SimCT: Recovering Lost Supervision for Cross-Tokenizer On-Policy Distillation

Simple Cross-Tokenizer On-Policy Distillation

Jie Sun

University of Science and Technology of China

October 2026

Outline

01 Background and Problem

Vocabulary and sequence mismatch discard supervision

02 SimCT: Method and Theory

Minimal aligned units, candidate scoring, and granularity

03 Results and Mechanism Analysis

Main results, supervision recovery, and training cost

On-Policy Distillation: Workflow and Loss

1 Student rollout
Sample a response to the prompt

2 Teacher feedback
Score the student-generated prefix

3 Student update
Minimize loss with a fixed teacher

With a shared tokenizer, compare next-token distributions at each student-generated prefix

$$L_{\mathrm{OPD}}(h) = \mathrm{KL}[p_{\mathrm{S}}(\cdot \mid h) \,\|\, p_{\mathrm{T}}(\cdot \mid h)]$$

h Prompt and student-generated text

$p_{\mathrm{S}}(\cdot \mid \mathrm{h})$ Student next-token distribution

$p_{\mathrm{T}}(\cdot \mid \mathrm{h})$ Teacher distribution at the same prefix

$L_{\mathrm{OPD}}(\mathrm{h})$ Distillation loss at this prefix

KL Divergence between two distributions

$\cdot$ All candidate tokens in the shared vocabulary

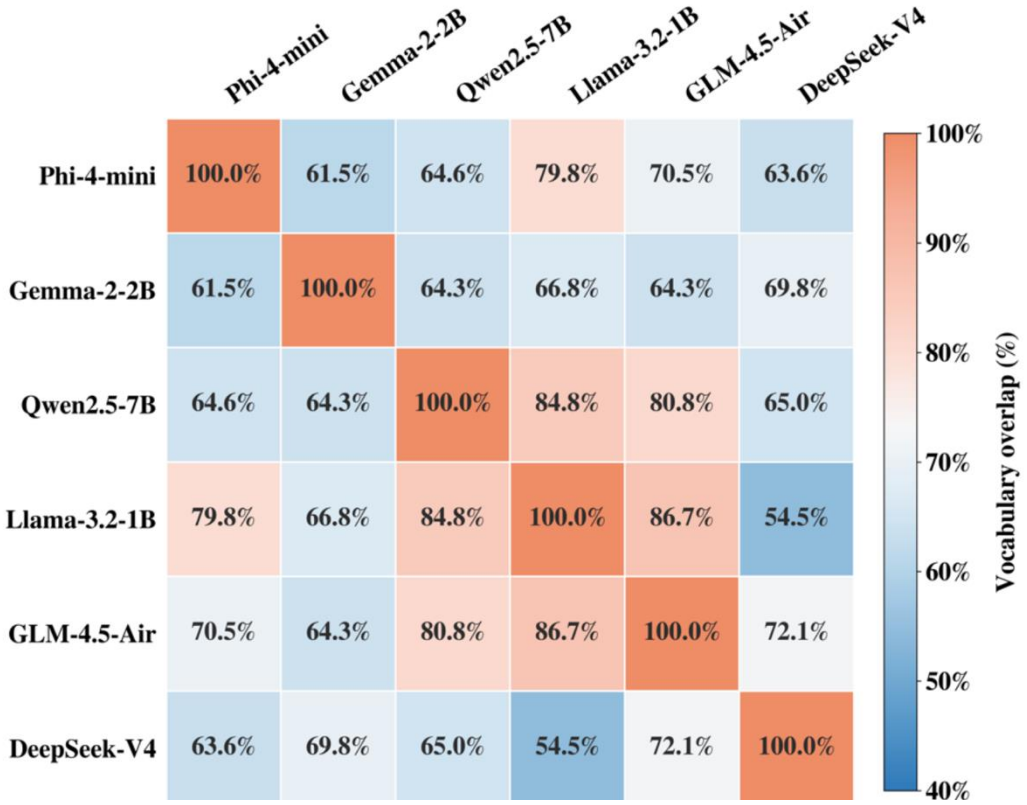
OPD provides teacher supervision on student-generated prefixes

Can teacher and student probabilities still be compared when tokenizers differ?

