ICML 2026

scDataset

Scalable Data Loading for Deep Learning on Large-Scale Single-Cell Omics

Davide D'Ascenzo
University of Milan · Politecnico di Torino

Sebastiano Cultrera di Montesano
Broad Institute of MIT & Harvard

UNIVERSITÀ DEGLI STUDI DI MILANO
Politecnico di Torino
ERIC AND WENDY SCHMIDT CENTER AT BROAD INSTITUTE
ICML International Conference On Machine Learning

Single-cell atlases now exceed memory

scRNA-seq atlases have scaled to hundreds of millions of cells. They no longer fit in RAM, so training must read the data from disk.

100M

cells in Tahoe-100M, a perturbation screen of 50 cancer lines × 380 drugs × 3 doses

314 GB

on disk in compressed sparse AnnData; over 1 TB once converted to a dense format

14

experimental plates of ~7M cells, stored in order and never shuffled

Loading the data is a bottleneck

RANDOM SAMPLING

Diverse, but unusably slow

One random disk access per cell gives unbiased minibatches, but random I/O at atlas scale crawls at ~20 samples/sec*.

58 days

for a single epoch over Tahoe-100M (AnnLoader)

SEQUENTIAL STREAMING

Fast, but biased

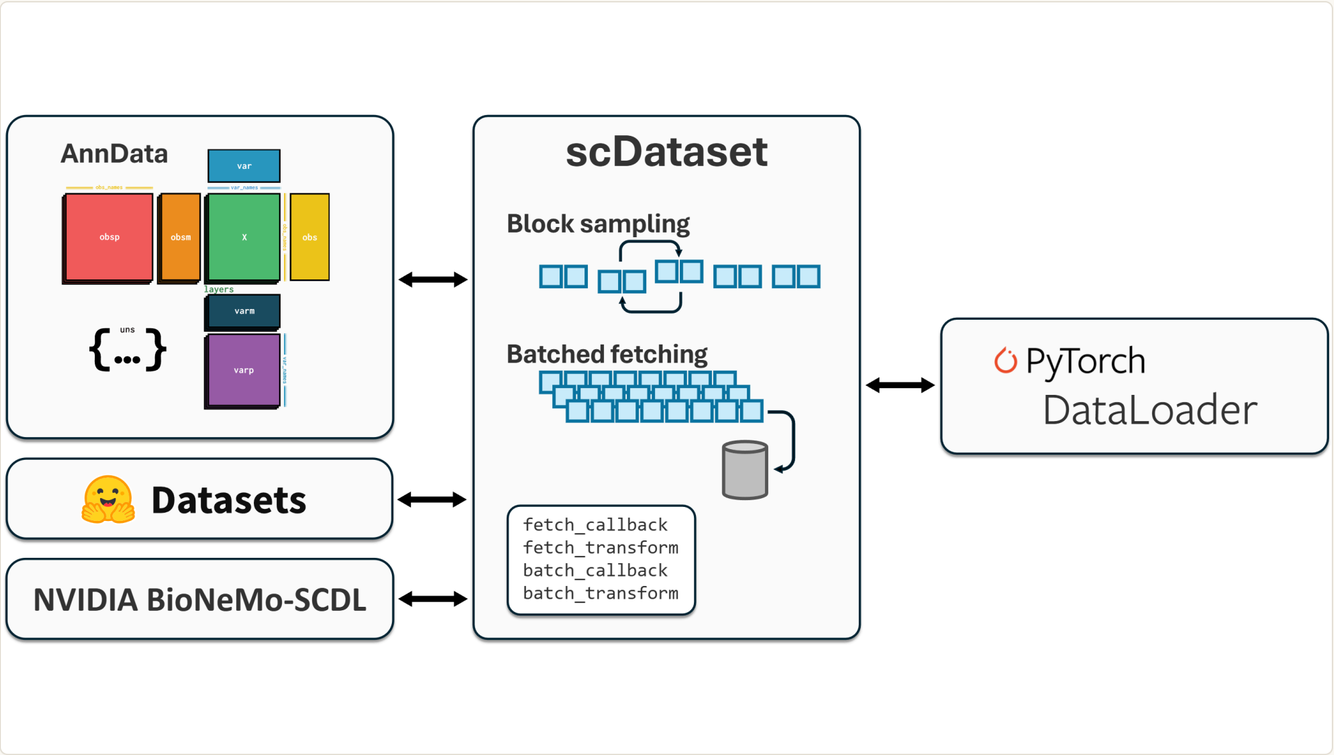
Reading cells in order is fast, but feeds the model one plate at a time.

