EVOLUTION-AWARE DISTILLATION VIA OPTIMAL TRANSPORT FOR LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

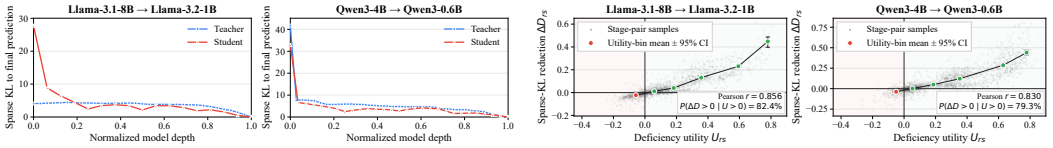
Knowledge distillation (KD) has become a widely adopted approach for compressing large language models (LLMs) for efficient deployment and inference. Existing LLM distillation methods can be broadly categorized into black-box and white-box paradigms. Black-box distillation transfers teacher-generated outputs but cannot fully exploit internal knowledge, while white-box distillation leverages logits or intermediate representations for finer-grained knowledge transfer. However, existing intermediate distillation methods usually rely on static layer-wise alignment and still suffer from two major limitations. First, predefined layer correspondence ignores that teacher and student predictions may evolve at different rates across model depth, causing layers at mismatched evolution stages to be aligned. Second, even with appropriate correspondence, indiscriminately transferring teacher evolution signals overlooks whether such knowledge actually complements the deficiencies of the student model. To address these issues, we propose an **ev**olution-**a**ware **k**nowledge **d**istillation framework (**EvaKD**) via optimal transport for LLMs, which dynamically aligns prediction evolution between teacher and student models and selectively transfers beneficial knowledge. Specifically, first, to overcome rigid layer alignment, we introduce prediction-evolution optimal transport, which projects intermediate representations into the prediction space, characterizes depthwise evolution through prediction changes between adjacent states, and establishes a soft many-to-many correspondence between teacher and student evolution via optimal transport. Second, to avoid indiscriminate knowledge transfer, we further propose deficiency-aware truncated distillation, which evaluates transported teacher evolution according to the remaining prediction deficiency of the student and retains only transport signals that contribute to closing this deficiency. Extensive experiments across multiple benchmarks and LLM families show that our method consistently outperforms strong LLM distillation baselines, improving ROUGE-L scores by up to 7.01% and accuracy scores by 7.4%. Our code is at https://anonymous.4open.science/r/EvaKD-CFCF/.

1 INTRODUCTION

Large language models (LLMs) have demonstrated remarkable capabilities across a broad range of natural language processing tasks, including text generation (Liu et al., 2025), code generation (Eghbali & Pradel, 2024), and complex reasoning (Monteiro et al., 2024). However, the rapidly increasing model scale introduces substantial computation and inference costs, which severely limit the deployment of LLMs in resource-constrained scenarios (Xu et al., 2024; Zhong et al., 2024). To address this challenge, knowledge distillation (KD) (Hinton et al., 2015) has become an important compression technique for LLMs, where a compact student model learns from a larger teacher model to retain its capabilities with considerably fewer parameters and faster inference (Ko et al., 2026).

Existing knowledge distillation methods for LLMs can be broadly categorized into black-box and white-box paradigms (Xu et al., 2024; Zhong et al., 2024). Black-box distillation typically transfers teacher-generated outputs, such as responses or chain-of-thought rationales, but cannot directly exploit the internal knowledge of the teacher (Ho et al., 2023; Hsieh et al., 2023). In contrast, white-box distillation accesses internal model signals to provide more fine-grained supervision (Agarwal et al., 2024; Gu et al., 2024; Ko et al., 2024), and can be further divided into output-layer distillation and intermediate-layer distillation. Specifically, output-layer distillation transfers knowledge through

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



(a) Prediction evolution. (b) Deficiency-aware transfer.

Figure 1: **Motivation analysis of prediction evolution and deficiency-aware transfer.** (a) **Prediction evolution.** The sparse KL divergence to the final prediction varies differently along normalized depth for teachers and students, showing that similar relative depths can correspond to different prediction-refinement stages and motivating evolution-aware alignment. (b) **Deficiency-aware transfer.** Each gray point denotes a teacher-student evolution pair (r, s), where U_{rs} measures the agreement between teacher evolution and student deficiency, and ΔD_{rs} denotes the sparse-KL reduction. Colored markers show utility-bin means with 95% confidence intervals. The positive correlation and high $P(\Delta D_{rs} > 0 \mid U_{rs} > 0)$ indicate that structurally corresponding teacher evolution can contribute differently across stages, motivating selective transfer based on student deficiency.

the final predictive distributions of the teacher (Ko et al., 2025; Wang et al., 2025), while the final-layer supervision overlooks the rich knowledge progressively formed across model depth (Zhang et al., 2024b). To exploit the internal knowledge, intermediate-layer distillation further transfers intermediate representations or predictions from teacher layers to student layers (Liang et al., 2023; Gong et al., 2025), providing finer-grained supervision beyond the final output of the teacher model.

Despite its potential, existing intermediate distillation still faces two fundamental challenges. **First, predefined layer correspondence can mismatch the prediction evolution of teacher and student models.** Specifically, existing methods align intermediate layers by identical or proportional depth, assuming that similar relative positions represent comparable learning stages. However, teacher and student models differ in depth, scale, and capacity, causing their predictions to evolve at different rates. Consequently, similar relative depths may correspond to different prediction-refinement stages. As shown in Figure 1(a), comparable prediction transitions can occur at different depths in the teacher and student. Therefore, rigid layer mappings may align inconsistent evolution stages, motivating evolution-based correspondence beyond layer position. **Second, even properly aligned teacher evolution is not necessarily beneficial to the student.** Generally, intermediate distillation transfers teacher knowledge once a correspondence is established. However, different student stages exhibit different remaining prediction deficiencies, making some teacher evolution redundant or inconsistent with the required correction direction. As illustrated in Figure 1(b), structurally similar teacher evolution can therefore contribute differently across student stages. Consequently, uniform transfer introduces redundant or conflicting supervision, motivating the deficiency-aware selection.

To address these issues, we propose **EvaKD**, an **ev**olution-**a**ware **k**nowledge **d**istillation framework for LLMs via optimal transport, to enable flexible and selective intermediate knowledge transfer. Specifically, EvaKD projects intermediate representations into a sparse prediction space and then aligns and transfers knowledge according to prediction evolution and student deficiency. First, to address rigid layer correspondence, we propose prediction-evolution optimal transport, which models consecutive prediction changes and establishes soft teacher-student correspondence based on evolution similarity and relative depth. Second, to avoid indiscriminate knowledge transfer, we propose deficiency-aware truncated distillation, which effectively evaluates aligned teacher evolution against student deficiency and retains beneficial transport mass for intermediate distillation.

We extensively evaluate EvaKD on domain-agnostic and domain-specific instruction-following benchmarks using Llama3 and Qwen3 teacher-student pairs with ten baselines for LLM distillation. The results demonstrate that EvaKD consistently improves over representative output-layer and intermediate-layer distillation baselines across different model families and tasks by up to 7.01% scores and 7.4% accuracy scores. Our main contributions are summarized as follows:

- We observe that teacher and student predictions evolve differently across depth and that aligned teacher evolution contributes unevenly across student stages, motivating EvaKD, an evolution-aware distillation framework for flexible and selective knowledge transfer.
- To address rigid layer correspondence, we propose prediction-evolution optimal transport, which aligns teacher and student evolution steps according to the prediction changes.

- To avoid indiscriminate knowledge transfer, we propose deficiency-aware truncated distillation, which retains teacher evolution that is beneficial to the current student deficiency.
- Extensive empirical results across multiple model families and benchmarks demonstrate that EvaKD consistently outperforms strong distillation baselines, validating the effectiveness of evolution-aware correspondence and deficiency-aware selective knowledge transfer.

2 Related Work

Knowledge distillation (KD) (Hinton et al., 2015) has become a widely adopted approach for compressing large language models (LLMs), which can be categorized into black-box and white-box distillation (Xu et al., 2024). **(I) Black-box distillation** treats the teacher as a data generator and trains the student using teacher-produced responses (Liu et al., 2026; He et al., 2026), including chain-of-thought or reasoning-oriented supervision (Lin et al., 2020; Zhuang et al., 2025; Kang et al., 2025). These methods are applicable when teacher internals are inaccessible and mainly transfer observable behaviors. **(II) White-box distillation** directly accesses internal teacher signals and can be further divided into *output-layer distillation* and *intermediate-layer distillation*. *Output-layer distillation* transfers knowledge through the final predictive distribution of the teacher, and has been studied with different divergence objectives (Gu et al., 2024; Agarwal et al., 2024; Ko et al., 2024; 2025; Wang et al., 2025), sampling strategies (Gao et al., 2025; Zhang et al., 2025; Zhu et al., 2026; Wu et al., 2026; Zhang et al., 2026a), reinforcement learning (Zhang et al., 2026b; Fu et al., 2026; Ko et al., 2026), and self-distillation (Zhao et al., 2026; Yang et al., 2026). *Intermediate-layer distillation* further exploits internal representations or layer-wise prediction signals to provide finer-grained supervision beyond final outputs (Liang et al., 2023; Zhang et al., 2024b; Gong et al., 2025). However, existing intermediate-layer methods often rely on predefined layer mappings and overlook prediction evolution and student deficiency. In contrast, our EvaKD learns flexible evolution-based correspondence and selectively transfers beneficial knowledge according to the student deficiency.

3 Methodology

3.1 Preliminary and Problem Statement

Knowledge Distillation for LLMs. Let $\mathcal{D} = \{(\mathbf{x}, \mathbf{y})\}$ denote the distillation dataset, where $\mathbf{x}$ is an input sequence and $\mathbf{y} = (y_1, \ldots, y_{|\mathbf{y}|})$ is the target response with length $|\mathbf{y}|$. We consider a teacher model $\mathcal{T}$ with $L^{\mathcal{T}}$ Transformer layers and a student model $\mathcal{S}$ with $L^{\mathcal{S}}$ layers. At token position t, let $\mathbf{p}_t^{\mathcal{T}}$ and $\mathbf{q}_t^{\mathcal{S}}$ denote the final-layer predictive distributions of the teacher and student over vocabulary $\mathcal{V}$, respectively. For any token $v \in \mathcal{V}$, $p_t^{\mathcal{T}}(v)$ and $q_t^{\mathcal{S}}(v)$ denote the scalar probabilities assigned to v by $\mathbf{p}_t^{\mathcal{T}}$ and $\mathbf{q}_t^{\mathcal{S}}$, respectively. In particular, $q_t^{\mathcal{S}}(y_t)$ denotes the probability assigned by the student to the target token y_t. The supervised fine-tuning objective of the student model is defined as follows:

$$\mathcal{L}_{\mathrm{SFT}} = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}[-\sum_{t=1}^{|\mathbf{y}|} \log q_t^{\mathcal{S}}(y_t)] \tag{1}$$

Let $\mathrm{KL}(\cdot\|\cdot)$ denote the Kullback-Leibler divergence (Kullback & Leibler, 1951). Then, the standard logit-based knowledge distillation objective of the student model is defined as follows:

$$\mathcal{L}_{\mathrm{KD}} = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}[\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} \mathrm{KL}(\mathbf{p}_t^{\mathcal{T}}\|\mathbf{q}_t^{\mathcal{S}})] \tag{2}$$

Intermediate Distillation for LLMs. Beyond final-layer knowledge distillation, intermediate distillation further transfers the internal knowledge. Let $\mathbf{f}_{t,l}^{\mathcal{T}}$ and $\mathbf{f}_{t,m}^{\mathcal{S}}$ denote the intermediate signals at token position t from the l-th teacher layer and the m-th student layer, respectively, where the intermediate signals can be hidden representations or their projected predictions. Let $\mathcal{A}$ denote a predefined set of teacher-student layer pairs (l, m), and let $\mathrm{Dist}(\cdot,\cdot)$ denote the discrepancy function between two intermediate signals. The intermediate distillation objective is defined as follows:

$$\mathcal{L}_{\mathrm{INT}} = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}[\frac{1}{|\mathbf{y}||\mathcal{A}|}\sum_{t=1}^{|\mathbf{y}|}\sum_{(l,m)\in\mathcal{A}} \mathrm{Dist}(\mathbf{f}_{t,l}^{\mathcal{T}}, \mathbf{f}_{t,m}^{\mathcal{S}})] \tag{3}$$

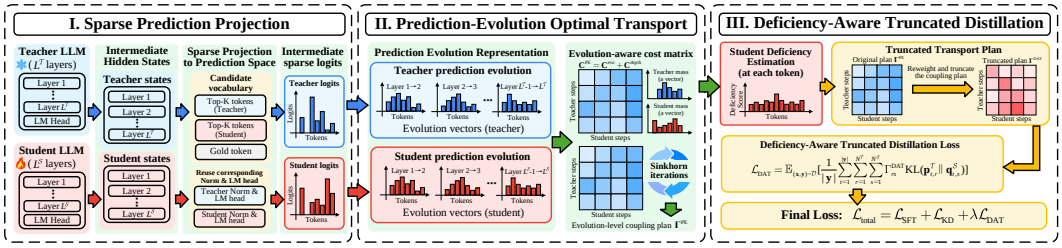


Figure 2: Overview of the EvaKD framework.

