Dense Cross-Tokenizer Distillation via Semantic Optimal Local Alignment

Mengyu Zheng[1] **Zheyuan Bai**[3] **Yuchuan Tian**[2] **Chong Zhu**[3]
Tianyu Guo[3] **Hanting Chen**[3*] **Yunhe Wang**[3*]

[1]Beijing Jiaotong University [2]Peking University [3]Huawei Noah's Ark Lab
myzheng@bjtu.edu.cn {chenhanting,yunhe.wang}@huawei.com

Abstract

Cross-tokenizer knowledge distillation lacks a shared coordinate system between teacher and student vocabularies, and existing methods each drop one of the two ingredients comparison needs: shape-based objectives preserve density but discard token identity, lexical alignment restores identity but uses surface form as a proxy for meaning, and text-space matching compresses the teacher signal into chunk-level events. We introduce SOLA (Semantic Optimal Local Alignment), which reconstructs the missing coordinate system explicitly: it partitions inputs into decoded-text blocks via Minimal Complete Correspondence, estimates a cross-vocabulary correspondence cost from calibration text, and runs entropic optimal transport on a bidirectional local support retrieved around each teacher prediction. Under matched training and decoding protocols, SOLA outperforms direct cross-tokenizer baselines on Dolly instruction following, an UltraChat general-instruction benchmark, and code and math domain transfer. Component ablations show gains from successive construction stages, and method-specific diagnostics across six methods show a positive coverage–score association.

1 Introduction

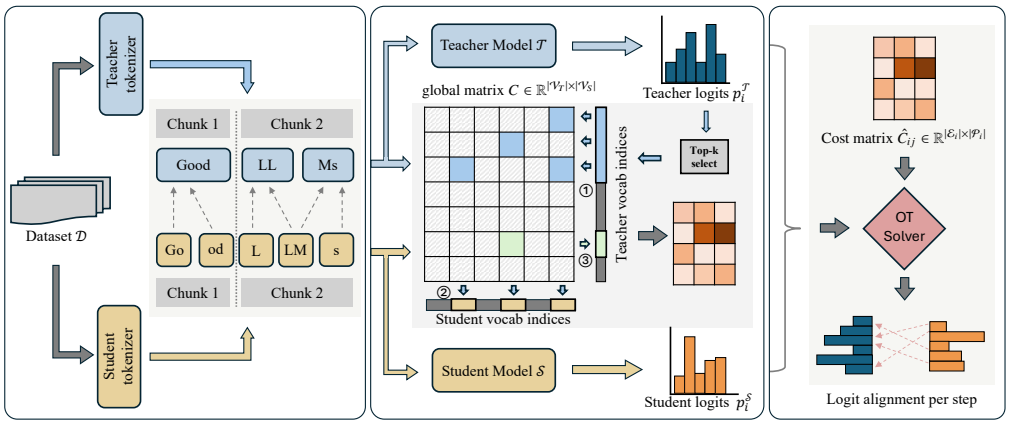
When teacher and student language models [53, 37] use different tokenizers, standard knowledge distillation [20, 25, 48, 56] breaks: the two predicted distributions no longer share a coordinate system. This case is now the rule rather than the exception. LLaMA [42], Qwen [50], Gemma [41], GPT-style, and domain-specific model families use different vocabularies, segmentation rules [30, 40, 54], normalization conventions, and byte/subword boundaries. The resulting cross-tokenizer setting changes the probability space on which distillation is defined.

The core difficulty is a mismatch of reference frames, meaning the tokenizer-indexed probability coordinates. A teacher distribution is supported on $\mathcal{V}_T$, while the student distribution is supported on $\mathcal{V}_S$; when $\mathcal{V}_T \neq \mathcal{V}_S$, the i-th coordinate of one logit vector need not refer to the same linguistic unit as the i-th coordinate of the other. The problem is compounded along the sequence axis: the same decoded string may be represented as one token by one tokenizer and as several subwords by another. Thus, cross-tokenizer distillation must solve two problems at once: it must synchronize teacher and student positions, and it must decide which teacher-side probability mass can meaningfully supervise which student-side tokens.

Existing approaches each address part of this problem and pay for it elsewhere. Shape-based and rank-based cross-tokenizer methods such as ULD and MultiLevelOT [7, 16] avoid index matching by comparing only sorted-shape statistics, which preserves probability density but discards token identity, since two probability vectors can have the same sorted shape while assigning mass to

*Corresponding Authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).



(a) MCC and Intra-Chunk Alignment (b) Data-Driven Semantic Cost Construction (c) OT Optimization

Figure 1: SOLA aligns heterogeneous teacher and student token spaces by first synchronizing their decoded text into local correspondence blocks, then constructing a sparse cross-vocabulary support on which dense logit distillation is performed. This support lets the student inherit both the teacher's active probability mass and the local alternatives that make the distribution informative.

unrelated tokens. Lexical alignment methods such as MinED [44] and CDM [9] restore explicit teacher–student correspondences, but edit distance [32] is an unreliable proxy for meaning, especially for short or ambiguous subwords. Text-space methods such as ALM [35] sidestep cross-vocabulary logits altogether by comparing decoded chunks, trading the teacher's dense distribution for a sparser chunk-level signal. Each of these failure modes drops one of the two ingredients distillation needs. Identity-level correspondence means matching token realizations rather than probability ranks; it tells the student which teacher alternatives to imitate; distributional density tells it how much mass each alternative carries. Identity without density reduces to top-1 transfer, discarding the alternatives that make distillation informative. Density without identity supervises the student's mass with teacher signals that may refer to different linguistic units. The cross-tokenizer setting therefore demands a comparison object that preserves both—exactly the requirement existing methods give up.

Recovering both ingredients requires reconstructing the missing local coordinate system before any distribution comparison is attempted. SOLA (Semantic Optimal Local Alignment) does this in three derived steps. The sequence axis is synchronized first, by partitioning teacher and student tokenizations along decoded-text-equality boundaries (Minimal Complete Correspondence, MCC), so subsequent comparisons take place within blocks with matching decoded text. Inside each block, identity-level correspondence is supplied by a data-derived cross-vocabulary cost estimated from calibration text, with hub correction to suppress generic high-frequency matches. This cost estimates context-marginalized calibration associations. Each training position then activates a compact local support around the teacher's currently high-mass tokens, on which SOLA runs an entropic optimal transport problem. The result keeps the dense logit information that shape and rank objectives have, restricts attention to plausible correspondences in the way lexical methods aim for, and avoids the chunk-level compression of text-space methods.

Under matched training and evaluation protocols across three settings (Dolly instruction-following, an UltraChat-style general-instruction benchmark, and code and math domain transfer), SOLA outperforms direct cross-tokenizer baselines including the strongest concurrent method, SeDi. Component ablations under a matched training recipe show gains from successive construction stages, and method-specific diagnostics across six methods show a positive coverage–score association.

2 Related Work

Cross-tokenizer KD methods differ mainly in how they handle the absent shared probability support.

Shape and lexical methods. Shape-based objectives compare probability ranks or sorted mass and treat tokenizer mismatch as invariance to token identity (ULD [7], MultiLevelOT [16]); they preserve dense mass but allow arbitrary identity-level mismatches. Lexical methods recover explicit pairs through edit distance [32] or DTW-style [4] monotone alignment (MinED [44], CDM [9]); they

restore identity but let surface form stand in for meaning, and they conflate sequence-level positioning with token-level correspondence by running monotone alignment over the full token sequence. SOLA's MCC partition separates the two: decoded-text-equality boundaries handle sequence-level synchronization, leaving vocabulary-level correspondence as a separate dense problem inside each block, and the entropic OT on a constructed local support replaces the uniform-cost or surface-form assumptions both families rely on.

Text-space matching and tokenizer adaptation. ALM [35] moves comparison into text space by matching the probability of decoded chunks, trading dense logits for chunk-level events. TokAlign [33] adapts the student vocabulary before applying token-level KD. Both pay a representational cost (compressing the teacher signal or modifying the student) that SOLA avoids by performing dense matching on a constructed local support with the original tokenizers.

Concurrent CTKD and other distillation objectives. The closest concurrent work is SeDi [45] (distribution-aware cross-vocabulary transfer) and DSKDv2 [52] (dual-space projection with exact token alignment). SOLA differs by separating three objects often coupled together: a reusable empirical correspondence cost, instance-level teacher and student marginals, and an information-projection objective on a bidirectionally retrieved local support. This separation gives the ablation handles a single dual-space projection lacks. Broader LLM distillation objectives such as reverse-KL [21], skew-KL [29], concrete score matching [27], on-policy distillation [1], and Wasserstein policy regularization [36] offer complementary choices of loss, cost, and sampling policy.

3 Method

Knowledge distillation is usually written as if teacher and student probabilities live on the same finite space. Cross-tokenizer distillation violates this assumption. The teacher assigns mass to $\mathcal{V}_T$, the student assigns mass to $\mathcal{V}_S$, and the two tokenizers may also split the same text into different numbers of positions. SOLA treats this as a support-construction problem: before comparing dense distributions, it first constructs a local support of teacher–student token pairs on which comparison is meaningful.

3.1 A Support View of Cross-Tokenizer Distillation

Let T be a teacher model with tokenizer τ_T and vocabulary $\mathcal{V}_T$, and let S_θ be a student model with tokenizer τ_S and vocabulary $\mathcal{V}_S$. Given text x, the tokenizers produce

$$\mathbf{t} = \tau_T(x) = (t_1, \dots, t_n), \qquad \mathbf{s} = \tau_S(x) = (s_1, \dots, s_m). \tag{1}$$

At teacher position i and student position j, the temperature-scaled predictive distributions are $p_i^T \in \Delta^{|\mathcal{V}_T|-1}$ and $p_j^S \in \Delta^{|\mathcal{V}_S|-1}$, where $\Delta^{d-1} = \{p \in \mathbb{R}^d_{\geq 0} : \sum_k p_k = 1\}$. If the vocabularies were shared, standard KD would minimize

$$\mathcal{L}_{\mathrm{KD}} = \sum_i \mathrm{KL}\Big(p_i^T \,\|\, p_i^S\Big). \tag{2}$$

This loss assumes that the same coordinate denotes the same token on both sides. When $\mathcal{V}_T \neq \mathcal{V}_S$, that coordinate system no longer exists.