Catastrophic

forgetting & model collapse at plate boundaries

* Throughput measured on an NVIDIA DGX Station.

scDataset: fast, randomized, and drop-in



A PyTorch IterableDataset that bridges any indexable backend to DataLoader via four callback hooks.

A quasi-random sampler that keeps the diversity of random sampling at near-streaming speed.

- AnnData-native, no format conversion
- Quasi-random sampling
- Memory-efficient & multi-worker

Two mechanisms: block sampling + batched fetching

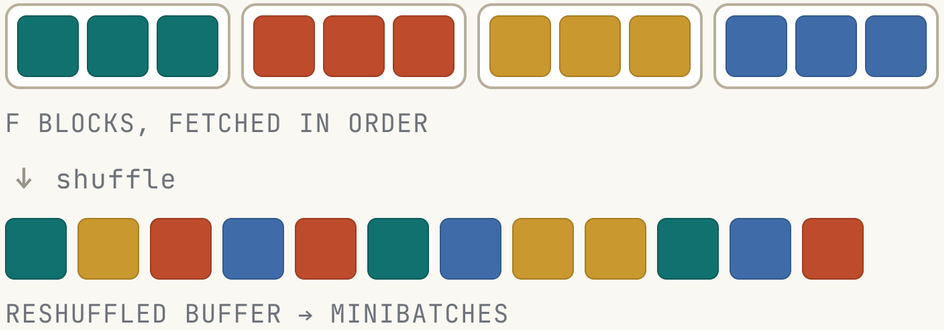
Block sampling

`m → ⌈m / b⌉ disk reads`

Sample contiguous **blocks of b = 3 cells** from the ordered data and read each block in a single sequential pass. Random disk reads fall from `m` to `m/b`.

Batched fetching

`fetch f blocks, shuffle in memory`



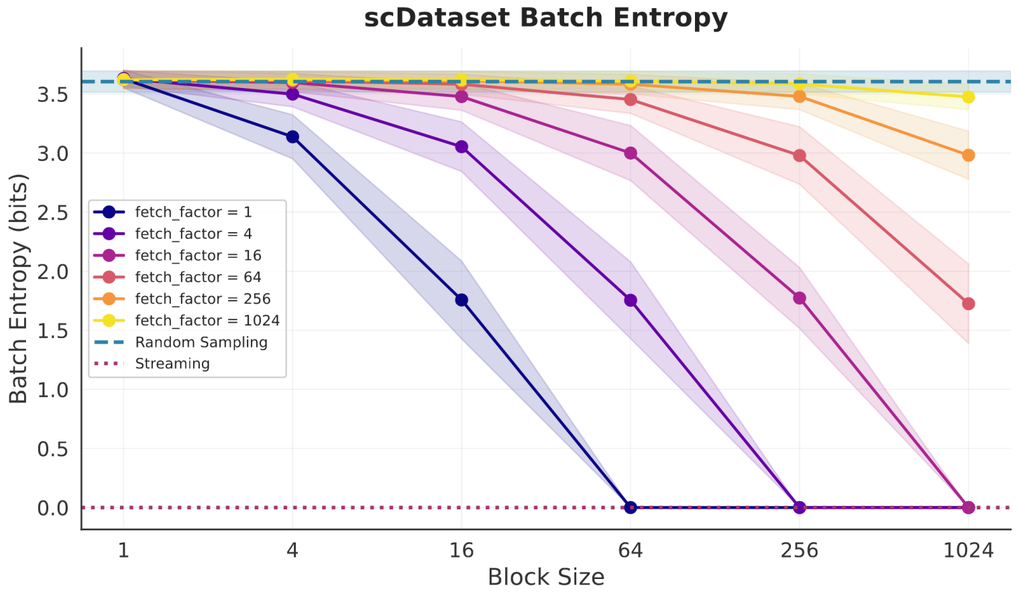
Fetch **f blocks** at once, shuffle the buffer in memory, then cut it into minibatches. Each minibatch now mixes cells from many blocks rather than one.