3.2 Framework

The framework of EvaKD is shown in Figure 2. First, we project intermediate representations into a sparse prediction space using the original normalization module and LM head to substantially reduce vocabulary computation. Second, we introduce prediction-evolution optimal transport, which models depthwise prediction changes to establish more flexible correspondence between teacher and student stages. Finally, we perform deficiency-aware truncated distillation, which filters teacher evolution based on student deficiency and redistributes useful transport mass for knowledge transfer.

3.3 Sparse Prediction Projection

Directly projecting every intermediate hidden state onto the full vocabulary introduces substantial computation and memory overhead. Meanwhile, most vocabulary tokens carry negligible probability for the current prediction. We therefore construct a token-specific candidate vocabulary using final predictions that are already available during standard knowledge distillation. Specifically, let $\mathbf{z}_t^{\mathcal{T}}$ and $\mathbf{z}_t^{\mathcal{S}}$ denote the final full-vocabulary logits of the teacher and student at token position t, respectively. Let K denote the number of top-ranked tokens selected from each model, and $\mathrm{TopK}(\mathbf{z}, K)$ returns the K tokens with the largest logits in $\mathbf{z}$. We construct the candidate vocabulary as follows:

$$\mathcal{C}_t = \mathrm{TopK}(\mathbf{z}_t^{\mathcal{T}}, K) \cup \mathrm{TopK}(\mathbf{z}_t^{\mathcal{S}}, K) \cup \{y_t\} \tag{4}$$

where duplicated tokens are removed. Therefore, $|\mathcal{C}_t| \leq 2K + 1$. The candidate set covers teacher-preferred tokens, student-preferred tokens, and the ground-truth token, preserving the prediction alternatives most relevant to distillation. Then, let $\mathbf{h}_{t,l}^{\mathcal{T}}$ and $\mathbf{h}_{t,l}^{\mathcal{S}}$ denote the hidden states produced by layer l of the teacher and student, respectively. Rather than introducing learned projection modules, we reuse the native normalization functions and LM heads of the two models. Let $\mathrm{Norm}^{\mathcal{T}}$ and $\mathrm{Norm}^{\mathcal{S}}$ denote the original final normalization functions of the teacher and student, respectively. Moreover, $\mathbf{W}_{\mathcal{C}_t}^{\mathcal{T}}$ and $\mathbf{W}_{\mathcal{C}_t}^{\mathcal{S}}$ denote the rows of their original LM-head matrices corresponding to tokens in $\mathcal{C}_t$. Finally, the resulting sparse intermediate logits $\mathbf{z}_{t,l}^{\mathcal{T}}$ and $\mathbf{z}_{t,l}^{\mathcal{S}}$ can be computed as follows:

$$\mathbf{z}_{t,l}^{\mathcal{T}} = \mathbf{W}_{\mathcal{C}_t}^{\mathcal{T}} \mathrm{Norm}^{\mathcal{T}}(\mathbf{h}_{t,l}^{\mathcal{T}}), \quad \mathbf{z}_{t,l}^{\mathcal{S}} = \mathbf{W}_{\mathcal{C}_t}^{\mathcal{S}} \mathrm{Norm}^{\mathcal{S}}(\mathbf{h}_{t,l}^{\mathcal{S}}) \tag{5}$$

In this manner, teacher and student intermediate states are expressed in a task-relevant prediction space without explicitly aligning their hidden dimensions.

3.4 Prediction-Evolution Optimal Transport

We regard the prediction change between two consecutive layers as a local evolution step. Let $N^{\mathcal{T}} = L^{\mathcal{T}} - 1$ and $N^{\mathcal{S}} = L^{\mathcal{S}} - 1$ denote the numbers of evolution steps in the teacher and student, respectively. For the r-th teacher evolution step, where $r \in \{1, \ldots, N^{\mathcal{T}}\}$, and the s-th student evolution step, where $s \in \{1, \ldots, N^{\mathcal{S}}\}$, we define their prediction changes as:

$$\Delta\mathbf{z}_{t,r}^{\mathcal{T}} = \mathbf{z}_{t,r+1}^{\mathcal{T}} - \mathbf{z}_{t,r}^{\mathcal{T}}, \quad \Delta\mathbf{z}_{t,s}^{\mathcal{S}} = \mathbf{z}_{t,s+1}^{\mathcal{S}} - \mathbf{z}_{t,s}^{\mathcal{S}} \tag{6}$$

These evolution vectors characterize how the prediction changes as the teacher and student move deeper through their networks. To measure whether two evolution steps represent similar prediction refinement, we first define an evolution cost. Let $\cos(\cdot, \cdot)$ denote cosine similarity between two vectors. The evolution cost between teacher step r and student step s is defined as:

$$C_{rs}^{\mathrm{evo}} = 1 - \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\Big[\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|}\cos(\Delta\mathbf{z}_{t,r}^{\mathcal{T}}, \Delta\mathbf{z}_{t,s}^{\mathcal{S}})\Big] \tag{7}$$

A smaller C_{rs}^{evo} indicates that the two evolution steps exhibit more consistent prediction changes. Besides evolution consistency, we introduce a normalized depth cost to preserve the relative ordering of model depth. The depth cost between steps r and s is defined as:

$$C_{rs}^{\text{depth}} = (\frac{r}{N^{\mathcal{T}}} - \frac{s}{N^{\mathcal{S}}})^2 \tag{8}$$

A smaller C_{rs}^{depth} indicates that the two evolution steps occur at more similar relative depths. We combine these two complementary costs without introducing an additional weighting coefficient. The prediction-evolution transport cost is defined as:

$$C_{rs}^{\text{PE}} = C_{rs}^{\text{evo}} + C_{rs}^{\text{depth}} \tag{9}$$

By calculating C_{rs}^{PE} for all teacher-student evolution pairs, we obtain the cost matrix $\mathbf{C}^{\text{PE}} \in \mathbb{R}^{N^{\mathcal{T}} \times N^{\mathcal{S}}}$. We then establish soft correspondence between teacher and student evolution steps through optimal transport. Let $\mathbf{\Gamma}^{\text{PE}} \in \mathbb{R}_{+}^{N^{\mathcal{T}} \times N^{\mathcal{S}}}$ denote the nonnegative transport matrix, where Γ_{rs}^{PE} represents the transport mass from the r-th teacher evolution step to the s-th student evolution step. We use uniform teacher and student marginals $\mathbf{a}^{\mathcal{T}}$ and $\mathbf{b}^{\mathcal{S}}$, whose elements are $a_r^{\mathcal{T}} = 1/N^{\mathcal{T}}$ and $b_s^{\mathcal{S}} = 1/N^{\mathcal{S}}$, respectively. Let $\mathbf{1}_n$ denote an n-dimensional all-ones vector, ε denote the entropy regularization coefficient, and $\langle \cdot, \cdot \rangle$ denote the Frobenius inner product between two matrices. The optimal transport problem is formulated as:

$$\mathbf{\Gamma}^{\text{PE}} = \underset{\mathbf{\Pi} \geq 0, \mathbf{\Pi} \mathbf{1}_{N^{\mathcal{S}}} = \mathbf{a}^{\mathcal{T}}, \mathbf{\Pi}^{\top} \mathbf{1}_{N^{\mathcal{T}}} = \mathbf{b}^{\mathcal{S}}}{\arg\min} [\langle \mathbf{\Pi}, \mathbf{C}^{\text{PE}} \rangle + \varepsilon \sum_{r=1}^{N^{\mathcal{T}}} \sum_{s=1}^{N^{\mathcal{S}}} \Pi_{rs} (\log \Pi_{rs} - 1)] \tag{10}$$

We solve Eq. 10 using standard log-domain Sinkhorn iterations and obtain the transport plan $\mathbf{\Gamma}^{\text{PE}}$. Detailed optimization steps of the Sinkhorn algorithm are provided in Algorithm 1 of Appendix B.1.

3.5 Deficiency-Aware Truncated Distillation

Although $\mathbf{\Gamma}^{\text{PE}}$ establishes flexible evolution-level correspondence between teacher and student models, some aligned teacher evolution may be less consistent with the current needs of the student. To further address this issue, we propose deficiency-aware truncated distillation, which estimates the remaining prediction deficiency of the student and selectively truncates the transport mass in $\mathbf{\Gamma}^{\text{PE}}$ according to its directional compatibility with this deficiency. Specifically, for the s-th evolution step, we use its destination state at layer $s+1$ and the final teacher prediction to characterize the remaining prediction deficiency. Let $\mathbf{d}_{t,s}^{\mathcal{S}}$ denote the deficiency vector of the student at position t:

$$\mathbf{d}_{t,s}^{\mathcal{S}} = \mathbf{z}_{t,L^{\mathcal{T}}}^{\mathcal{T}} - \mathbf{z}_{t,s+1}^{\mathcal{S}} \tag{11}$$

The vector $\mathbf{d}_{t,s}^{\mathcal{S}}$ represents the prediction-space direction required to move the current student state toward the final teacher prediction. We evaluate whether the r-th teacher evolution step follows this required direction. Let U_{rs} denote the deficiency utility between teacher step r and student step s:

$$U_{rs} = \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim \mathcal{D}} [\frac{1}{|\mathbf{y}|} \sum_{t=1}^{|\mathbf{y}|} \cos(\Delta \mathbf{z}_{t,r}^{\mathcal{T}}, \mathbf{d}_{t,s}^{\mathcal{S}})] \tag{12}$$

A positive U_{rs} indicates that the corresponding teacher evolution direction is aligned with the remaining deficiency of the student, whereas a non-positive value indicates a lack of positive directional alignment. We then use the deficiency utility to directly refine the transport plan $\mathbf{\Gamma}^{\text{PE}}$. Let $[x]_+ = \max(0, x)$ denote the positive-part function. For each teacher-student evolution pair (r, s), we truncate the original transport mass according to its deficiency utility:

$$\widetilde{\Gamma}_{rs} = \Gamma_{rs}^{\text{PE}} [U_{rs}]_+ \tag{13}$$

Thus, a correspondence receives transport mass only when it is supported by both the prediction-evolution alignment encoded in Γ_{rs}^{PE} and the student deficiency measured by U_{rs}. Pairs with $U_{rs} \leq 0$ are removed, while positive-utility correspondences retain transport mass proportional to their deficiency utility. To maintain a normalized transport distribution, we further normalize the retained mass. Let $\mathbf{\Gamma}^{\text{DAT}}$ denote the resulting deficiency-aware transport plan, whose element Γ_{rs}^{DAT} is:

$$\Gamma_{rs}^{\text{DAT}} = \frac{\widetilde{\Gamma}_{rs}}{\sum_{r'=1}^{N^{\mathcal{T}}} \sum_{s'=1}^{N^{\mathcal{S}}} \widetilde{\Gamma}_{r's'}} \tag{14}$$

Accordingly, U_{rs} is used as a directional compatibility criterion to guide transport selection, while the selected teacher destination states provide the subsequent distillation targets. Therefore, $\mathbf{\Gamma}^{\mathrm{DAT}}$ preserves the evolution-level correspondence learned by $\mathbf{\Gamma}^{\mathrm{PE}}$ while removing transport paths whose evolution directions are not positively aligned with student deficiency.

