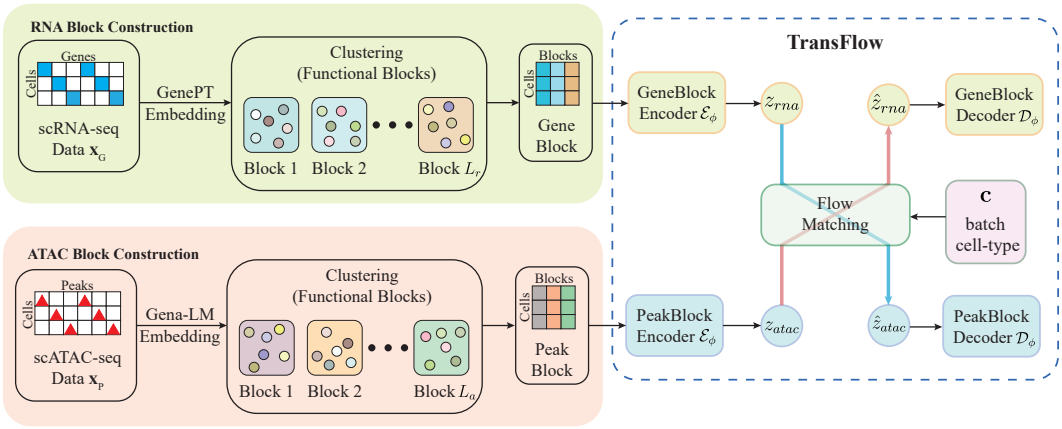

Figure 1: **Overview of the TransFlow framework.** The model performs bidirectional cross-modality translation through a flow-based formulation. **(Left)** High-dimensional scRNA-seq and scATAC-seq inputs are organized into **GeneBlocks** and **PeakBlocks** using GenePT Chen & Zou (2024) and Gena-LM Fishman et al. (2025) embeddings, respectively. **(Right)** The core of TransFlow is a shared Flow Matching module operating in the latent space. Encoders $\mathcal{E}_\phi$ project the feature blocks into latent representations. Guided by cell-type and batch conditions $\mathbf{c}$, the model learns a continuous vector field between modalities, enabling direct cross-modal translation in both directions.

perspective is complementary to existing approaches and is particularly suited to paired cross-modality translation tasks.

Flow matching and transport-based modeling. Flow Matching Liu et al. (2022) provides a general framework for learning continuous transformations between distributions, and has been extended in various directions, including bidirectional formulations such as BiFM Dai et al. (2026). Related ideas also appear in optimal transport and recent applications such as CellFlow. Our work builds on these developments, focusing on how transport-based formulations can be instantiated in paired multi-omics translation with structured representations, rather than proposing a new transport algorithm itself.

3 Method

3.1 Problem Setting and Modeling Tasks

We consider paired single-cell multi-omics measurements collected from the same cell population, forming a dataset $\mathcal{D} = \{(x_i^{\mathrm{RNA}}, x_i^{\mathrm{ATAC}}, \mathbf{c}_i)\}_{i=1}^N$, where x^{RNA} and x^{ATAC} denote transcriptomic and chromatin accessibility profiles, respectively, and $\mathbf{c}_i$ represents auxiliary biological or technical conditions such as cell type or batch.

Our primary goal is to model **conditional cross-modality translation** in both directions:

$$p_\theta(x^{\mathrm{ATAC}} \mid x^{\mathrm{RNA}}, \mathbf{c}), \quad p_\theta(x^{\mathrm{RNA}} \mid x^{\mathrm{ATAC}}, \mathbf{c}). \tag{1}$$

Unlike standard conditional generation, where samples are generated from a fixed prior distribution, this setting provides *paired observations* from two structured domains. This allows us to consider formulations in which the observed source modality directly defines the starting point of generation, rather than serving only as an auxiliary condition.

In addition to cross-modality translation, we also evaluate conditional single-modality generation $p_\theta(x^{\mathrm{RNA}} \mid \mathbf{c})$ and $p_\theta(x^{\mathrm{ATAC}} \mid \mathbf{c})$ as auxiliary tasks. However, the main focus of our method is on paired cross-modality transport.

3.2 Distribution-to-Distribution Transport Formulation

We formalize cross-modality translation as a conditional transport problem between structured latent distributions. Let $m_s, m_t \in \{\mathrm{RNA}, \mathrm{ATAC}\}$ denote the source and target modalities with $m_s \neq m_t$. Given paired observations, modality-specific encoders map inputs to latent representations z^{m_s} and z^{m_t}.

A key design choice in our formulation is to treat the encoded source modality z^{m_s} as the *initial state* of the generative process, and the encoded target modality z^{m_t} as the *terminal state*. This differs from standard noise-based formulations, where the initial state is sampled from an uninformative prior.

Following the flow matching framework, we define a family of intermediate states along a continuous path between the paired endpoints:

$$z_t^{m_s \to m_t} = (1-t)z^{m_s} + tz^{m_t}, \quad t \in [0,1]. \tag{2}$$

The corresponding target velocity field is defined as:

$$u^{m_s \to m_t} = z^{m_t} - z^{m_s}. \tag{3}$$

While this linear interpolation follows standard flow matching practice, its role here is to enable supervised learning of a *conditional transport field* between paired modalities. The key distinction from noise-to-data generation lies in the endpoints: both z^{m_s} and z^{m_t} are structured and paired, allowing the model to learn how samples move between modality-specific manifolds rather than from noise to data.

This formulation naturally supports bidirectional translation by swapping the roles of (m_s, m_t), and provides a consistent framework for incorporating paired supervision along the entire transport path rather than only at the final output.

3.3 TransFlow Architecture

TransFlow instantiates the above transport formulation with a modular architecture tailored to the structural heterogeneity of single-cell multi-omics data. As illustrated in Figure 1, the model consists of two key components: (i) modality-specific block-wise encoders and decoders, and (ii) a shared bidirectional conditional flow network.

3.3.1 Block-wise Latent Representation

Single-cell RNA and ATAC profiles are characterized by extreme dimensionality and heterogeneous feature semantics. To address this, TransFlow draws inspiration from the SAVE framework and adopts its modular philosophy, generalizing it to the chromatin modality, establishing a unified, block-wise representation for multi-omics data.

RNA Block Construction. For the transcriptomic modality, adhering to the SAVE paradigm, we utilize **GenePT** Chen & Zou (2024) to generate high-dimensional semantic embeddings for genes based on their textual descriptions and pathway annotations. These genes are subsequently partitioned into L_r functional blocks via clustering, ensuring that each block encapsulates a distinct gene module with shared functional semantics. This clustering is fixed during training. The encoder processes genes block-wise, mapping each gene block into a latent token of dimension D, yielding a sequence-structured representation $z^{\mathrm{RNA}} = [z_1^{\mathrm{RNA}}, \ldots, z_{L_r}^{\mathrm{RNA}}]$.

ATAC Block Construction. For chromatin accessibility, we adopt a sequence-centric strategy. Each ATAC peak is mapped to its underlying DNA sequence, and we extract sequence embeddings using **Gena-LM** Fishman et al. (2025), a pre-trained DNA language model. Peaks are clustered in this sequence embedding space to form L_a chromatin blocks. The ATAC encoder maps each chromatin block into a latent token, producing $z^{\mathrm{ATAC}} = [z_1^{\mathrm{ATAC}}, \ldots, z_{L_a}^{\mathrm{ATAC}}]$.

This block-wise structure reduces effective dimensionality while preserving biological modularity and providing shared-dimensional latent representations for the downstream flow.

3.3.2 CONDITIONAL FLOW NETWORK

To disentangle biological heterogeneity from technical variation, we explicitly condition the flow on auxiliary attributes. Let b denote the batch index and y denote the cell type annotation. We project these discrete indicators into learnable embedding vectors $\mathbf{e}_{batch}$ and $\mathbf{e}_{type}$, respectively. These are concatenated to form a context vector $\mathbf{c} = [\mathbf{e}_{batch}, \mathbf{e}_{type}]$. This condition $\mathbf{c}$ is injected into the velocity network via adaptive layer normalization, allowing the model to capture systematic variation across biological and technical contexts without requiring separate models for each condition.

3.3.3 SHARED BIDIRECTIONAL FLOW MATCHING

To implement the transport formulation described above, we adopt a bidirectional flow matching design based on BiFM Dai et al. (2026). Rather than training two independent single-direction flow models, TransFlow realizes RNA→ATAC and ATAC→RNA translation with a single shared velocity network. The RNA and ATAC VAEs first produce latent token sequences z^{RNA} and z^{ATAC}.

The denoising network is shared across both translation directions. Following BiFM, it receives the current latent state, the current time t, the paired interval endpoint t', and the biological condition embedding.

Concretely, for an interval $0 \le t_s < t_e \le 1$ on the paired latent path, TransFlow constructs two interval states and evaluates the same network in opposite orders:

$$\begin{aligned} z_{t_s} &= (1-t_s)z^{\mathrm{RNA}} + t_s z^{\mathrm{ATAC}}, \qquad z_{t_e} = (1-t_e)z^{\mathrm{RNA}} + t_e z^{\mathrm{ATAC}}, \\ \hat{u}_{\mathrm{R}\to\mathrm{A}} &= v_\theta(z_{t_s}, t_s, t_e, \mathbf{c}), \qquad \hat{u}_{\mathrm{A}\to\mathrm{R}} = v_\theta(z_{t_e}, t_e, t_s, \mathbf{c}). \end{aligned} \tag{4}$$

Here the same condition embedding $\mathbf{c}$ is used for both directions; the ordered time pair (t, t') determines the translation orientation.

3.4 OPTIMIZATION

TransFlow is trained using a two-stage strategy that decouples representation learning from transport modeling.

Stage I: Modality-Specific Representation Learning. We independently train block-wise VAEs for RNA and ATAC modalities. Each VAE is optimized with a modality-specific reconstruction term and a shared Gaussian latent prior:

$$\begin{aligned} \mathcal{L}^m_{\mathrm{VAE}} &= \lambda_{\mathrm{rec}}\mathcal{L}^m_{\mathrm{rec}} + \lambda_{\mathrm{KL}}\mathcal{L}^m_{\mathrm{KL}}, \quad m \in \{\mathrm{RNA}, \mathrm{ATAC}\}, \\ \mathcal{L}^{\mathrm{RNA}}_{\mathrm{rec}} &= \mathcal{L}_{\mathrm{ZINB}}\left(x^{\mathrm{RNA}}; \boldsymbol{\mu}, \boldsymbol{\theta}, \boldsymbol{\pi}\right), \\ \mathcal{L}^{\mathrm{ATAC}}_{\mathrm{rec}} &= \mathcal{L}_{\mathrm{BCE}}\left(\hat{x}^{\mathrm{ATAC}}, x^{\mathrm{ATAC}}\right). \end{aligned} \tag{5}$$

For RNA, the decoder predicts a nonnegative output $\hat{x}^{\mathrm{RNA}}$ together with ZINB parameters $(\boldsymbol{\mu}, \boldsymbol{\theta}, \boldsymbol{\pi})$. The former is used for reconstruction, generation, and cross-modality translation, while the latter parameterizes a count-space likelihood:

$$\mathcal{L}_{\mathrm{ZINB}}(x; \boldsymbol{\mu}, \boldsymbol{\theta}, \boldsymbol{\pi}) = -\sum_j \log\left[\pi_j \mathbb{I}(x_j = 0) + (1-\pi_j)\mathrm{NB}(x_j; \mu_j, \theta_j)\right]. \tag{6}$$

For ATAC, binary cross-entropy remains appropriate because the chromatin profiles are binarized. After pretraining, encoder/decoder parameters are frozen.