A provable bound on minibatch diversity

Measuring diversity as label entropy, the expected minibatch entropy is sandwiched around the true distribution $H(p)$:

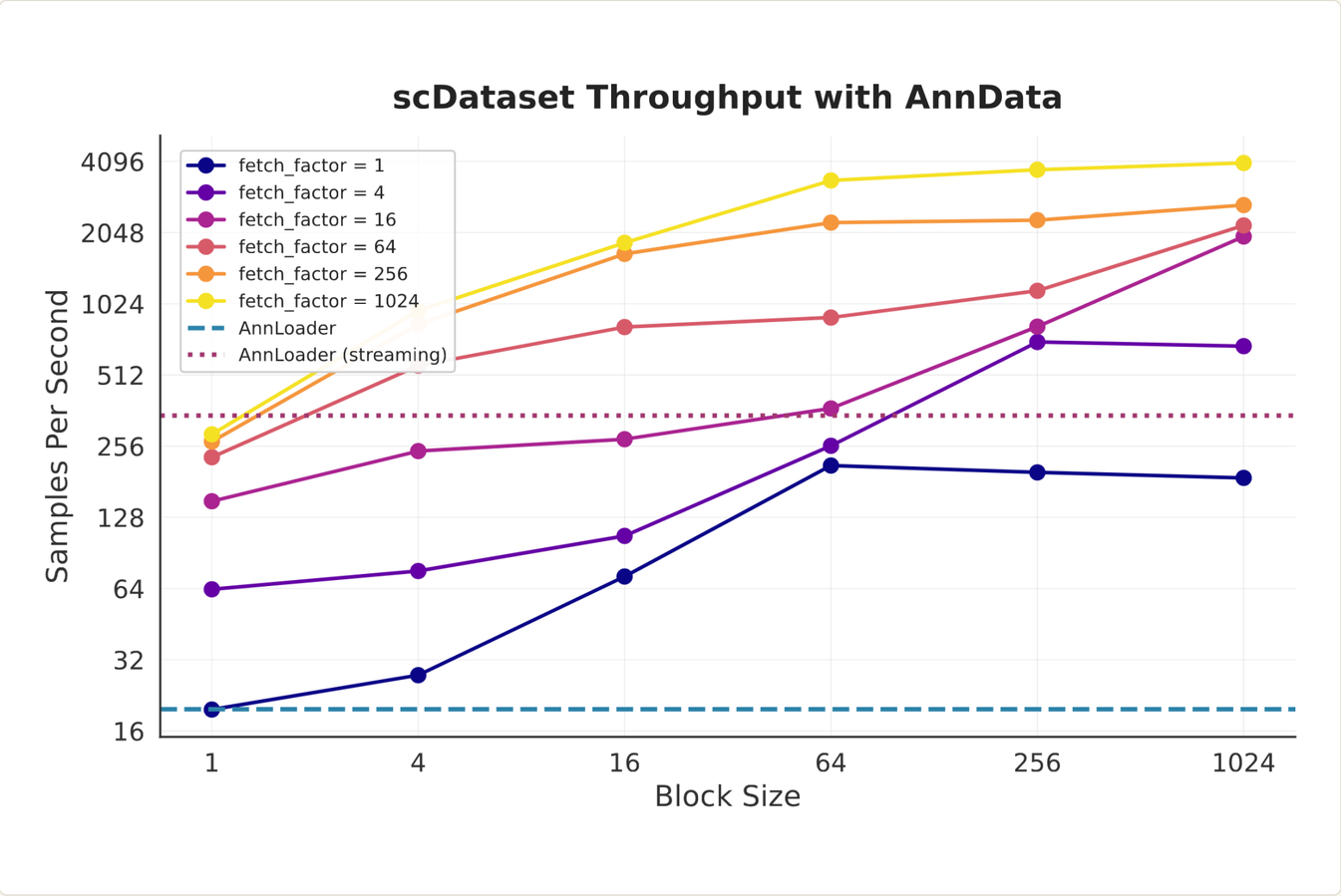
$$H(p) - \frac{(K-1)\ b}{2m\ \ln 2} \leq \mathbb{E}[H] \leq H(p) - \frac{K-1}{2m\ \ln 2}$$