SOLA replaces the missing coordinate system with an explicit local support $\Omega_{ij} \subseteq \mathcal{V}_T \times \mathcal{V}_S$. For each locally aligned position pair (i, j), SOLA restricts the teacher and student distributions to this support and computes a transport loss:

$$\mathcal{L}_{\mathrm{SOLA}} = \mathcal{L}_{\mathrm{task}} + \lambda \sum_{(i,j)\in\mathcal{A}(x)} W_\varepsilon\Big(\hat{p}_i^T, \hat{p}_j^S; \hat{C}_{ij}\Big). \tag{3}$$

Here $\mathcal{A}(x)$ is the set of local teacher–student position pairs constructed by MCC, and $\Omega_{ij} \subseteq \mathcal{V}_T \times \mathcal{V}_S$ is the local support of teacher–student token correspondences for pair (i, j). Writing $\Omega_{ij}^T := \{u : \exists v,\ (u, v) \in \Omega_{ij}\}$ and $\Omega_{ij}^S := \{v : \exists u,\ (u, v) \in \Omega_{ij}\}$ for the teacher- and student-side projections of Ω_{ij}, the local marginals

$$\hat{p}_i^T(u) = \frac{p_i^T(u)\,\mathbf{1}[u \in \Omega_{ij}^T]}{\sum_{u'\in\Omega_{ij}^T} p_i^T(u')}, \qquad \hat{p}_j^S(v) = \frac{p_j^S(v)\,\mathbf{1}[v \in \Omega_{ij}^S]}{\sum_{v'\in\Omega_{ij}^S} p_j^S(v')} \tag{4}$$

restrict the teacher and student distributions to those projections. We record how much teacher mass the support retains through the coverage variable

$$\rho_i := \sum_{u \in \Omega_{ij}^T} p_i^T(u), \tag{5}$$

which equals 1 when the retrieved support exhausts the active teacher distribution and approaches 0 as truncation discards teacher mass; ρ_i is reported as a per-method diagnostic in Sec. 4.4. Finally, $\hat{C}_{ij}$ is the local cost matrix, and W_ε is the entropic optimal-transport objective over Ω_{ij}. The task loss is student-side next-token cross-entropy on response tokens, $\mathcal{L}_{\text{task}} = -\sum_{t \in \mathcal{I}_{\text{resp}}} \log p_{S_\theta}(y_t \mid x_{\text{prompt}}, y_{<t})$; prompt and padding tokens are masked. The SOLA term transfers dense teacher mass through a calibration-derived cross-tokenizer support.

This objective admits an information-projection reading. With the compatibility kernel $K_{uv} = \exp(-\hat{C}_{ij}(u,v)/\varepsilon)$ on Ω_{ij} and $Z_K = \sum_{(u,v)} K_{uv}$, substituting $\hat{C}_{uv} = -\varepsilon \log K_{uv}$ into Eq. (14) shows that its optimizer coincides with $\arg\min_{\Gamma \in \Pi(\hat{p}_i^T, \hat{p}_j^S)} \mathrm{KL}(\Gamma \,\|\, K/Z_K)$; the objectives differ by a positive scale and an additive constant. The kernel weights token pairs while the current local probabilities fix the marginals: a uniform kernel weights every pair equally, whereas a nonuniform kernel favors particular correspondences.

3.2 Semantic Optimal Local Alignment

A useful local support has to be text-local, identity-aware, and contextual at the same time. Text-locality confines supervision to teacher and student blocks with matching decoded text. Identity-awareness means candidate student tokens are selected because they are plausible realizations of teacher tokens, not because their probabilities have similar ranks; without it, shape matching can pair unrelated tokens. Contextuality means a globally plausible token pair may still be wrong for the current distribution if the teacher's active mass is on a different sense; without it, a static cost overfits to a frequency-weighted average of senses. SOLA addresses these requirements in order with three steps that together define the support on which Eq. (3) is evaluated: local text synchronization establishes text-locality, cross-vocabulary cost estimation supplies token associations, and instance-specific retrieval uses the current teacher probabilities to weight them.

Local text synchronization. The sequence axis is aligned by decoded text rather than by token index. Let D_T and D_S be the two detokenizers. A boundary pair (i, j) is synchronized when

$$D_T(t_{1:i}) = D_S(s_{1:j}). \tag{6}$$

The interval between two consecutive synchronized boundaries is a Minimal Complete Correspondence block: the two spans decode to the same string and contain no smaller synchronized prefix pair. Let $\mathcal{B}(x)$ be the set of synchronized boundary pairs. The MCC partition is what distinguishes sequence-level synchronization in SOLA from prior monotone-token alignment: earlier CTKD methods such as MinED [44] and CDM [9] run DTW over the entire teacher–student token sequence under an edit-distance or hand-designed cost, conflating sequence-level positioning with token-level correspondence; the decoded-text-equality boundary in Eq. (6) instead isolates the sequence-level problem to a partition that is exact whenever a shared decoded prefix exists. Inside each MCC block we then apply a classical monotone alignment procedure (DTW [4], as used for cross-tokenizer correspondence by [9, 44]) to obtain a token-position path, and define $\mathcal{A}(x) := \bigcup_{\text{blocks}} \mathrm{DTW\text{-}path}(\text{block})$ as the union of these paths over all blocks. With this scaffold in place, vocabulary-level correspondence becomes a separate, dense problem inside each block, and the next two paragraphs construct the support on which that problem is solved.

Cross-vocabulary geometry. SOLA estimates a ground cost from a calibration corpus $\mathcal{D}_{\text{cal}}$. Running both tokenizers and applying MCC gives local teacher–student token co-occurrences:

$$f_{uv} = \sum_{x \in \mathcal{D}_{\text{cal}}} \sum_{(i,j) \in \mathcal{A}(x)} \mathbf{1}[t_i = u]\, \mathbf{1}[s_j = v], \qquad u \in \mathcal{V}_T,\ v \in \mathcal{V}_S. \tag{7}$$

These counts yield a data-derived semantic cost: an empirical model of which teacher–student token pairs co-occur in matched local contexts, aggregated across calibration contexts. Table 4 compares this semantic cost (+ cost) with a uniform-cost configuration. To avoid turning frequent student

tokens into universal hubs, SOLA normalizes by the student-side marginal

$$q_v = \frac{\sum_u f_{uv}}{\sum_{u',v'} f_{u'v'}} \tag{8}$$

and forms the row distribution

$$R_{uv} = \frac{(f_{uv} + \delta_f)/(q_v + \delta_q)^\alpha}{\sum_{v'} (f_{uv'} + \delta_f)/(q_{v'} + \delta_q)^\alpha}. \tag{9}$$

Here δ_f and δ_q denote count and marginal-frequency smoothing, respectively. We refer to the exponent α as the *hub-correction* parameter; setting $\alpha{=}0$ disables the correction and reproduces a frequency-biased cost, while $\alpha{>}0$ down-weights pairs in which the student side is a generic high-frequency token. The global cost is

$$C_{uv} = -\log(R_{uv} + \delta_C). \tag{10}$$

The fixed floor $\delta_C > 0$ keeps this logarithm finite. Low cost means that a student token is a common realization of the teacher token in matched local text contexts, after correcting for generic high-frequency matches.

The cost matrix is fixed during distillation. At each position, the current teacher distribution selects the active anchors, and the two models' current probabilities, renormalized on the retrieved sets, provide the local marginals. The same cost therefore serves transport problems that change with the predictions.

Contextual support. The global cost provides a reusable vocabulary geometry, but each training position only needs the part of that geometry touched by the teacher's current distribution. SOLA performs two retrieval steps: a *forward retrieval* that picks student candidates linked to the teacher's active intents, and a *backward expansion* that adds back teacher tokens those candidates could also realize. The active teacher intents are

$$\mathcal{K}_i = \mathrm{TopK}(p_i^T, K), \tag{11}$$

the forward-retrieved student candidates are

$$\mathcal{P}_i = \{v \in \mathcal{V}_S : \exists u \in \mathcal{K}_i,\ f_{uv} > 0\}, \tag{12}$$

and the backward-expanded teacher set is

$$\mathcal{E}_i = \mathcal{K}_i \cup \{u \in \mathcal{V}_T : \exists v \in \mathcal{P}_i,\ f_{uv} > 0\}. \tag{13}$$

The local support is $\Omega_{ij} = \mathcal{E}_i \times \mathcal{P}_i$. To keep the per-step Sinkhorn cost bounded when forward and backward retrieval produce many candidates, we cap each side to at most κ tokens (default $\kappa{=}256$, distinct from the TopK parameter K above) by retaining each active anchor's lowest-cost partner first. Backward expansion adds reverse-associated teacher rows: a student token can co-occur with different teacher tokens across calibration contexts. It changes the row set and the renormalized teacher marginal, with added rows weighted by their current teacher probabilities. Hub correction down-weights generic high-frequency destinations in the expanded support.

Algorithm 1 gives the complete procedure, including capping, finite sentinel costs for unobserved pairs, and empty-support fallback. Each minibatch retrieves the local sets, renormalizes the current probabilities over them, and solves the local coupling. The transport gradient reaches the student through its renormalized distribution on $\mathcal{P}_i$. With the support $\Omega_{ij} = \mathcal{E}_i \times \mathcal{P}_i$ constructed as above, $\Omega_{ij}^T = \mathcal{E}_i$ and $\Omega_{ij}^S = \mathcal{P}_i$. The local marginals in Eq. (4) renormalize p_i^T over $\mathcal{E}_i$ and p_j^S over $\mathcal{P}_i$, while Eq. (5) sums p_i^T over $\mathcal{E}_i$ before local renormalization. The transport term in Eq. (3) is

$$W_\varepsilon(\hat{p}_i^T, \hat{p}_j^S; \hat{C}_{ij}) = \min_{\Gamma \in \Pi(\hat{p}_i^T, \hat{p}_j^S)} \left[\langle \Gamma, \hat{C}_{ij} \rangle + \varepsilon \sum_{u,v} \Gamma_{uv}(\log \Gamma_{uv} - 1) \right], \tag{14}$$

which is the entropic optimal transport problem [43, 38, 55] solved by Sinkhorn iterations [17] on a small local matrix.

Two intermediate configurations are useful as ablation anchors and are reported alongside the full method in Sec. 4.5: *ULD + MCC* applies ULD's sorted-shape loss inside MCC blocks while skipping cost/backward/hub correction; *MCC + uniform OT* replaces the calibration-derived cost with a uniform $\hat{C}_{ij}$ on the same blocks. They evaluate the MCC scaffold and the uniform-cost OT objective as successive stages of the ablation.

To make the support's role precise, we contrast the empirical construction above with an idealized cross-vocabulary correspondence.

Definition 3.1 (Ideal correspondence in context)**.** Fix an MCC block B beginning at synchronized prefixes $t_{<i}$ and $s_{<j}$ with common decoded context x. Let $\chi_{x,B}(\mathbf{u}, \mathbf{v})$ indicate whether teacher and student token sequences appended to these prefixes yield the same decoded continuation. In the one-token case, write $\varphi_{x,B}(u) = \{v : \chi_{x,B}((u), (v)) = 1\}$. If these sets are nonempty on $\mathrm{supp}(\hat{p}_i^T)$, define the ideal pushforward of the local teacher distribution as

$$\hat{p}_i^{T\to S}(v) := \sum_{u:\, v\in\varphi_{x,B}(u)} \frac{\hat{p}_i^T(u)}{|\varphi_{x,B}(u)|}. \tag{15}$$

The relation $\chi_{x,B}$ defines the contextual analysis target. The empirical score R estimates a corpus-level association $r_{\alpha,\mathcal{D}}$ under vanishing frequency smoothing. Supp. Sec. A gives its population target and convergence conditions, including the fixed-smoothing case. The next proposition characterizes an ideal target on a fixed support.