Stage II: Bidirectional Conditional Flow Optimization. After VAE pretraining, the modality-specific encoders provide paired latent sequences $(z^{\mathrm{RNA}}, z^{\mathrm{ATAC}})$ for flow learning. We sample the intermediate states as defined in Eq. 4 to calculate the corresponding interval-normalized velocity fields:

$$u_{\mathrm{R}\to\mathrm{A}} = \frac{z_{t_e} - z_{t_s}}{t_e - t_s}, \qquad u_{\mathrm{A}\to\mathrm{R}} = \frac{z_{t_s} - z_{t_e}}{t_e - t_s}. \tag{7}$$

The BiFM loss consists of three terms: two directional translation losses and the consistency loss from the original BiFM formulation Dai et al. (2026):

$$\begin{aligned} \mathcal{L}_{\mathrm{R}\to\mathrm{A}} &= \|\hat{u}_{\mathrm{R}\to\mathrm{A}} - u_{\mathrm{R}\to\mathrm{A}}\|_2^2, \\ \mathcal{L}_{\mathrm{A}\to\mathrm{R}} &= \|\hat{u}_{\mathrm{A}\to\mathrm{R}} - u_{\mathrm{A}\to\mathrm{R}}\|_2^2, \\ \mathcal{L}_{\mathrm{con}} &= \|\hat{u}_{\mathrm{R}\to\mathrm{A}} + \hat{u}_{\mathrm{A}\to\mathrm{R}}\|_2^2, \\ \mathcal{L}_{\mathrm{BiFM}} &= \mathcal{L}_{\mathrm{R}\to\mathrm{A}} + \mathcal{L}_{\mathrm{A}\to\mathrm{R}} + \lambda_{\mathrm{con}}\mathcal{L}_{\mathrm{con}}. \end{aligned} \tag{8}$$

The first two terms regress the interval velocities for RNA→ATAC and ATAC→RNA, respectively. The consistency term enforces that the predicted forward and reverse directions on the same latent segment are negatives of each other.

Cross-Modality Inference. Given an observed source profile x^{m_s}, the corresponding modality-specific encoder first produces a source latent representation z^{m_s}. Starting from this latent state, we numerically integrate the shared velocity network over a predefined time grid in the direction associated with $m_s \to m_t$. The terminal latent representation is then passed to the target-modality decoder to produce the translated feature profile $\hat{x}^{m_t}$. No target-modality measurement is used during inference. The complete interval-Euler procedure is provided in Algorithm 2.

4 Experiments

We evaluate TransFlow on paired RNA–ATAC datasets to study conditional cross-modality translation under severe modality mismatch. Rather than focusing on unconditional generation, our experiments emphasize how well a model can transform one structured modality into another under paired supervision. This setting provides a controlled testbed for analyzing distribution-level alignment, feature-level fidelity, and label consistency.

4.1 Experimental Setup

Datasets. We evaluate our proposed model on three benchmark paired multi-omics datasets: **OpenProblems BMMC**, **PBMC10k** 10x Genomics (2021), and **MiniAtlas** Wu et al. (2025). The OpenProblems BMMC dataset, sourced from the NeurIPS 2021 Multimodal Integration Challenge Lance et al. (2022), includes both cell type and batch annotations; it serves as our primary testbed for assessing multi-conditional modeling. The PBMC10k dataset, containing only cell type labels, functions as a complementary benchmark for examining model behavior under a fixed technical context. Finally, we incorporate a subset of MiniAtlas as a supplementary benchmark containing a broader range of tissues and cell types. We use it to evaluate translation performance under greater biological heterogeneity rather than as a separate out-of-distribution test. For BMMC and PBMC10k, all standard and supplementary evaluations use the fixed train/test splits described in Appendix D.2.

These datasets cover a range of biological complexity and heterogeneity, allowing us to examine model behavior under different levels of variation. Detailed descriptions, preprocessing protocols, and data splits are provided in Appendix D.1.

Baselines. We position TransFlow against representative methods spanning distinct modeling paradigms for single-cell multi-omics translation. Within joint latent-variable models, MultiVI Ashuach et al. (2023) provides a probabilistic VAE framework for multimodal integration with missing modalities, while BABEL Wu et al. (2021) and scButterfly Cao et al. (2024) directly address cross-modality prediction through modality-specific encoders and decoders. We also consider graph-based prediction through scMoGNN Wen et al. (2022), diffusion-based conditional generation and translation through scDiffusion-X Luo et al. (2025), and optimal-transport GENOT Klein et al. (2024). These baselines are representative of different modeling paradigms rather than exhaustive, and provide a broad comparison across latent-variable, graph-based, diffusion, and transport-based approaches.

Conditioning Settings. In the standard translation experiments, TransFlow uses the available cell-type and batch metadata as auxiliary conditions. We therefore interpret label-transfer accuracy in

this setting as measuring whether the translated profiles remain consistent with the supplied biological condition, rather than as evidence that cell identity is inferred solely from the source modality. In the held-out-cell-type experiment, cell-type labels are not supplied to TransFlow as model conditions, allowing us to evaluate transfer based on the observed source modality without explicit cell-type input.

Evaluation Metrics. We evaluate translation quality using class-macro distributional, feature-level, and label-transfer metrics. For distributional alignment, the standard benchmark reports class-macro W_2 and class-macro linear-time RBF-MMD^2 in an outer-train-fitted PCA representation, using the first 50 principal components for the main comparison. For feature-level fidelity, we report row-wise Pearson correlation for ATAC→RNA translation and global AUPRC for RNA→ATAC translation. For label consistency, we report label-transfer accuracy and the signed accuracy gap Δ Acc, for which lower values are preferred. Detailed definitions and computation procedures are provided in Appendix D.7.

4.2 Bidirectional Cross-Modality Translation

Performance summaries for the standard benchmark are reported in Table 1. The main comparison uses the 50-PC entries from the outer-train-fitted class-macro evaluation; the full PCA sensitivity analysis and expanded label-transfer scores are provided in Appendix H. Additional qualitative visualizations and unimodal generation results are provided in Appendices F and E, respectively.

Quantitative Cross-Modal Translation. As shown in Table 1, TransFlow achieves the lowest distributional scores across the BMMC and PBMC10k comparisons, while feature-level and label-transfer results remain metric dependent. We also evaluate all methods on MiniAtlas as an additional benchmark. For BMMC and PBMC10k, we repeat the experiments across multiple independent runs and report the mean and sample standard deviation; detailed multiple-run results are provided in Appendix H. A training-set visualization of decoded transport states is provided in Appendix 4.

In the standard metadata-conditioned setting, the label-transfer results indicate that the translated profiles are consistent with the supplied biological conditions. Because cell-type metadata are available to TransFlow in this setting, we treat these values as conditional-consistency measures. The held-out-cell-type experiment in Section 4.3 provides a stricter evaluation without explicit cell-type inputs. We retain the signed Δ Acc convention, with lower values preferred.

Table 1: **Bidirectional cross-modality translation performance.** We report class-macro W_2 and class-macro linear-time RBF-MMD^2 together with feature-level fidelity and label-transfer metrics. Lower distributional metrics and signed Δ Acc are preferred, whereas higher feature and accuracy metrics are preferred. Best results are bolded.

Model	ATAC → RNA					RNA → ATAC				
	$W_2\downarrow$	RBF-MMD$^2\downarrow$	Row Cor. ↑	Acc ↑	Δ Acc ↓	$W_2\downarrow$	RBF-MMD$^2\downarrow$	AUPRC ↑	Acc ↑	Δ Acc ↓
(a) BMMC Dataset										
BABEL	15.3628 ± 0.2816	0.1163 ± 0.0078	0.4625 ± 0.0082	0.7592 ± 0.0077	0.1122 ± 0.0077	12.9377 ± 0.0695	0.1991 ± 0.0013	0.1558 ± 0.0030	0.6510 ± 0.0074	0.1873 ± 0.0074
scMoGNN	23.6424 ± 0.0068	0.3411 ± 0.0001	0.4236 ± 0.0001	0.1674 ± 0.0000	0.7040 ± 0.0000	14.5017 ± 0.1983	0.3094 ± 0.0058	0.1963 ± 0.0008	0.1631 ± 0.0093	0.6751 ± 0.0093
scDiffusion-X	35.6264 ± 0.0808	0.5998 ± 0.0045	0.4503 ± 0.0005	0.6551 ± 0.0043	0.2162 ± 0.0043	9.7800 ± 0.1395	0.1018 ± 0.0024	0.1972 ± 0.0005	0.6877 ± 0.0120	0.1506 ± 0.0120
GENOT	26.0312 ± 0.3022	0.1772 ± 0.0023	0.3547 ± 0.0064	0.1297 ± 0.0049	0.7417 ± 0.0049	15.1771 ± 0.1320	0.1674 ± 0.0063	0.1419 ± 0.0015	0.1100 ± 0.0031	0.7282 ± 0.0031
scButterfly	11.7498 ± 0.3434	0.0309 ± 0.0018	**0.5038 ± 0.0014**	0.7921 ± 0.0146	0.0792 ± 0.0146	7.5469 ± 0.1864	0.0576 ± 0.0060	**0.2321 ± 0.0005**	0.8205 ± 0.0019	0.0178 ± 0.0019
MultiVI	31.8923 ± 0.1221	0.5125 ± 0.0035	0.4726 ± 0.0004	0.7463 ± 0.0061	0.1251 ± 0.0061	54.0841 ± 0.2142	1.0269 ± 0.0030	0.2205 ± 0.0005	0.8299 ± 0.0019	0.0083 ± 0.0019
TransFlow	**11.2189 ± 0.0061**	**0.0095 ± 0.0003**	0.4225 ± 0.0000	**0.9248 ± 0.0018**	**-0.0534 ± 0.0018**	**6.0772 ± 0.1215**	**0.0274 ± 0.0040**	0.1829 ± 0.0015	**0.8385 ± 0.0037**	**-0.0002 ± 0.0037**
(b) PBMC10k Dataset										
BABEL	12.9113 ± 0.3699	0.1708 ± 0.0126	0.5819 ± 0.0073	0.8285 ± 0.0454	0.1085 ± 0.0454	16.6523 ± 0.7766	0.2440 ± 0.0230	0.3640 ± 0.0030	0.8565 ± 0.0105	0.0540 ± 0.0105
scMoGNN	20.6909 ± 0.0078	0.4555 ± 0.0004	0.6066 ± 0.0001	0.1073 ± 0.0566	0.8297 ± 0.0566	20.3993 ± 0.0210	0.3546 ± 0.0014	0.4066 ± 0.0002	0.1447 ± 0.0014	0.7658 ± 0.0014
scDiffusion-X	24.0286 ± 0.0344	0.5399 ± 0.0026	0.6172 ± 0.0014	0.7322 ± 0.0140	0.2048 ± 0.0140	12.7300 ± 0.3638	0.0955 ± 0.0129	0.4231 ± 0.0015	0.7227 ± 0.0120	0.1878 ± 0.0120
GENOT	17.4633 ± 0.4999	0.1407 ± 0.0098	0.6203 ± 0.0025	0.2837 ± 0.0114	0.6533 ± 0.0114	17.7758 ± 0.6423	0.1677 ± 0.0046	0.3944 ± 0.0049	0.2767 ± 0.0076	0.6338 ± 0.0076
scButterfly	16.3459 ± 0.5211	0.2494 ± 0.0158	0.6128 ± 0.0021	0.5087 ± 0.0360	0.4283 ± 0.0360	24.3341 ± 0.3040	0.5224 ± 0.0070	0.4321 ± 0.0027	0.4267 ± 0.0023	0.4838 ± 0.0023
MultiVI	20.0813 ± 0.2316	0.4356 ± 0.0091	0.6313 ± 0.0004	0.7115 ± 0.0253	0.2255 ± 0.0253	46.0057 ± 0.0393	0.9884 ± 0.0018	**0.4485 ± 0.0015**	0.7562 ± 0.0184	0.1543 ± 0.0184
TransFlow	**9.2378 ± 0.0823**	**0.0507 ± 0.0014**	**0.6333 ± 0.0006**	**0.9833 ± 0.0015**	**-0.0463 ± 0.0015**	**8.0183 ± 0.0782**	**0.0361 ± 0.0093**	0.4135 ± 0.0029	**0.8822 ± 0.0038**	**0.0283 ± 0.0038**
(c) MiniAtlas Dataset										
BABEL	15.286	1.640	0.581	0.777	0.124	25.401	1.773	0.263	0.666	0.203
scMoGNN	24.972	4.135	0.427	0.088	0.814	23.920	2.078	0.272	0.086	0.783
scDiffusion-X	22.354	1.231	0.188	0.068	0.809	20.832	1.385	0.087	0.056	0.813
GENOT	32.491	1.597	0.284	0.055	0.847	29.280	1.536	0.176	0.045	0.824
scButterfly	10.015	1.240	**0.640**	0.825	0.076	16.617	1.285	0.359	0.774	0.095
MultiVI	41.008	2.153	0.516	0.462	0.440	46.018	1.996	0.359	0.849	0.020
TransFlow (Ours)	**5.971**	**0.952**	0.586	**0.933**	**-0.032**	**9.754**	**1.003**	**0.367**	**0.905**	**-0.036**

Exploratory Block-Level Analysis. We use the block-wise representation to inspect associations between RNA and ATAC blocks. Figure 2(a) shows heterogeneous association strengths across block pairs. We select one high-scoring block pair and report motif enrichment for the peaks assigned to its ATAC block in Figure 2(b), where ETS-family motifs appear among the top-ranked results. This

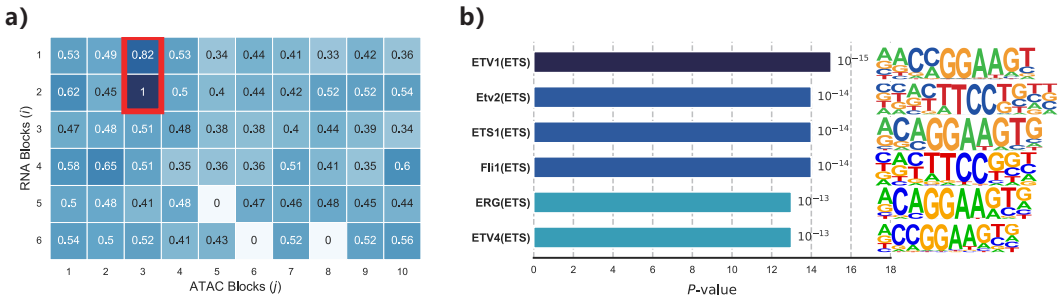

Figure 2: **Exploratory block-level association and motif enrichment.** **(a)** Association scores between transcriptomic (rows) and chromatin accessibility (columns) blocks. The highlighted entry denotes the selected block pair used for subsequent analysis. **(b)** Motif-enrichment results for peaks assigned to the selected ATAC block, with ETS-family motifs among the top-ranked results. This analysis illustrates a possible use of the block-wise representation and is not intended as mechanistic validation.

analysis illustrates how the representation can support block-level inspection, but it does not establish causal or mechanistically validated regulatory relationships.