The gap shrinks as the fetch factor grows. Practical guideline: $b \leq f / 2$ recovers near-random diversity, even on plate-ordered data.



Minibatch label entropy vs. block size, at increasing fetch factor f. Dashed line: random sampling.

Over two orders of magnitude faster



204×

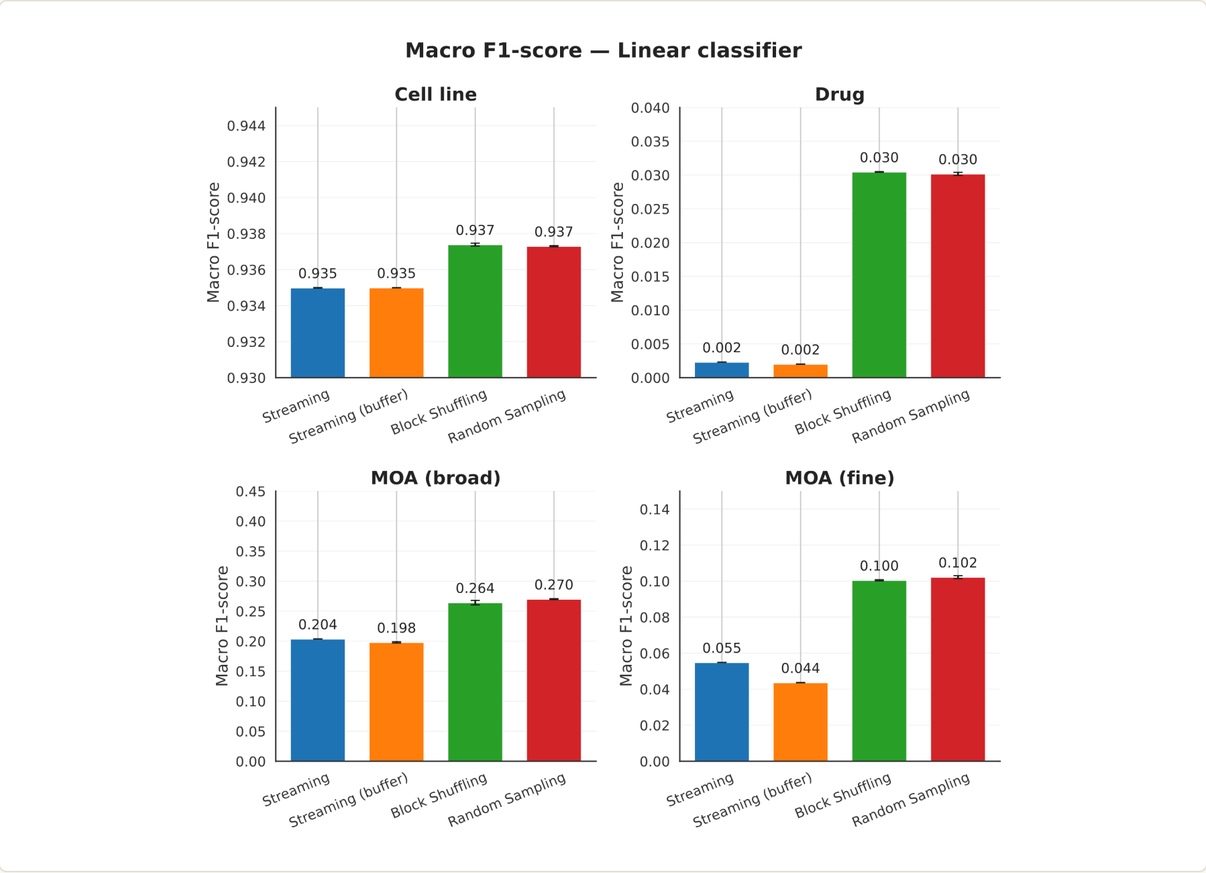
throughput over AnnLoader, working directly on AnnData, no conversion