Tokenizer Mismatch

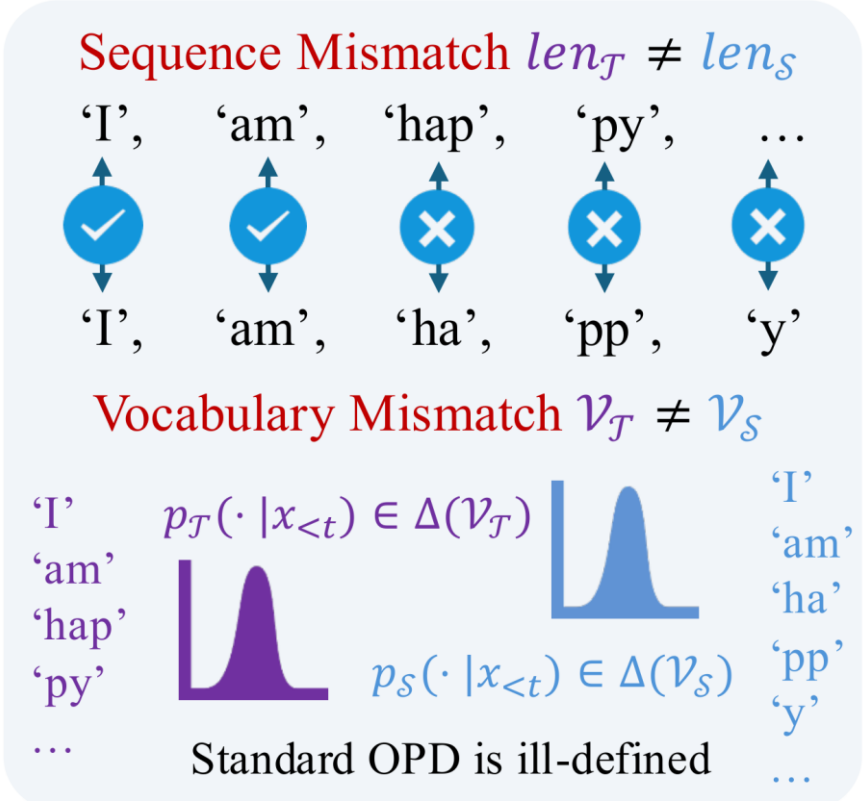
Vocabulary mismatch

Teacher candidates outside the shared vocabulary



Sequence mismatch

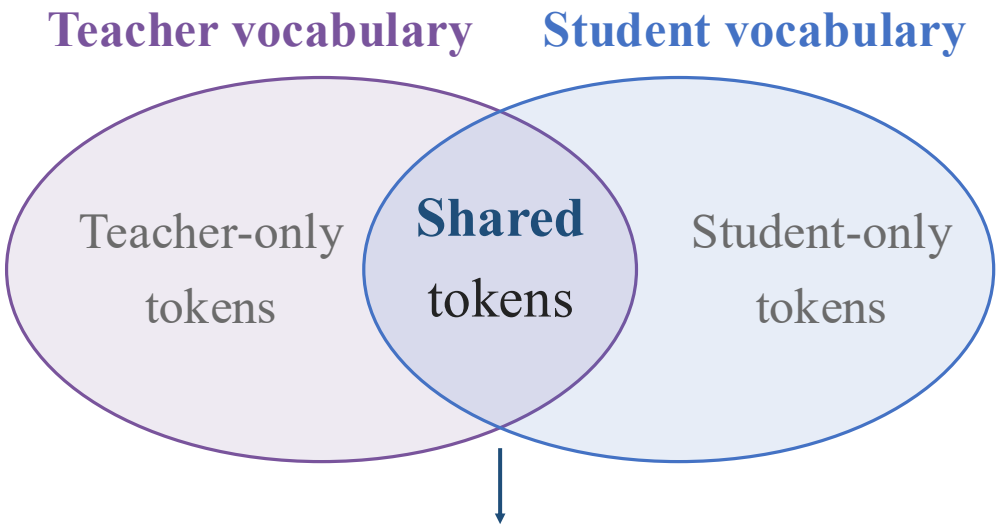
No one-to-one token alignment for the same text



Vocabulary and boundary differences make exact token matching discard usable teacher supervision

Limitations of Existing Supervision Interfaces

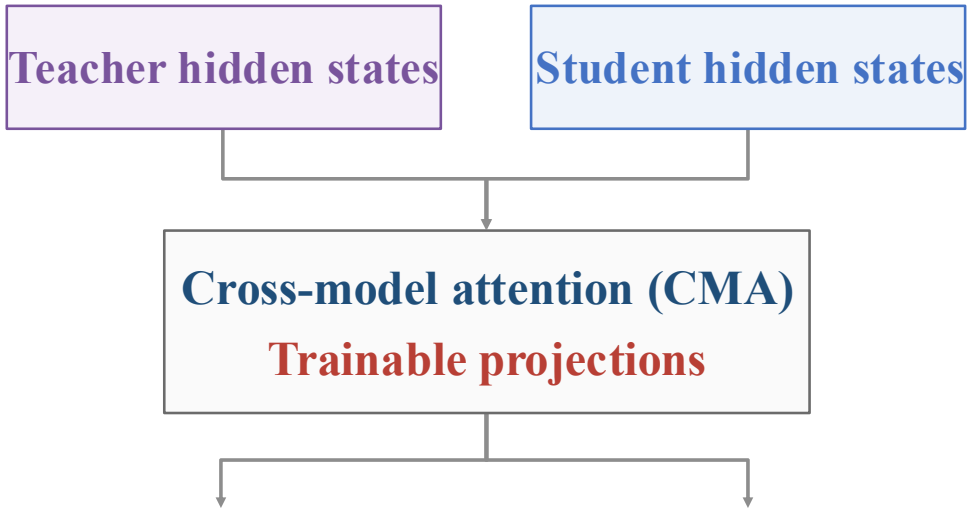
SimpleOPD: Shared tokens only



Compare only shared-token probabilities

Discards non-shared and unaligned supervision

DSKD: Learned cross-model mapping



Shared student head

Shared teacher head

Adds projection and alignment parameters

SimpleOPD restricts supervision. DSKD learns an additional cross-model mapping

Can existing tokenizer boundaries directly recover the missing supervision?