4.3 Generalization to Held-Out Cell Types

To evaluate whether the learned mapping transfers beyond cell types observed during training, we construct a held-out-cell-type experiment on the BMMC dataset. Cell-type labels are not supplied to TransFlow as auxiliary model conditions in this experiment. Translation is therefore based on the observed source modality without explicit cell-type input. We compare TransFlow with the baselines under the same held-out split.

Held-Out Cell-Type Generalization. As summarized in Table 2, TransFlow obtains the best results among the compared methods across the reported distributional, feature-level, and label-transfer metrics. These results provide evidence that the learned source-to-target mapping transfers to the held-out populations in this BMMC split. Because the evaluation is limited to one dataset and one held-out protocol, we interpret it as evidence of held-out-cell-type transfer rather than broad out-of-distribution robustness.

Table 2: **Held-out-cell-type translation performance on BMMC.** We report distributional alignment (W_2, RBF-MMD^2), feature-level fidelity (Row Cor. and AUPRC), and label-transfer results (Acc and the signed accuracy gap Δ Acc). Lower Δ Acc is preferred. Best results are bolded.

Model	ATAC → RNA					RNA → ATAC				
	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	Row Cor. $\uparrow$	Acc $\uparrow$	Δ Acc $\downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	AUPRC $\uparrow$	Acc $\uparrow$	Δ Acc $\downarrow$
BABEL	20.703	2.455	0.451	0.405	0.451	18.854	2.449	0.155	0.382	0.411
scMoGNN	24.689	4.291	0.396	0.094	0.762	16.456	2.687	0.185	0.099	0.694
GENOT	30.106	2.193	0.321	0.076	0.780	19.716	2.107	0.107	0.062	0.731
scButterfly	18.610	2.339	0.500	0.450	0.406	12.417	1.955	0.229	0.458	0.335
MultiVI	35.522	2.666	0.463	0.413	0.443	54.646	3.016	0.205	0.450	0.343
TransFlow (Ours)	**12.633**	**1.832**	**0.508**	**0.512**	**0.344**	**11.301**	**1.766**	**0.255**	**0.460**	**0.332**

4.4 Ablation Studies

To better understand the empirical contributions of different components in TransFlow, we conduct controlled comparisons on the BMMC dataset. We compare the full model with (i) a noise-initialized flow-matching variant, (ii) two independent directional flow-matching models, (iii) an unstructured latent representation, and (iv) direct latent regression without continuous flow integration. The last variant trains two independent MLPs for RNA→ATAC and ATAC→RNA using squared-error regression between paired source and target latents.

Source Initialization and Continuous Transport. As shown in Table 3, the *Noise-Initialized FM* variant performs substantially worse than TransFlow across both translation directions. This variant retains the flow-matching architecture and receives the same source representation as conditioning, but replaces the source latent at the initial state with Gaussian noise. The comparison therefore supports the use of source initialization rather than attributing the difference only to source-information availability. The *Direct Latent MLP* variant also shows a substantial performance decrease. This variant uses two independent deterministic MLPs to map the pretrained source latent directly to its paired target latent. Within the evaluated architecture, the comparison suggests that learning the mapping through a continuous velocity field provides an additional empirical benefit over direct latent regression.

Shared Bidirectional Modeling. The *Independent Directional FMs* variant produces higher W_2 and RBF-MMD^2 and lower RNA→ATAC AUPRC than the shared model, but achieves higher label-transfer accuracy and lower signed accuracy gaps. The shared formulation is therefore not uniformly superior across all metrics. Instead, parameter sharing favors distributional alignment and peak-level recovery in this experiment, whereas the independent models retain stronger class-discriminative signals under the label-transfer classifier.

Effect of the Block-wise Structure. The *Unstructured Latent* variant performs similarly to the full model. It obtains marginally lower W_2 in both directions and the same RNA row-wise correlation, whereas the block-wise model yields slightly better RBF-MMD^2, AUPRC, and label-transfer accuracy. These differences are small and mixed across metrics. We therefore do not interpret the block-wise structure as the primary source of the quantitative improvements; its main role is to provide a structured representation for organizing features and conducting block-level analyses.

Table 3: **Ablation studies on BMMC.** We compare TransFlow with an unstructured latent representation, independent directional flow models, noise-initialized flow matching, and two independent direct latent MLPs. We report distributional alignment (W_2, RBF-MMD^2), feature-level fidelity (Row Cor. and AUPRC), and label-transfer results (Acc and the signed accuracy gap Δ Acc). Lower Δ Acc is preferred. Best results are bolded.

Model Variant	ATAC → RNA					RNA → ATAC				
	$W_2 \downarrow$	RBF-$\mathrm{MMD}^2 \downarrow$	Row Cor. ↑	Acc ↑	Δ Acc ↓	$W_2 \downarrow$	RBF-$\mathrm{MMD}^2 \downarrow$	AUPRC ↑	Acc ↑	Δ Acc ↓
TransFlow	9.170	**1.026**	**0.427**	0.920	-0.044	7.345	**0.981**	**0.245**	0.861	-0.021
Unstructured Latent	**9.130**	1.027	**0.427**	0.918	-0.042	**7.098**	0.982	0.243	0.858	-0.018
Independent Directional FMs	12.651	1.293	0.424	**0.955**	**-0.078**	10.334	1.188	0.207	**0.925**	**-0.084**
Noise-Initialized FM	25.334	1.476	0.366	0.517	0.358	16.528	1.261	0.144	0.602	0.237
Direct Latent MLP	35.960	4.719	0.325	0.071	0.805	20.295	4.376	0.130	0.015	0.824

5 Discussion

We introduced TransFlow as a source-initialized flow-matching formulation for paired RNA–ATAC translation. Across the evaluated benchmarks, the method shows its clearest advantage in population-level distributional alignment, whereas improvements in paired feature recovery are less uniform. The controlled comparisons indicate that source initialization and continuous flow-based transport contribute more strongly to the observed differences than the block-wise representation, whose quantitative effect is small and mixed across metrics.

Several limitations remain. Cell-type-unconditioned translation is evaluated only in the BMMC held-out-cell-type experiment. The independently pretrained and frozen VAEs may limit fine-grained feature recovery, particularly for sparse ATAC profiles. Finally, the block-wise representation yields mixed quantitative differences, and the motif analysis should be interpreted as an exploratory case study rather than mechanistic biological validation.

AI use statement

In this work, we used generative AI tools to assist with drafting early versions of the manuscript, revising its wording, modifying code, and running code. The authors reviewed all AI-assisted text

and code changes and take responsibility for the final content of this work, including any text, claims, or artifacts produced with the aid of generative AI.

Reproducibility statement

To support reproducibility, we provide an anonymous implementation linked in Appendix A. The appendix documents the public datasets and existing assets used in this study, including their sources and available license information (Appendix A.1); dataset preprocessing and train/test splits (Appendix D.2); model architecture, optimization settings, computing resources, and baseline configurations (Appendices C, D.3, and D.5); and the training, sampling, and evaluation procedures (Algorithms 1–2 and Appendix D.7). We report means and sample standard deviations across multiple independent runs for BMMC and PBMC10k, with expanded PCA-dimension and label-transfer results in Appendix H; the MiniAtlas results are reported as a single-run supplementary experiment.

Ethics statement

This study uses publicly available single-cell multi-omics datasets and does not involve new data collection. Data sources and available license information are provided in Appendix A. TransFlow outputs are computational predictions that may reflect errors or biases in the training data and should be validated experimentally before supporting biological or clinical conclusions. Additional discussion is provided in Appendix B.

References

10x Genomics. PBMC from a Healthy Donor - Granulocytes Removed Through Cell Sorting (10k), Epi Multiome Dataset Analyzed Using Cell Ranger ARC 2.0.0. 10x Genomics, 2021. Published May 3, 2021. Available at: https://www.10xgenomics.com/datasets/pbmc-from-a-healthy-donor-granulocytes-removed-through-cell-sorting-10-k-1-standar

Tal Ashuach, Mariano I. Gabitto, Rohan V. Koodli, Giuseppe-Antonio Saldi, Michael I. Jordan, and Nir Yosef. MultiVI: Deep generative model for the integration of multimodal data. *Nature Methods*, 20(8):1222–1231, August 2023. ISSN 1548-7091, 1548-7105. doi: 10.1038/s41592-023-01909-9.

Kai Cao, Qiyu Gong, Yiguang Hong, and Lin Wan. A unified computational framework for single-cell data integration with optimal transport. *Nature Communications*, 13(1):7419, December 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-35094-8.

Yichuan Cao, Xiamiao Zhao, Songming Tang, Qun Jiang, Sijie Li, Siyu Li, and Shengquan Chen. scButterfly: A versatile single-cell cross-modality translation method via dual-aligned variational autoencoders. *Nature Communications*, 15(1):2973, April 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-47418-x.

Yiqun Chen and James Zou. GenePT: A Simple But Effective Foundation Model for Genes and Cells Built From ChatGPT, March 2024.

Yasong Dai, Zeeshan Hayder, David Ahmedt-Aristizabal, and Hongdong Li. Bifm: Bidirectional flow matching for few-step image editing and generation. *arXiv preprint arXiv:2603.24942*, 2026. doi: 10.48550/arXiv.2603.24942. URL https://arxiv.org/abs/2603.24942. Accepted in CVPR 2026.

Jiayuan Ding, Renming Liu, Hongzhi Wen, Wenzhuo Tang, Zhaoheng Li, Julian Venegas, Runze Su, Dylan Molho, Wei Jin, Yixin Wang, et al. Dance: A deep learning library and benchmark platform for single-cell analysis. *Genome Biology*, 25(1):1–28, 2024.

Veniamin Fishman, Yuri Kuratov, Aleksei Shmelev, Maxim Petrov, Dmitry Penzar, Denis Shepelin, Nikolay Chekanov, Olga Kardymon, and Mikhail Burtsev. Gena-lm: a family of open-source foundational dna language models for long sequences. *Nucleic Acids Research*, 53(2):gkae1310, 01 2025. ISSN 0305-1048. doi: 10.1093/nar/gkae1310. URL https://doi.org/10.1093/nar/gkae1310.

Adam Gayoso, Romain Lopez, Galen Xing, Pierre Boyeau, Valeh Valiollah Pour Amiri, Justin Hong, Katherine Wu, Michael Jayasuriya, Edouard Mehlman, Maxime Langevin, Yining Liu, Jules Samaran, Gabriel Misrachi, Achille Nazaret, Oscar Clivio, Chenling Xu, Tal Ashuach, Mariano Gabitto, Mohammad Lotfollahi, Valentine Svensson, Eduardo da Veiga Beltrame, Vitalii Kleshchevnikov, Carlos Talavera-López, Lior Pachter, Fabian J. Theis, Aaron Streets, Michael I. Jordan, Jeffrey Regier, and Nir Yosef. A Python library for probabilistic analysis of single-cell omics data. *Nature Biotechnology*, 40(2):163–166, February 2022. ISSN 1546-1696. doi: 10.1038/s41587-021-01206-w.