3.6 Distillation Objective

Given $\mathbf{\Gamma}^{\mathrm{DAT}}$, we use its transport mass to weight the distillation between aligned teacher and student prediction states. Let τ denote the distillation temperature. The sparse predictive distributions at the destination states of teacher evolution step r and student evolution step s are denoted by $\mathbf{p}^{\mathcal{T}}_{t,r}$ and $\mathbf{q}^{\mathcal{S}}_{t,s}$, respectively, and are computed as:

$$\mathbf{p}^{\mathcal{T}}_{t,r} = \mathrm{softmax}(\mathbf{z}^{\mathcal{T}}_{t,r+1}/\tau), \qquad \mathbf{q}^{\mathcal{S}}_{t,s} = \mathrm{softmax}(\mathbf{z}^{\mathcal{S}}_{t,s+1}/\tau) \tag{15}$$

We then define the deficiency-aware truncated distillation loss by weighting each teacher-student prediction discrepancy with its corresponding transport mass $\Gamma^{\mathrm{DAT}}_{rs}$:

$$\mathcal{L}_{\mathrm{DAT}} = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\Big[\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|}\sum_{r=1}^{N^{\mathcal{T}}}\sum_{s=1}^{N^{\mathcal{S}}}\Gamma^{\mathrm{DAT}}_{rs}\mathrm{KL}(\mathbf{p}^{\mathcal{T}}_{t,r}\|\mathbf{q}^{\mathcal{S}}_{t,s})\Big] \tag{16}$$

Finally, let λ denote the weight of the proposed intermediate distillation objective. The overall training objective is defined as:

$$\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{SFT}} + \mathcal{L}_{\mathrm{KD}} + \lambda\mathcal{L}_{\mathrm{DAT}} \tag{17}$$

Through this formulation, $\mathbf{\Gamma}^{\mathrm{PE}}$ first determines where knowledge should be transferred according to prediction evolution, while deficiency-aware truncation further determines what knowledge should be retained according to the current needs of the student. The resulting $\mathbf{\Gamma}^{\mathrm{DAT}}$ directly weights the intermediate distillation loss and guides selective knowledge transfer to the student.

4 Experiments

4.1 Experimental Settings

Datasets. We conduct experiments on domain-agnostic and domain-specific benchmarks. (I) For domain-agnostic evaluation, we perform distillation on Databricks-Dolly-15k (**Dolly**) (Conover et al., 2023) and evaluate on Dolly, Self-Instruct (**SelfInst**) (Wang et al., 2023), Vicuna Evaluation (**Vicuna**) (Chiang et al., 2023), Super-Natural Instructions (**S-NI**) (Wang et al., 2022), and Unnatural Instructions (**UnNI**) (Honovich et al., 2023). (II) For domain-specific evaluation, we consider two knowledge-intensive domains. (1) **Medical**: we perform distillation on GenMedGPT (Li et al., 2023) and evaluate on three medical subsets of MMLU (Hendrycks et al., 2021), including clinical knowledge (**CK**), professional medicine (**PM**), and college medicine (**CM**). (2) **Legal**: we perform distillation on Lawyer-Instruct (Alignment-Lab-AI, 2024) and USTerms (Chalkidis et al., 2023), and evaluate on the international law (**IL**) and professional law (**PL**) subsets of MMLU. In addition, we report results on reasoning benchmarks, including **GSM8K** (Cobbe et al., 2021), **MBPP** (Austin et al., 2021), and **HumanEval** (Chen et al., 2021) in Appendix C.1 to evaluate reasoning abilities.

Models. We evaluate EvaKD across different LLM families and model scales. In main experiments, we use **Llama3.1-8B** (Grattafiori et al., 2024) and **Qwen3-4B** (Yang et al., 2025) as teacher models, and **Llama3.2-1B** (Grattafiori et al., 2024) and **Qwen3-0.6B** (Yang et al., 2025) as their corresponding student models. Accordingly, we consider two main teacher-student pairs: **Llama3.1-8B** $\rightarrow$ **Llama3.2-1B** and **Qwen3-4B** $\rightarrow$ **Qwen3-0.6B**. To further examine the larger teacher, we perform **Llama2-13B** (Touvron et al., 2023) $\rightarrow$ **TinyLlama-1.1B** (Zhang et al., 2024a) in Appendix C.2.

Baselines. We compare EvaKD with ten representative LLM knowledge distillation methods. (I) **Output-layer distillation** transfers knowledge primarily through the final predictions of the teacher, including **KD** (Hinton et al., 2015), **SeqKD** (Kim & Rush, 2016), **ImitKD** (Lin et al., 2020), **GKD** (Agarwal et al., 2024), **MiniLLM** (Gu et al., 2024), **Distillm** (Ko et al., 2024), **Distillm-2** (Ko et al., 2025), and **ABKD** (Wang et al., 2025). (II) **Intermediate-layer distillation** further exploits intermediate representations of the teacher and includes **TED** (Liang et al., 2023) and **FDD** (Gong et al., 2025). More implementation details of our method and baselines are listed in Appendix B.3.

Table 1: Main results with ROUGE-L (%) on domain-agnostic datasets.

	Llama3.1-8B → Llama3.2-1B						Qwen3-4B → Qwen3-0.6B					
Methods	**Dolly**	**SelfInst**	**S-NI**	**UnNI**	**Vicuna**	**Avg.**	**Dolly**	**SelfInst**	**S-NI**	**UnNI**	**Vicuna**	**Avg.**
Teacher	31.21	25.17	38.17	37.72	19.82	30.42	34.53	30.21	48.22	47.26	24.03	36.85
Student	23.94	14.73	28.63	27.01	15.92	22.05	27.07	22.09	41.26	43.35	18.95	30.54
KD	25.72	15.19	29.06	28.19	16.19	22.87	28.53	23.16	42.52	43.74	19.79	31.55
SeqKD	26.47	16.08	30.57	31.06	16.42	24.12	28.42	23.71	41.98	43.47	18.43	31.20
ImitKD	25.73	15.81	26.71	27.31	17.95	22.70	28.75	23.14	42.64	43.64	18.47	31.33
GKD	26.95	16.60	31.76	30.61	17.06	24.60	28.16	23.68	42.68	43.86	18.39	31.35
MiniLLM	28.25	18.10	34.36	33.41	17.61	26.35	28.31	23.69	43.02	43.66	18.44	31.42
Distillm	28.33	19.23	37.60	35.65	18.24	27.81	28.49	23.57	42.81	44.76	18.75	31.68
Distillm-2	28.42	19.32	37.68	35.74	18.41	27.91	28.62	23.41	42.14	44.98	18.65	31.56
ABKD	28.50	19.28	37.59	35.83	18.42	27.92	28.47	23.51	42.46	44.22	18.23	31.38
TED	28.34	19.52	36.19	35.67	17.83	27.51	28.97	23.95	42.45	44.68	19.89	31.99
FDD	28.96	20.45	37.02	37.11	17.27	28.16	29.21	24.08	42.71	45.02	20.34	32.27
EvaKD	**30.02**	**23.23**	**38.61**	**38.19**	**18.49**	**29.71**	**29.56**	**24.77**	**43.07**	**45.63**	**21.23**	**32.85**

Table 2: Main results with accuracy on domain-specific datasets.

	Llama3.1-8B → Llama3.2-1B						Qwen3-4B → Qwen3-0.6B					
Methods	**CK**	**PM**	**CM**	**PL**	**IL**	**Avg.**	**CK**	**PM**	**CM**	**PL**	**IL**	**Avg.**
Teacher	0.59	0.78	0.57	0.44	0.63	0.602	0.65	0.68	0.73	0.49	0.82	0.674
Student	0.33	0.24	0.31	0.22	0.18	0.256	0.49	0.31	0.41	0.18	0.59	0.396
KD	0.36	0.26	0.33	0.27	0.25	0.294	0.53	0.31	0.45	0.20	0.61	0.420
SeqKD	0.35	0.29	0.34	**0.29**	0.26	0.306	0.53	0.29	0.45	0.23	0.61	0.422
ImitKD	0.38	0.34	0.35	0.26	0.25	0.316	0.55	0.33	0.46	0.21	0.63	0.436
GKD	0.39	0.29	0.32	**0.29**	0.27	0.312	0.55	0.35	0.42	0.23	0.59	0.428
MiniLLM	0.37	0.28	0.33	0.27	0.24	0.298	0.53	0.33	0.45	0.23	0.57	0.422
Distillm	0.41	0.31	0.35	0.26	0.26	0.318	0.55	0.36	0.47	0.24	0.57	0.438
Distillm-2	0.43	0.28	0.36	0.28	0.28	0.326	0.57	0.38	0.47	0.25	0.57	0.448
ABKD	0.42	0.32	0.34	0.28	0.26	0.324	0.57	**0.39**	0.46	0.25	0.59	0.452
TED	0.43	0.33	0.36	0.28	0.32	0.344	0.58	0.34	0.46	0.22	0.61	0.442
FDD	0.42	0.32	0.37	0.27	0.31	0.338	0.56	0.37	0.47	0.24	0.63	0.454
EvaKD	**0.44**	**0.35**	**0.39**	**0.29**	**0.37**	**0.368**	**0.61**	**0.39**	**0.51**	**0.27**	**0.64**	**0.484**

4.2 Experimental Results

Main Results on Domain-Agnostic Benchmarks. To evaluate EvaKD on domain-agnostic settings, we perform knowledge distillation from Llama3.1-8B to Llama3.2-1B and from Qwen3-4B to Qwen3-0.6B, with the results reported in Table 1. Overall, EvaKD achieves the best performance across all benchmarks and obtains the highest average ROUGE-L on both teacher-student pairs, demonstrating consistent effectiveness across different model families and scales. For Llama3.2-1B, EvaKD achieves an average ROUGE-L of 29.71, outperforming the strongest intermediate-layer baseline FDD by +1.55↑ and the strongest output-layer baseline ABKD by +1.79↑. Similarly, on Qwen3-0.6B, EvaKD reaches 32.85 on average, surpassing FDD by +0.58↑ and Distillm by +1.17↑. Moreover, EvaKD shows strong generalization beyond the distillation corpus. For example, on Llama3.2-1B, EvaKD improves SelfInst, S-NI, and UnNI by +2.78↑, +0.93↑, and +1.08↑, respectively, over the strongest baselines. These gains on held-out benchmarks indicate that EvaKD improves not only in-domain distillation performance but also transferability to unseen scenarios.

Also, we conduct a GPT-5-based pairwise evaluation on 100 sampled examples, as shown in Figure 3. For Llama3.2-1B, EvaKD wins against different baselines in 51%–62% of the comparisons, with the highest win rate of 62% against FDD. For Qwen3-0.6B, the win rates range from 47% to 55%, and EvaKD obtains more wins than losses against nearly all baselines. These results further show that improvements in ROUGE-L also reflect in generated response quality under LLM-based evaluation.

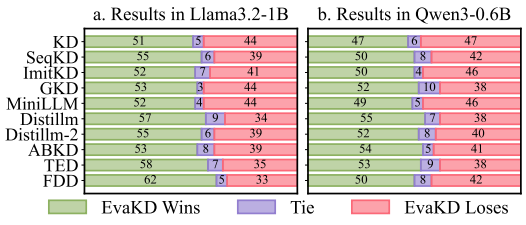


Figure 3: Main results with GPT-5 scores of one hundred samples on the domain-agnostic datasets.

Table 3: Ablation results with ROUGE-L (%) on domain-agnostic benchmarks.