Proposition 3.2 (Ideal target under fixed support)**.** *Fix a teacher marginal $a = \hat{p}_i^T$, a finite graph G with nonempty row neighborhoods $G(u)$, finite costs C on G, and $\varepsilon > 0$. Minimize the entropic objective in Eq.* (14) *over couplings supported on G with row marginal a and a free student marginal. The unique optimizer spreads each teacher row with Gibbs weights proportional to* $\exp(-C_{uv}/\varepsilon)$. *If $G(u) = \varphi_{x,B}(u)$ at a fixed (x, B) and C is constant within each row, its student marginal $b^\star(C)$ equals the ideal pushforward in Eq.* (15)*.*

On the same support G, $b^\star(C)$ is also the unique minimizer of $b \mapsto W_\varepsilon(a, b; C)$: optimizing over b frees the column marginal. This characterizes the minimum of the transport term for fixed a, G, C, and ε. In particular, the target has the explicit form (Supp. Eq. (20))

$$b_v^\star(C) = \sum_{u:\, v\in G(u)} a_u \frac{\exp(-C_{uv}/\varepsilon)}{\sum_{w\in G(u)} \exp(-C_{uw}/\varepsilon)}.$$

Each term is the teacher mass of u multiplied by its normalized compatibility with v. For SOLA's local Cartesian support, taking $G = \mathcal{E}_i \times \mathcal{P}_i$ defines this target over the retrieved student candidates.

The shared-vocabulary identity map is a simple instance of Proposition 3.2: if $\varphi_{x,B}(u) = \{u\}$ and G contains exactly these edges, the ideal pushforward and $b^\star(C)$ both coincide with a. If $G(u) = \varphi_{x,B}(u)$ lists several student realizations and C is constant on $G(u)$, the free-marginal optimum splits a_u uniformly over them. Supp. Sec. A proves this characterization.

With $C_{uv} = -\log(R_{uv} + \delta_C)$ on observed pairs, their Gibbs weights are proportional to $(R_{uv} + \delta_C)^{1/\varepsilon}$. Calibration associations therefore determine the relative weights of observed student candidates in each teacher row, and the target combines the row distributions using teacher probabilities a_u. The entropy strength controls how concentrated these allocations are. Smaller ε gives greater weight to lower-cost correspondences, while larger ε distributes mass more evenly within each row. Thus, the correspondence scores and current teacher probabilities jointly determine the target.

For fixed G, a, and $\varepsilon > 0$, Supp. Eq. (21) bounds the target change $\|b^\star(\widehat{C}) - b^\star(C)\|_1$ by $2\tanh(\|\widehat{C} - C\|_{\infty,G}/(2\varepsilon))$. The bound depends on the maximum cost error over G relative to the entropy strength. Its linear upper bound, $\|\widehat{C} - C\|_{\infty,G}/\varepsilon$, shows how cost accuracy and regularization jointly control target variation. Combined with the Gibbs form, this describes the balance between concentrating mass on preferred correspondences and controlling sensitivity to estimated costs.

From the ideal target to empirical diagnostics. Proposition 3.2 characterizes the student-side target induced by a fixed teacher marginal, support, cost, and entropy strength. In SOLA, bidirectional retrieval constructs the local support and retained teacher marginal, while calibration supplies the cross-vocabulary costs. We report teacher-mass coverage and hubness, measuring retained teacher mass and renormalized student mass on frequent destinations within the support, respectively. Coverage is positively associated with downstream score across six methods, and each successive construction stage adds a positive increment.

4 Experiments

The experiments are organized into four blocks. First, we evaluate SOLA on two instruction-following settings: a Dolly-style cross-tokenizer pair (LLaMA2-7B→GPT-2-124M, ROUGE-L) and a general-instruction benchmark (Qwen3-8B→GPT-2, zero-shot accuracy). Second, we test whether the same

Table 1: Main results for distillation from LLaMA2-7B to GPT-2-124M in the Dolly instruction-following setting. Best results are shown in **bold**.

Method	Dolly	Self-Inst.	Vicuna	S-NI	UnNI	Avg.
Teacher	32.20	24.93	22.33	37.11	41.51	31.62
SFT	$20.21_{\pm 0.40}$	$9.04_{\pm 0.30}$	$14.18_{\pm 0.27}$	$17.43_{\pm 0.34}$	$18.07_{\pm 0.55}$	$15.79_{\pm 0.25}$
MinED	$22.97_{\pm 0.36}$	$10.59_{\pm 0.28}$	$14.64_{\pm 0.25}$	$19.33_{\pm 0.32}$	$19.24_{\pm 0.50}$	$17.35_{\pm 0.23}$
ULD	$23.12_{\pm 0.32}$	$10.77_{\pm 0.25}$	$13.87_{\pm 0.23}$	$19.03_{\pm 0.28}$	$20.35_{\pm 0.45}$	$17.43_{\pm 0.20}$
MLOT	$22.89_{\pm 0.34}$	$11.03_{\pm 0.27}$	$14.58_{\pm 0.24}$	$20.49_{\pm 0.30}$	$20.39_{\pm 0.48}$	$17.88_{\pm 0.22}$
ALM	$23.64_{\pm 0.28}$	$11.52_{\pm 0.22}$	$15.21_{\pm 0.20}$	$20.83_{\pm 0.26}$	$21.02_{\pm 0.42}$	$18.44_{\pm 0.18}$
SeDi	$24.19_{\pm 0.26}$	$12.64_{\pm 0.20}$	$16.46_{\pm 0.18}$	$21.78_{\pm 0.22}$	$21.88_{\pm 0.36}$	$19.39_{\pm 0.15}$
SOLA (Ours)	$\mathbf{25.41}_{\pm 0.22}$	$\mathbf{13.27}_{\pm 0.16}$	$\mathbf{17.18}_{\pm 0.15}$	$\mathbf{22.94}_{\pm 0.18}$	$\mathbf{22.83}_{\pm 0.30}$	$\mathbf{20.33}_{\pm 0.13}$

Table 2: Zero-shot results for distillation from Qwen3-8B to GPT-2 in the general instruction setting. Best results are shown in **bold**.

Method	ARC-C	ARC-E	BoolQ	IFEval	MMLU	PIQA	Avg.
Teacher	55.63	83.54	86.82	42.09	72.92	76.77	69.63
SFT	$18.37_{\pm 0.70}$	$42.43_{\pm 0.50}$	$46.00_{\pm 0.62}$	$9.31_{\pm 1.00}$	$21.05_{\pm 0.38}$	$61.02_{\pm 0.50}$	$33.03_{\pm 0.26}$
MinED	$18.95_{\pm 0.58}$	$43.15_{\pm 0.45}$	$41.50_{\pm 0.50}$	$9.15_{\pm 0.85}$	$22.10_{\pm 0.36}$	$61.40_{\pm 0.50}$	$32.71_{\pm 0.22}$
ULD	$19.03_{\pm 0.62}$	$43.86_{\pm 0.45}$	$39.30_{\pm 0.55}$	$8.99_{\pm 0.92}$	$22.92_{\pm 0.32}$	$61.59_{\pm 0.45}$	$32.62_{\pm 0.23}$
MLOT	$18.92_{\pm 0.60}$	$42.65_{\pm 0.52}$	$38.99_{\pm 0.52}$	$9.05_{\pm 0.88}$	$22.78_{\pm 0.34}$	$61.70_{\pm 0.43}$	$32.34_{\pm 0.22}$
ALM	$18.84_{\pm 0.55}$	$43.28_{\pm 0.40}$	$42.03_{\pm 0.48}$	$9.17_{\pm 0.80}$	$22.64_{\pm 0.30}$	$61.76_{\pm 0.40}$	$32.95_{\pm 0.20}$
SeDi	$19.05_{\pm 0.50}$	$43.40_{\pm 0.42}$	$43.85_{\pm 0.45}$	$11.00_{\pm 0.72}$	$22.90_{\pm 0.28}$	$62.00_{\pm 0.36}$	$33.70_{\pm 0.18}$
SOLA	$\mathbf{20.18}_{\pm 0.45}$	$\mathbf{44.45}_{\pm 0.38}$	$\mathbf{47.50}_{\pm 0.42}$	$\mathbf{13.91}_{\pm 0.65}$	$\mathbf{23.50}_{\pm 0.26}$	$\mathbf{62.59}_{\pm 0.32}$	$\mathbf{35.36}_{\pm 0.17}$

mechanism transfers to code and math, using task-specialized teachers. Third, we analyze where SOLA improves alignment, focusing on alignment efficiency and its correlation with downstream quality. Finally, we ablate the method components and key hyperparameters.

4.1 Protocol and Baselines

Dolly instruction-following setting. The first experiment uses Dolly-style instruction distillation [15] following the canonical SeDi setup [45], with the cross-tokenizer pair LLaMA2-7B [42]→GPT-2-124M [39] that exhibits a SentencePiece-to-byte-BPE mismatch between teacher and student. We evaluate on five instruction-following benchmarks: Dolly [15], Self-Inst. [47], Vicuna [11], S-NI [46], and UnNI [26], and report ROUGE-L and F1. All methods share the same teacher outputs, student initialization, training data, validation split, decoding rule, and checkpoint-selection rule. KD methods share the KD update budget; SFT/KD step counts are listed separately in Supp. Tab. 9. The Dolly and general-instruction tables report means $\pm$ std over n=5 seeds.

General-instruction setting. The second experiment uses Qwen3-8B [50] as teacher and GPT-2 [39] as student, trained on UltraChat [19], with zero-shot evaluation on ARC-C/E [13], BoolQ [12], IFEval [57], MMLU [24], and PIQA [6]. The same matched-protocol rules apply. An additional Dolly-v2-3B→OPT-125M pair (different tokenizer regime) is reported in Supp. Sec. B.

Code and math setting. For code and math, we use the SeDi domain settings: DeepSeek-Coder-6.7B [22] to Pythia-160M [5] for code, and DeepSeek-R1-Distill-Llama-8B [18] to Pythia-410M for math. We evaluate on HumanEval [8] and MBPP [2] for code, and on MetaMath [51], GSM8K [14], MATH [23], and ORCA [34] for math, reporting ROUGE-L throughout following SeDi's domain-evaluation convention. The teacher is chosen to be task-specialized, so the experiment tests whether SOLA transfers a strong domain-specific teacher signal rather than only improving general instruction following. Within each domain, methods share the student and evaluation protocol; KD methods use matched update budgets.