Zhen He, Shuofeng Hu, Yaowen Chen, Sijing An, Jiahao Zhou, Runyan Liu, Junfeng Shi, Jing Wang, Guohua Dong, Jinhui Shi, Jiaxin Zhao, Le Ou-Yang, Yuan Zhu, Xiaochen Bo, and Xiaomin Ying. Mosaic integration and knowledge transfer of single-cell multimodal data with MIDAS. *Nature Biotechnology*, 42(10):1594–1605, October 2024. ISSN 1087-0156, 1546-1696. doi: 10.1038/s41587-023-02040-y.

Dominik Klein, Théo Uscidda, Fabian J Theis, and marco cuturi. GENOT: Entropic (gromov) wasserstein flow matching with applications to single-cell genomics. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=hjspWd7jvg.

Christopher Lance, Malte D. Luecken, Daniel B. Burkhardt, Robrecht Cannoodt, Pia Rautenstrauch, Anna Laddach, Aidyn Ubingazhibov, Zhi-Jie Cao, Kaiwen Deng, Sumeer Khan, Qiao Liu, Nikolay Russkikh, Gleb Ryazantsev, Uwe Ohler, NeurIPS 2021 Multimodal data integration competition participants, Angela Oliveira Pisco, Jonathan Bloom, Smita Krishnaswamy, and Fabian J. Theis. Multimodal single cell data integration challenge: Results and lessons learned. In Douwe Kiela, Marco Ciccone, and Barbara Caputo (eds.), *Proceedings of the NeurIPS 2021 Competitions and Demonstrations Track*, volume 176 of *Proceedings of Machine Learning Research*, pp. 162–176. PMLR, 06–14 Dec 2022. URL https://proceedings.mlr.press/v176/lance22a.html.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *CoRR*, abs/2209.03003, 2022. doi: 10.48550/ARXIV.2209.03003. URL https://doi.org/10.48550/arXiv.2209.03003.

Erpai Luo, Lei Wei, Minsheng Hao, Xuegong Zhang, and Qiao Liu. Multi-modal diffusion model with dual-cross-attention for multi-omics data generation and translation. *bioRxiv : the preprint server for biology*, 2025. doi: 10.1101/2025.02.27.640020. URL https://www.biorxiv.org/content/early/2025/10/08/2025.02.27.640020.

Kodai Minoura, Ko Abe, Hyunha Nam, Hiroyoshi Nishikawa, and Teppei Shimamura. A mixture-of-experts deep generative model for integrated analysis of single-cell multiomics data. *Cell Reports Methods*, 1(5):100071, September 2021. ISSN 26672375. doi: 10.1016/j.crmeth.2021.100071.

Alessandro Palma, Till Richter, Hanyi Zhang, Manuel Lubetzki, Alexander Tong, Andrea Dittadi, and Fabian J Theis. Multi-modal and multi-attribute generation of single cells with CFGen. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=3MnMGLctKb.

Xin Tang, Jiawei Zhang, Yichun He, Xinhe Zhang, Zuwan Lin, Sebastian Partarrieu, Emma Bou Hanna, Zhaolin Ren, Hao Shen, Yuhong Yang, Xiao Wang, Na Li, Jie Ding, and Jia Liu. Explainable multi-task learning for multi-modality biological data analysis. *Nature Communications*, 14 (1):2546, May 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-37477-x.

Hongzhi Wen, Jiayuan Ding, Wei Jin, Yiqi Wang, Yuying Xie, and Jiliang Tang. Graph Neural Networks for Multimodal Single-Cell Data Integration. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 4153–4163, Washington DC USA, August 2022. ACM. ISBN 978-1-4503-9385-0. doi: 10.1145/3534678.3539213.

Juncheng Wu, Changxin Wan, Zhicheng Ji, Yuyin Zhou, and Wenpin Hou. Epifoundation: A foundation model for single-cell atac-seq via peak-to-gene alignment. *bioRxiv*, 2025. doi: 10.1101/2025.02.05.636688. URL https://www.biorxiv.org/content/early/2025/02/08/2025.02.05.636688.

Kevin E. Wu, Kathryn E. Yost, Howard Y. Chang, and James Zou. BABEL enables cross-modality translation between multiomic profiles at single-cell resolution. *Proceedings of the National Academy of Sciences*, 118(15):e2023070118, April 2021. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.2023070118.

Ran Zhang, Laetitia Meng-Papaxanthos, Jean-Philippe Vert, and William Stafford Noble. Semi-supervised single-cell cross-modality translation using polarbear. *bioRxiv : the preprint server for biology*, 2021. doi: 10.1101/2021.11.18.467517.

Ziqi Zhang, Haoran Sun, Ragunathan Mariappan, Xi Chen, Xinyu Chen, Mika S. Jain, Mirjana Efremova, Sarah A. Teichmann, Vaibhav Rajan, and Xiuwei Zhang. scMoMaT jointly performs single cell mosaic integration and multi-modal bio-marker detection. *Nature Communications*, 14 (1):384, January 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-36066-2.

A SOFTWARE AND DATA

The implementation of the proposed model is available for review. The source code can be accessed at the following anonymous repository:

`https://anonymous.4open.science/r/TransFlow-D2A1/`

All datasets used in this study are publicly available.

A.1 LICENSES AND TERMS FOR EXISTING ASSETS

We used existing public datasets, baseline implementations, software packages, and pretrained embedding resources for benchmarking and feature grouping. The sources and available license or terms-of-use information are listed below. When an exact release, commit, or asset-specific license was not recorded in the current manuscript, we mark it for author confirmation rather than inferring it.

Datasets.

- **OpenProblems BMMC.** We used the BMMC multiome dataset from the NeurIPS 2021 Multimodal Single-Cell Data Integration Challenge Lance et al. (2022). The OpenProblems competition documentation states that its data are licensed under CC BY; a Figshare record for the NeurIPS 2021 dataset also lists CC BY 4.0. Source: `https://openproblems.bio/competitions/neurips_2021`.
- **PBMC10k.** We used the 10x Genomics PBMC multiome dataset, "PBMC from a Healthy Donor – No Cell Sorting (10k)" 10x Genomics (2021), analyzed with Cell Ranger ARC 2.0.0. 10x Genomics lists this dataset under the Creative Commons Attribution 4.0 International (CC BY 4.0) license. Source: `https://www.10xgenomics.com/datasets/pbmc-from-a-healthy-donor-no-cell-sorting-10-k-1-standard-2-0-0`.
- **MiniAtlas.** We used MiniAtlas from EpiFoundation Wu et al. (2025). The Hugging Face dataset card for `UCSC-VLAA/MiniAtlas` lists the license as Apache-2.0. Source: `https://huggingface.co/datasets/UCSC-VLAA/MiniAtlas`.

Baseline implementations and software packages.

- **BABEL and scMoGNN through DANCE.** We used the DANCE Ding et al. (2024) implementations for BABEL Wu et al. (2021) and scMoGNN Wen et al. (2022). The DANCE repository is distributed under the BSD 2-Clause License. Source: `https://github.com/OmicsML/dance`.
- **MultiVI through scvi-tools.** We used the official `MULTIVI` implementation Ashuach et al. (2023) from `scvi-tools` Gayoso et al. (2022). The `scvi-tools` repository is distributed under the BSD 3-Clause License. Source: `https://github.com/scverse/scvi-tools`.
- **GENOT.** We used the official GENOT repository Klein et al. (2024) for bidirectional modality translation. The GENOT repository is distributed under the MIT License. Source: `https://github.com/MUCDK/genot`.
- **scDiffusion-X.** We used the official scDiffusion-X implementation Luo et al. (2025) for diffusion-based generation and translation. The repository is distributed under the MIT License. Source: `https://github.com/EperLuo/scDiffusion-X`.
- **CFGen.** We used the official CFGen implementation Palma et al. (2025) for the unimodal generation benchmark. The repository is distributed under the MIT License. Source: `https://github.com/theislab/cfgen`.
- **scButterfly.** We used the official scButterfly implementation Cao et al. (2024) for RNA-ATAC translation. The repository is distributed under the MIT License. Source: `https://github.com/BioX-NKU/scButterfly`.

Pretrained embedding resources.

- **GenePT.** We used GenePT gene embeddings Chen & Zou (2024) for transcriptomic feature grouping. The Zenodo record for the GenePT embeddings is released under CC BY 4.0. Source: `https://zenodo.org/records/10833191`.
- **Gena-LM.** We used Gena-LM Fishman et al. (2025) to embed DNA sequences associated with ATAC peaks. The GENA-LM source repository is distributed under the MIT License. The Hugging Face model card for `AIRI-Institute/gena-lm-bigbird-base-t2t` does not list a separate model-weight license in its metadata. Sources: `https://github.com/AIRI-Institute/GENA_LM` and `https://huggingface.co/AIRI-Institute/gena-lm-bigbird-base-t2t`.

B Societal Impacts

TransFlow may support exploratory analyses when one single-cell modality is unavailable and help prioritize follow-up experiments for studying cell states and regulatory programs. Potential negative impacts mainly arise from overinterpreting generated omics profiles as experimentally validated measurements: inaccurate predictions could mislead downstream biological conclusions, clinical hypotheses, or resource allocation. Dataset bias may also affect performance for underrepresented tissues, donors, or cell states, and applications involving human omics data should continue to follow privacy safeguards and require experimental validation before biomedical use.

C Model Details

C.1 Condition Embedding and Adaptive Normalization

To rigorously inject conditional information into the generative process, we use the same conditioning mechanism as SAVE, which employs an Adaptive Layer Normalization (adaLN) mechanism to modulate network dynamics based on temporal and biological contexts. The biological attributes comprise cell type y and batch index b, which are projected via learnable lookup tables to obtain $\mathbf{e}_{type} \in \mathbb{R}^D$ and $\mathbf{e}_{batch} \in \mathbb{R}^D$. Conditioned on this vector, we compute the modulated output via the affine transformation:

$$\text{adaLN}(\mathbf{h}, \mathbf{c}) = \boldsymbol{\gamma}(\mathbf{c}) \odot \text{LayerNorm}(\mathbf{h}) + \boldsymbol{\beta}(\mathbf{c}), \tag{9}$$

where $\boldsymbol{\gamma}(\mathbf{c}), \boldsymbol{\beta}(\mathbf{c}) \in \mathbb{R}^D$ represent the scale and shift parameters learned from $\mathbf{c}$, and $\odot$ denotes the element-wise Hadamard product. Additionally, a dimension-wise gating factor $\boldsymbol{\alpha}(\mathbf{c})$ is regressed to scale the residual branches. This SAVE-specific mechanism ensures that the generative trajectory is dynamically adapted to the specified biological context while maintaining stable training dynamics through zero-initialization.

C.2 Encoder-Decoder Architecture

The core of Block-wise VAE is built upon the Diffusion Transformer (DiT) backbone, adapted for sequence-based single-cell data processing.

Encoder $\mathcal{E}_\phi$. The encoder $\mathcal{E}_\phi$ initiates the encoding process by transforming the high-dimensional input $\mathbf{x} \in \mathbb{R}^{D_{in}}$ into a sequence of feature embeddings $\mathbf{H}_0 \in \mathbb{R}^{L \times D}$, where L denotes the number of functional blocks. This tokenization is augmented with learnable positional embeddings $\mathbf{P}_{pos}$ to preserve the biological sequence order. The embedding sequence is then propagated through a stack of N_{enc} DiT blocks. Within each l-th block, we employ an adaptive computation mechanism where the hidden state update is governed by:

$$\mathbf{H}_l = \mathbf{H}'_l + \boldsymbol{\alpha}_l(\mathbf{c}) \odot \text{MLP}(\text{adaLN}(\mathbf{H}'_l, \mathbf{c})), \tag{10}$$

with the intermediate state $\mathbf{H}'_l$ derived from the attention mechanism. Crucially, the $\text{adaLN}(\cdot, \mathbf{c})$ operation applies adaptive layer normalization modulated by the condition vector $\mathbf{c}$, while $\boldsymbol{\alpha}_l(\mathbf{c})$ acts as a zero-initialized dimension-wise gating factor for residual connections to ensure stable training dynamics. The final hidden state $\mathbf{H}_{Nenc}$ is projected by dual MLPs to yield the mean $\boldsymbol{\mu} \in \mathbb{R}^{L \times D}$ and log-variance $\log \boldsymbol{\sigma}^2 \in \mathbb{R}^{L \times D}$, parameterizing the block-wise latent distribution.