	Llama3.1-8B → Llama3.2-1B						Qwen3-4B → Qwen3-0.6B					
Methods	**Dolly**	**SelfInst**	**S-NI**	**UnNI**	**Vicuna**	**Avg.**	**Dolly**	**SelfInst**	**S-NI**	**UnNI**	**Vicuna**	**Avg.**
EvaKD	**30.02**	**23.23**	**38.61**	**38.19**	**18.49**	**29.71**	**29.56**	**24.77**	**43.07**	**45.63**	**21.23**	**32.85**
w/o PEOT	28.46	21.32	36.43	36.55	16.36	27.82	27.57	19.72	38.79	41.58	20.05	29.54
w/o DAT	29.32	22.29	36.99	37.26	17.42	28.66	28.14	22.86	42.73	42.42	20.56	31.34

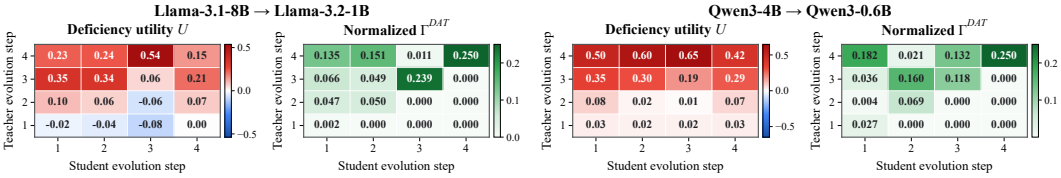


Figure 4: Analysis of Deficiency-Aware Knowledge Selection.

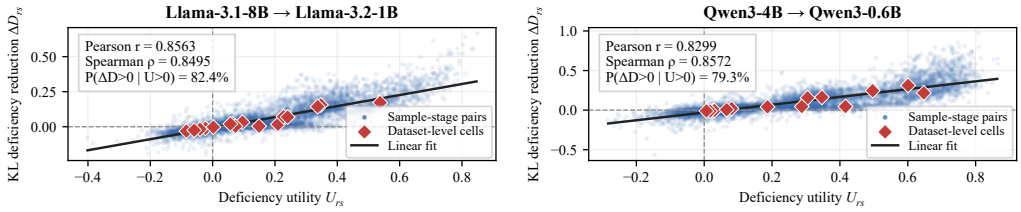
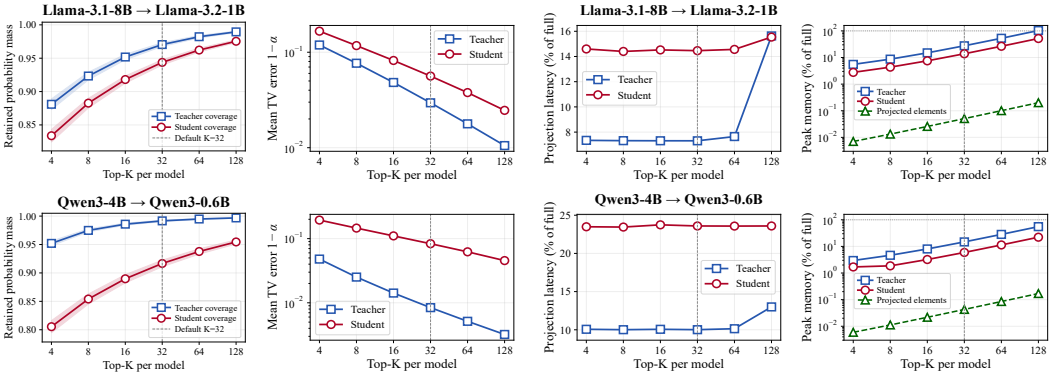


Figure 5: Analysis of Utility and Deficiency Reduction.

Main Results on Domain-Specific Benchmarks. To evaluate EvaKD in knowledge-intensive domains, we conduct distillation on medical and legal datasets, with the results reported in Table 2. Overall, EvaKD achieves the highest average accuracy on both teacher-student pairs. For Llama3.2-1B, EvaKD reaches an average accuracy of 0.368, outperforming the strongest baseline TED by +0.024↑ and the strongest output-layer baseline ABKD by +0.044↑. Similarly, on Qwen3-0.6B, EvaKD achieves 0.484 on average, surpassing FDD by +0.030↑ and ABKD by +0.032↑. Moreover, EvaKD consistently improves performance across both medical and legal tasks. For example, on Llama3.2-1B, it improves PM, CM, and IL by +0.01↑, +0.02↑, and +0.05↑ over the strongest baselines, respectively. On Qwen3-0.6B, it further achieves gains of +0.03↑ on CK, +0.04↑ on CM, and +0.02↑ on PL. These results indicate that evolution-aware and deficiency-aware intermediate transfer remains consistently effective for preserving fine-grained knowledge in the specialized domains.

Ablation Results. We evaluate the two core components in EvaKD: **PEOT** denotes prediction-evolution optimal transport, and **DAT** denotes deficiency-aware truncated distillation. Results are reported in Table 3. First, removing PEOT substantially degrades performance on both teacher-student pairs. The average ROUGE-L drops from 29.71 to 27.82 on Llama3.2-1B and from 32.85 to 29.54 on Qwen3-0.6B, decreases of 1.89 and 3.31 points, respectively. This confirms that evolution-aware correspondence is critical for avoiding mismatched intermediate supervision. Second, removing DAT also reduces performance, with the averages decreasing from 29.71 to 28.66 and from 32.85 to 31.34, respectively. These drops of 1.05 and 1.51 points indicate that deficiency-aware selection provides complementary benefits beyond correspondence learning. Overall, these results verify that both the PEOT and DAT components are essential to the effectiveness and robustness of EvaKD.

Analysis of Deficiency-Aware Knowledge Selection. To examine how DAT selects teacher evolution, Figure 4 visualizes the deficiency utility U_{rs} and the resulting normalized transport mass $\Gamma_{rs}^{\mathrm{DAT}}$. Larger U_{rs} indicates stronger alignment between teacher evolution and student deficiency, while larger $\Gamma_{rs}^{\mathrm{DAT}}$ indicates greater retained transport weight. First, utility varies substantially across teacher evolution steps. For Llama3, 4 of 16 pairs have negative utility, and the first teacher step is mostly low or negative. For Qwen3, the first two teacher steps show lower utility than later steps. This confirms that aligned teacher evolution contributes unequally across student stages and should not be transferred uniformly. Second, DAT concentrates mass on higher-utility teacher evolution. The last two teacher steps account for approximately 90.1% and 89.9% of the normalized DAT mass on Llama3 and Qwen3, respectively, with the final step contributing 54.7% and 58.5%. This indicates that DAT prioritizes evolution signals that better complement the remaining student deficiency.



(a) Vocabulary coverage. (b) Sparse projection accuracy-efficiency trade-off.

Figure 6: Analysis of sparse prediction projection.

Analysis of Utility and Deficiency Reduction. Figure 5 examines whether the deficiency utility U_{rs} reflects the benefit of teacher evolution. Each blue point denotes a sample-stage pair, red diamonds denote aggregated cells, and the black line is the linear fit. A positive ΔD_{rs} indicates that teacher evolution reduces the sparse KL divergence to the final teacher prediction. First, U_{rs} is strongly correlated with deficiency reduction, with Pearson correlations of 0.8563 and 0.8299 and Spearman correlations of 0.8495 and 0.8572 on Llama3 and Qwen3, respectively. Second, when $U_{rs} > 0$, 82.4% and 79.3% of updates yield positive KL reduction. These results show that U_{rs} is a reliable indicator of beneficial teacher evolution and supports deficiency-aware selection in DAT.

Analysis of Sparse Prediction Projection. Figure 6 analyzes the accuracy-efficiency trade-off of sparse prediction projection under different candidate sizes K. In Figure 6(a), retained probability measures the probability mass covered by the candidate vocabulary, while the corresponding TV error reflects the approximation error. In Figure 6(b), projection latency and peak memory are reported relative to full-vocabulary projection. First, increasing K steadily improves vocabulary coverage while reducing approximation error. At the default $K = 32$, the candidate vocabulary already retains 91.66%-99.16% of the prediction probability across teacher and student models, indicating that a compact vocabulary preserves most predictive information. Further increasing K to 64 yields only limited coverage gains, showing diminishing returns. Second, sparse projection substantially reduces computation and memory overhead. At $K = 32$, its projection latency is only 7.31%-23.59% of full projection, while peak memory is reduced to 5.92%-27.55%. These results demonstrate that $K = 32$ provides a balance between prediction fidelity and projection efficiency.

Analysis of Depth-wise Deficiency Utility. To examine whether teacher information is equally useful across student depths, we evaluate the PEOT-weighted positive transport ratio and average deficiency utility at four normalized student depths on the Dolly validation set. The positive transport ratio measures the fraction of PEOT mass assigned to teacher evolution with $U_{rs} > 0$, while the average utility reflects its overall compatibility with student deficiency. As shown in Figure 7, both metrics generally increase with student depth. For example, the positive ratio on Llama3 rises from 33.80% to 95.77%, while the average utility increases from -0.010 to 0.193. These results show that conflicting teacher evolution is concentrated in earlier stages, supporting depth-adaptive knowledge selection in DAT.

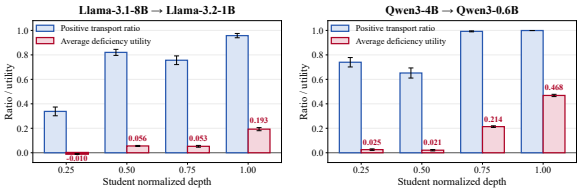


Figure 7: Analysis of depth-wise deficiency utility.

5 Conclusion

In this work, we propose EvaKD, an evolution-aware distillation framework for LLMs that enables flexible and selective intermediate knowledge transfer. To address rigid layer correspondence, we propose prediction-evolution optimal transport, which aligns teacher and student evolution according to depthwise prediction changes. To avoid indiscriminate transfer, we propose deficiency-aware truncated distillation, which selectively retains teacher evolution based on student deficiency. Extensive experiments across diverse benchmarks and models demonstrate the effectiveness of EvaKD.

AI USE STATEMENT

In this work, we used generative AI tools solely for language polishing, including improving grammar, clarity, readability, and academic expression of the manuscript. In addition, we used GPT-5 as an automatic evaluator to assess the quality of generated responses in the evaluation experiments. We have not used generative AI tools for generating research ideas, developing the methodology, implementing experimental code, analyzing experimental results, or drawing scientific conclusions; other disclosure tasks are not applicable to this work. We have carefully reviewed and revised all AI-assisted text to ensure its accuracy, originality, and consistency with the intended technical content. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *ICLR*, 2024.

Alignment-Lab-AI. Lawyer-instruct. Hugging Face Datasets, 2024. URL https://huggingface.co/datasets/Alignment-Lab-AI/Lawyer-Instruct.

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

Ilias Chalkidis, Nicolas Garneau, Catalina Goanta, Daniel Katz, and Anders Søgaard. LeXFiles and LegalLAMA: Facilitating English multinational legal language model development. In *ACL*, 2023.

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.

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, 2023. URL https://lmsys.org/blog/2023-03-30-vicuna/.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

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, 2023.

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 *AAAI*, 2025.

Aryaz Eghbali and Michael Pradel. De-hallucinator: Iterative grounding for llm-based code completion. *arXiv preprint arXiv:2401.01701*, 2024.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Jiacai Liu, Zhuo Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. In *COLM*, 2026.

Shiping Gao, Fanqi Wan, Jiajian Guo, Xiaojun Quan, and Qifan Wang. Advantage-guided distillation for preference alignment in small language models. In *ICLR*, 2025.

Guoqiang Gong, Jiaxing Wang, Jin Xu, Deping Xiang, Zicheng Zhang, Leqi Shen, Yifeng Zhang, Junhua Shu, Zhaolong Xing, Zhen Chen, Pengzhang Liu, and Ke Zhang. Beyond logits: Aligning feature dynamics for effective knowledge distillation. In *ACL*, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *ICLR*, 2024.

Bowei He, Yankai Chen, Xiaokun Zhang, Linghe Kong, Philip S Yu, Xue Liu, and Chen Ma. Pedagogically-inspired data synthesis for language model knowledge distillation. In *ICLR*, 2026.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *ICLR*, 2021.

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

Namgyu Ho, Laura Schmid, and Se-Young Yun. Large language models are reasoning teachers. In *ACL*, 2023.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. Unnatural instructions: Tuning language models with (almost) no human labor. In *ACL*, 2023.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of ACL*, 2023.

Minki Kang, Jongwon Jeong, Seanie Lee, Jaewoong Cho, and Sung Ju Hwang. Distilling llm agent into small models with retrieval and code tools. In *NeurIPS*, 2025.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *EMNLP*, 2016.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. Distillm: towards streamlined distillation for large language models. In *ICML*, 2024.

Jongwoo Ko, Tianyi Chen, Sungnyun Kim, Tianyu Ding, Luming Liang, Ilya Zharkov, and Se-Young Yun. Distillm-2: A contrastive approach boosts the distillation of llms. In *ICML*, 2025.