Baselines. We compare SOLA against SFT (no distillation), ULD [7], MultiLevelOT (MLOT) [16], MinED [44], ALM [35], and SeDi [45], all under the experimental setup of SeDi [45]. SeqKD [28], CDM [9], DSKDv2 [52], and TokAlign [33] are in Supplement Sec. B.

Table 3: Results for distillation in the code and math settings. Best results are shown in **bold**.

Method	Code			Math				
	HumanEval	MBPP	Avg.	MetaMath	GSM8K	MATH	ORCA	Avg.
SFT	18.43	16.42	17.43	30.98	22.63	21.06	24.54	24.80
ULD	22.17	18.74	20.46	32.59	23.03	20.09	25.96	25.42
MLOT	21.34	18.51	19.93	33.18	23.23	20.49	26.32	25.81
MinED	21.71	18.21	19.96	32.44	23.56	19.44	24.96	25.10
ALM	22.18	18.82	20.50	33.42	23.81	20.62	26.21	26.02
SeDi	22.77	19.45	21.11	34.91	24.99	20.93	28.05	27.22
SOLA	**23.84**	**20.05**	**21.95**	**35.84**	**25.97**	**21.66**	**28.65**	**28.03**

Selection criteria. Each method is tuned within an equal-size hyperparameter grid covering its exposed knobs (e.g., distillation strength λ, temperature, Top-K, support cap, hub-correction α). For Dolly, the final hyperparameter configuration maximizes mean ROUGE-L on a held-out validation slice disjoint from all five test benchmarks, averaged over n=5 seeds. Within that configuration, the checkpoint maximizes validation ROUGE-L along the training trajectory. Selection rules are fixed before test evaluation; native support policies are detailed in Supp. Sec. D.

Calibration corpus. The cross-vocabulary count table f_{uv} in Eq. (7), and therefore the global cost matrix C in Eq. (10), is estimated once per (T, S) setting on a calibration slice $\mathcal{D}_{\text{cal}}$ before distillation starts. $\mathcal{D}_{\text{cal}}$ is drawn from the training data of the same setting (Dolly training portion for the Dolly suite, UltraChat training portion for the general-instruction suite, the SeDi domain-training portions for code and math), capped at 40M tokens after parallel tokenization by both τ_T and τ_S. The slice is disjoint from all validation and test splits used in evaluation. The count table is reusable across seeds and grid points. For each chosen cost configuration (including α and smoothing), C is computed before training and held fixed across seeds; changing a cost parameter requires recomputing C from the same counts.

4.2 Instruction Following

We test SOLA's instruction-following capability under two settings: a Dolly-style cross-tokenizer pair and an UltraChat general-instruction benchmark.

Dolly. On the LLaMA2-7B→GPT-2-124M pair (Tab. 1), SOLA recovers more than a quarter of the SFT-to-teacher headroom and beats the strongest non-SOLA baseline SeDi by $+0.94$ Avg ROUGE-L, while ULD closes only about a tenth of the same gap. The gain is consistent across all five sub-benchmarks rather than driven by any single column.

UltraChat. On the Qwen3-8B→GPT-2 zero-shot suite (Tab. 2), four direct CTKD baselines (ULD, MLOT, MinED, and ALM) have lower average scores than SFT, with ALM 0.08 points below the reference. SeDi is the first non-SOLA method to cross into positive transfer. SOLA wins all six benchmarks and exceeds SeDi by $+1.66$ on average. The contrast–widespread negative transfer for shape and lexical baselines versus a positive SOLA gain in the same setting–is consistent with the support-construction reading: when the teacher vocabulary is large and the student vocabulary small, identity-blind matching routes mass onto cross-vocabulary candidates the student cannot represent, while support-restricted transport keeps mass on reachable correspondences. Additional results on a near-shared-tokenizer pair and two more (T, S) combinations testing teacher and student scale appear in Supp. Secs. B and F; both corroborate the main-paper trends.

4.3 Code and Math

Tab. 3 reports both blocks. On the code suite (DeepSeek-Coder-6.7B→Pythia-160M, $41.9\times$ capacity gap), SOLA reaches 21.95 Avg ROUGE-L versus 21.11 for SeDi ($+0.84$), with the larger margin on HumanEval ($+1.07$) than on MBPP ($+0.60$). On the math suite (DeepSeek-R1-Distill-Llama-8B→Pythia-410M, $2.55\times$ vocabulary gap), SOLA reaches 28.03 versus 27.22 for SeDi ($+0.81$), with per-benchmark gains of $+0.93/+0.98/+0.73/+0.60$ on MetaMath/GSM8K/MATH/ORCA. MATH is the one benchmark where every non-SOLA CTKD method underperforms SFT in mean

Figure 2: Diagnostic coverage versus ROUGE-L on Dolly.

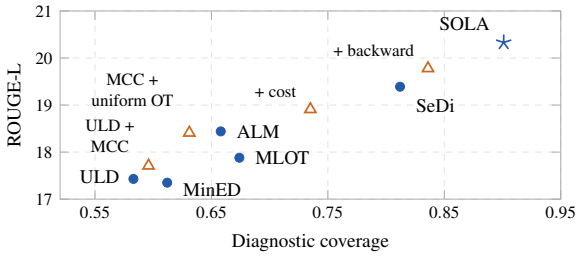


Table 4: Component ablation on Dolly; each "+" adds one module.

Configuration	Score	Δ
SFT (no distillation)	15.79	—
ULD reference	17.43	+1.64
ULD + MCC	17.71	+1.92
MCC + uniform OT	18.41	+2.62
+ cost	18.91	+3.12
+ backward	19.78	+3.99
+ hub corr. **(SOLA)**	**20.33**	**+4.54**

(21.06 SFT versus 19.44–20.93 for ULD/MLOT/MinED/ALM/SeDi), and SOLA is the only method whose mean shifts positively against SFT (21.66, +0.60). The MATH cell echoes the small-student UltraChat negative-transfer pattern: when the teacher's high-mass tokens are domain-specific and the student vocabulary cannot match them, identity-blind objectives spread mass onto cross-vocabulary noise, while support-restricted transport keeps mass inside reachable correspondences.

4.4 Alignment Efficiency and Score Correlation

A central question is how retained teacher mass relates to distillation performance. We measure coverage using ρ_i (Eq. 5) for full SOLA and method-specific post-hoc supports for the baselines. Definitions and per-method values, including hubness, are in Supp. Sec. E.

Coverage–score correlation. The six methods in Supp. Tab. 11 show a positive association between diagnostic coverage and ROUGE-L, with Spearman rank correlation r_s=0.89 (n=6). Paired resampling of these six methods gives a 95% percentile bootstrap interval of $[0.20, 1.00]$. Fig. 2 also shows four ablations as a reference overlay, with horizontal readings from the original diagnostic figure and scores from Tab. 4; data sources, diagnostic supports, and resampling details are in Supp. Sec. E. Component gains are evaluated by the sequential ablation in Sec. 4.5. Within SOLA, the Top-K sweep peaks at $K = 64$ over the tested range $K = 16$–128 (Supp. Sec. D).

4.5 Ablations

We read Tab. 4 as an additive buildup on SFT. ULD-style shape matching first establishes the share attributable to any cross-tokenizer signal at all (+1.64). The next two rows disentangle the MCC scaffold from the comparison objective: running ULD's sorted-shape loss *inside* MCC blocks adds only +0.28, while replacing the shape objective with uniform-cost OT on the same blocks adds a further +0.70. These increments are conditional on this sequence of changes. The remaining three rows decompose the support-construction package built on top of MCC. The semantic cost, backward expansion, and hub correction each add a further increment in the same range. Backward expansion has the largest conditional increment (+0.87) among the five SOLA additions. It adds reverse-associated teacher rows weighted by their current probabilities. Hub correction down-weights high-frequency destinations. SOLA has the lowest hubness among the six methods in Supp. Tab. 11. All five SOLA additions have positive conditional increments, ranging from +0.28 to +0.87. Hyperparameter sensitivity and additional ablations are in the supplement.

4.6 Sensitivity to the calibration corpus

We varied both calibration size and domain. The size conditions use nested 50% and 10% prefixes of one shuffled in-domain manifest, so each smaller set is a strict subset of the larger one. For domain, we rebuilt the matrix from UltraChat (instruction-style) and WikiText-103 (general text), each truncated to the full in-domain token budget. Training data, seeds, and the downstream protocol remain fixed; only the calibration matrix changes.

As shown in Tab. 5, halving the corpus reduces the score by 0.09; the 10% subset reduces it by 0.72 and raises the empty-support rate from 1.13% to 5.41%. This pattern is consistent with early saturation of frequent associations in this setting, while long-tail coverage remains data-limited. At the same

Table 5: Sensitivity to calibration corpus size and domain.

Calibration	Budget	Score	Coverage	Hubness	Empty %
Full in-domain	100%	20.33	0.90	0.09	1.13
Half in-domain	50%	20.24	0.88	0.09	1.86
Tenth in-domain	10%	19.61	0.80	0.13	5.41
Instruction OOD (UltraChat)	100%	20.05	0.85	0.11	2.71
General-text OOD (WikiText-103)	100%	19.16	0.72	0.16	9.02

Table 6: Training efficiency and performance in the Dolly setting.

Method	ms/step	tok/s	Peak allocated (GiB)	Preprocess (min)	Artifact (MiB)	Avg ↑
SFT	163.7	25,024	4.75	—	—	15.79
MinED	688.4	5,951	20.85	38.66	0.18	17.35
ALM	803.7	5,098	21.11	online	—	18.44
SeDi	1,048.9	3,906	23.46	online	—	19.39
SOLA	**963.2**	**4,253**	**22.78**	**47.71**	**72.68**	**20.33**

token budget, instruction-style OOD calibration loses 0.28 points, whereas general-text calibration loses 1.17 and reduces coverage to 0.72. The result supports transfer to this instruction-style corpus and quantifies the degradation under a larger domain shift.

4.7 Training Efficiency

Tab. 6 compares the training efficiency and resource usage of SOLA with those of the baselines. We profile distillation from LLaMA2-7B to GPT-2-124M on a single A100 GPU in BF16, with a micro-batch size of 4, two gradient accumulation steps, and sequences padded to 512 tokens. For each method, we launch five independent processes, each with 50 warm-up steps followed by 200 synchronized measurement steps. Compared with SeDi, SOLA improves the average score in the Dolly setting by 0.94 points while reducing training time per step by 8.17% and peak allocated GPU memory by 0.68 GiB. MinED and ALM have shorter step times but score 2.98 and 1.89 points lower, respectively. SOLA requires 47.71 minutes of preprocessing for this teacher–student pair and cost configuration. The counts are reusable across grid points, while the cost matrix is reusable across seeds and settings that leave its parameters unchanged. Extrapolating the measured step times to 10,010 KD steps per seed over five seeds gives amortized totals of 851.2 minutes for SOLA (including preprocessing) and 875.0 minutes for SeDi. These results show that SOLA's optimal transport formulation over local supports improves distillation performance while keeping training time and memory costs manageable.