Outline

01 Background and Problem

Vocabulary and sequence mismatch discard supervision

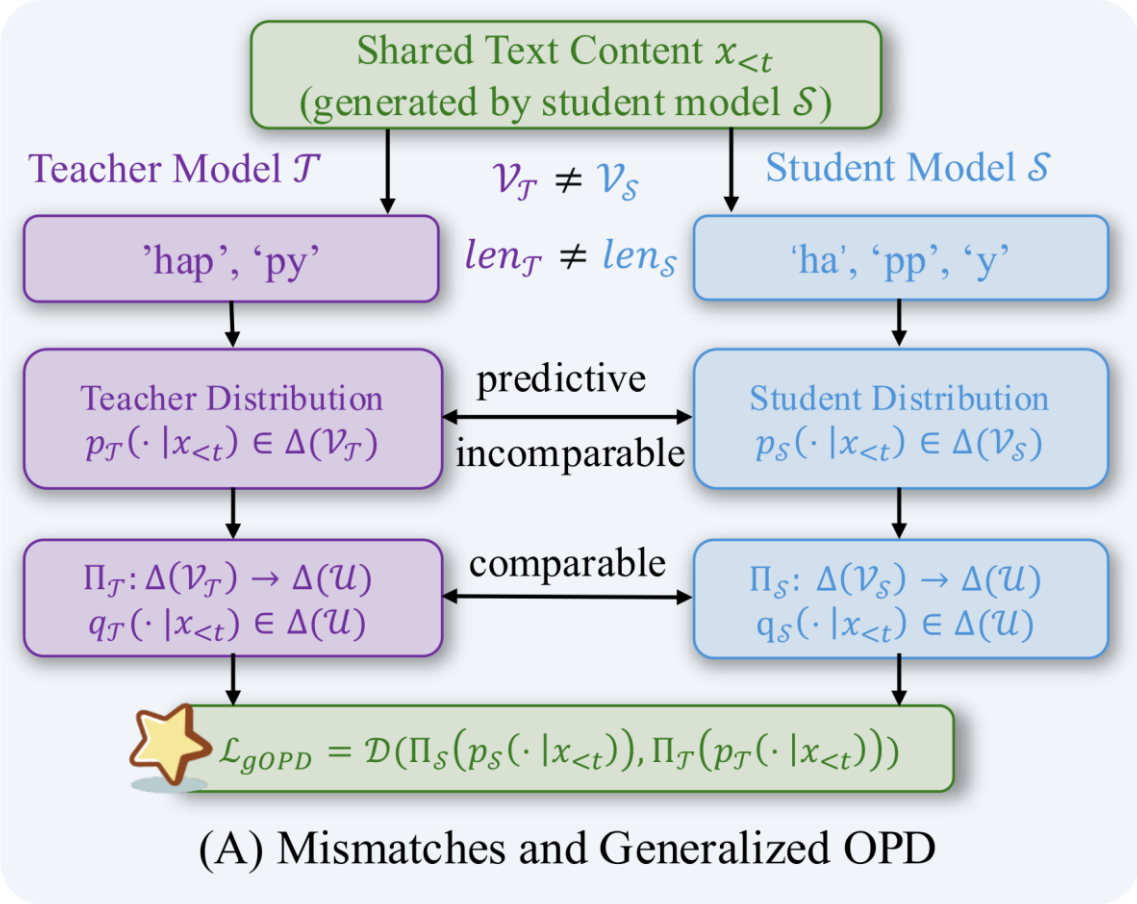
02 SimCT: Method and Theory

Minimal aligned units, candidate scoring, and granularity

03 Results and Mechanism Analysis

Main results, supervision recovery, and training cost

SimCT: A Common Supervision Space



(A) Mismatches and Generalized OPD

Retained + recovered supervision

$$U_{\text{SimCT}} = (V_{\text{T}} \cap V_{\text{S}}) \cup A$$

U_{SimCT} Common candidate supervision space

V_{T} / V_{S} Teacher / student vocabularies

$V_{\text{T}} \cap V_{\text{S}}$ Existing shared tokens

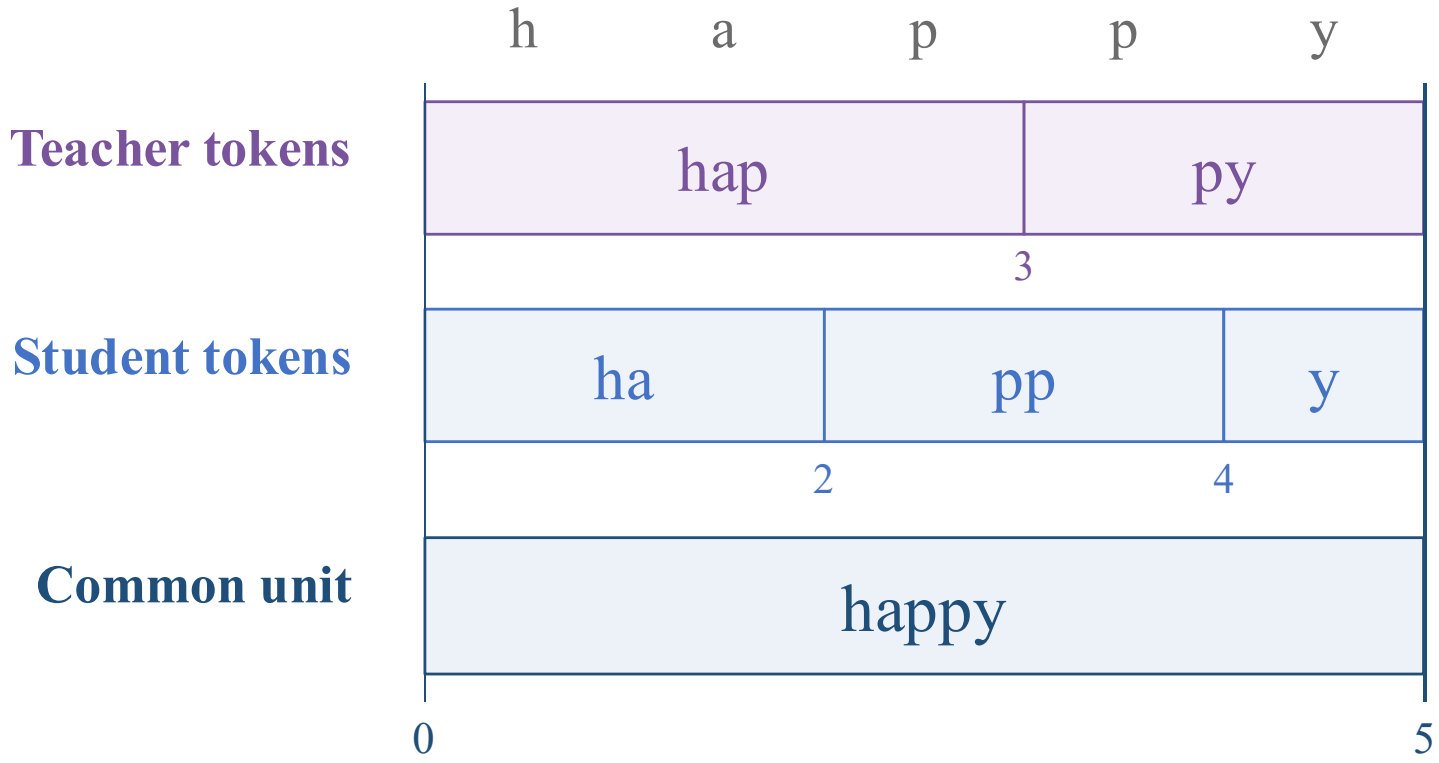
A Set of minimal aligned units

Recover the same text across different tokenizations

Keep shared-token supervision and recover the same text under different tokenizations

Constructing Minimal Aligned Units

Align two tokenizations of “happy” by character position



Split at shared boundaries

Shared boundaries: 0 and 5

Cuts at 2, 3, or 4 split a token in at least one tokenizer

Thus, “happy” is one MAU

An MAU is the finest span made of complete tokens under both fixed tokenizations

Any further split cuts a token in at least one tokenizer

Scoring the Same Unit with Two Models

The same candidate unit: u = happy

Teacher T: hap | py

Average 2 conditional log-probabilities

Student S: ha | pp | y

Average 3 conditional log-probabilities

$$s_M(u \mid h) = (1/k) \sum_{j=1,\ldots,k} \log p_M(v_j \mid h, v_{<j})$$

M Teacher T or student S

s_M Score of candidate unit u

u The same text unit to be scored

h Prompt and student-generated text

k Number of tokens in M's realization of u

v_j The j-th token within the unit

$v_{<j}$ Preceding tokens within the unit

p_M Model M's conditional token probability

Both models score the same text using their own token sequences

Shared tokens use log-probability. Multi-token units use mean log-probability

Candidate Scores and OPD Training

1 Student rollout → **2 Shared candidates** → **3 Candidate scoring** → **4 Filter and update**

Use student prefixes | Shared tokens + MAUs | Normalize on the same set | Keep the teacher fixed