Decoder $\mathcal{D}_\psi$. The decoder $\mathcal{D}_\psi$ mirrors the encoder's architecture to facilitate high-fidelity reconstruction. It accepts the sampled latent sequence $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \text{diag}(\boldsymbol{\sigma}^2))$ as input, projecting it back to the hidden dimension D. The sequence is processed through N_{dec} DiT blocks, which share the same adaptive modulation mechanism as the encoder. The resulting sequence is finally mapped back to the original feature space dimension via block-specific linear heads.

Implementation of Block-wise VAE. The input features are grouped based on their semantic embeddings derived from a foundation model. This clustering ensures that functionally related features are identified and collectively assigned to a global sequence. Formally, we define a permutation $\mathbf{P}$ to reorder the raw input $\mathbf{x} \in \mathbb{R}^{D_{in}}$ into a reorganized vector $\tilde{\mathbf{x}} = \mathbf{x}\mathbf{P}^\top$, such that features belonging to the same functional group are positioned contiguously.

When the reordered sequence $\tilde{\mathbf{x}}$ is fed into the encoder, it is partitioned into several segments according to a predefined block size M. Each segment $\mathbf{x}^k \in \mathbb{R}^M$ is then compressed into a single latent token $\mathbf{z}_0^k \in \mathbb{R}^D$ through the encoder$\mathcal{E}_\phi$.

C.3 Flow Matching and Denoising Model Details

The Flow Matching component is parameterized by a shared interval-conditioned velocity network $v_\theta(\mathbf{z}_t, t, t', \mathbf{c})$, tailored to regress bidirectional transport between RNA and ATAC latent sequences. The model accepts the current latent state $\mathbf{z}_t$, the current time coordinate t, the next time coordinate t', and the biological condition vector $\mathbf{c}_{bio}$ derived from batch and cell type embeddings. To incorporate the interval direction, we use separate time embedders for t and t' and sum their outputs with the biological condition embedding to form a unified modulation vector. During training, the condition is randomly masked to support classifier-free guidance; during inference, conditional and unconditional velocity predictions are combined according to the guidance weight. The latent state is projected into the hidden dimension and processed through a backbone of $N = 5$ stacked DiT blocks. The core of the backbone relies on the **adaLN-Zero** mechanism to inject the conditional signal $\mathbf{c}$. Unlike standard layer normalization, the normalization parameters—scale $\boldsymbol{\gamma}_l(\mathbf{c})$ and shift $\boldsymbol{\beta}_l(\mathbf{c})$—along with a gating scalar $\boldsymbol{\alpha}_l(\mathbf{c})$, are regressed directly from $\mathbf{c}$ via block-specific MLPs. Formally, the output $\mathbf{H}_l$ of the l-th DiT block is computed as:

$$\mathbf{H}_l = \mathbf{H}_{l-1} + \boldsymbol{\alpha}_l(\mathbf{c}) \odot \mathcal{F}_{block}(\text{adaLN}(\mathbf{H}_{l-1}, \mathbf{c})), \tag{11}$$

where $\text{adaLN}(\mathbf{h}, \mathbf{c}) = \boldsymbol{\gamma}_l(\mathbf{c}) \odot \text{LayerNorm}(\mathbf{h}) + \boldsymbol{\beta}_l(\mathbf{c})$, and $\mathcal{F}_{block}$ denotes the standard self-attention or feed-forward operation. Crucially, we initialize the final projection layer of the MLP predicting $\boldsymbol{\alpha}_l$ to zero. Consequently, at the initialization stage, $\boldsymbol{\alpha}_l(\mathbf{c}) \to 0$, rendering each residual block an approximate identity function $\mathbf{H}_l \approx \mathbf{H}_{l-1}$. This zero-initialization strategy ensures stable gradient propagation and allows the network to progressively learn the complex transport dynamics from a well-behaved starting point.

D Experiment Details

D.1 Dataset Details

We utilize the following three datasets to evaluate our model across different scales and conditional complexities:

- **OpenProblems BMMC:** Sourced from the NeurIPS 2021 Multimodal Integration Challenge, this dataset profiles human bone marrow mononuclear cells. It provides both cell-type annotations and batch labels, allowing us to evaluate metadata-conditioned bidirectional cross-modality translation.
- **PBMC10k:** This is a standard dataset from 10x Genomics containing paired RNA and ATAC profiles of human peripheral blood mononuclear cells. Unlike BMMC, this dataset only provides cell type annotations, restricting our evaluation to single-condition settings. It serves as a complementary benchmark for examining the model's ability to capture fine-grained biological heterogeneity under a fixed technical context, without the confounding factor of batch effects.

- **MiniAtlas:** MiniAtlas is a single-cell multi-omics dataset containing over 100,000 cells with paired scRNA-seq and scATAC-seq measurements across 19 tissues and 56 cell types. We use this cohort as a supplementary benchmark with broader biological heterogeneity.

D.2 Data Preprocessing

To ensure rigorous reproducibility and fair benchmarking, we implemented a unified data processing pipeline applied consistently across all datasets (OpenProblems BMMC, PBMC10k and MiniAtlas), and established a common feature space for all models by performing feature selection prior to model-specific formatting. For BMMC and PBMC10k, we use the fixed train/test splits throughout the experiments. Specifically for MiniAtlas, a randomly selected subset comprising 20% of the total cell population was extracted to serve as the primary cohort for both model training and subsequent performance evaluation.

RNA Modality. The raw gene expression counts were processed as follows:

- **Normalization & Log-transformation:** Raw counts were library-size normalized (scaling total counts per cell to 10^4) and then transformed using $\log(1 + x)$.
- **Feature Selection:** Highly Variable Genes (HVGs) were selected based on the dispersion of the normalized and log-transformed data.
- **Scaling:** The features were further scaled using `MaxAbsScaler` to map values into the $[0, 1]$ interval, preserving sparsity while stabilizing training.

ATAC Modality. The chromatin accessibility data was processed as follows:

- **Binarization:** Due to the sparsity and near-binary nature of single-cell chromatin accessibility, raw peak counts were binarized.
- **Feature Selection:** Consistent with the RNA modality, highly variable peaks were selected based on the variability of the processed profiles.

Model-Specific Input Configurations While the set of selected features (genes and peaks) was identical across all experiments, the data scaled was adapted to adhere to the official requirements of each baseline method:

- **TransFlow:** The RNA VAE used the raw counts matrix for the ZINB likelihood. The ATAC VAE used the binarized peak profiles.
- **BABEL & scMoGNN & scButterfly:** These predictive models were fed with the normalized and log-transformed data matrices (without MaxAbsScaling), consistent with their standard input protocols.
- **scDiffusion-X & CFGen:** Following their official implementations, these generative baselines were trained using raw count matrices (subset to the same selected HVGs).
- **MultiVI:** Following the input requirements of `scvi-tools`, MultiVI was trained on raw integer RNA counts and binarized ATAC peak profiles. Specifically, the RNA modality was registered using the raw count matrix, whereas the ATAC modality used the binarized peak matrix.
- **GENOT:** GENOT requires modality-specific 30-dimensional PCA embeddings rather than raw or binarized feature matrices. Paired ATAC and RNA cells were aligned by barcode, projected separately to PCA space, and used for bidirectional translation. Predictions were generated in PCA space and inverse-transformed to the original feature space.

D.3 Experimental Setup and Computing Resources

All models were implemented using PyTorch. We utilized the AdamW optimizer with a decoupled weight decay of 2.5×10^{-5}, and the learning rate was set to 1×10^{-4} with a cosine annealing schedule. We set the VAE reconstruction weight $\lambda_{\text{rec}} = 1$ and the KL divergence weight $\lambda_{\text{KL}} = 0.5$. For BiFM, the consistency weight was set to $\lambda_{\text{con}} = 0.1$. Across datasets, RNA VAEs used the

raw-count layer for the ZINB likelihood, ATAC VAEs used binarized peak profiles. VAE training used batch sizes of 1024 for RNA and 512 for ATAC, while BiFM used a batch size of 1024. Training was performed on NVIDIA 3090 (24GB) GPUs with approximately 100GB of host memory.

For BMMC, the RNA and ATAC block sizes were set to 1,200 genes and 8,000 peaks, respectively; the VAE stage required approximately 1.5 hours and the BiFM stage required another 1.5 hours. For PBMC10k, both RNA and ATAC block sizes were set to 4,000, with approximately 0.5 hours for VAE training and 0.5 hours for BiFM training. For MiniAtlas, the RNA and ATAC block sizes were set to 2,000 genes and 8,000 peaks, respectively; the VAE and BiFM stages each required approximately 2.5 hours. The reported runs used VAE pre-training up to 200 epochs and BiFM training for approximately 3,200 iterations.

D.4 Training and Sampling Algorithms

We provide the pseudo-code for the two-stage training process and the sampling procedure used in TransFlow. The training stage is shown in Algorithm 1, while the sampling stage is presented in Algorithm 2. In the main text, z denotes the latent representation of an individual sample; in the algorithms below, bold symbols such as $\mathbf{z}$ denote mini-batch latent tensors.

Algorithm 1 TransFlow Training Algorithm

Input: Paired multi-omics dataset $\mathcal{D} = \{(\mathbf{x}_i^{\mathrm{RNA}}, \mathbf{x}_i^{\mathrm{ATAC}}, \mathbf{c}_i)\}$, Pre-training epochs I_{vae}, Flow iterations I_{flow}
// Stage 1: Pre-train Block-wise VAEs
for $epoch = 1$ **to** I_{vae} **do**
  Optimize $\mathcal{L}_{VAE}^{\mathrm{RNA}}$ and $\mathcal{L}_{VAE}^{\mathrm{ATAC}}$ independently (Eq. 5)
end for
Freeze encoder and decoder parameters for both modalities
// Stage 2: Train Shared Bidirectional Flow Matching
for $iter = 1$ **to** I_{flow} **do**
  Sample mini-batch of pairs $(\mathbf{x}_i^{\mathrm{RNA}}, \mathbf{x}_i^{\mathrm{ATAC}}, \mathbf{c}) \sim \mathcal{D}$
  Encode latents: $\mathbf{z}^{\mathrm{RNA}} \leftarrow \mathcal{E}^{\mathrm{RNA}}(\mathbf{x}_i^{\mathrm{RNA}})$, $\mathbf{z}^{\mathrm{ATAC}} \leftarrow \mathcal{E}^{\mathrm{ATAC}}(\mathbf{x}_i^{\mathrm{ATAC}})$
  Sample interval endpoints $0 \le t_s < t_e \le 1$
  $\mathbf{z}_{t_s} \leftarrow (1 - t_s)\mathbf{z}^{\mathrm{RNA}} + t_s\mathbf{z}^{\mathrm{ATAC}}$
  $\mathbf{z}_{t_e} \leftarrow (1 - t_e)\mathbf{z}^{\mathrm{RNA}} + t_e\mathbf{z}^{\mathrm{ATAC}}$
  $\mathbf{u}_{\mathrm{R}\to\mathrm{A}} \leftarrow (\mathbf{z}_{t_e} - \mathbf{z}_{t_s})/(t_e - t_s)$
  $\mathbf{u}_{\mathrm{A}\to\mathrm{R}} \leftarrow (\mathbf{z}_{t_s} - \mathbf{z}_{t_e})/(t_e - t_s)$
  Forward prediction: $\hat{\mathbf{u}}_{\mathrm{R}\to\mathrm{A}} \leftarrow v_\theta(\mathbf{z}_{t_s}, t_s, t_e, \mathbf{c})$
  Reverse prediction: $\hat{\mathbf{u}}_{\mathrm{A}\to\mathrm{R}} \leftarrow v_\theta(\mathbf{z}_{t_e}, t_e, t_s, \mathbf{c})$
  $\mathcal{L}_{\mathrm{R}\to\mathrm{A}} \leftarrow \|\hat{\mathbf{u}}_{\mathrm{R}\to\mathrm{A}} - \mathbf{u}_{\mathrm{R}\to\mathrm{A}}\|^2$
  $\mathcal{L}_{\mathrm{A}\to\mathrm{R}} \leftarrow \|\hat{\mathbf{u}}_{\mathrm{A}\to\mathrm{R}} - \mathbf{u}_{\mathrm{A}\to\mathrm{R}}\|^2$
  $\mathcal{L}_{\mathrm{con}} \leftarrow \|\hat{\mathbf{u}}_{\mathrm{R}\to\mathrm{A}} - (-\hat{\mathbf{u}}_{\mathrm{A}\to\mathrm{R}})\|^2$
  // Joint Optimization
  Update θ by minimizing $\mathcal{L}_{total} = \mathcal{L}_{\mathrm{R}\to\mathrm{A}} + \mathcal{L}_{\mathrm{A}\to\mathrm{R}} + \lambda_{\mathrm{con}}\mathcal{L}_{\mathrm{con}}$
end for

D.5 Baseline Configurations

To ensure a fair comparison, we configured the baseline models following their official implementations while aligning key hyperparameters where possible.