5 Discussion

Limitations. SOLA targets the regime where teacher–student tokenization disagreements are concentrated on short spans, which captures the common SentencePiece, BPE, and byte-BPE pairs that dominate open-source language models today. Long mixed-script or byte-fallback regions where this assumption fails (multilingual transfer with non-Latin scripts, cross-script pairs, and inputs dominated by byte-fallback bytes) are out of scope, as are on-policy sampling and 70B-scale teachers. Within scope, the cross-vocabulary support is estimated once from a calibration corpus, so substantial domain shift between calibration and downstream tasks degrades alignment quality.

Conclusion. Cross-tokenizer distillation is ill-defined under heterogeneous vocabularies: token-level KD has no shared coordinate system, and shape-only matching preserves probability density at the cost of token identity. We introduced SOLA, a structured-support objective that synchronizes teacher and student tokenizations into local correspondence blocks via Minimal Complete Correspondence, retrieves an auditable cross-vocabulary support around each teacher prediction, and applies dense entropic optimal transport only inside that support. Across cross-tokenizer instruction following, general-instruction distillation, and code and math transfer, SOLA outperforms direct cross-tokenizer baselines. Ablations demonstrate that the sequential introduction of each component yields performance gains, indicating that every component contributes to the performance.

Acknowledgements

This work was supported by Beijing Jiaotong University.

References

[1] Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024.

[2] Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

[3] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

[4] Donald J Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *Proceedings of the 3rd international conference on knowledge discovery and data mining*, pages 359–370, 1994.

[5] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of the International Conference on Machine Learning*, 2023.

[6] Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439, 2020.

[7] Nicolas Boizard, Kevin El Haddad, Céline Hudelot, and Pierre Colombo. Towards cross-tokenizer distillation: the universal logit distillation loss for llms. *arXiv preprint arXiv:2402.12030*, 2024.

[8] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

[9] Yijie Chen, Yijin Liu, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. Enhancing cross-tokenizer knowledge distillation with contextual dynamical mapping. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8005–8018, 2025.

[10] Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. Dialogsum: A real-life scenario dialogue summarization dataset. *arXiv preprint arXiv:2105.06762*, 2021.

[11] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. `https://lmsys.org/blog/2023-03-30-vicuna/`, 2023.

[12] Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*, 2019.

[13] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

[14] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[15] Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. Free dolly: Introducing the world's first truly open instruction-tuned llm. `https://www.databricks.com/blog/2023/04/12/dolly-first-open-commercially-viable-instruction-tuned-llm`, 2023.

[16] Xiao Cui, Mo Zhu, Yulei Qin, Liang Xie, Wengang Zhou, and Houqiang Li. Multi-level optimal transport for universal cross-tokenizer knowledge distillation on language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23724–23732, 2025.

[17] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

[18] DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[19] Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*, 2023.

[20] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International journal of computer vision*, 129(6):1789–1819, 2021.

[21] Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. *arXiv preprint arXiv:2306.08543*, 2023.

[22] Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Y. K. Wu, Y. K. Li, et al. Deepseek-coder: When the large language model meets programming – the rise of code intelligence. *arXiv preprint arXiv:2401.14196*, 2024.

[23] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021.

[24] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

[25] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[26] Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. Unnatural instructions: Tuning language models with (almost) no human labor. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2023.

[27] Yeongmin Kim, Donghyeok Shin, Mina Kang, Byeonghu Na, and Il-Chul Moon. Distillation of large language models via concrete score matching. In *The Fourteenth International Conference on Learning Representations*, 2026.

[28] Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 1317–1327, 2016.

[29] Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. Distillm: Towards streamlined distillation for large language models. In *Proceedings of the 41st International Conference on Machine Learning*, pages 24872–24895, 2024.

[30] Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. *arXiv preprint arXiv:1804.10959*, 2018.

[31] Matthew Lamm, Jennimaria Palomaki, Chris Alberti, Daniel Andor, Eunsol Choi, Livio Baldini Soares, and Michael Collins. Qed: A framework and dataset for explanations in question answering. *Transactions of the Association for computational Linguistics*, 9:790–806, 2021.

[32] VI Lcvenshtcin. Binary coors capable or 'correcting deletions, insertions, and reversals. In *Soviet physics-doklady*, volume 10, 1966.

[33] Chong Li, Jiajun Zhang, and Chengqing Zong. Tokalign: Efficient vocabulary adaptation via token alignment. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pages 4109–4126, 2025.

[34] Wing Lian, Bleys Goodson, Eugene Pentland, Austin Cook, Chanvichet Vong, and Teknium. Openorca: An open dataset of gpt augmented flan reasoning traces. `https://huggingface.co/Open-Orca/OpenOrca`, 2023.

[35] Benjamin Minixhofer, Ivan Vulić, and Edoardo M Ponti. Cross-tokenizer distillation via approximate likelihood matching. 2025.

[36] Byeonghu Na, Hyungho Na, Yeongmin Kim, Suhyeon Jo, HeeSun Bae, Mina Kang, and Il-Chul Moon. Semantic-aware wasserstein policy regularization for large language model alignment. In *The Fourteenth International Conference on Learning Representations*, 2026.

[37] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models. *ACM Transactions on Intelligent Systems and Technology*, 16(5):1–72, 2025.

[38] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[39] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[40] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 1715–1725, 2016.

[41] Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

[42] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[43] Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2008.

[44] Fanqi Wan, Xinting Huang, Deng Cai, Xiaojun Quan, Wei Bi, and Shuming Shi. Knowledge fusion of large language models. *arXiv preprint arXiv:2401.10491*, 2024.

[45] Huazheng Wang, Yongcheng Jing, Haifeng Sun, Jingyu Wang, Jianxin Liao, Leszek Rutkowski, and Dacheng Tao. Bridging the tokenizer gap: Semantics and distribution-aware knowledge transfer for unbiased cross-tokenizer distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 33494–33502, 2026.

[46] Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, et al. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2022.

[47] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2023.

[48] Xiaohan Xu, Ming Li, Chongyang Tao, Tao Shen, Reynold Cheng, Jinyang Li, Can Xu, Dacheng Tao, and Tianyi Zhou. A survey on knowledge distillation of large language models. *arXiv preprint arXiv:2402.13116*, 2024.

[49] Ying Xu, Dakuo Wang, Mo Yu, Daniel Ritchie, Bingsheng Yao, Tongshuang Wu, Zheng Zhang, Toby Jia-Jun Li, Nora Bradford, Branda Sun, et al. Fantastic questions and where to find them: Fairytaleqa–an authentic dataset for narrative comprehension. *arXiv preprint arXiv:2203.13947*, 2022.

[50] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[51] Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for large language models. In *Proceedings of the International Conference on Learning Representations*, 2024.

[52] Xue Zhang, Songming Zhang, Yunlong Liang, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. A dual-space framework for general knowledge distillation of large language models. *arXiv preprint arXiv:2504.11426*, 2025.

[53] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2), 2023.

[54] Mengyu Zheng, Hanting Chen, Tianyu Guo, Chong Zhu, Binfan Zheng, Chang Xu, and Yunhe Wang. Enhancing large language models through adaptive tokenizers. *Advances in Neural Information Processing Systems*, 37:113545–113568, 2024.

[55] Mengyu Zheng, Zhiwei Hao, Yehui Tang, and Chang Xu. Visual prompting via partial optimal transport. In *European Conference on Computer Vision*, pages 1–18. Springer, 2024.

[56] Mengyu Zheng, Yehui Tang, Zhiwei Hao, Kai Han, Yunhe Wang, and Chang Xu. Adapt without forgetting: Distill proximity from dual teachers in vision-language models. In *European Conference on Computer Vision*, pages 109–125. Springer, 2024.

[57] Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

Appendix

A Calibration Estimation and Fixed-Support Analysis

Population target and convergence. Consider N i.i.d. aligned pairs from a fixed distribution $P_{\mathcal{D}}$ on finite vocabularies, with fixed $\alpha \geq 0$. Restrict the estimator to $\mathcal{V}_S^+ = \{v : P_{\mathcal{D}}(v) > 0\}$ and teacher rows with $P_{\mathcal{D}}(u) > 0$. The population association is

$$r_{\alpha,\mathcal{D}}(v \mid u) = \frac{P_{\mathcal{D}}(u,v)/P_{\mathcal{D}}(v)^{\alpha}}{\sum_{v' \in \mathcal{V}_S^+} P_{\mathcal{D}}(u,v')/P_{\mathcal{D}}(v')^{\alpha}}, \qquad v \in \mathcal{V}_S^+. \tag{16}$$

The empirical version can use observed student support, which equals $\mathcal{V}_S^+$ eventually almost surely in this finite-vocabulary model. Write $\widehat{P}_N(u,v) = f_{uv}/N$ and $\widehat{q}_N(v) = \sum_u \widehat{P}_N(u,v)$. Define

$$\widehat{r}_N(v \mid u) = \frac{(\widehat{P}_N(u,v) + \eta_{f,N})/(\widehat{q}_N(v) + \eta_{q,N})^{\alpha}}{\sum_{v' \in \mathcal{V}_S^+} (\widehat{P}_N(u,v') + \eta_{f,N})/(\widehat{q}_N(v') + \eta_{q,N})^{\alpha}}. \tag{17}$$

The count-scale form in Eq. (9) has $\eta_{f,N} = \delta_f/N$ and $\eta_{q,N} = \delta_q$ on the same support. If both frequency-scale smoothers vanish, the strong law gives $\widehat{P}_N \to P_{\mathcal{D}}$ and $\widehat{q}_N \to P_{\mathcal{D}}(v)$. All included student marginals are positive, and every included teacher row has a positive normalization denominator. Continuity of division and row normalization therefore gives $\widehat{r}_N \to r_{\alpha,\mathcal{D}}$ uniformly on this finite support. Define $\widehat{C}_N = -\log(\widehat{r}_N + \delta_C)$ and $C_{\mathcal{D}} = -\log(r_{\alpha,\mathcal{D}} + \delta_C)$. With fixed $\delta_C > 0$, the logarithm is $1/\delta_C$-Lipschitz on $[\delta_C, \infty)$, so

$$\|\widehat{C}_N - C_{\mathcal{D}}\|_{\infty} \leq \delta_C^{-1} \|\widehat{r}_N - r_{\alpha,\mathcal{D}}\|_{\infty} \longrightarrow 0. \tag{18}$$

For full-vocabulary normalization, the same zero-extended limit holds with the additional conditions $\eta_{q,N} > 0$ and $\eta_{f,N}/\eta_{q,N}^{\alpha} \to 0$: each zero-marginal student token then contributes a vanishing smoothed weight. For fixed count smoothing δ_f and fixed $\delta_q > 0$, the limit instead replaces $P_{\mathcal{D}}(v)^{\alpha}$ by $(P_{\mathcal{D}}(v) + \delta_q)^{\alpha}$ throughout Eq. (16).