Scores define candidate distributions

$$q_M(u \mid h) = \exp(s_M(u \mid h)) / Z_M(h)$$

$Z_M(h)$ Sum of exp(score) over all candidates

q_M Distribution normalized over the candidate set

Position filter: include a position in the loss only when $1 / Z_T(h) \leq 2$

Loss in the common space

$$L_{SimCT}(h) = KL[q_S \parallel q_T]$$

q_S / q_T Student / teacher distributions

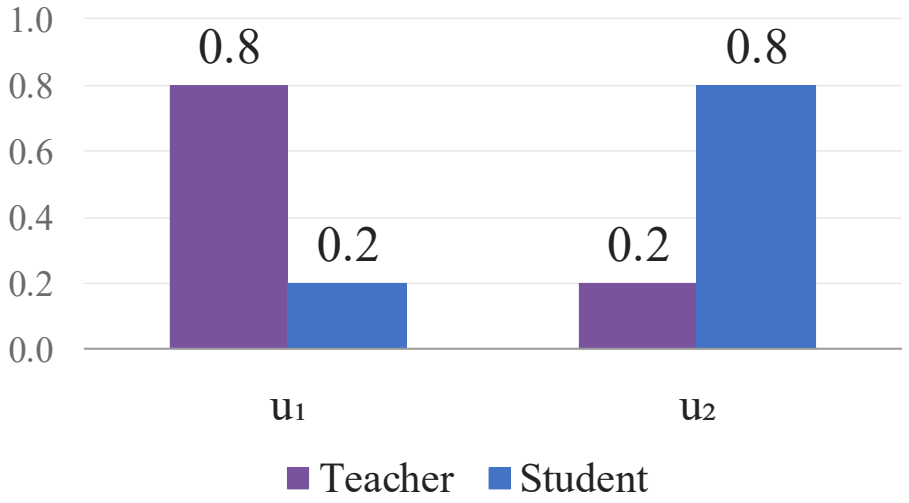
Aggregate retained positions to update the student

Normalize candidate scores into comparable distributions, then apply the OPD loss

Why Minimal Supervision Granularity Matters

Example: group two candidates and sum their probabilities within the group

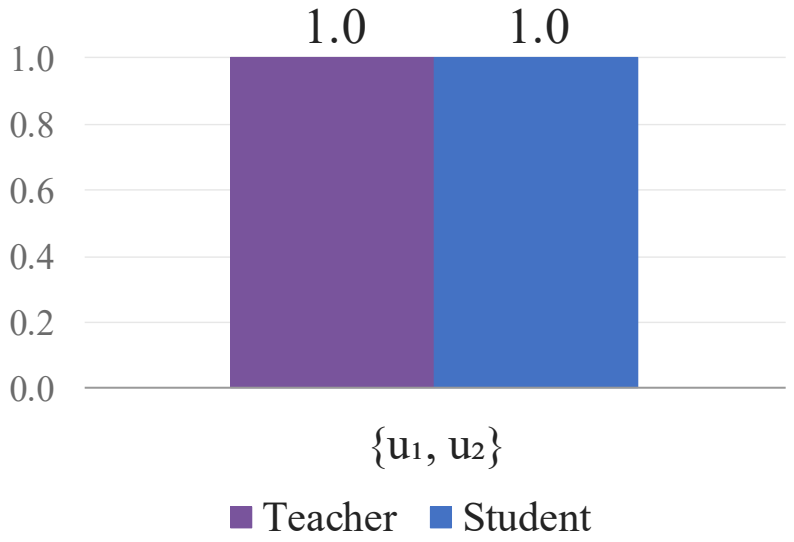
Before merging: two units



KL > 0

Sum probabilities

After merging: one group



KL = 0

$$\mathrm{KL}(q_S \,\|\, q_T) \geq \mathrm{KL}(Q_S \,\|\, Q_T)$$ q: fine distribution Q: grouped distribution

Coarsening can erase teacher–student preferences within a group

A smaller KL may reflect lost supervision signal

Outline

01 Background and Problem

Vocabulary and sequence mismatch discard supervision

02 SimCT: Method and Theory

Minimal aligned units, candidate scoring, and granularity

03 Results and Mechanism Analysis

Main results, supervision recovery, and training cost

Experimental Setup

Three teacher–student pairs with different tokenizers

Teacher	Student
Qwen2.5-7B-Instruct	Gemma-2-2B-IT
Qwen2.5-7B-Instruct	Phi-4-mini-Instruct
Phi-4-mini-Instruct	Gemma-2-2B-IT

Benchmarks and baselines

Math: GSM8K and MATH-500

Code: MBPP and LiveCodeBench-v6

SFT, SimpleOPD, ALM, GOLD, DSKD

Common training and evaluation protocol

Same SFT start within each pair and seed

Identical training setup and hardware

5 training seeds, one evaluation per checkpoint

Report pass@1 mean and standard deviation

Compare math and code performance across three pairs under a common training protocol

Main Results across Models and Tasks

Mean pass@1 over four benchmarks (%)

Method	Qwen → Gemma	Qwen → Phi	Phi → Gemma
SFT	30.94	51.30	31.11
SimpleOPD	31.12	52.02	31.49
ALM	31.42	53.41	31.62
GOLD	31.98	53.90	32.07
DSKD	32.00	53.88	31.85
SimCT	**32.80**	**55.06**	**32.58**