BABEL & scMoGNN: We utilized the official implementations of BABEL and scMoGNN provided in the `dance` package. BABEL was configured as hidden dim=64 with domain-specific encoders/decoders. Training minimized the composite RMSE loss with weight=1.0, using the Adam optimizer ($lr = 0.01$, batch size=64) for up to 500 epochs, with early stopping (patience=20). scMoGNN constructs a heterogeneous graph integrating pathway information. The architecture employs a 4-layer ScMoGNN with residual connections and alpha-weighted pathway aggregation. Optimization used AdamW ($lr = 10^{-2}$, weight decay=10^{-5}) with an exponential schedule (decay=0.99), trained for max 15,000 epochs (batch size=1,000, patience=200).

Algorithm 2 Interval-Euler Sampling for Cross-Modality Translation

Input: Source data $\mathbf{x}_{src}$, target decoder $\mathcal{D}_{tgt}$, condition $\mathbf{c}$, direction d, steps N
Encode source: $\mathbf{z}_{src} \leftarrow \mathcal{E}_{src}(\mathbf{x}_{src})$
Initial state: $\mathbf{z}_{t_0} \leftarrow \mathbf{z}_{src}$
Define time grid $\{t_i\}_{i=0}^{N}$: increasing for RNA→ATAC, decreasing for ATAC→RNA
for $i = 0$ **to** $N - 1$ **do**
  $\Delta t = t_{i+1} - t_i$
  $\mathbf{v}_i = v_\theta(\mathbf{z}_{t_i}, t_i, t_{i+1}, \mathbf{c})$
  $\mathbf{z}_{t_{i+1}} \leftarrow \mathbf{z}_{t_i} + |\Delta t| \cdot \mathbf{v}_i$
end for
Decode target: $\hat{\mathbf{x}}_{tgt} \leftarrow \mathcal{D}_{tgt}(\mathbf{z}_{t_N})$
Return $\hat{\mathbf{x}}_{tgt}$

scDiffusion-X: To evaluate scDiffusion-X, we adopted a three-stage workflow comprising autoencoder pre-training, conditional generation, and cross-modality translation. First, a learnt binarized autoencoder was trained for 300 epochs to compress multimodal data into a latent space. Subsequently, for generation, we trained a cell-type-conditioned diffusion model (128 channels, 1,000 diffusion steps) for 800,000 steps, utilizing a linear noise schedule and DDIM sampling. For cross-modality translation, we employed a normal model configuration (160 channels) rather than the official large setting. Translation inference was performed using DDPM sampling with a classifier guidance scale of 3.0, averaging results over five independent runs to ensure robustness.

CFGen: We implemented CFGen using a two-stage training strategy for the unimodal generation experiment. In the first stage, we trained a joint autoencoder to compress the high-dimensional single-cell data into a latent representation. The encoder configuration utilized a joint architecture with binarization enabled to effectively handle the sparsity and discrete nature of single-cell features. This autoencoder was trained for 300 epochs to ensure robust latent representations. In the second stage, we trained the primary generative model CFGen for 1500 epochs as the offical implementations, utilizing the latent space established by the pre-trained autoencoder checkpoints. The generative backbone employed a large ResNet architecture as the denoising module, which was also configured with binarization enabled to align with the encoder's latent structure. The training objective was set to Conditional Flow Matching with a Gaussian prior (Noise-to-Data).

MultiVI: We implemented MultiVI using the official `MULTIVI` model from `scvi-tools`. The model was configured with a 20-dimensional latent space, two encoder layers, and two decoder layers. Other architectural settings followed the `scvi-tools` defaults. Training was performed for up to 500 epochs with early stopping enabled, using a batch size of 1,024. Optimization used AdamW with the default MultiVI training parameters in `scvi-tools` ($lr = 10^{-4}$, weight decay=10^{-3}, $\epsilon = 10^{-8}$), with a 50-epoch KL warmup and adversarial modality mixing enabled. For bidirectional translation, we constructed modality-specific test inputs by setting MultiVI's modality index to RNA-only for RNA-to-ATAC prediction and ATAC-only for ATAC-to-RNA prediction.

GENOT: We used the bidirectional modality translation pipeline implemented in the official GENOT repository. Two independent models were trained for each split, one for ATAC→RNA and one for RNA→ATAC. Each model used an MLP vector field with latent, condition, and time embedding dimension 256, 8 SiLU layers per block, and 128 sinusoidal time-encoding frequencies. Mini-batches of 1,024 source and target cells were matched with a balanced entropic Gromov–Wasserstein solver using squared Euclidean intra-modality costs, mean cost scaling, and $\varepsilon = 0.01$. The inner linear OT solver was Sinkhorn with momentum value 1.0 starting at iteration 25. GENOT was trained for 3,000 iterations per direction without early stopping, using the pure GW setting and the default AdamW optimizer ($lr = 10^{-4}$, weight decay=10^{-10}). The training objective was the L2 flow-matching loss between the predicted vector field and the matched Gaussian-noise-to-target displacement.

scButterfly: We used the official scButterfly implementation for RNA-ATAC translation and selected the basic scButterfly-B variant without data augmentation. We used the default dual-aligned VAE architecture, including modality-specific encoders and decoders, chromosome-split ATAC

blocks, and a shared 128-dimensional stochastic translator. RNA autoencoder pre-training, ATAC autoencoder pre-training, and joint bidirectional translation were performed with a batch size of 1024. All other architectural choices, loss terms, optimizer settings, KL warm-up, early stopping, checkpointing, and inference procedures followed the official scButterfly repository.

D.6 Standardized Evaluation Space

To ensure a fair comparison, all quantitative evaluations and visualizations were performed in the unified normalized and log-transformed space. Model outputs generated in method-specific spaces were first transformed back to the selected gene or peak feature space before evaluation. GENOT predictions, which were produced in modality-specific PCA spaces, were inverse-transformed to the original selected feature spaces. For MultiVI, RNA predictions were rescaled to a target library size of 10,000 and transformed with $log(1 + x)$. For models outputting raw counts, including scDiffusion-X and CFGen, generated RNA outputs were post-processed with library-size normalization and log-transformation before comparison with the held-out ground truth, while generated ATAC outputs were converted to the corresponding peak-accessibility representation used for evaluation.

D.7 Metrics Details

We assess standard cross-modality translation using class-macro distributional metrics, paired feature-level metrics, and label-transfer metrics. The PCA reference and label-transfer probe are fitted using outer-train data only and then applied unchanged to real-test and generated-test profiles.

Class-Macro W_2. For each dataset and target modality, a 200-component PCA basis is fitted on outer-train profiles. The main table uses the first 50 components. Within each test cell type, we compute the exact empirical 2-Wasserstein distance with squared Euclidean cost and uniform empirical masses, using at most 1,000 cells per class. The same selected indices are used for real and generated profiles. If P_c and Q_c denote the real and generated empirical distributions for cell type c, the reported class-macro score is

$$W_{2,\mathrm{macro}}(P,Q) = \frac{1}{C}\sum_{c=1}^{C}\left(\min_{\pi_c \in \Pi(P_c,Q_c)} \sum_{i,j} (\pi_c)_{ij} \|\mathbf{x}_{i,c} - \mathbf{y}_{j,c}\|^2\right)^{1/2}. \tag{12}$$

We solve each empirical transport problem using the POT exact EMD implementation. Lower values indicate closer distributions.

Class-Macro Linear-Time RBF-MMD^2. The PCA representation and class grouping are the same as for class-macro W_2. For each PCA dimension, the reference bandwidth σ_0 is the median pairwise Euclidean distance among at most 1,024 outer-train PCA scores. We use Gaussian kernels with bandwidths $a\sigma_0$ for $a \in \{0.5, 1, 2\}$ and compute the unbiased linear-time estimator within each cell type:

$$\widehat{\mathrm{MMD}}^2 = \frac{1}{m}\sum_{i=1}^{m}\left[k(\mathbf{x}_{2i-1},\mathbf{x}_{2i}) + k(\mathbf{y}_{2i-1},\mathbf{y}_{2i}) - k(\mathbf{x}_{2i-1},\mathbf{y}_{2i}) - k(\mathbf{x}_{2i},\mathbf{y}_{2i-1})\right]. \tag{13}$$

The three bandwidth estimates are averaged within each class and then equally across classes. This finite-sample unbiased estimate may be slightly negative. Lower values indicate closer distributions.

Feature-Level Fidelity. For ATAC→RNA translation, Row Cor. is the mean Pearson correlation between each generated RNA profile and its paired real profile across genes. For RNA→ATAC translation, AUPRC uses the real binary peak-accessibility labels and generated accessibility scores; the main table reports the global AUPRC over all cell–peak entries. Higher values indicate better paired feature recovery.

Label Transfer Accuracy. For each dataset and target modality, we fit one probe using outer-train target profiles and outer-train cell-type labels. The probe uses 50-component PCA, standardizes the PCA scores, and fits logistic regression with maximum 3,000 iterations and random state 0. The same

frozen probe is applied to real-test and generated-test profiles. We report generated-profile accuracy in the main comparison and use

$$\Delta\mathrm{Acc} = \mathrm{Acc}_{\mathrm{real}} - \mathrm{Acc}_{\mathrm{gen}}, \tag{14}$$

where lower values are preferred. A positive value indicates lower generated-profile accuracy than the real-test reference, while a negative value indicates higher generated-profile accuracy.

E Unimodal Generation Results

While formulated for cross-modality transport, TransFlow's architecture can also be used for conditional generation over individual modalities. We report this auxiliary experiment to examine its compatibility with a prior-to-data generation setting, independently of the main cross-modal translation task.

For single-modality generation, we use the same modality-specific latent space learned by the corresponding VAE, but replace the observed source modality with a standard Gaussian latent sample. Given a biological condition vector $\mathbf{c}$, the conditional flow maps this prior sample toward the target modality distribution, and the corresponding decoder reconstructs the generated RNA or ATAC profile. This setting recovers the standard prior-to-data generation task as an auxiliary use of TransFlow, while the main model remains optimized for paired source-to-target translation.

For this appendix-only generation benchmark, we compare against scDiffusion-X and CFGen. Implementation details and hyperparameter settings for evaluated baselines are reported in Appendix D.5.

Table 4: **Unimodal generative performance on BMMC.** We report distributional alignment (W_2, RBF-MMD^2), feature-level fidelity (Row Cor. for RNA and AUPRC for ATAC), and label-transfer results (Gen Acc and the signed accuracy gap Δ Acc). Lower Δ Acc is preferred. Best results are highlighted in bold.

Model	RNA					ATAC				
	$W_2\downarrow$	RBF-$\mathrm{MMD}^2\downarrow$	Row Cor. $\uparrow$	Gen Acc $\uparrow$	Δ Acc $\downarrow$	$W_2\downarrow$	RBF-$\mathrm{MMD}^2\downarrow$	AUPRC $\uparrow$	Gen Acc $\uparrow$	Δ Acc $\downarrow$
scDiffusion-X	15.359	1.304	0.243	0.878	-0.001	12.266	1.142	0.056	0.831	0.008
CFGen	18.080	1.352	0.168	0.892	-0.020	11.418	**1.132**	0.053	0.880	-0.040
TransFlow (Ours)	**11.500**	**1.279**	**0.312**	**0.956**	**-0.080**	**11.204**	1.220	**0.142**	**0.942**	**-0.101**

Quantitative Fidelity on the BMMC Benchmark. As summarized in Table 4, TransFlow achieves strong unimodal generation performance on BMMC. For RNA generation, it obtains the lowest W_2 and RBF-MMD^2, the highest row-wise correlation, and the highest generated label-transfer accuracy. For ATAC generation, TransFlow achieves the lowest W_2 and the highest AUPRC and generated label-transfer accuracy, while CFGen obtains a slightly lower RBF-MMD^2. These results indicate that TransFlow can preserve feature-level structure and downstream annotation utility in this single-modality generation benchmark.