Proof of Proposition 3.2. With only the row marginal fixed, the objective separates over teacher rows. Rows with $a_u = 0$ are identically zero. For $a_u > 0$, write $\Gamma_{uv} = a_u q_u(v)$ for $v \in G(u)$, where q_u is a probability vector. Discarding terms constant in q_u, the row objective is

$$a_u \left[\sum_{v \in G(u)} q_u(v) C_{uv} + \varepsilon \sum_{v \in G(u)} q_u(v) \log q_u(v) \right]. \tag{19}$$

Strict convexity and the normalization constraint give the unique Gibbs row $q_u(v) \propto \exp(-C_{uv}/\varepsilon)$. Thus

$$\Gamma_{uv}^C = a_u \frac{e^{-C_{uv}/\varepsilon}}{\sum_{v' \in G(u)} e^{-C_{uv'}/\varepsilon}}, \qquad b_v^{\star}(C) = \sum_u \Gamma_{uv}^C. \tag{20}$$

A constant cost within each $G(u)$ makes this row uniform, giving the stated uniform-split marginal on the prescribed hard support. Optimizing Eq. (14) jointly over the coupling and its student marginal yields this same feasible set, so $b^{\star}(C)$ uniquely minimizes the transport term over b. □

Stability under estimated costs. For the same fixed G, a, and $\varepsilon > 0$, let $b^{\star}(C)$ be the free-marginal target in Eq. (20). For two finite cost matrices on G,

$$\|b^{\star}(\widehat{C}) - b^{\star}(C)\|_1 \leq 2 \tanh\left(\frac{\|\widehat{C} - C\|_{\infty,G}}{2\varepsilon} \right) \leq \frac{\|\widehat{r} - r\|_{\infty,G}}{\varepsilon \delta_C}, \tag{21}$$

where the second inequality assumes nonnegative associations $r, \widehat{r}$, $C = -\log(r + \delta_C)$, $\widehat{C} = -\log(\widehat{r} + \delta_C)$, and fixed $\delta_C > 0$.

Proof. Fix a row, write $p = q_u^C$, and set $w_v = \exp[-(\widehat{C}_{uv} - C_{uv})/\varepsilon]$, $Z = \mathbb{E}_p w$. Then $q_u^{\widehat{C}}(v) = p(v) w_v / Z$. Let $m = \min_v w_v > 0$ and $M = \max_v w_v$. If $m = M$, the row is unchanged. Otherwise,

convexity of $|w - Z|$ on $[m, M]$ gives

$$\|q_u^{\widehat{C}} - q_u^C\|_1 = \frac{\mathbb{E}_p|w - Z|}{Z} \le \frac{2(M - Z)(Z - m)}{Z(M - m)} \le 2\frac{\sqrt{M} - \sqrt{m}}{\sqrt{M} + \sqrt{m}} = 2\tanh\left(\frac{\log(M/m)}{4}\right). \tag{22}$$

The second inequality follows from $Z + Mm/Z \ge 2\sqrt{Mm}$. Since $\log(M/m) \le 2\|\widehat{C} - C\|_{\infty,G}/\varepsilon$, mixing rows with probability weights a_u and applying the triangle inequality proves the first bound in Eq. (21). The second follows from $\tanh t \le t$ for $t \ge 0$ and the log-Lipschitz inequality in Eq. (18). □

B Full Baseline Sweep on Dolly

Table 7 reports results for four additional baselines: CDM, DSKDv2, TokAlign+KD, and SeqKD/KD.

Table 7: Full baseline sweep on Dolly. Best in **bold**.

Method	Dolly		Self-Inst.		Vicuna		S-NI		UnNI		Avg.	
	R	F1	R	F1	R	F1	R	F1	R	F1	R	F1
LLaMA2-7B → GPT-2-124M												
SeqKD/KD	21.85	17.81	9.92	6.20	14.21	10.59	18.63	9.22	19.42	12.39	16.81	11.24
CDM	23.30	19.11	10.96	6.88	15.12	11.31	19.81	9.87	20.61	13.23	17.96	12.08
DSKDv2	23.17	18.98	**11.86**	**7.47**	**15.56**	**11.67**	**20.83**	**10.44**	**20.92**	**13.39**	**18.47**	**12.39**
TokAlign+KD	**23.71**	**19.37**	11.42	7.16	15.34	11.46	20.51	10.26	20.84	13.32	18.36	12.31
Dolly-v2-3B → OPT-125M												
SeqKD/KD	20.41	16.39	8.94	6.71	13.62	10.35	13.85	5.93	15.62	10.11	14.49	9.90
CDM	22.74	18.33	9.89	7.47	14.71	11.24	16.19	6.99	17.10	11.13	16.13	11.03
DSKDv2	**23.05**	**18.60**	**10.65**	**8.05**	**15.05**	**11.51**	**16.67**	**7.23**	**17.15**	**11.18**	**16.51**	**11.31**
TokAlign+KD	22.75	18.29	10.18	7.67	14.65	11.16	16.27	7.00	17.04	11.06	16.18	11.04

The strongest extended baseline is DSKDv2, the dual-space projection method, at 18.47 Avg ROUGE-L on LLaMA2-7B→GPT-2-124M and 16.51 on Dolly-v2-3B→OPT-125M. TokAlign+KD's tokenizer-adaptation strategy reaches 18.36 and 16.18 on the same two pairs, improving over the ULD-style shape baseline by 0.93 and 0.78, respectively. The corresponding SOLA scores are 20.33 and 18.21 (Tab. 1 and Tab. 8). The two extended baselines reduce the SOLA margin by about a third compared with ULD/MLOT/MinED. SOLA retains the best Avg cell on both pairs and on both ROUGE-L and F1.

Vocabulary gap analysis: complete results. Tab. 8 reports the full Dolly evaluation suite for the high-mismatch pair from the main paper and the near-shared pair (Dolly-v2-3B→OPT-125M, SMR 96.8%, VMR 87.4%), where teacher and student already share a byte-level BPE family.

Cross-pair comparison. The SOLA-vs-SeDi Avg gap is $+0.94$ ROUGE-L on the high-mismatch pair and $+0.59$ on the near-shared pair. SOLA's lead over the mean of {MinED, ULD, MLOT} is 2.78 ROUGE-L on the high-mismatch pair and 2.69 on the near-shared pair. SOLA wins every per-benchmark column on both ROUGE-L and F1 in both regimes ($20/20$ cells across the two blocks), so the consistency holds at the per-benchmark level rather than only at the average.

Per-seed variation and scope of comparisons. Tables 1 and 2 report means and standard deviations over five training seeds. SOLA has a higher mean than SeDi in all $5/5$ high-mismatch Dolly, $5/5$ near-shared Dolly, and $6/6$ general-instruction benchmark cells reported in Tables 1, 8, and 2. These 16 benchmark comparisons reuse each method's checkpoints and seeds within each setting.

C The details of the experimental settings

All methods within a setting share initialization, data manifest, seeds (13/21/42/87/101), validation/decoding, and checkpoint selection. KD methods also share the KD optimizer-step budget.

Table 8: Dolly suite across vocab-gap regimes (ROUGE-L / F1). Best in **bold**.

Method	Dolly		Self-Inst.		Vicuna		S-NI		UnNI		Avg.	
	R	F1	R	F1	R	F1	R	F1	R	F1	R	F1
High-mismatch: LLaMA2-7B → GPT-2-124M (SMR 81.2%, VMR 27.9%)												
SFT	20.21	16.65	9.04	6.32	14.18	11.86	17.43	7.91	18.07	11.22	15.79	10.79
MinED	22.97	17.23	10.59	7.13	14.64	10.51	19.33	9.28	19.24	12.39	17.35	11.31
ULD	23.12	17.42	10.77	7.67	13.87	10.38	19.03	9.05	20.35	12.68	17.43	11.44
MLOT	22.89	18.52	11.03	7.93	14.58	11.84	20.49	9.30	20.39	13.32	17.88	12.18
ALM	23.64	18.95	11.52	7.81	15.21	11.04	20.83	9.97	21.02	13.61	18.44	12.28
SeDi	24.19	19.79	12.64	7.95	16.46	12.33	21.78	10.88	21.88	14.03	19.39	13.00
SOLA (Ours)	**25.41**	**20.52**	**13.27**	**8.55**	**17.18**	**12.59**	**22.94**	**11.18**	**22.83**	**14.92**	**20.33**	**13.55**
Near-shared: Dolly-v2-3B → OPT-125M (SMR 96.8%, VMR 87.4%)												
SFT	19.25	15.56	7.88	6.45	13.45	10.74	11.92	5.23	13.61	8.82	13.22	9.36
MinED	20.72	17.09	9.67	6.79	13.89	11.71	15.10	7.02	16.78	10.17	15.23	10.56
ULD	20.94	17.42	9.85	7.02	14.06	10.58	15.14	7.54	17.00	10.32	15.40	10.58
MLOT	22.43	17.37	9.77	7.43	14.40	11.32	16.20	7.63	16.80	10.59	15.92	10.87
ALM	22.95	18.61	10.46	7.51	14.74	11.84	17.13	7.65	17.66	10.99	16.59	11.32
SeDi	24.05	19.36	11.11	8.38	16.34	12.47	18.29	7.88	18.33	11.92	17.62	12.00
SOLA (Ours)	**24.51**	**19.92**	**11.83**	**8.46**	**16.71**	**13.05**	**18.83**	**8.34**	**19.18**	**12.21**	**18.21**	**12.40**

Table 9 lists SFT and KD step counts and learning rates separately. Each baseline is tuned within an equal-size grid over its own tunable hyperparameters. SOLA's remaining coefficients are validation-selected per setting.

Table 9: Training configurations across the experimental settings.

Setting	Teacher → student	Train data	Max len.	Effective batch	SFT/KD steps	SFT/KD LR	SOLA K/λ
Dolly	LLaMA2-7B → GPT-2-124M	Dolly 11,435	512	8	4,290/10,010	2e-5/1e-4	$64/0.5$
UltraChat	Qwen3-8B → GPT-2	train_sft 207,865	512	8	25,984/51,968	1e-5/5e-5	$96/0.5$
Code	DeepSeek-Coder-6.7B → Pythia-160M	CodeM 9,000	512	8	3,375/7,875	2e-5/1e-4	$64/0.7$
Math	R1-Distill-Llama-8B → Pythia-410M	MetaMath 10,000	512	8	3,750/8,750	2e-5/8e-5	$96/0.5$

D Native Support Policies and the Top-K Comparison

Teacher Top-K seeds SOLA's local proposal, which is then expanded and capped to form the transport support. At Dolly $K = 64$, the raw anchors contain about 0.58 of teacher probability mass, while the full SOLA support after expansion and capping retains 0.901 (Table 11). The baseline support policies are summarized in Table 10. These policies describe the native training objectives; Sec. E defines the post-hoc alignment probes.