Mean relative gain over SFT

+6.0%

Qwen → Gemma +6.0%

Qwen → Phi +7.3%

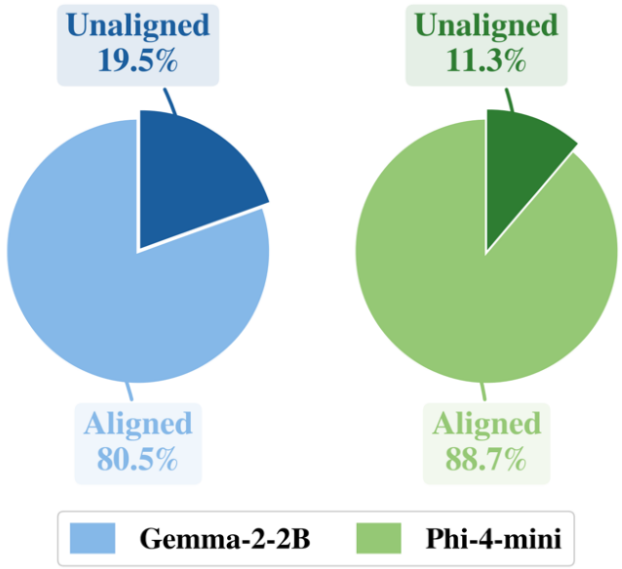
Phi → Gemma +4.7%

Average four benchmarks per pair, then average the three relative gains over SFT

SimCT achieves the best student result on all four benchmarks for all three model pairs

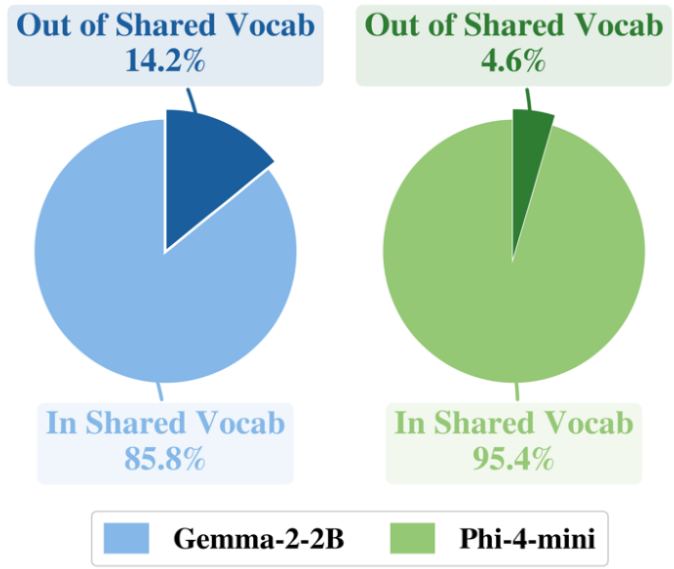
Both Sources of Lost Supervision Improve Learning

Sequence mismatch



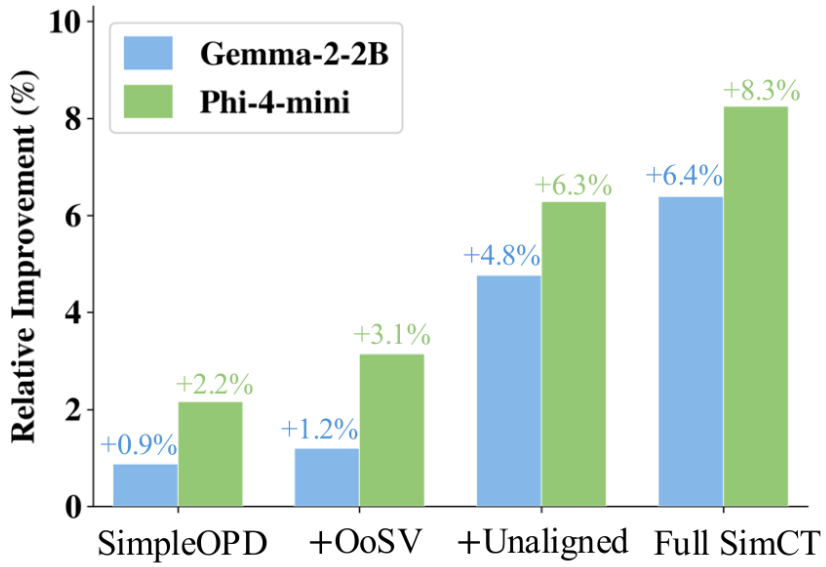
(A) Token Alignment Ratio

Outside the shared vocabulary



(B) Top-100 Shared Vocab Coverage

Partial and full recovery



(C) Supervision Recovery Decomposition

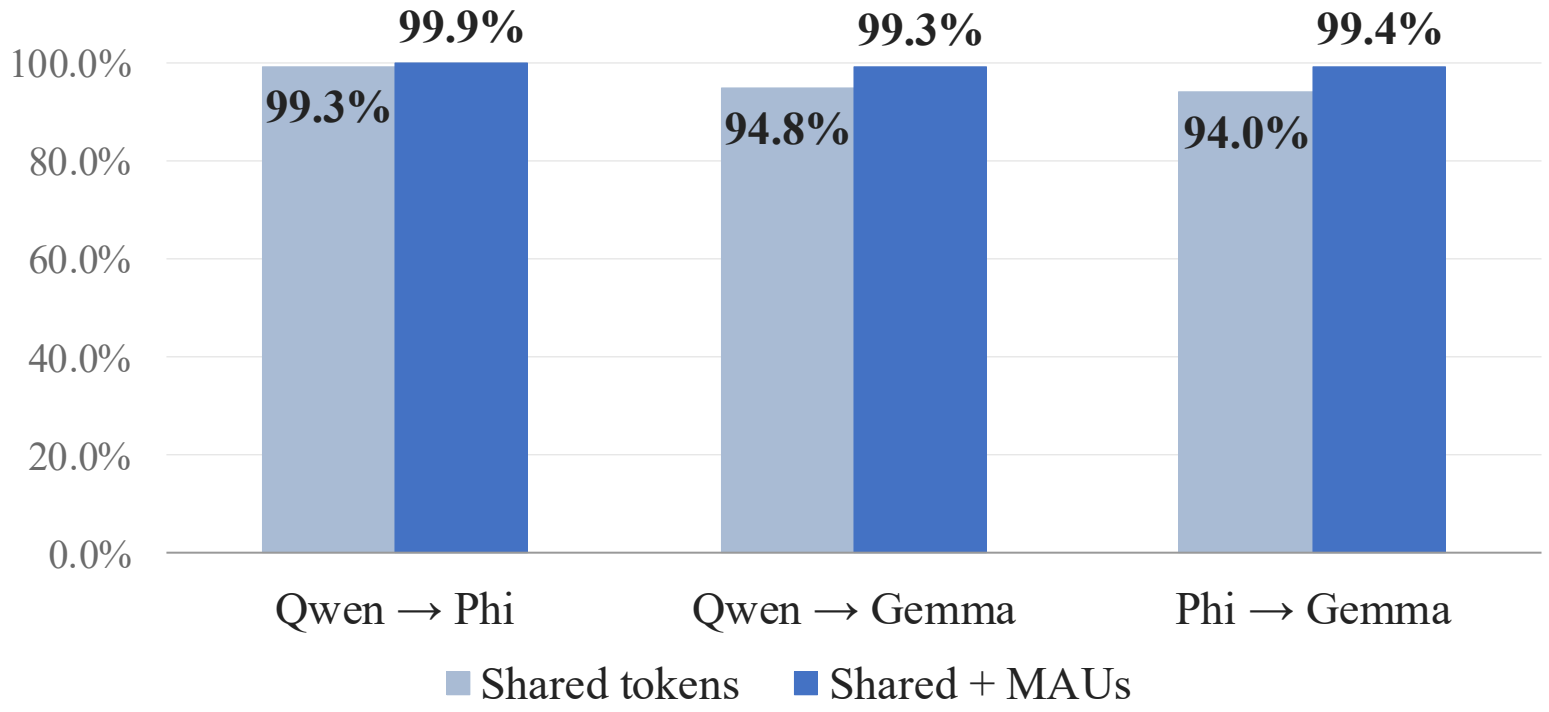
OoSV: non-shared vocabulary supervision at aligned positions. Unaligned: supervision from non-1:1 units.