Jongwoo Ko, Sara Abdali, Young Jin Kim, Tianyi Chen, and Pashmina Cameron. Scaling reasoning efficiently via relaxed on-policy distillation. *arXiv preprint arXiv:2603.11137*, 2026.

Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

Yunxiang Li, Zihan Li, Kai Zhang, Ruilong Dan, Steve Jiang, and You Zhang. Chatdoctor: A medical chat model fine-tuned on a large language model meta-ai (llama) using medical domain knowledge. *Cureus*, 15(6), 2023.

Chen Liang, Simiao Zuo, Qingru Zhang, Pengcheng He, Weizhu Chen, and Tuo Zhao. Less is more: task-aware layer-wise distillation for language model compression. In *ICML*, 2023.

Alexander Lin, Jeremy Wohlwend, Howard Chen, and Tao Lei. Autoregressive knowledge distillation through imitation learning. In *EMNLP*, 2020.

Kaiyuan Liu, Shaotian Yan, Rui Miao, Bing Wang, Chen Shen, Jun Zhang, and Jieping Ye. Where did this sentence come from? tracing provenance in LLM reasoning distillation. In *ICLR*, 2026.

Mushui Liu, Yuhang Ma, Zhen Yang, Jun Dan, Yunlong Yu, Zeng Zhao, Zhipeng Hu, Bai Liu, and Changjie Fan. Llm4gen: Leveraging semantic representation of llms for text-to-image generation. In *AAAI*, 2025.

Joao Monteiro, Pierre-Andre Noel, Etienne Marcotte, Sai Rajeswar Mudumba, Valentina Zantedeschi, David Vazquez, Nicolas Chapados, Chris Pal, and Perouz Taslakian. Repliqa: A question-answering dataset for benchmarking llms on unseen reference content. In *NeurIPS*, 2024.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Guanghui Wang, Zhiyong Yang, Zitai Wang, Shi Wang, Qianqian Xu, and Qingming Huang. Abkd: Pursuing a proper allocation of the probability mass in knowledge distillation via α-β-divergence. In *ICML*, 2025.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. *arXiv preprint arXiv:2204.07705*, 2022.

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 *ACL*, 2023.

Yecheng Wu, Song Han, and Hai Cai. Lightning opd: Efficient post-training for large reasoning models with offline on-policy distillation. *arXiv preprint arXiv:2604.13010*, 2026.

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.

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.

Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled rlvr. *arXiv preprint arXiv:2604.03128*, 2026.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. Tinyllama: An open-source small language model. *arXiv preprint arXiv:2401.02385*, 2024a.

Songming Zhang, Xue Zhang, Zengkui Sun, Yufeng Chen, and Jinan Xu. Dual-space knowledge distillation for large language models. In *EMNLP*, 2024b.

Songming Zhang, Xue Zhang, Tong Zhang, Bojie Hu, Yufeng Chen, and Jinan Xu. Alignditil: Token-level language model alignment as adaptive policy distillation. In *ACL*, 2025.

Xinsen Zhang, Zhenkai Ding, Tianjun Pan, Run Yang, Chun Kang, Xue Xiong, and Jingnan Gu. Opsdl: On-policy self-distillation for long-context language models. *arXiv preprint arXiv:2604.17535*, 2026a.

Zhaoyang Zhang, Shuli Jiang, Yantao Shen, Yuting Zhang, Dhananjay Ram, Shuo Yang, Zhuowen Tu, Wei Xia, and Stefano Soatto. Reinforcement-aware knowledge distillation for llm reasoning. *arXiv preprint arXiv:2602.22495*, 2026b.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

Qihuang Zhong, Liang Ding, Li Shen, Juhua Liu, Bo Du, and Dacheng Tao. Revisiting knowledge distillation for autoregressive language models. In *ACL*, 2024.

Wenhong Zhu, Ruobing Xie, Rui Wang, and Pengfei Liu. Hybrid policy distillation for LLMs. In *ICML*, 2026.

Xianwei Zhuang, Zhihong Zhu, Zhichang Wang, Xuxin Cheng, and Yuexian Zou. Unicott: A unified framework for structural chain-of-thought distillation. In *ICLR*, 2025.

APPENDIX CONTENTS

A Theoretical Analysis **14**
A.1 Properties of Sparse Prediction Projection 14
A.2 Properties of Prediction-Evolution Optimal Transport 16
A.3 Properties of Deficiency-Aware Truncated Distillation 20

B Methodological and Experimental Details **25**
B.1 Details of Optimization of Prediction-Evolution Optimal Transport 25
B.2 Details of Datasets 27
B.3 Details of Baselines 28
B.4 Details of Implementation Settings 29
B.5 Details of Analysis Experiments 29

C Additional Experimental Results **32**
C.1 Results on Reasoning Benchmarks 32
C.2 Results on Larger Teacher Models 32
C.3 Results on Heterogeneous Distillation 33
C.4 Analysis of Cross-Task Prediction Evolution 33
C.5 Case Study 36

A Theoretical Analysis

In this section, we provide theoretical analyses of the three main components of EvaKD. We first characterize the approximation induced by sparse prediction projection, then establish the existence and structure of the prediction-evolution optimal transport solution, and finally analyze the effect of deficiency-aware truncation on transport utility.

A.1 Properties of Sparse Prediction Projection

This section analyzes the sparse prediction projection introduced in Sec. 3. We first show that the sparse prediction is exactly the full-vocabulary prediction conditioned on the candidate vocabulary. We then quantify the approximation induced by discarding the remaining vocabulary and characterize its relation to full-vocabulary teacher-student distillation. Finally, we show that the depthwise prediction changes used in prediction-evolution optimal transport are exactly preserved on the selected coordinates.

Setup. Fix an input-output pair $(\mathbf{x}, \mathbf{y})$, token position t, and candidate vocabulary $\mathcal{C}_t$ defined in Eq. 4. For model $\mathcal{X} \in \{\mathcal{T}, \mathcal{S}\}$, let $\mathbf{W}^{\mathcal{X}} \in \mathbb{R}^{|\mathcal{V}| \times d_{\mathcal{X}}}$ denote the original LM head, and define the hypothetical full-vocabulary intermediate logits:

$$\overline{\mathbf{z}}^{\mathcal{X}}_{t,l} = \mathbf{W}^{\mathcal{X}} \operatorname{Norm}^{\mathcal{X}}(\mathbf{h}^{\mathcal{X}}_{t,l}). \tag{18}$$

Let $\mathbf{R}_t \in \{0,1\}^{|\mathcal{C}_t| \times |\mathcal{V}|}$ be the coordinate-selection matrix associated with $\mathcal{C}_t$, such that $\mathbf{W}^{\mathcal{X}}_{\mathcal{C}_t} = \mathbf{R}_t \mathbf{W}^{\mathcal{X}}$. For temperature $\tau > 0$, define:

$$\boldsymbol{\pi}^{\mathcal{X}}_{t,l} = \operatorname{softmax}(\overline{\mathbf{z}}^{\mathcal{X}}_{t,l}/\tau), \qquad \alpha^{\mathcal{X}}_{t,l} = \sum_{v \in \mathcal{C}_t} \pi^{\mathcal{X}}_{t,l}(v). \tag{19}$$

Exact conditionalization of sparse prediction. The first result clarifies what information is changed by sparse projection. It shows that selecting rows from the original LM head does not modify any retained logit, and the only effect of sparse softmax is to renormalize the full distribution within $\mathcal{C}_t$.

Proposition 1 (Exact conditionalization of sparse prediction)**.** *For any model $\mathcal{X}$, token t, and layer l, the sparse logits satisfy:*

$$\mathbf{z}^{\mathcal{X}}_{t,l} = \mathbf{R}_t \overline{\mathbf{z}}^{\mathcal{X}}_{t,l}. \tag{20}$$

Let

$$\boldsymbol{\pi}^{\mathcal{X},\mathcal{C}}_{t,l} = \operatorname{softmax}(\mathbf{z}^{\mathcal{X}}_{t,l}/\tau). \tag{21}$$

Then, for every $v \in \mathcal{C}_t$:

$$\pi^{\mathcal{X},\mathcal{C}}_{t,l}(v) = \frac{\pi^{\mathcal{X}}_{t,l}(v)}{\alpha^{\mathcal{X}}_{t,l}} = \pi^{\mathcal{X}}_{t,l}(v \mid v \in \mathcal{C}_t). \tag{22}$$

Proof. Using $\mathbf{W}^{\mathcal{X}}_{\mathcal{C}_t} = \mathbf{R}_t \mathbf{W}^{\mathcal{X}}$ and Eq. 18:

$$\begin{aligned} \mathbf{z}^{\mathcal{X}}_{t,l} &= \mathbf{W}^{\mathcal{X}}_{\mathcal{C}_t} \operatorname{Norm}^{\mathcal{X}}(\mathbf{h}^{\mathcal{X}}_{t,l}) \\ &= \mathbf{R}_t \mathbf{W}^{\mathcal{X}} \operatorname{Norm}^{\mathcal{X}}(\mathbf{h}^{\mathcal{X}}_{t,l}) = \mathbf{R}_t \overline{\mathbf{z}}^{\mathcal{X}}_{t,l}. \end{aligned} \tag{23}$$

Hence, each retained sparse logit equals its full-vocabulary counterpart. Therefore, for $v \in \mathcal{C}_t$:

$$\begin{aligned} \pi^{\mathcal{X},\mathcal{C}}_{t,l}(v) &= \frac{\exp(\overline{z}^{\mathcal{X}}_{t,l}(v)/\tau)}{\sum_{u \in \mathcal{C}_t} \exp(\overline{z}^{\mathcal{X}}_{t,l}(u)/\tau)} \\ &= \frac{\dfrac{\exp(\overline{z}^{\mathcal{X}}_{t,l}(v)/\tau)}{\sum_{w \in \mathcal{V}} \exp(\overline{z}^{\mathcal{X}}_{t,l}(w)/\tau)}}{\displaystyle\sum_{u \in \mathcal{C}_t} \frac{\exp(\overline{z}^{\mathcal{X}}_{t,l}(u)/\tau)}{\sum_{w \in \mathcal{V}} \exp(\overline{z}^{\mathcal{X}}_{t,l}(w)/\tau)}} \\ &= \frac{\pi^{\mathcal{X}}_{t,l}(v)}{\alpha^{\mathcal{X}}_{t,l}}, \end{aligned} \tag{24}$$

which is exactly the conditional distribution in Eq. 22. □

Thus, sparse projection does not alter the relative probabilities of retained tokens. In particular, for any $u, v \in \mathcal{C}_t$:

$$\frac{\pi^{\mathcal{X},\mathcal{C}}_{t,l}(v)}{\pi^{\mathcal{X},\mathcal{C}}_{t,l}(u)} = \frac{\pi^{\mathcal{X}}_{t,l}(v)}{\pi^{\mathcal{X}}_{t,l}(u)}. \tag{25}$$

Truncation error and its relation to distillation. Proposition 1 shows that prediction inside $\mathcal{C}_t$ is exact after conditioning. The next result quantifies the information discarded outside $\mathcal{C}_t$ and connects the sparse teacher-student KL divergence to its full-vocabulary counterpart.