F Additional Qualitative Results

In this section, we provide supplementary qualitative visualizations to support the quantitative findings presented in Section 4.2.

Figure 3 provides qualitative UMAP visualizations for cross-modality translation on BMMC. We compare TransFlow with BABEL, scMoGNN, and scDiffusion-X, displaying embeddings by cell-type annotation and by data source.

G Additional Analyses

This section presents the newly added decoded-trajectory visualization.

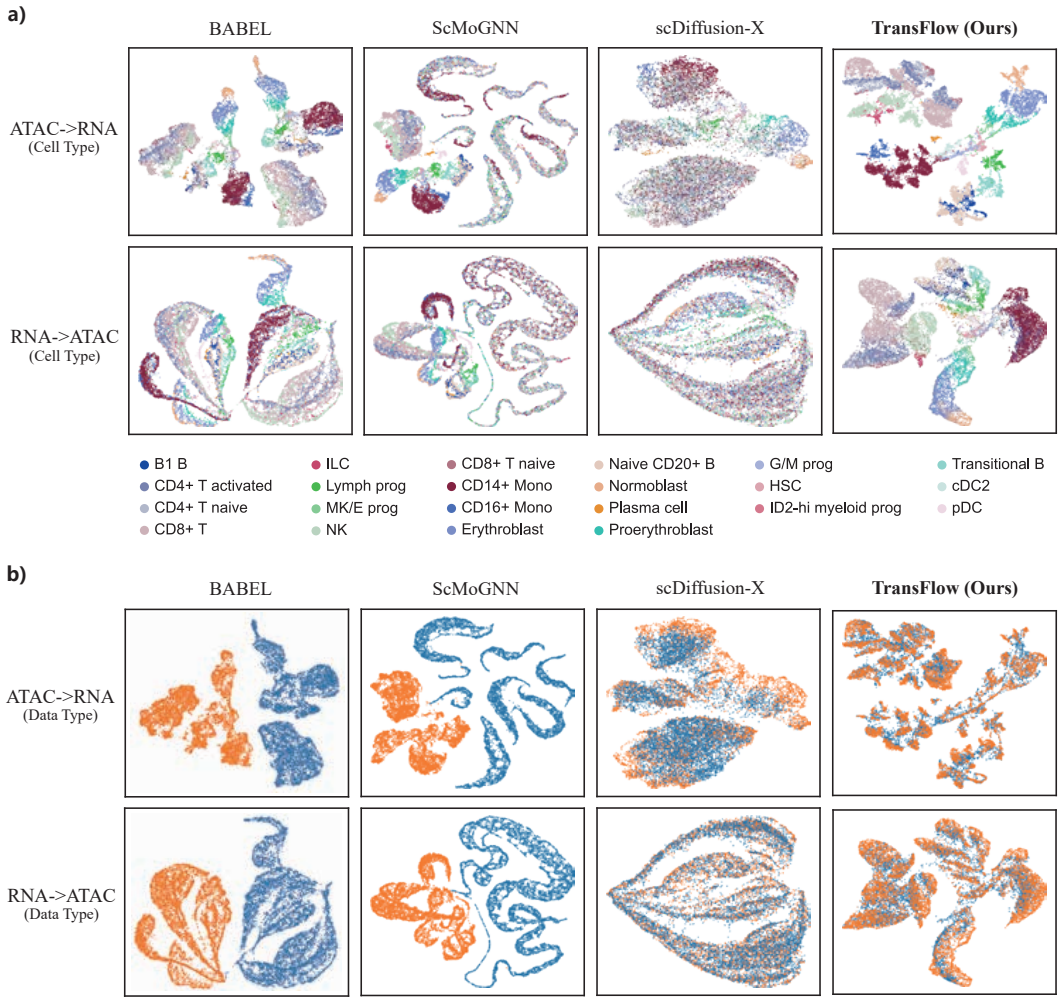

Figure 3: **Qualitative evaluation of cross-modality translation on the BMMC dataset.** Columns correspond to different methods (BABEL, scMoGNN, scDiffusion-X, and TransFlow). **(a)** UMAP embeddings for ATAC→RNA and RNA→ATAC translation, colored by cell-type annotation. **(b)** The same embeddings colored by data source (**blue**: generated; **orange**: reference). These visualizations provide a qualitative view of cell-type organization and generated–reference overlap. Because UMAP can distort global distances, we interpret them together with the quantitative distributional and feature-level metrics.

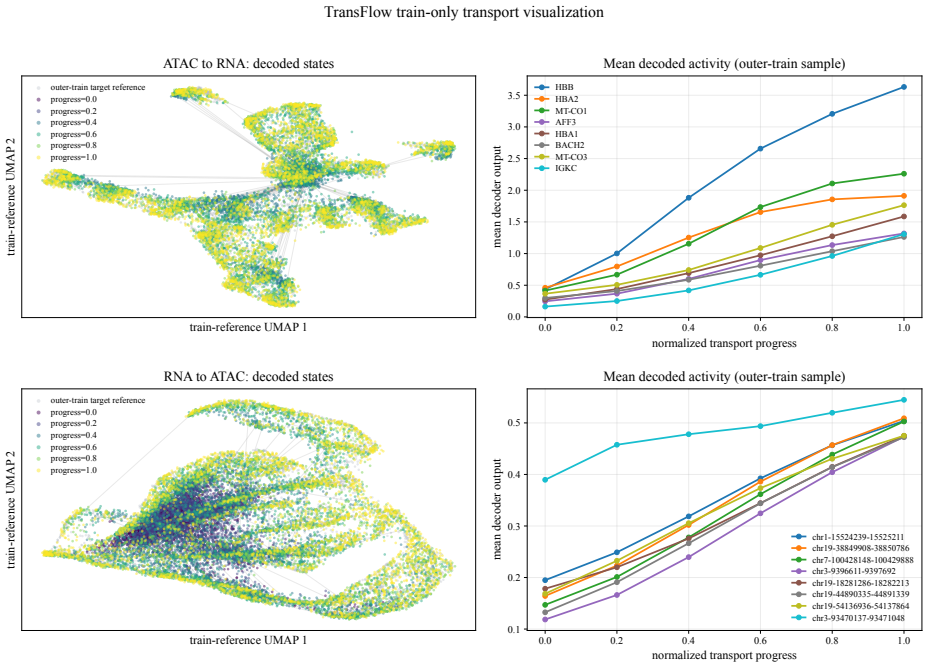

Figure 4: **Decoded transport trajectories on BMMC.** Top: ATAC→RNA; bottom: RNA→ATAC. Points show decoded states at selected normalized integration progress values in a target-modality UMAP fitted on the balanced outer-train visualization subset. The accompanying curves show mean decoded activity for selected target features. This is a visualization of model integration on training cells, not a biological developmental trajectory.

G.1 Decoded Transport Trajectories

Figure 4 visualizes decoded states at selected integration steps for both translation directions. This illustration uses a balanced outer-train subset and a projection fitted on that same subset; it does not represent a measured developmental trajectory.

H Additional Metrics

This appendix reports additional PCA-dimension sensitivity and label-transfer metrics for BMMC and PBMC10k. The preprocessing and fixed train/test splits are the same as those described for these datasets in the main experimental setup. Values are aggregated across multiple independent runs and are reported as mean ± sample standard deviation. The PCA analysis reports class-macro W_2 and class-macro linear-time RBF-MMD^2 at 20, 30, 50, 100, and 200 principal components. The label-transfer analysis reports accuracy, balanced accuracy, macro-F1, and weighted-F1 on generated profiles, together with the corresponding real-test reference scores.

H.1 Metric Computation

For each dataset and target modality, a 200-component PCA basis is fitted on the outer-train profiles from a fixed reference run and applied unchanged to the outer-train, real-test, and generated-test profiles. The sensitivity analysis retains the first $k \in \{20, 30, 50, 100, 200\}$ components. For each test cell type, exact empirical W_2 is computed using all cells up to a maximum of 1,000, with the same selected cell indices for real and generated profiles; the reported value is the unweighted mean across cell types. The class-macro linear-time RBF-MMD^2 uses a bandwidth estimated from at most 1,024 outer-train PCA scores and the three scales $0.5\sigma_0$, σ_0, and $2\sigma_0$. The three-scale estimates are averaged within each cell type and then equally across cell types.

Table 5: **PCA-dimension sensitivity on BMMC.** Entries are mean ± sample standard deviation across multiple independent runs.

(a) ATAC → RNA

Model	20 PCs		30 PCs		50 PCs		100 PCs		200 PCs	
	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$
BABEL	13.0302 ± 0.2601	0.1029 ± 0.0068	14.0860 ± 0.2779	0.1108 ± 0.0076	15.3628 ± 0.2816	0.1163 ± 0.0078	17.4271 ± 0.3159	0.1189 ± 0.0086	20.5363 ± 0.2702	0.1208 ± 0.0070
scMoGNN	22.0908 ± 0.0064	0.3581 ± 0.0000	22.8349 ± 0.0069	0.3539 ± 0.0000	23.6424 ± 0.0068	0.3411 ± 0.0001	24.9324 ± 0.0073	0.3128 ± 0.0002	27.0680 ± 0.0078	0.2756 ± 0.0002
scDiffusion-X	34.1178 ± 0.0717	0.6405 ± 0.0047	34.8630 ± 0.0762	0.6242 ± 0.0045	35.6264 ± 0.0808	0.5998 ± 0.0045	36.5811 ± 0.0812	0.5530 ± 0.0042	38.1241 ± 0.0804	0.4898 ± 0.0037
GENOT	24.3525 ± 0.3243	0.1926 ± 0.0029	25.3014 ± 0.3123	0.1871 ± 0.0027	26.0312 ± 0.3022	0.1772 ± 0.0023	27.2131 ± 0.2901	0.1579 ± 0.0017	29.1916 ± 0.2718	0.1365 ± 0.0011
scButterfly	9.3460 ± 0.3353	0.0234 ± 0.0008	10.3574 ± 0.3500	0.0267 ± 0.0012	11.7498 ± 0.3434	0.0309 ± 0.0018	**14.0961 ± 0.2985**	0.0358 ± 0.0018	**17.5089 ± 0.2495**	0.0437 ± 0.0017
MultiVI	30.1885 ± 0.1151	0.5406 ± 0.0035	30.9326 ± 0.1167	0.5282 ± 0.0034	31.8923 ± 0.1221	0.5125 ± 0.0035	33.0108 ± 0.1242	0.4741 ± 0.0033	34.7342 ± 0.1204	0.4206 ± 0.0031
TransFlow	**8.3681 ± 0.0169**	**0.0092 ± 0.0003**	**9.4759 ± 0.0072**	**0.0091 ± 0.0002**	**11.2189 ± 0.0061**	**0.0095 ± 0.0003**	14.5800 ± 0.0201	**0.0095 ± 0.0001**	19.1930 ± 0.0339	**0.0095 ± 0.0002**

(b) RNA → ATAC

Model	20 PCs		30 PCs		50 PCs		100 PCs		200 PCs	
	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$
BABEL	12.6683 ± 0.0761	0.2044 ± 0.0019	12.7907 ± 0.0757	0.2028 ± 0.0018	12.9377 ± 0.0695	0.1991 ± 0.0013	13.2674 ± 0.0619	0.1904 ± 0.0009	13.8956 ± 0.0607	0.1777 ± 0.0004
scMoGNN	14.2880 ± 0.2011	0.3232 ± 0.0062	14.3896 ± 0.1999	0.3189 ± 0.0060	14.5017 ± 0.1983	0.3094 ± 0.0058	14.7593 ± 0.1946	0.2881 ± 0.0054	15.2541 ± 0.1880	0.2562 ± 0.0047
scDiffusion-X	9.4692 ± 0.1419	0.1035 ± 0.0027	9.6082 ± 0.1412	0.1030 ± 0.0025	9.7800 ± 0.1395	0.1018 ± 0.0024	10.1725 ± 0.1354	0.0993 ± 0.0022	10.8968 ± 0.1285	0.0956 ± 0.0019
GENOT	14.6551 ± 0.1394	0.1768 ± 0.0072	15.0585 ± 0.1333	0.1718 ± 0.0063	15.1771 ± 0.1320	0.1674 ± 0.0063	15.4490 ± 0.1291	0.1583 ± 0.0064	15.9681 ± 0.1242	0.1457 ± 0.0065
scButterfly	7.1686 ± 0.1964	0.0562 ± 0.0059	7.3308 ± 0.1924	0.0573 ± 0.0059	7.5469 ± 0.1864	0.0576 ± 0.0060	8.0327 ± 0.1738	0.0579 ± 0.0059	8.9077 ± 0.1554	0.0586 ± 0.0057
MultiVI	53.8704 ± 0.2190	1.0485 ± 0.0025	54.0265 ± 0.2154	1.0330 ± 0.0028	54.0841 ± 0.2142	1.0269 ± 0.0030	54.1620 ± 0.2136	1.0178 ± 0.0031	54.3109 ± 0.2125	1.0000 ± 0.0033
TransFlow	**5.5555 ± 0.1248**	**0.0293 ± 0.0042**	**5.7805 ± 0.1240**	**0.0287 ± 0.0041**	**6.0772 ± 0.1215**	**0.0274 ± 0.0040**	**6.7197 ± 0.1152**	**0.0250 ± 0.0037**	**7.8055 ± 0.1065**	**0.0226 ± 0.0033**

Table 6: **PCA-dimension sensitivity on PBMC10k.** Entries are mean ± sample standard deviation across multiple independent runs.