Right: independent ablation runs, with improvement relative to the unadapted Base student.

Both discarded signals help. Full SimCT performs best by recovering both

MAUs Increase Teacher Supervision Coverage

Coverage of the original teacher next-token probability mass



Mean coverage over all positions

95.6% → 99.5%

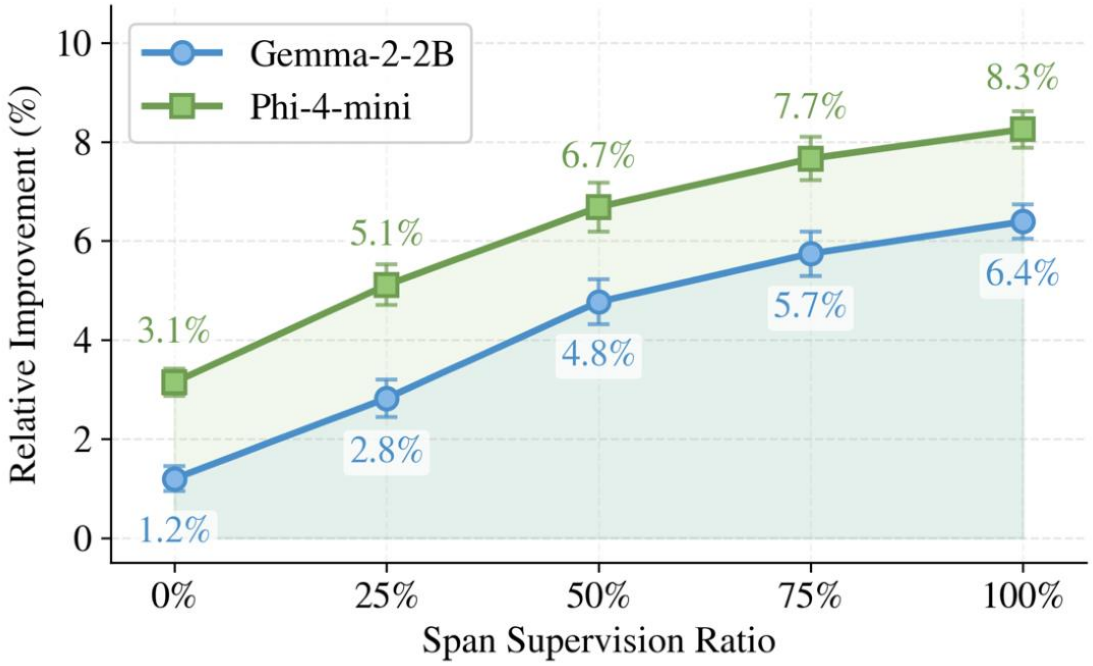
With MAUs, mean uncovered mass is 0.5%

263,817 on-policy rollout positions

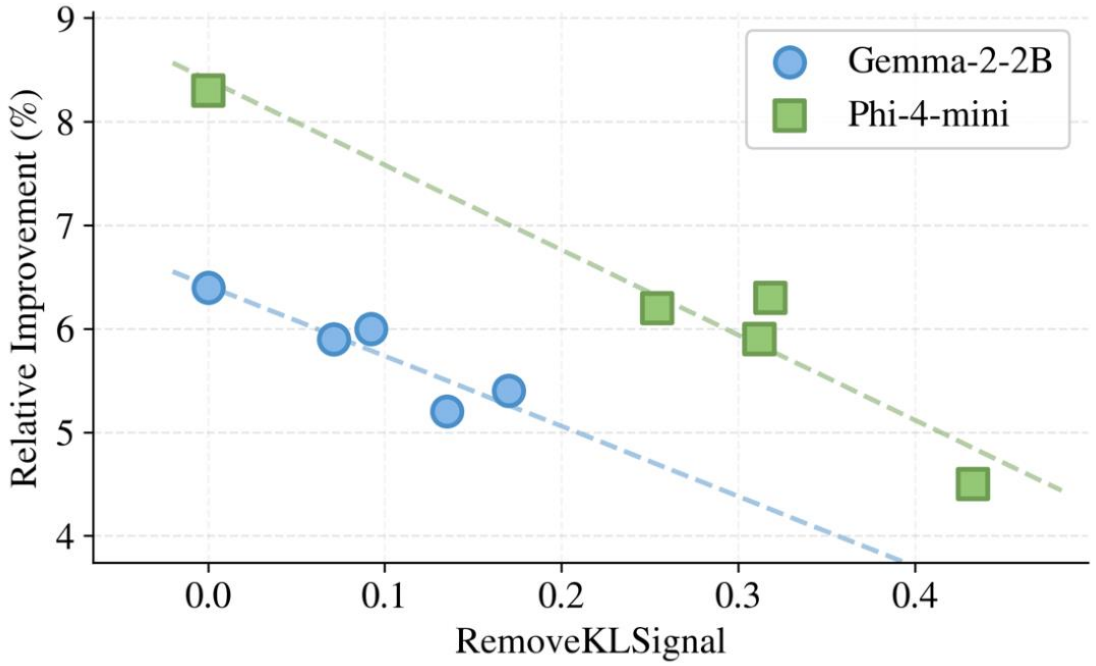
MAUs raise mean original teacher support coverage from 95.6% to 99.5%