Theorem 1 (Error characterization and KL decomposition)**.** *Define the zero-extended sparse distribution:*

$$\widetilde{\pi}^{\mathcal{X},\mathcal{C}}_{t,l}(v) = \begin{cases} \pi^{\mathcal{X},\mathcal{C}}_{t,l}(v), & v \in \mathcal{C}_t, \\ 0, & v \notin \mathcal{C}_t. \end{cases} \tag{26}$$

Then

$$d_{\mathrm{TV}}\left(\boldsymbol{\pi}^{\mathcal{X}}_{t,l}, \widetilde{\boldsymbol{\pi}}^{\mathcal{X},\mathcal{C}}_{t,l}\right) = 1 - \alpha^{\mathcal{X}}_{t,l}, \qquad \mathrm{KL}\left(\widetilde{\boldsymbol{\pi}}^{\mathcal{X},\mathcal{C}}_{t,l} \| \boldsymbol{\pi}^{\mathcal{X}}_{t,l}\right) = -\log \alpha^{\mathcal{X}}_{t,l}. \tag{27}$$

Moreover, let P and Q denote arbitrary full-vocabulary teacher and student distributions at token t, and define

$$\alpha = P(\mathcal{C}_t), \qquad \beta = Q(\mathcal{C}_t). \tag{28}$$

If $\mathcal{C}_t \neq \mathcal{V}$, then

$$\begin{aligned} \mathrm{KL}(P\|Q) &= \mathrm{KL}\big(\mathrm{Ber}(\alpha)\|\,\mathrm{Ber}(\beta)\big) \\ &\quad + \alpha \mathrm{KL}(P_{\mathcal{C}}\|Q_{\mathcal{C}}) + (1-\alpha)\mathrm{KL}(P_{\mathcal{C}^c}\|Q_{\mathcal{C}^c}), \end{aligned} \tag{29}$$

where $P_{\mathcal{C}}$ and $Q_{\mathcal{C}}$ are the corresponding conditional distributions on $\mathcal{C}_t$. Consequently,

$$\alpha \mathrm{KL}(P_{\mathcal{C}}\|Q_{\mathcal{C}}) \leq \mathrm{KL}(P\|Q). \tag{30}$$

Proof. Using Proposition 1,

$$\begin{aligned} 2d_{\mathrm{TV}} &= \sum_{v\in\mathcal{C}_t} \left| \pi^{\mathcal{X}}_{t,l}(v) - \frac{\pi^{\mathcal{X}}_{t,l}(v)}{\alpha^{\mathcal{X}}_{t,l}} \right| + \sum_{v\notin\mathcal{C}_t} \pi^{\mathcal{X}}_{t,l}(v) \\ &= \alpha^{\mathcal{X}}_{t,l}\left(\frac{1}{\alpha^{\mathcal{X}}_{t,l}} - 1\right) + \left(1 - \alpha^{\mathcal{X}}_{t,l}\right) \\ &= 2\left(1 - \alpha^{\mathcal{X}}_{t,l}\right), \end{aligned} \tag{31}$$

which proves the first identity in Eq. 27. Similarly,

$$\begin{aligned} \mathrm{KL}\left(\widetilde{\boldsymbol{\pi}}^{\mathcal{X},\mathcal{C}}_{t,l} \| \boldsymbol{\pi}^{\mathcal{X}}_{t,l}\right) &= \sum_{v\in\mathcal{C}_t} \frac{\pi^{\mathcal{X}}_{t,l}(v)}{\alpha^{\mathcal{X}}_{t,l}} \log \frac{1}{\alpha^{\mathcal{X}}_{t,l}} \\ &= -\log \alpha^{\mathcal{X}}_{t,l}, \end{aligned} \tag{32}$$

proving the second identity.

For the teacher–student decomposition, write:

$$\mathrm{KL}(P\|Q) = \sum_{v\in\mathcal{C}_t} P(v) \log \frac{P(v)}{Q(v)} + \sum_{v\notin\mathcal{C}_t} P(v) \log \frac{P(v)}{Q(v)}. \tag{33}$$

Since $P(v) = \alpha P_{\mathcal{C}}(v)$ and $Q(v) = \beta Q_{\mathcal{C}}(v)$ inside $\mathcal{C}_t$, the first term becomes:

$$\sum_{v\in\mathcal{C}_t} P(v) \log \frac{P(v)}{Q(v)} = \alpha \log \frac{\alpha}{\beta} + \alpha \mathrm{KL}(P_{\mathcal{C}}\|Q_{\mathcal{C}}). \tag{34}$$

Likewise, outside $\mathcal{C}_t$:

$$\begin{aligned} \sum_{v\notin\mathcal{C}_t} P(v) \log \frac{P(v)}{Q(v)} &= (1-\alpha) \log \frac{1-\alpha}{1-\beta} \\ &\quad + (1-\alpha)\mathrm{KL}(P_{\mathcal{C}^c}\|Q_{\mathcal{C}^c}). \end{aligned} \tag{35}$$

Adding the two expressions gives Eq. 29. Since every KL term is nonnegative, Eq. 30 follows directly. □

Theorem 1 shows that the approximation is controlled solely by the omitted probability mass. In particular, if $\alpha_{t,l}^{\mathcal{X}} \geq 1-\delta$ for $0 \leq \delta < 1$, then

$$d_{\mathrm{TV}} \leq \delta, \qquad \mathrm{KL}\left(\widetilde{\boldsymbol{\pi}}_{t,l}^{\mathcal{X},\mathcal{C}} \| \boldsymbol{\pi}_{t,l}^{\mathcal{X}}\right) \leq -\log(1-\delta). \tag{36}$$

At the final layer, because the model-specific top-K set is contained in $\mathcal{C}_t$,

$$\alpha_{t,L^{\mathcal{X}}}^{\mathcal{X}} \geq \sum_{v \in \mathrm{TopK}(\mathbf{z}_t^{\mathcal{X}}, K)} \pi_{t,L^{\mathcal{X}}}^{\mathcal{X}}(v). \tag{37}$$

Preservation of prediction evolution. The previous results concern individual prediction states, whereas PEOT operates on differences between adjacent layers. The final result shows that sparse projection preserves these evolution vectors exactly on the retained vocabulary coordinates.

Proposition 2 (Coordinate preservation of prediction evolution)**.** *Define the full-vocabulary and sparse evolution vectors as:*

$$\Delta\overline{\mathbf{z}}_{t,l}^{\mathcal{X}} = \overline{\mathbf{z}}_{t,l+1}^{\mathcal{X}} - \overline{\mathbf{z}}_{t,l}^{\mathcal{X}}, \qquad \Delta\mathbf{z}_{t,l}^{\mathcal{X}} = \mathbf{z}_{t,l+1}^{\mathcal{X}} - \mathbf{z}_{t,l}^{\mathcal{X}}. \tag{38}$$

Then

$$\Delta\mathbf{z}_{t,l}^{\mathcal{X}} = \mathbf{R}_t \Delta\overline{\mathbf{z}}_{t,l}^{\mathcal{X}}. \tag{39}$$

If $\mathbf{R}_t^c$ selects the complementary coordinates $\mathcal{V} \setminus \mathcal{C}_t$, then

$$\left\|\Delta\overline{\mathbf{z}}_{t,l}^{\mathcal{X}}\right\|_2^2 = \left\|\Delta\mathbf{z}_{t,l}^{\mathcal{X}}\right\|_2^2 + \left\|\mathbf{R}_t^c \Delta\overline{\mathbf{z}}_{t,l}^{\mathcal{X}}\right\|_2^2. \tag{40}$$

Proof. By Eq. 20,

$$\begin{aligned} \Delta\mathbf{z}_{t,l}^{\mathcal{X}} &= \mathbf{R}_t \overline{\mathbf{z}}_{t,l+1}^{\mathcal{X}} - \mathbf{R}_t \overline{\mathbf{z}}_{t,l}^{\mathcal{X}} \\ &= \mathbf{R}_t \left(\overline{\mathbf{z}}_{t,l+1}^{\mathcal{X}} - \overline{\mathbf{z}}_{t,l}^{\mathcal{X}}\right) = \mathbf{R}_t \Delta\overline{\mathbf{z}}_{t,l}^{\mathcal{X}}, \end{aligned} \tag{41}$$

which proves Eq. 39.

Since $\mathbf{R}_t$ and $\mathbf{R}_t^c$ partition the vocabulary coordinates,

$$\mathbf{R}_t^\top \mathbf{R}_t + (\mathbf{R}_t^c)^\top \mathbf{R}_t^c = \mathbf{I}. \tag{42}$$

Therefore, for $\mathbf{a} = \Delta\overline{\mathbf{z}}_{t,l}^{\mathcal{X}}$,

$$\begin{aligned} \|\mathbf{a}\|_2^2 &= \mathbf{a}^\top \left[\mathbf{R}_t^\top \mathbf{R}_t + (\mathbf{R}_t^c)^\top \mathbf{R}_t^c\right] \mathbf{a} \\ &= \|\mathbf{R}_t \mathbf{a}\|_2^2 + \|\mathbf{R}_t^c \mathbf{a}\|_2^2, \end{aligned} \tag{43}$$

which yields Eq. 40. □

Summary. The above analysis establishes three properties of sparse prediction projection. First, it preserves the original LM-head logits exactly on $\mathcal{C}_t$ and produces the corresponding full-vocabulary prediction conditioned on this candidate set. Second, its approximation error is determined by the excluded probability mass, while the sparse teacher–student KL constitutes an exact conditional component of the full KL divergence. Third, the adjacent-layer prediction changes used by PEOT are exactly preserved on the selected coordinates, with all discarded evolution confined to the complementary vocabulary dimensions.

A.2 Properties of Prediction-Evolution Optimal Transport

This section analyzes the prediction-evolution optimal transport (PEOT) introduced in Sec. 3.4. We first establish that the entropy-regularized transport problem is well posed and admits a unique positive solution with the Sinkhorn scaling structure. We then quantify how entropy regularization trades exact transport-cost minimization for soft many-to-many correspondence. Finally, we show that the resulting transport plan is stable against perturbations in the estimated prediction-evolution cost matrix.

Setup. Let

$$n = N^{\mathcal{T}}, \qquad m = N^{\mathcal{S}}, \tag{44}$$

and let $\mathbf{C}^{\mathrm{PE}} \in \mathbb{R}^{n\times m}$ denote the prediction-evolution cost matrix defined in Eq. 9. We assume that the cosine similarities used to construct $\mathbf{C}^{\mathrm{PE}}$ are well defined, such that every entry C^{PE}_{rs} is finite. Since cosine similarity lies in $[-1, 1]$ and the normalized depth difference lies in $[-1, 1]$, we have:

$$0 \le C^{\mathrm{evo}}_{rs} \le 2, \qquad 0 \le C^{\mathrm{depth}}_{rs} \le 1, \qquad 0 \le C^{\mathrm{PE}}_{rs} \le 3. \tag{45}$$

Let $\mathbf{a}^{\mathcal{T}} \in \mathbb{R}^n$ and $\mathbf{b}^{\mathcal{S}} \in \mathbb{R}^m$ denote the uniform marginals, i.e., $a^{\mathcal{T}}_r = 1/n$ and $b^{\mathcal{S}}_s = 1/m$. Define the feasible transport polytope as:

$$\mathcal{U} = \left\{ \mathbf{\Pi} \in \mathbb{R}^{n\times m}_{+} : \mathbf{\Pi}\mathbf{1}_m = \mathbf{a}^{\mathcal{T}},\ \mathbf{\Pi}^{\top}\mathbf{1}_n = \mathbf{b}^{\mathcal{S}} \right\}. \tag{46}$$

Every feasible transport plan has total mass:

$$\sum_{r=1}^{n}\sum_{s=1}^{m} \Pi_{rs} = 1. \tag{47}$$

For $\varepsilon > 0$, define:

$$F_{\mathbf{C}}(\mathbf{\Pi}) = \langle \mathbf{\Pi}, \mathbf{C} \rangle + \varepsilon \sum_{r=1}^{n}\sum_{s=1}^{m} \Pi_{rs}(\log \Pi_{rs} - 1), \tag{48}$$

with the convention $0 \log 0 = 0$. The PEOT plan is therefore:

$$\mathbf{\Gamma}^{\mathrm{PE}} = \arg\min_{\mathbf{\Pi}\in\mathcal{U}} F_{\mathbf{C}^{\mathrm{PE}}}(\mathbf{\Pi}). \tag{49}$$

Well-posedness and Sinkhorn structure. Before using $\mathbf{\Gamma}^{\mathrm{PE}}$ as a soft layer-evolution correspondence, it is necessary to verify that Eq. 49 has a well-defined and unique solution. The following theorem further derives the multiplicative structure underlying the Sinkhorn iterations used in our implementation.