We compare native support policies under matched within-setting training and tuning budgets. ULD scores 17.43 on Dolly, versus 19.39 for SeDi and 20.33 for SOLA.

The existing sweep in Fig. 3 gives $18.45/19.62/20.33/20.21/19.98$ at $K = 16/32/64/96/128$. Every tested value exceeds ULD, and $K = 32$ through 128 exceed SeDi, a fourfold anchor-size range.

Table 10: Native teacher-signal policies for the direct baselines and SOLA. Top-K values specify training inputs or retrieval anchors.

Method	Teacher signal consumed
ULD [7]	Full distribution, sorted and zero-padded
SeDi [45]	Top-100 logits
MLOT [16]	Top-50 logits
MinED [44]	Top-10 cached logits with a fixed single-valued map
ALM [35]	Chunk-level likelihoods
SOLA	Top-64 anchors for Dolly/code and Top-96 for UltraChat/math, followed by support expansion and capping

E Per-Method Alignment Efficiency on Dolly

Fig. 2 shows the six method-level coverage–score pairs in Table 11 and a reference overlay of four intermediate configurations from Table 4. The overlay's horizontal coordinates are digitized from the original diagnostic figure and reported to three decimals; the scores follow Table 4. The six tabled method pairs form the correlation and resampling sample. The + cost stage uses the semantic cost in Eq. (10). Table 14 records the plotting inputs and their sources.

Diagnostic definitions for Table 11. Tab. 11 reports mass coverage and hubness on a held-out Dolly split. For full SOLA, coverage is Eq. (5) after retrieval and capping, using the original teacher probabilities before local renormalization. Baseline coverage uses the method-specific post-hoc probes below. Hubness measures the mass assigned to high-frequency tokens by the student distribution renormalized over the support:

$$\text{Hub.} = \mathbb{E}_{(i,j)\in\mathcal{A}}\left[\sum_{v\in\mathcal{H}_{ij}} \hat{p}_j^S(v)\right], \quad \mathcal{H}_{ij} = \underset{h}{\text{TopFreq}}\big(\Omega_{ij}^S, q\big), \tag{23}$$

where q is the student-side marginal of the calibration counts in Eq. (8) and TopFreq_h returns the top-h fraction of Ω_{ij}^S ranked by q, with $h{=}10\%$ rounded up. Lower Hub. means less mass on high-frequency student tokens within the support and therefore less degeneration toward generic hubs. The Score column reports the corresponding downstream ROUGE-L on the Dolly suite.

Computation for non-support methods. The baseline diagnostics use post-hoc probes. For ULD and MLOT, the diagnostic teacher probes contain the top 64 and 96 logits, respectively; coverage is their original teacher mass and hubness is computed on the corresponding student top-logit probes. Their training objectives consume the full distribution for ULD and Top-50 logits for MLOT (Table 10).

MinED uses a fixed single-valued map m. Its teacher Top-64 diagnostic probe Q_i reaches the distinct student destinations $D_i = \{m(u) : u \in Q_i \cap \text{dom}(m)\}$, with $\mathbb{E}|D_i| = 48$. Define the pullback $A_i^{\text{diag}} = \{u \in \text{dom}(m) : m(u) \in D_i\}$. Coverage sums the original teacher mass over this set: $\rho_i = \sum_{u\in A_i^{\text{diag}}} p_i^T(u)$.

For ALM, coverage is teacher mass on tokens whose decoded surface lies inside an active chunk, with hubness on chunk-resolved student tokens. SOLA and SeDi use their token-level alignment diagnostics. The coverage–score correlation summarizes these method-specific measurements.

Per-method comparison. Under these diagnostics, SOLA has the highest coverage and lowest hubness, alongside the highest Dolly score. Across the six methods, diagnostic coverage is positively rank-correlated with Dolly score ($r_s{=}0.89$, $n{=}6$).

Correlation and resampling. This correlation, also reported in Sec. 4.4 and Fig. 2, uses the six pairs in Table 11. The method-level bootstrap treats these six pairs as exchangeable resampling units. We draw 100,000 samples of six pairs with replacement (NumPy PCG64, seed 42), retaining each method's coverage and score together. In each sample, we recompute Spearman's r_s using average ranks for ties. The 99,988 resamples with variation in both coordinates enter the percentile

Table 11: Alignment efficiency on Dolly. Best in **bold**.

Method	$\rho \uparrow$	Hub.$\downarrow$	Score$\uparrow$
MinED	0.612	0.241	17.35
ULD	0.583	0.295	17.43
MLOT	0.674	0.182	17.88
ALM	0.658	0.207	18.44
SeDi	0.812	0.124	19.39
SOLA	**0.901**	**0.087**	**20.33**

calculation; the remaining 12 have constant coordinates. The 2.5th and 97.5th percentiles give the 95% paired percentile bootstrap interval $[0.20, 1.00]$ for the observed r_s=0.89.

F Additional Instruction-Following and Task-Specific Benchmarks

We complement the main Dolly results with general-instruction and task-specific evaluations on additional teacher–student pairs and NLG domains, demonstrating that SOLA's gain is robust across teacher scale, student scale, and task structure. The general-instruction sweep uses UltraChat-style training [19] and zero-shot evaluation on ARC-C/E [13], BoolQ [12], IFEval [57], MMLU [24], and PIQA [6], with Qwen3-8B [50] and Qwen1.5-1.8B [3] as teachers and Gemma-2-2B [41] or GPT-2 [39] as students. The task-specific NLG suite covers QED [31], DialSum [10], and FQA [49].

Table 12: General-instruction across three (T, S) scales. Best in **bold**.

Student	Method	ARC-C	ARC-E	BoolQ	IFEval	MMLU	PIQA	Avg.
Teacher: Qwen3-8B								
Teacher Reference		55.63	83.54	86.82	42.09	72.92	76.77	69.63
Gemma-2-2B	SFT	44.07	71.46	79.12	32.79	43.19	74.18	57.47
	MinED	43.55	72.12	79.25	33.65	42.35	74.45	57.56
	ULD	43.83	72.66	79.39	34.45	42.57	74.70	57.93
	MLOT	42.46	74.42	79.10	34.09	41.51	75.20	57.79
	ALM	43.82	73.19	79.54	33.91	42.58	74.73	57.96
	SeDi	44.05	74.50	79.85	34.20	42.85	75.10	58.43
	SOLA (Ours)	**44.45**	**75.16**	**80.50**	**34.93**	**43.47**	**75.71**	**59.04**
GPT-2	SFT	18.37	42.43	46.00	9.31	21.05	61.02	33.03
	MinED	18.95	43.15	41.50	9.15	22.10	61.40	32.71
	ULD	19.03	43.86	39.30	8.99	22.92	61.59	32.62
	MLOT	18.92	42.65	38.99	9.05	22.78	61.70	32.34
	ALM	18.84	43.28	42.03	9.17	22.64	61.76	32.95
	SeDi	19.05	43.40	43.85	11.00	22.90	62.00	33.70
	SOLA (Ours)	**20.18**	**44.45**	**47.50**	**13.91**	**23.50**	**62.59**	**35.36**
Teacher: Qwen1.5-1.8B								
Teacher Reference		33.70	62.63	70.09	32.37	46.10	73.07	52.99
GPT-2	SFT	18.37	42.43	46.00	9.31	21.05	61.02	33.03
	MinED	19.15	43.50	40.80	10.15	22.50	61.65	32.96
	ULD	19.37	44.49	39.54	8.75	22.91	61.37	32.74
	MLOT	19.54	44.02	39.11	11.15	22.92	61.81	33.09
	ALM	19.63	43.02	41.43	10.26	23.14	61.82	33.22
	SeDi	19.85	44.20	42.10	10.95	23.05	62.30	33.74
	SOLA (Ours)	**20.22**	**44.53**	**46.51**	**11.22**	**23.29**	**62.89**	**34.78**

Impact of student scale. Holding the teacher fixed at Qwen3-8B in Tab. 12, the high-capacity Gemma-2-2B student starts from a much stronger SFT floor (57.47 Avg) than GPT-2 (33.03). SOLA improves over SFT by $+1.57$ on Gemma-2-2B and $+2.33$ on GPT-2, and over the strongest non-SOLA baseline (SeDi) by $+0.61$ and $+1.66$ respectively. Two qualitative differences across the two students are worth noting. On Gemma-2-2B every direct CTKD baseline already beats SFT, so SOLA's $+1.57$ is the additional margin on top of an already positive ranking. On GPT-2 four of the

five direct baselines have average scores below the SFT reference (ULD/MLOT/MinED/ALM), and SOLA delivers the largest positive gain among the direct CTKD methods in this setting.

Impact of teacher scale. Holding the student fixed at GPT-2 and varying the teacher, Qwen3-8B yields SOLA 35.36 and Qwen1.5-1.8B yields SOLA 34.78, a -0.58 change against a -16.64-point teacher Avg change (69.63 vs. 52.99). With the weaker teacher, ULD (32.74, -0.29 vs. SFT) and MinED (32.96, -0.07) sit at or below SFT, MLOT and ALM are slightly above, and SOLA delivers $+1.75$ over SFT and $+1.04$ over the next-best SeDi. SOLA wins every per-benchmark column in both teacher blocks, so the per-cell sweep holds across teacher scales.

Table 13: Task-specific NLG (ROUGE-L). Best in **bold**.

Method	QED	DialSum	FQA	Avg.
SFT	48.30	44.50	33.65	42.15
ULD	48.60	44.90	33.82	42.44
MLOT	49.15	45.10	33.95	42.73
ALM	49.40	45.30	34.05	42.92
SOLA	**50.72**	**45.83**	**34.22**	**43.59**

These results extend the support-construction story to (i) a stronger student (Gemma-2-2B), (ii) a wider teacher set (Qwen3-8B and Qwen1.5-1.8B), and (iii) task-specific NLG domains (QED, DialSum, FQA), showing that SOLA's gain is consistent across teacher scale, student scale, and task structure rather than tied to the SeDi instruction setup of the main paper.

G Additional Ablation Analysis

Table 14 restates the main-paper ablation with increments relative to the preceding row; the main-paper Table 4 reports increments relative to SFT. Each SOLA addition has a positive conditional increment (+0.28 to +0.87 points) along this sequence of configurations.

H SOLA Training Algorithm

The full per-minibatch SOLA training procedure used in all main and supplement experiments is shown as Algorithm 1. It implements the pipeline in Sec. 3.2, including support capping, finite sentinel costs for unobserved pairs, and empty-support fallback.