(a) ATAC → RNA

Model	20 PCs		30 PCs		50 PCs		100 PCs		200 PCs	
	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$
BABEL	11.6803 ± 0.4388	0.1797 ± 0.0154	12.2799 ± 0.3921	0.1792 ± 0.0136	12.9113 ± 0.3699	0.1708 ± 0.0126	14.2659 ± 0.3681	0.1596 ± 0.0117	16.3530 ± 0.4074	0.1431 ± 0.0118
scMoGNN	20.0640 ± 0.0078	0.5054 ± 0.0005	20.3624 ± 0.0079	0.4868 ± 0.0005	20.6909 ± 0.0078	0.4555 ± 0.0004	21.3751 ± 0.0077	0.3991 ± 0.0003	22.5757 ± 0.0081	0.3278 ± 0.0003
scDiffusion-X	23.0987 ± 0.0295	0.5824 ± 0.0026	23.6177 ± 0.0351	0.5694 ± 0.0027	24.0286 ± 0.0344	0.5399 ± 0.0026	24.7062 ± 0.0339	0.4814 ± 0.0024	25.8450 ± 0.0309	0.4038 ± 0.0020
GENOT	16.3807 ± 0.5242	0.1590 ± 0.0102	17.0675 ± 0.5116	0.1521 ± 0.0103	17.4633 ± 0.4999	0.1407 ± 0.0098	18.2737 ± 0.4786	0.1220 ± 0.0087	19.6694 ± 0.4457	0.1003 ± 0.0072
scButterfly	14.7210 ± 0.5531	0.2454 ± 0.0174	15.7604 ± 0.5321	0.2611 ± 0.0167	16.3459 ± 0.5211	0.2494 ± 0.0158	17.3847 ± 0.4936	0.2242 ± 0.0140	19.0560 ± 0.4593	0.1910 ± 0.0117
MultiVI	19.1363 ± 0.2703	0.4714 ± 0.0113	19.6226 ± 0.2562	0.4610 ± 0.0104	20.0813 ± 0.2316	0.4356 ± 0.0091	20.8419 ± 0.2288	0.3834 ± 0.0084	22.1445 ± 0.2132	0.3173 ± 0.0070
TransFlow	**7.7017 ± 0.0924**	**0.0507 ± 0.0015**	**8.4024 ± 0.0896**	**0.0519 ± 0.0015**	**9.2378 ± 0.0823**	**0.0507 ± 0.0014**	**10.7673 ± 0.0716**	**0.0479 ± 0.0012**	**13.1146 ± 0.0554**	**0.0447 ± 0.0010**

(b) RNA → ATAC

Model	20 PCs		30 PCs		50 PCs		100 PCs		200 PCs	
	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$	$W_2 \downarrow$	RBF-MMD$^2 \downarrow$
BABEL	16.4321 ± 0.7902	0.2572 ± 0.0245	16.5255 ± 0.7851	0.2522 ± 0.0239	16.6523 ± 0.7766	0.2440 ± 0.0230	16.9451 ± 0.7568	0.2259 ± 0.0211	17.5038 ± 0.7233	0.1982 ± 0.0182
scMoGNN	20.2392 ± 0.0212	0.3772 ± 0.0015	20.3149 ± 0.0210	0.3690 ± 0.0015	20.3993 ± 0.0210	0.3546 ± 0.0014	20.6043 ± 0.0208	0.3237 ± 0.0013	21.0037 ± 0.0206	0.2776 ± 0.0011
scDiffusion-X	12.4655 ± 0.3716	0.1004 ± 0.0135	12.5879 ± 0.3675	0.0988 ± 0.0132	12.7300 ± 0.3638	0.0955 ± 0.0129	13.0706 ± 0.3537	0.0887 ± 0.0121	13.7120 ± 0.3371	0.0778 ± 0.0107
GENOT	17.2966 ± 0.6521	0.1766 ± 0.0050	17.6716 ± 0.6461	0.1723 ± 0.0048	17.7758 ± 0.6423	0.1677 ± 0.0046	18.0276 ± 0.6334	0.1579 ± 0.0043	18.5123 ± 0.6172	0.1421 ± 0.0041
scButterfly	24.0296 ± 0.3070	0.5422 ± 0.0073	24.2309 ± 0.3050	0.5380 ± 0.0072	24.3341 ± 0.3040	0.5224 ± 0.0070	24.5338 ± 0.3018	0.4863 ± 0.0066	24.9025 ± 0.2980	0.4298 ± 0.0061
MultiVI	45.9260 ± 0.0391	1.0145 ± 0.0019	45.9677 ± 0.0392	1.0045 ± 0.0019	46.0057 ± 0.0393	0.9884 ± 0.0018	46.0984 ± 0.0392	0.9515 ± 0.0017	46.2817 ± 0.0392	0.8893 ± 0.0015
TransFlow	**7.5839 ± 0.0850**	**0.0375 ± 0.0099**	**7.7808 ± 0.0820**	**0.0373 ± 0.0097**	**8.0183 ± 0.0782**	**0.0361 ± 0.0093**	**8.5685 ± 0.0721**	**0.0339 ± 0.0084**	**9.5484 ± 0.0634**	**0.0303 ± 0.0070**

For label transfer, a frozen probe is fitted on outer-train target profiles and cell-type labels. It applies 50-component PCA, standardization, and logistic regression with a maximum of 3,000 iterations. The same probe is applied to real-test and generated-test profiles. We report accuracy, balanced accuracy, macro-F1, and weighted-F1. Across multiple independent runs, means and sample standard deviations are computed with denominator $n - 1$.

H.2 PCA-Dimension Sensitivity

The following tables report class-macro W_2 and class-macro linear-time RBF-MMD^2 across PCA dimensions. Lower values are preferred. Best values within each metric column are shown in bold.

H.3 Label-Transfer Metrics

The following tables report generated-profile label-transfer scores and the corresponding real-test reference scores. Best generated-profile values within each metric column are shown in bold.

Table 7: **Label-transfer metrics on BMMC.** Entries are mean ± sample standard deviation across multiple independent runs.

(a) ATAC → RNA

Model	Gen. Acc.	Gen. Bal. Acc.	Gen. Macro-F1	Gen. W-F1
BABEL	0.7592 ± 0.0077	0.5443 ± 0.0127	0.5586 ± 0.0175	0.7338 ± 0.0116
scMoGNN	0.1674 ± 0.0000	0.0455 ± 0.0000	0.0130 ± 0.0000	0.0480 ± 0.0000
scDiffusion-X	0.6551 ± 0.0043	0.5319 ± 0.0057	0.5128 ± 0.0074	0.6590 ± 0.0032
GENOT	0.1297 ± 0.0049	0.0625 ± 0.0013	0.0616 ± 0.0007	0.1271 ± 0.0031
scButterfly	0.7921 ± 0.0146	0.6369 ± 0.0240	0.6677 ± 0.0271	0.7789 ± 0.0163
MultiVI	0.7463 ± 0.0061	0.7295 ± 0.0083	0.6774 ± 0.0105	0.7473 ± 0.0066
TransFlow	**0.9248 ± 0.0018**	**0.8886 ± 0.0033**	**0.9064 ± 0.0029**	**0.9239 ± 0.0019**

(b) RNA → ATAC

Model	Gen. Acc.	Gen. Bal. Acc.	Gen. Macro-F1	Gen. W-F1
BABEL	0.6510 ± 0.0074	0.4276 ± 0.0123	0.4007 ± 0.0245	0.5963 ± 0.0056
scMoGNN	0.1631 ± 0.0093	0.0692 ± 0.0024	0.0210 ± 0.0008	0.0596 ± 0.0008
scDiffusion-X	0.6877 ± 0.0120	0.5123 ± 0.0202	0.5281 ± 0.0255	0.6724 ± 0.0140
GENOT	0.1100 ± 0.0031	0.0577 ± 0.0024	0.0567 ± 0.0022	0.1149 ± 0.0036
scButterfly	0.8205 ± 0.0019	0.6384 ± 0.0062	0.6478 ± 0.0062	0.8037 ± 0.0022
MultiVI	0.8299 ± 0.0019	**0.7545 ± 0.0066**	0.7373 ± 0.0076	0.8244 ± 0.0019
TransFlow	**0.8385 ± 0.0037**	0.7352 ± 0.0056	**0.7759 ± 0.0036**	**0.8330 ± 0.0042**

Table 8: **Label-transfer metrics on PBMC10k.** Entries are mean ± sample standard deviation across multiple independent runs.

(a) ATAC → RNA

Model	Gen. Acc.	Gen. Bal. Acc.	Gen. Macro-F1	Gen. W-F1
BABEL	0.8285 ± 0.0454	0.7331 ± 0.0558	0.7402 ± 0.0597	0.8137 ± 0.0539
scMoGNN	0.1073 ± 0.0566	0.0714 ± 0.0000	0.0136 ± 0.0068	0.0241 ± 0.0179
scDiffusion-X	0.7322 ± 0.0140	0.6520 ± 0.0108	0.6468 ± 0.0077	0.7298 ± 0.0118
GENOT	0.2837 ± 0.0114	0.1856 ± 0.0125	0.1856 ± 0.0115	0.2859 ± 0.0106
scButterfly	0.5087 ± 0.0360	0.3580 ± 0.0414	0.3101 ± 0.0382	0.4132 ± 0.0376
MultiVI	0.7115 ± 0.0253	0.6314 ± 0.0058	0.5710 ± 0.0162	0.6477 ± 0.0290
TransFlow	**0.9833 ± 0.0015**	**0.9236 ± 0.0013**	**0.9160 ± 0.0013**	**0.9787 ± 0.0015**

(b) RNA → ATAC

Model	Gen. Acc.	Gen. Bal. Acc.	Gen. Macro-F1	Gen. W-F1
BABEL	0.8565 ± 0.0105	**0.7449 ± 0.0371**	**0.7545 ± 0.0393**	0.8454 ± 0.0128
scMoGNN	0.1447 ± 0.0014	0.0855 ± 0.0040	0.0279 ± 0.0012	0.0463 ± 0.0022
scDiffusion-X	0.7227 ± 0.0120	0.5861 ± 0.0173	0.6044 ± 0.0167	0.7145 ± 0.0092
GENOT	0.2767 ± 0.0076	0.1843 ± 0.0081	0.1862 ± 0.0094	0.2807 ± 0.0076
scButterfly	0.4267 ± 0.0023	0.3165 ± 0.0071	0.2286 ± 0.0083	0.3054 ± 0.0044
MultiVI	0.7562 ± 0.0184	0.6010 ± 0.0341	0.5694 ± 0.0378	0.7155 ± 0.0244
TransFlow	**0.8822 ± 0.0038**	0.7378 ± 0.0108	0.7372 ± 0.0114	**0.8593 ± 0.0048**

Table 9: **Real-test reference label-transfer scores.** Entries are mean ± sample standard deviation across multiple independent runs.

Dataset	Target modality	Accuracy	Balanced accuracy	Macro-F1	Weighted-F1
BMMC	RNA	0.8713 ± 0.0000	0.8235 ± 0.0000	0.8350 ± 0.0000	0.8690 ± 0.0000
BMMC	ATAC	0.8383 ± 0.0000	0.7409 ± 0.0000	0.7692 ± 0.0000	0.8325 ± 0.0000
PBMC10k	RNA	0.9370 ± 0.0000	0.8875 ± 0.0000	0.8961 ± 0.0000	0.9356 ± 0.0000
PBMC10k	ATAC	0.9105 ± 0.0000	0.8523 ± 0.0000	0.8632 ± 0.0000	0.9062 ± 0.0000