Ablations on MAU Coverage and Granularity

Recovering more MAU supervision



Preserving fine-grained differences



Left: 0% MAU retains OoSV. Right: the x-axis measures KL signal removed by coarsening.
Independent ablations relative to Base. Error bars show standard deviations over 5 training runs.

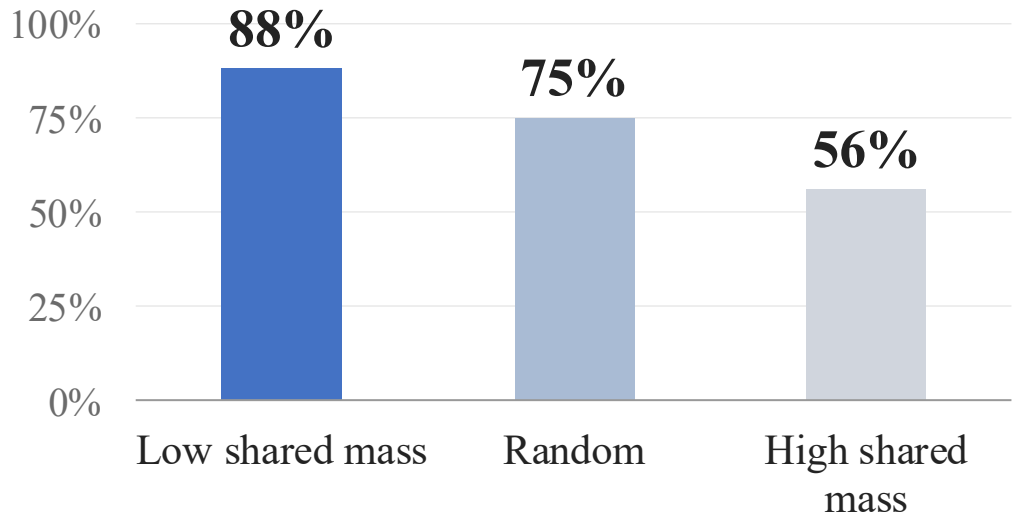
More recovered supervision improves performance. Coarsening removes useful local differences

Where Recovered Supervision Helps Most

Qwen → Gemma: compare recovery of the full gain using equal-size MAU subsets

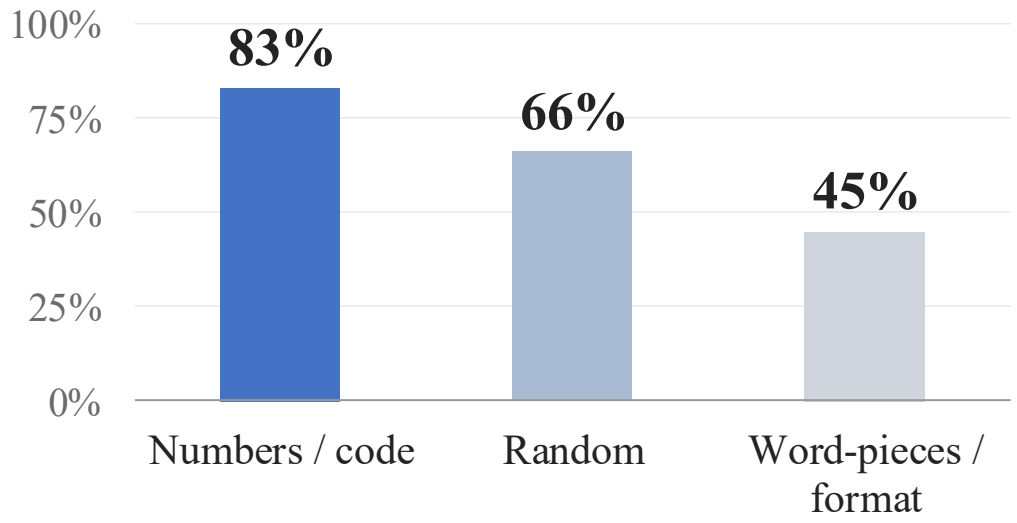
Positions with weak shared-token coverage

Each subset uses 50% of MAUs



Numbers, code, and operators

Each subset uses 37% of MAUs



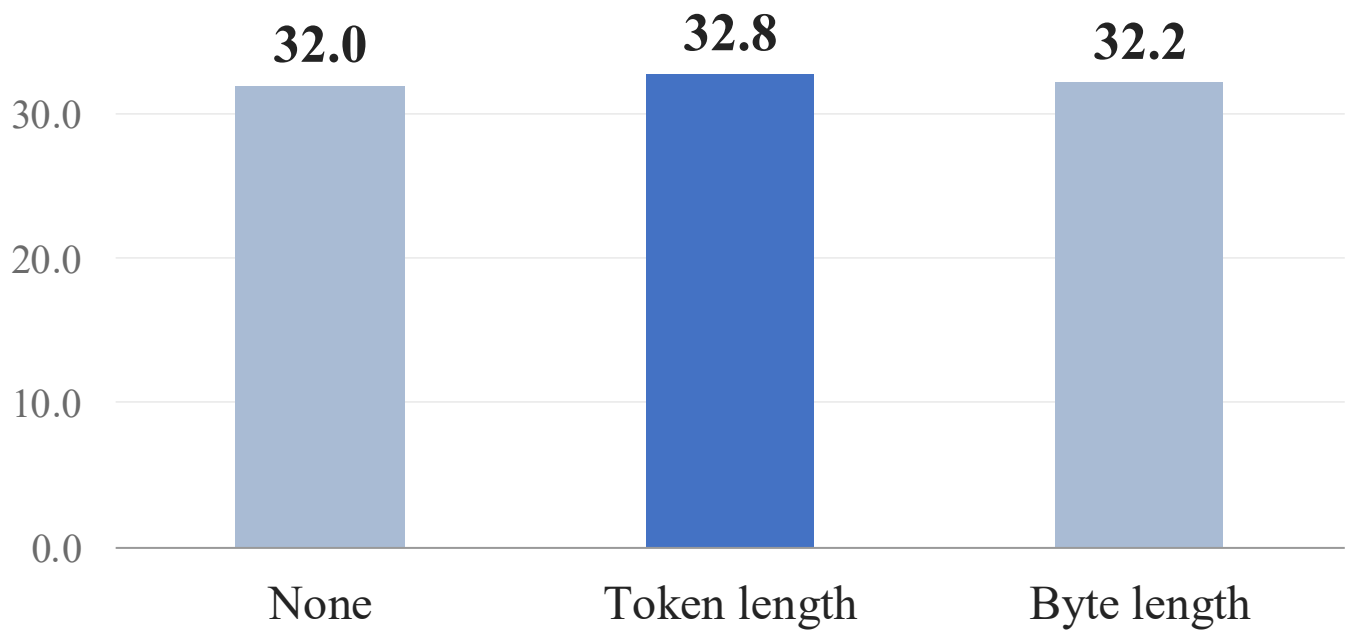
Gain recovery: SimpleOPD = 0%, Full SimCT = 100%. Each chart uses a matched MAU budget.