Theorem 2 (Existence, uniqueness, and Sinkhorn structure of PEOT)**.** *For any finite cost matrix $\mathbf{C}^{\mathrm{PE}}$ and any $\varepsilon > 0$, the optimization problem in Eq. 49 admits a unique solution $\mathbf{\Gamma}^{\mathrm{PE}}$. Moreover,*

$$\Gamma^{\mathrm{PE}}_{rs} > 0, \qquad \forall r, s, \tag{50}$$

and there exist positive scaling vectors $\mathbf{u} \in \mathbb{R}^n_+$ and $\mathbf{v} \in \mathbb{R}^m_+$ such that:

$$\mathbf{\Gamma}^{\mathrm{PE}} = \mathrm{diag}(\mathbf{u})\mathbf{K}\,\mathrm{diag}(\mathbf{v}), \qquad K_{rs} = \exp\left(-\frac{C^{\mathrm{PE}}_{rs}}{\varepsilon}\right). \tag{51}$$

The scaling vectors satisfy:

$$\mathbf{u} = \mathbf{a}^{\mathcal{T}} \oslash (\mathbf{K}\mathbf{v}), \qquad \mathbf{v} = \mathbf{b}^{\mathcal{S}} \oslash (\mathbf{K}^{\top}\mathbf{u}). \tag{52}$$

Proof. We first establish existence. The matrix

$$\mathbf{\Pi}^0 = \mathbf{a}^{\mathcal{T}}(\mathbf{b}^{\mathcal{S}})^{\top} \tag{53}$$

satisfies:

$$\begin{aligned} \mathbf{\Pi}^0\mathbf{1}_m &= \mathbf{a}^{\mathcal{T}}(\mathbf{b}^{\mathcal{S}})^{\top}\mathbf{1}_m = \mathbf{a}^{\mathcal{T}}, \\ (\mathbf{\Pi}^0)^{\top}\mathbf{1}_n &= \mathbf{b}^{\mathcal{S}}(\mathbf{a}^{\mathcal{T}})^{\top}\mathbf{1}_n = \mathbf{b}^{\mathcal{S}}, \end{aligned} \tag{54}$$

because both marginals sum to one. Hence $\mathcal{U}$ is nonempty. It is also closed and bounded because $\Pi_{rs} \ge 0$ and $\sum_{r,s} \Pi_{rs} = 1$. Therefore, $\mathcal{U}$ is compact.

The function

$$\phi(x) = x(\log x - 1) \tag{55}$$

is continuous on $[0, \infty)$ under $0 \log 0 = 0$ and strictly convex, since for $x > 0$,

$$\phi''(x) = \frac{1}{x} > 0. \tag{56}$$

Thus, the entropy term in Eq. 48 is strictly convex, whereas the transport-cost term is linear. Consequently, $F_{\mathbf{C}^{\mathrm{PE}}}$ is strictly convex on the convex set $\mathcal{U}$. Continuity on the compact feasible set guarantees existence, and strict convexity guarantees uniqueness.

We next show that the unique minimizer is strictly positive. Suppose, for contradiction, that $\Gamma^{\mathrm{PE}}_{rs} = 0$ for at least one pair (r, s). Since $\mathbf{\Pi}^0_{rs} = a^{\mathcal{T}}_r b^{\mathcal{S}}_s > 0$, define for $\theta \in (0, 1)$:

$$\mathbf{\Pi}(\theta) = (1-\theta)\mathbf{\Gamma}^{\mathrm{PE}} + \theta\mathbf{\Pi}^0. \tag{57}$$

By convexity of $\mathcal{U}$, $\mathbf{\Pi}(\theta) \in \mathcal{U}$. For any zero entry $\Gamma^{\mathrm{PE}}_{rs} = 0$, its contribution to the entropy difference satisfies:

$$\begin{aligned}\frac{\phi(\Pi_{rs}(\theta)) - \phi(0)}{\theta} &= \frac{\phi(\theta\Pi^0_{rs})}{\theta} \\ &= \Pi^0_{rs}\left[\log\theta + \log\Pi^0_{rs} - 1\right] \longrightarrow -\infty\end{aligned} \tag{58}$$

as $\theta \to 0^+$. For every positive entry of $\mathbf{\Gamma}^{\mathrm{PE}}$, the corresponding directional derivative is finite, and the linear transport-cost term also has a finite directional derivative because $\mathbf{C}^{\mathrm{PE}}$ is finite. Therefore,

$$\lim_{\theta\to 0^+} \frac{F_{\mathbf{C}^{\mathrm{PE}}}(\mathbf{\Pi}(\theta)) - F_{\mathbf{C}^{\mathrm{PE}}}(\mathbf{\Gamma}^{\mathrm{PE}})}{\theta} = -\infty, \tag{59}$$

which implies $F_{\mathbf{C}^{\mathrm{PE}}}(\mathbf{\Pi}(\theta)) < F_{\mathbf{C}^{\mathrm{PE}}}(\mathbf{\Gamma}^{\mathrm{PE}})$ for sufficiently small $\theta > 0$. This contradicts optimality, proving Eq. 50. Because the optimum is strictly positive, the nonnegativity constraints are inactive. Introduce Lagrange multipliers α_r and β_s for the marginal constraints. The Lagrangian is:

$$\begin{aligned}\mathcal{J} = &\sum_{r,s}\Pi_{rs}C^{\mathrm{PE}}_{rs} + \varepsilon\sum_{r,s}\Pi_{rs}(\log\Pi_{rs} - 1) \\ &+ \sum_r \alpha_r\left(a^{\mathcal{T}}_r - \sum_s \Pi_{rs}\right) + \sum_s \beta_s\left(b^{\mathcal{S}}_s - \sum_r \Pi_{rs}\right).\end{aligned} \tag{60}$$

The stationarity condition at the optimum gives:

$$\begin{aligned}0 = \frac{\partial\mathcal{J}}{\partial\Pi_{rs}} &= C^{\mathrm{PE}}_{rs} + \varepsilon\log\Gamma^{\mathrm{PE}}_{rs} - \alpha_r - \beta_s, \\ \Gamma^{\mathrm{PE}}_{rs} &= \exp\left(\frac{\alpha_r + \beta_s - C^{\mathrm{PE}}_{rs}}{\varepsilon}\right) \\ &= \exp(\alpha_r/\varepsilon)\exp(-C^{\mathrm{PE}}_{rs}/\varepsilon)\exp(\beta_s/\varepsilon).\end{aligned} \tag{61}$$

Defining $u_r = \exp(\alpha_r/\varepsilon)$, $v_s = \exp(\beta_s/\varepsilon)$, and $K_{rs} = \exp(-C^{\mathrm{PE}}_{rs}/\varepsilon)$ gives Eq. 51. Finally, substituting this form into the two marginal constraints yields:

$$\begin{aligned}u_r\sum_{s=1}^{m} K_{rs}v_s &= a^{\mathcal{T}}_r, \\ v_s\sum_{r=1}^{n} K_{rs}u_r &= b^{\mathcal{S}}_s,\end{aligned} \tag{62}$$

which is equivalent to Eq. 52. □

Theorem 2 shows that entropy regularization produces a unique transport plan with positive mass on every teacher-student evolution pair. The correspondence is therefore intrinsically soft rather than restricted to a predefined one-to-one layer mapping.

Effect of entropy regularization on evolution alignment. The previous theorem explains why PEOT yields a soft correspondence. However, entropy regularization changes the original minimum-cost transport problem. The following proposition quantifies this change and shows that the increase in prediction-evolution transport cost is explicitly controlled by ε.

Proposition 3 (Transport-cost gap induced by entropy regularization)**.** *Let*

$$\mathbf{\Gamma}^0 \in \arg\min_{\mathbf{\Pi}\in\mathcal{U}} \left\langle \mathbf{\Pi}, \mathbf{C}^{\mathrm{PE}} \right\rangle \tag{63}$$

be any unregularized optimal transport plan. Then

$$0 \le \left\langle \mathbf{\Gamma}^{\mathrm{PE}}, \mathbf{C}^{\mathrm{PE}} \right\rangle - \left\langle \mathbf{\Gamma}^0, \mathbf{C}^{\mathrm{PE}} \right\rangle \le \varepsilon \log(nm). \tag{64}$$

Proof. For any feasible transport plan, define its entropy as:

$$H(\mathbf{\Pi}) = -\sum_{r,s} \Pi_{rs} \log \Pi_{rs}. \tag{65}$$

Using $\sum_{r,s} \Pi_{rs} = 1$, Eq. 48 can be rewritten as:

$$F_{\mathbf{C}^{\mathrm{PE}}}(\mathbf{\Pi}) = \left\langle \mathbf{\Pi}, \mathbf{C}^{\mathrm{PE}} \right\rangle - \varepsilon H(\mathbf{\Pi}) - \varepsilon. \tag{66}$$

Since $\mathbf{\Gamma}^{\mathrm{PE}}$ minimizes this objective,

$$\left\langle \mathbf{\Gamma}^{\mathrm{PE}}, \mathbf{C}^{\mathrm{PE}} \right\rangle - \varepsilon H(\mathbf{\Gamma}^{\mathrm{PE}}) \le \left\langle \mathbf{\Gamma}^0, \mathbf{C}^{\mathrm{PE}} \right\rangle - \varepsilon H(\mathbf{\Gamma}^0).$$

Rearranging gives:

$$\left\langle \mathbf{\Gamma}^{\mathrm{PE}}, \mathbf{C}^{\mathrm{PE}} \right\rangle - \left\langle \mathbf{\Gamma}^0, \mathbf{C}^{\mathrm{PE}} \right\rangle \le \varepsilon \left[H(\mathbf{\Gamma}^{\mathrm{PE}}) - H(\mathbf{\Gamma}^0) \right].$$

A transport plan is a probability distribution over nm pairs. Hence,

$$0 \le H(\mathbf{\Pi}) \le \log(nm), \tag{67}$$

which applied to Eq. 67 yields:

$$\left\langle \mathbf{\Gamma}^{\mathrm{PE}}, \mathbf{C}^{\mathrm{PE}} \right\rangle - \left\langle \mathbf{\Gamma}^0, \mathbf{C}^{\mathrm{PE}} \right\rangle \le \varepsilon \log(nm). \tag{68}$$

The lower bound follows directly from the definition of $\mathbf{\Gamma}^0$ as a minimizer of the unregularized transport cost:

$$\left\langle \mathbf{\Gamma}^0, \mathbf{C}^{\mathrm{PE}} \right\rangle \le \left\langle \mathbf{\Gamma}^{\mathrm{PE}}, \mathbf{C}^{\mathrm{PE}} \right\rangle. \tag{69}$$

Combining the two inequalities proves Eq. 64. □

Proposition 3 provides an explicit interpretation of ε. A larger ε encourages a smoother transport plan, while the additional prediction-evolution matching cost relative to exact minimum-cost transport is bounded linearly by ε.

Stability under prediction-evolution cost perturbations. The entries of $\mathbf{C}^{\mathrm{PE}}$ are estimated by averaging prediction-evolution similarities over data and tokens. It is therefore important that small perturbations of this estimated cost do not cause arbitrarily large changes in the learned correspondence. The following theorem establishes such a stability guarantee.

Theorem 3 (Stability of PEOT under cost perturbations)**.** *Let $\mathbf{C}, \mathbf{C}' \in \mathbb{R}^{n\times m}$ be two finite cost matrices satisfying:*

$$\|\mathbf{C} - \mathbf{C}'\|_\infty \le \eta, \tag{70}$$

and let $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$ be their respective entropy-regularized optimal transport plans over the same feasible set $\mathcal{U}$ with regularization coefficient $\varepsilon > 0$. Then

$$\|\mathbf{\Gamma} - \mathbf{\Gamma}'\|_1 \le \frac{\eta}{\varepsilon}. \tag{71}$$

Proof. Define

$$h(\mathbf{\Pi}) = \sum_{r,s} \Pi_{rs} \log \Pi_{rs}. \tag{72}$$

By Theorem 2, both $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$ are strictly positive. For any two positive probability matrices $\mathbf{P}$ and $\mathbf{Q}$ with total mass one, the Bregman divergence induced by h is:

$$\begin{aligned} &h(\mathbf{P}) - h(\mathbf{Q}) - \langle \nabla h(\mathbf{Q}), \mathbf{P} - \mathbf{Q} \rangle \\ &= \sum_{r,s} P_{rs} \log \frac{P_{rs}}{Q_{rs}} = \mathrm{KL}(\mathbf{P} \| \mathbf{Q}). \end{aligned} \tag{73}$$

By Pinsker's inequality,

$$\mathrm{KL}(\mathbf{P}\|\mathbf{Q}) \geq \frac{1}{2}\|\mathbf{P}-\mathbf{Q}\|_1^2. \tag{74}$$

Hence the negative entropy h is 1-strongly convex with respect to the ℓ_1 norm on the probability simplex.