47× / 25×

on the HuggingFace and BioNeMo backends

Single-core throughput on AnnData, sweeping block size and fetch factor.

scDataset matches true random sampling



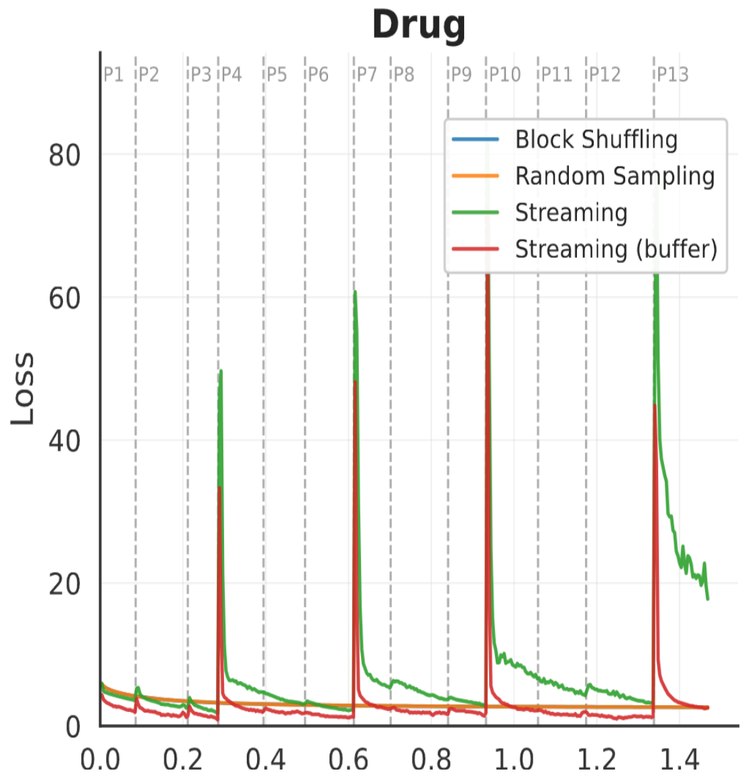
Macro F1 (mean ± std over 3 seeds) across four Tahoe-100M tasks; linear classifier.

Block shuffling (`b = 16, f = 256`) matches random sampling on every task, for both linear and MLP heads, with near-zero variance across seeds.

Both streaming variants keep up only on cell line, where the label is plate-invariant. On drug and mechanism of action they fall close to chance.

Block shuffling attains streaming throughput at random-sampling accuracy.

Sequential streaming destabilizes training



Drug-classification training loss. Dashed lines mark plate boundaries.

Each plate boundary is a distribution shift. Both streaming variants spike the loss at every boundary: the model forgets, then re-learns, plate after plate.

The usual fix is an in-memory shuffle buffer, `Streaming (buffer)`. But covering a plate would take a buffer of millions of cells, so in practice it spikes too. Block shuffling and random sampling stay flat.

0.002 streaming drug F1

0.030 block / random drug F1

TAKEAWAYS

Atlas-scale training on standard hardware

01 **Up to 204× the throughput** of the standard loader, working directly on AnnData files.

02 **Provably as diverse as random sampling**, and matches its accuracy on every task.

03 **Drop-in and scverse-native**: no conversion, and it works with HuggingFace, BioNeMo, LaminDB, and any custom backend.

```
pip install scdataset
```

github.com/scDataset/scDataset

davide.dascenzo@unimi.it · scultrer@broadinstitute.org

UNIVERSITÀ DEGLI STUDI DI MILANO

Politecnico di Torino

ERIC AND WENDY SCHMIDT CENTER AT BROAD INSTITUTE