I Hyperparameter Sensitivity

Figure 3 reports SOLA's ROUGE-L from the validation-budgeted sweep over Top-K, support cap κ, OT weight λ, entropy ε, and hub-correction α, together with the α=0 reference from the + backward ablation in Tab. 4 (19.78). The validation-selected configuration (K=64, κ=256, λ=0.5, ε=0.1, α=1.0) achieves 20.33. All 22 sweep points and the α=0 reference stay above the ULD reference (17.43), and SOLA drops at most 1.88 from its peak, so the defaults sit on a plateau rather than a knife edge. Ranked by peak-to-trough gap, Top-K is the most sensitive knob (1.88), followed by λ (1.41), κ (1.12), ε (0.88), and α (0.55). The shape of each curve aligns with the role of the corresponding knob in the support-construction pipeline: Top-K and κ underflow collapse mass coverage by retrieving too few candidates; λ underflow ($0.1\rightarrow18.92$) costs more than λ overflow ($2.0\rightarrow19.27$), suggesting the OT term is undersupplying gradient when downweighted but only partially overriding CE when overweighted; ε overflow ($0.5\rightarrow19.45$) erases the cross-vocabulary cost geometry while ε underflow ($0.05\rightarrow19.78$) stays close to the peak. At α=1, hub correction raises the score to 20.33 ($+0.55$); α=2 also scores 19.78.

J Visualization of Alignment Results

MCC Visualization. Figure 4 (a) illustrates the limitations of the baseline naive token-level alignment, where the rigid monotonicity constraint leads to severe positional dislocation in cross-tokenizer scenarios. For instance, when the teacher tokenizes "apple" into ["appl", "e"] while the

Table 14: Sequential ablation in the Dolly setting. Δ denotes the score change from the preceding row. The final column records plotting inputs to Fig. 2: ULD and full SOLA follow Table 11; superscript d marks digitized horizontal readings from the original diagnostic figure, retained as the reference overlay.

Variant	Score	Sequential Δ	Plotted coverage
SFT	15.79	—	—
ULD reference	17.43	$+1.64$	0.583
ULD + MCC	17.71	$+0.28$	0.596^d
MCC + uniform OT	18.41	$+0.70$	0.631^d
+ cost	18.91	$+0.50$	0.735^d
+ backward expansion	19.78	$+0.87$	0.836^d
+ hub correction (full SOLA)	20.33	$+0.55$	0.901

Algorithm 1 SOLA distillation for one minibatch

Input: minibatch $\mathcal{B}$, teacher T, student S_θ, fixed counts f, cost C, retrieval size K, support cap κ, OT weight λ, entropy weight ε.
Initialize $\mathcal{L}_{\mathrm{ot}} \leftarrow 0$.
for each text $x \in \mathcal{B}$ **do**
  Tokenize x as $\mathbf{t} = \tau_T(x)$ and $\mathbf{s} = \tau_S(x)$; compute MCC boundaries $\mathcal{B}(x)$ via Eq. (6) and form $\mathcal{A}(x)$ as the union of DTW paths inside the resulting blocks.
  Run T and S_θ to obtain local distributions $\{p_i^T\}$ and $\{p_j^S\}$.
  for each locally aligned pair $(i, j) \in \mathcal{A}(x)$ **do**
    Extract active teacher intents $\mathcal{K}_i \leftarrow \mathrm{TopK}(p_i^T, K)$.
    Retrieve student candidates $\mathcal{P}_i \leftarrow \{v : \exists u \in \mathcal{K}_i,\ f_{uv} > 0\}$.
    Expand teacher context $\mathcal{E}_i \leftarrow \mathcal{K}_i \cup \{u : \exists v \in \mathcal{P}_i,\ f_{uv} > 0\}$.
    if $\mathcal{P}_i = \emptyset$ **or** $\mathcal{E}_i = \emptyset$ **then**
      continue {skip OT for this position; CE supervision via $\mathcal{L}_{\mathrm{task}}$ still applies.}
    end if
    if $|\mathcal{P}_i| > \kappa$ or $|\mathcal{E}_i| > \kappa$ **then**
      Cap to κ: retain each $\mathcal{K}_i$ anchor's lowest-cost partner first, then fill by partner count, dropping all-forbidden rows/columns.
    end if
    Slice the support and cost: $\Omega_{ij} \leftarrow \mathcal{E}_i \times \mathcal{P}_i$, $\hat{C}_{ij} \leftarrow C[\mathcal{E}_i, \mathcal{P}_i]$; cells with $f_{uv} = 0$ receive a large finite sentinel cost.
    Set $a \leftarrow \mathrm{Normalize}(p_i^T|_{\mathcal{E}_i})$ and $b \leftarrow \mathrm{Normalize}(p_j^S|_{\mathcal{P}_i})$.
    Solve $\Gamma_{ij}^\star \leftarrow \arg\min_{\Gamma \in \Pi(a,b)} \langle \Gamma, \hat{C}_{ij} \rangle + \varepsilon \sum_{u,v} \Gamma_{uv}(\log \Gamma_{uv} - 1)$.
    Accumulate $\mathcal{L}_{\mathrm{ot}} \leftarrow \mathcal{L}_{\mathrm{ot}} + \langle \Gamma_{ij}^\star, \hat{C}_{ij} \rangle + \varepsilon \sum_{u,v} \Gamma_{ij,uv}^\star(\log \Gamma_{ij,uv}^\star - 1)$.
  end for
end for
Update θ using $\mathcal{L}_{\mathrm{task}}(\theta; \mathcal{B}) + \lambda \mathcal{L}_{\mathrm{ot}}$, with T, f, and C fixed.

student retains it as ["apple"], the baseline fails to synchronize the granularities, erroneously forcing the teacher's suffix ["e"] to align with the student's subsequent token ["a"]. This misalignment introduces significant noise, as the student receives supervision signals from text-unrelated tokens, disrupting the structural integrity of the sequence.

In contrast, Figure 4 (b) demonstrates the effectiveness of our Minimal Complete Correspondence (MCC) strategy. By identifying boundary pairs whose decoded prefixes match, MCC dynamically aggregates fragmented tokens (such as grouping ["appl", "e"]) into decoded-text-aligned blocks (dashed boxes) that map to the student's unified representation. The spans ["ke", "eps"] and ["kee", "ps"] likewise form one MCC block, as both decode to "keeps". This mechanism bridges the tokenization gap by establishing valid many-to-many correspondences, ensuring that the subsequent distillation process operates on text-synchronized units rather than superficial token positions.

K Broader Impact

SOLA enables distillation across heterogeneous tokenizers while retaining the original teacher and student vocabularies. It broadens the model pairs available for compression and can lower deployment

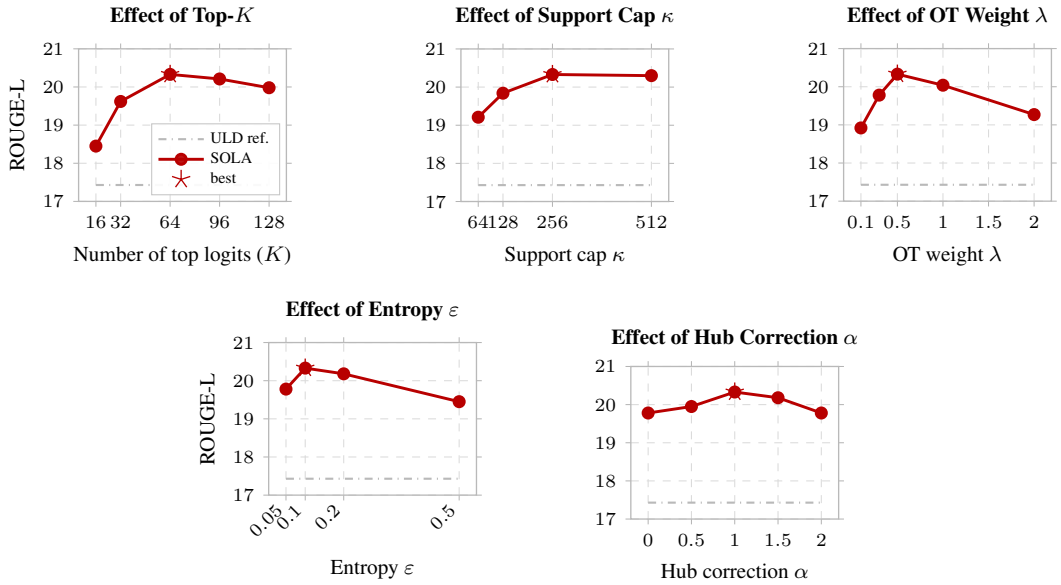
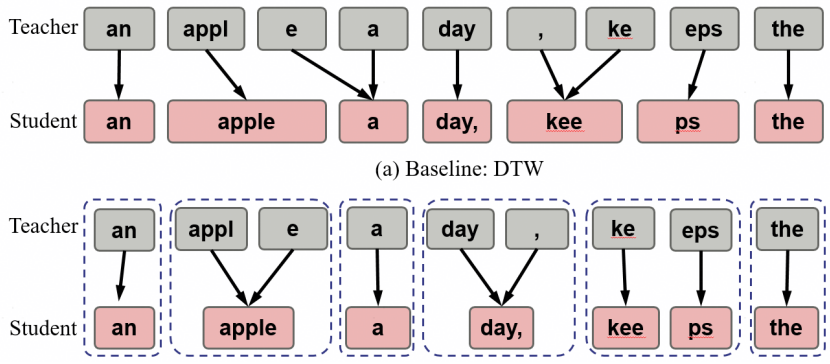


Figure 3: Hyperparameter sensitivity of SOLA on Dolly. Star: selected grid point. Dash-dot: ULD reference (17.43). The α=0 reference uses the + backward ablation in Tab. 4 (19.78).



(a) Baseline: DTW

(b) MCC: Minimal Complete Correspondence

Figure 4: Visualization of Alignment Strategies and MCC Effectiveness.

costs on resource-constrained hardware. Distillation can also transfer a teacher's biases, factual errors, and harmful response patterns to the student. Practitioners should evaluate the distilled student's safety, bias, and factuality before deployment.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction include the claims made in the paper

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The "limitations" section is included in the main text.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Proposition 3.2 states a fixed-support, free-marginal result. Appendix A provides its proof, the calibration-estimation assumptions, and the fixed-support stability bound.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: This paper discloses all the necessary information in the Experiments section and the Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Code will be released after notification.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: we specify the details of the experiments in the experiment section and the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report the std with 5 different seeds.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We report information such as computation cost.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: It complies with the Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discussed the broader impact in the appendix.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: the paper poses no such risks

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: the creators and original owners of all assets used in the paper are properly credited, and their licenses and terms of use are explicitly stated and fully respected.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: This paper does not introduce any new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: the core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.