Weak shared-token coverage and math/code content identify more useful recovered supervision

Ablation on Token-Length Normalization

Qwen → Gemma: vary only the length normalization used to score each MAU

Mean pass@1 over four benchmarks (%)



Mean token log-probability

$$s_M(u \mid h) = \ell_M(u \mid h) / k_M(u)$$

ℓ_M: sum of token log-probabilities

k_M: token count under model M

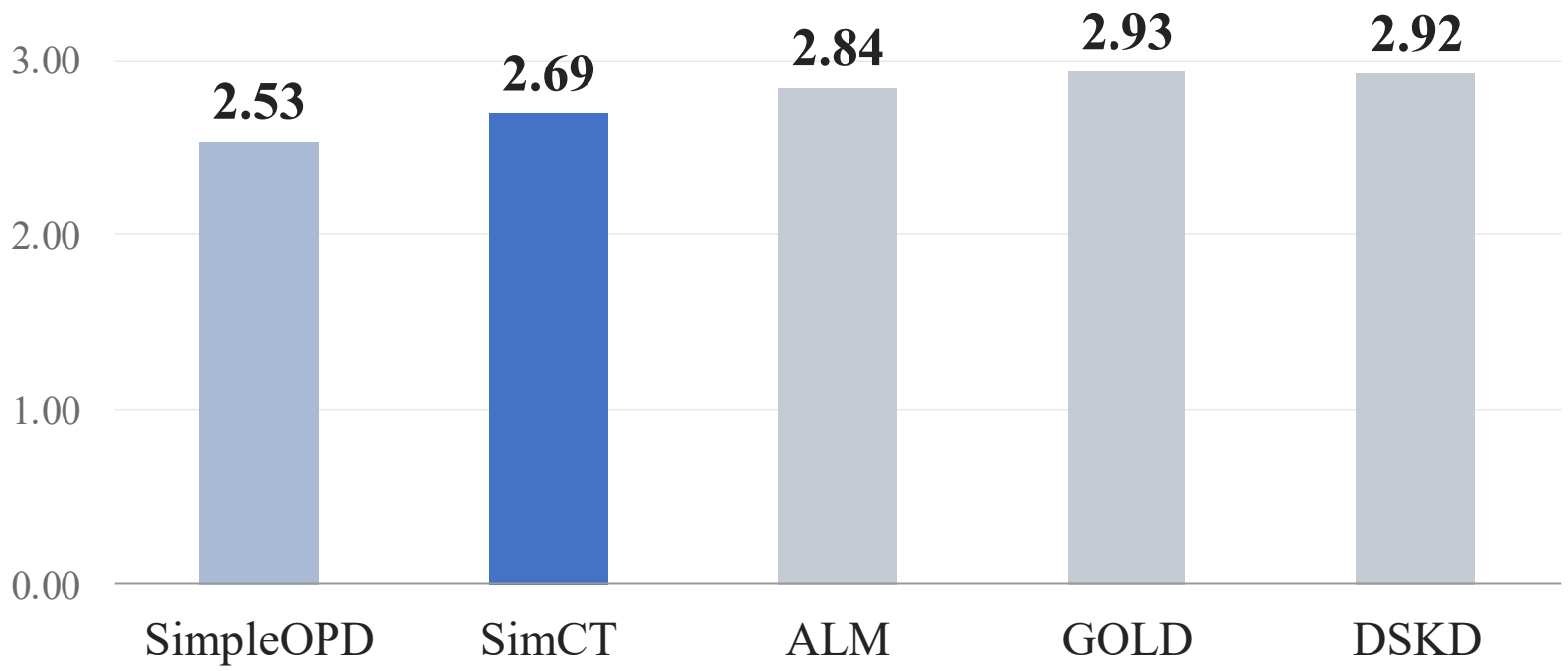
vs. no normalization: +0.8 pts

vs. byte normalization: +0.6 pts

Token-length normalization performs best on all four benchmarks for this model pair

Training Cost Relative to SimpleOPD

Mean end-to-end OPD training time across three model pairs (hours)



Relative to SimpleOPD

+6.3%

An extra 0.16 hours

About 9.6 minutes

No trainable mapping parameters

Same protocol on 8 × NVIDIA H20. Timing includes rollouts, teacher forward passes, alignment, and updates.

In this setup, SimCT recovers more supervision with about 6.3% additional training time

Summary

Problem: Tokenizer mismatch discards useful supervision

Non-shared predictions and text without one-to-one token alignment
both contain useful teacher feedback.

Method: Shared tokens + minimal aligned units

Score the same text under both models and apply OPD in a common space.
Keep the finest common granularity to retain local preference differences.

Results: Better math and code performance at modest extra cost

Mean relative gain over SFT: 6.0%. Original teacher support coverage: 99.5%.
Training time increases by 6.3% relative to SimpleOPD.

SimCT recovers lost supervision and preserves comparable, fine-grained differences