Consider:

$$F_{\mathbf{C}}(\mathbf{\Pi}) = \langle \mathbf{\Pi}, \mathbf{C}\rangle + \varepsilon h(\mathbf{\Pi}), \tag{75}$$

where the constant $-\varepsilon$ from Eq. 48 is omitted because it does not affect the optimizer. Since $\mathbf{\Gamma}$ minimizes $F_{\mathbf{C}}$ over the convex set $\mathcal{U}$, first-order optimality gives:

$$\langle \nabla F_{\mathbf{C}}(\mathbf{\Gamma}), \mathbf{\Gamma}' - \mathbf{\Gamma}\rangle \geq 0. \tag{76}$$

Combining Eq. 76 with the strong convexity obtained from Eq. 74,

$$F_{\mathbf{C}}(\mathbf{\Gamma}') - F_{\mathbf{C}}(\mathbf{\Gamma}) \geq \frac{\varepsilon}{2}\|\mathbf{\Gamma}' - \mathbf{\Gamma}\|_1^2. \tag{77}$$

Applying the same argument to $\mathbf{C}'$ and $\mathbf{\Gamma}'$ yields

$$F_{\mathbf{C}'}(\mathbf{\Gamma}) - F_{\mathbf{C}'}(\mathbf{\Gamma}') \geq \frac{\varepsilon}{2}\|\mathbf{\Gamma}' - \mathbf{\Gamma}\|_1^2. \tag{78}$$

Adding Eqs. 77 and 78 cancels the entropy terms and gives:

$$\begin{aligned}\varepsilon\|\mathbf{\Gamma}' - \mathbf{\Gamma}\|_1^2 &\leq \langle \mathbf{C} - \mathbf{C}', \mathbf{\Gamma}' - \mathbf{\Gamma}\rangle \\ &\leq \|\mathbf{C} - \mathbf{C}'\|_\infty \|\mathbf{\Gamma}' - \mathbf{\Gamma}\|_1 \\ &\leq \eta\|\mathbf{\Gamma}' - \mathbf{\Gamma}\|_1,\end{aligned} \tag{79}$$

where the second inequality follows from Hölder's inequality and the third from Eq. 70. If $\mathbf{\Gamma} = \mathbf{\Gamma}'$, Eq. 71 holds trivially. Otherwise, dividing Eq. 79 by $\varepsilon\|\mathbf{\Gamma}' - \mathbf{\Gamma}\|_1$ gives:

$$\|\mathbf{\Gamma}' - \mathbf{\Gamma}\|_1 \leq \frac{\eta}{\varepsilon}, \tag{80}$$

which proves Eq. 71. □

Theorem 3 shows that entropy regularization not only yields a soft correspondence but also controls its sensitivity to estimation noise in the prediction-evolution cost. For a fixed perturbation magnitude η, increasing ε makes the transport plan less sensitive to the estimated cost, consistent with its smoothing effect.

Summary. The above analysis establishes three properties of PEOT. First, the entropy-regularized transport problem has a unique strictly positive solution with the Sinkhorn scaling structure, formally supporting soft many-to-many evolution alignment. Second, the transport-cost deviation from unregularized optimal transport is bounded by $\varepsilon \log(N^{\mathcal{T}} N^{\mathcal{S}})$, quantifying the cost of entropy-induced smoothing. Third, perturbing the prediction-evolution cost matrix by at most η changes the learned transport plan by at most η/ε in ℓ_1 distance, providing an explicit stability guarantee for the estimated evolution correspondence.

A.3 Properties of Deficiency-Aware Truncated Distillation

This section analyzes the deficiency-aware truncated distillation (DAT) introduced in Sec. 3.5. We first show that DAT admits an exact optimization interpretation as a utility-aware reweighting of the PEOT plan. We then prove that the resulting transport plan never decreases the transport-weighted deficiency utility and derive an exact expression for its improvement. Finally, we establish a local geometric interpretation of the deficiency utility, showing that a positive utility identifies a teacher evolution direction that decreases the average student prediction deficiency to first order under a direct infinitesimal movement along this direction.

Setup. For compactness, let

$$\mathcal{I} = \{(r, s) : 1 \le r \le N^{\mathcal{T}},\ 1 \le s \le N^{\mathcal{S}}\} \tag{81}$$

denote the set of all teacher-student evolution pairs. For each $i = (r, s) \in \mathcal{I}$, define:

$$P_i = \Gamma^{\mathrm{PE}}_{rs}, \qquad U_i = U_{rs}, \qquad U_i^+ = [U_i]_+ = \max(0, U_i). \tag{82}$$

Since $\mathbf{\Gamma}^{\mathrm{PE}}$ satisfies normalized transport marginals,

$$P_i \ge 0, \qquad \sum_{i \in \mathcal{I}} P_i = 1. \tag{83}$$

Moreover, because U_i is an average cosine similarity,

$$-1 \le U_i \le 1, \qquad 0 \le U_i^+ \le 1. \tag{84}$$

Define the retained utility mass:

$$Z = \sum_{i \in \mathcal{I}} P_i U_i^+. \tag{85}$$

When $Z > 0$, Eq. 14 is equivalently written as:

$$Q_i = \Gamma^{\mathrm{DAT}}_{rs} = \frac{P_i U_i^+}{Z}. \tag{86}$$

When $Z = 0$, DAT follows the fallback rule in Sec. 3.5 and sets $\mathbf{\Gamma}^{\mathrm{DAT}} = \mathbf{\Gamma}^{\mathrm{PE}}$.

Optimization interpretation of deficiency-aware truncation. The truncation in Eq. 86 combines two factors: the original evolution correspondence P_i and the deficiency utility U_i^+. The following proposition is introduced to show that this multiplicative form is not arbitrary. It is the unique solution of a KL-regularized reweighting problem that preserves the PEOT correspondence while favoring transport paths with larger positive deficiency utility.

Proposition 4 (DAT as utility-aware KL reweighting)**.** *Assume $Z > 0$ and define the positive-utility support:*

$$\mathcal{I}_+ = \{i \in \mathcal{I} : U_i > 0\}. \tag{87}$$

Let

$$\rho = \sum_{i \in \mathcal{I}_+} P_i > 0, \qquad \overline{P}_i = \frac{P_i}{\rho}, \quad i \in \mathcal{I}_+, \tag{88}$$

be the PEOT distribution restricted and normalized on $\mathcal{I}_+$. Then $\mathbf{Q} = \{Q_i\}_{i \in \mathcal{I}_+}$ in Eq. 86 is the unique solution of:

$$\min_{\mathbf{R} \in \Delta(\mathcal{I}_+)} \left[\mathrm{KL}(\mathbf{R} \| \overline{\mathbf{P}}) - \sum_{i \in \mathcal{I}_+} R_i \log U_i \right], \tag{89}$$

where $\Delta(\mathcal{I}_+)$ denotes the probability simplex over $\mathcal{I}_+$.

Proof. For $i \in \mathcal{I}_+$, we have $U_i > 0$, so $\log U_i$ is finite. Expanding the objective in Eq. 89 gives:

$$\begin{aligned} \mathcal{J}(\mathbf{R}) &= \sum_{i \in \mathcal{I}_+} R_i \log \frac{R_i}{\overline{P}_i} - \sum_{i \in \mathcal{I}_+} R_i \log U_i \\ &= \sum_{i \in \mathcal{I}_+} R_i \log \frac{R_i}{\overline{P}_i U_i}. \end{aligned} \tag{90}$$

Define:

$$A = \sum_{j \in \mathcal{I}_+} \overline{P}_j U_j, \qquad R_i^\star = \frac{\overline{P}_i U_i}{A}. \tag{91}$$

Since $\sum_i R_i = 1$, Eq. 90 becomes:

$$\begin{aligned}\mathcal{J}(\mathbf{R}) &= \sum_{i\in\mathcal{I}_+} R_i \log \frac{R_i}{AR_i^\star} \\ &= \sum_{i\in\mathcal{I}_+} R_i \log \frac{R_i}{R_i^\star} - \log A \sum_{i\in\mathcal{I}_+} R_i \\ &= \mathrm{KL}(\mathbf{R}\|\mathbf{R}^\star) - \log A. \end{aligned} \tag{92}$$

Because KL divergence is nonnegative,

$$\mathcal{J}(\mathbf{R}) \geq -\log A, \tag{93}$$

with equality if and only if $\mathbf{R} = \mathbf{R}^\star$. Hence $\mathbf{R}^\star$ is the unique minimizer.

It remains to show that $\mathbf{R}^\star$ equals the DAT distribution. Using Eq. 88,

$$\begin{aligned} R_i^\star &= \frac{(P_i/\rho)U_i}{\sum_{j\in\mathcal{I}_+}(P_j/\rho)U_j} \\ &= \frac{P_iU_i}{\sum_{j\in\mathcal{I}_+} P_jU_j}. \end{aligned} \tag{94}$$

Since $U_i > 0$ on $\mathcal{I}_+$ and $U_i^+ = 0$ outside $\mathcal{I}_+$,

$$\sum_{j\in\mathcal{I}_+} P_jU_j = \sum_{j\in\mathcal{I}} P_jU_j^+ = Z. \tag{95}$$

Therefore,

$$R_i^\star = \frac{P_iU_i^+}{Z} = Q_i, \tag{96}$$

which proves the result. □

Proposition 4 shows that DAT performs a principled posterior reweighting of the PEOT correspondence: the KL term preserves proximity to the original transport structure, while the second term favors paths with larger positive deficiency utility.

Improvement of transport-weighted deficiency utility. The previous proposition explains the form of DAT, but the central purpose of truncation is to retain transport paths that better match the current student deficiency. The following theorem directly quantifies this effect and shows that DAT cannot decrease the average deficiency utility of the original PEOT plan.

Theorem 4 (Monotonic improvement of deficiency utility)**.** *Let*

$$\mathbb{E}_{\mathrm{PE}}[U] = \sum_{i\in\mathcal{I}} P_iU_i \tag{97}$$

and

$$\mathbb{E}_{\mathrm{DAT}}[U] = \sum_{i\in\mathcal{I}} Q_iU_i \tag{98}$$

denote the transport-weighted average utilities under PEOT and DAT, respectively. If $Z > 0$, then

$$\mathbb{E}_{\mathrm{DAT}}[U] - \mathbb{E}_{\mathrm{PE}}[U] = \frac{\mathrm{Var}_{\mathbf{P}}(U^+)}{\mathbb{E}_{\mathbf{P}}[U^+]} + \mathbb{E}_{\mathbf{P}}[U^-] \geq 0, \tag{99}$$

where

$$U^- = [-U]_+ = \max(0, -U). \tag{100}$$

Consequently,

$$\mathbb{E}_{\mathrm{DAT}}[U] \geq \mathbb{E}_{\mathrm{PE}}[U]. \tag{101}$$

If $Z = 0$, the fallback $\mathbf{\Gamma}^{\mathrm{DAT}} = \mathbf{\Gamma}^{\mathrm{PE}}$ gives equality.

Proof. Assume first that $Z > 0$. Since $Q_i > 0$ only when $U_i > 0$, we have $U_i = U_i^+$ on the support of $\mathbf{Q}$. Using Eq. 86,

$$\begin{aligned}\mathbb{E}_{\mathrm{DAT}}[U] &= \sum_{i\in\mathcal{I}} Q_i U_i \\ &= \sum_{i\in\mathcal{I}} \frac{P_i U_i^+}{Z} U_i \\ &= \frac{\sum_{i\in\mathcal{I}} P_i (U_i^+)^2}{\sum_{i\in\mathcal{I}} P_i U_i^+} \\ &= \frac{\mathbb{E}_{\mathbf{P}}[(U^+)^2]}{\mathbb{E}_{\mathbf{P}}[U^+]}. \end{aligned} \tag{102}$$

Using

$$\mathbb{E}_{\mathbf{P}}[(U^+)^2] = \mathrm{Var}_{\mathbf{P}}(U^+) + \left(\mathbb{E}_{\mathbf{P}}[U^+]\right)^2, \tag{103}$$

Eq. 102 becomes:

$$\mathbb{E}_{\mathrm{DAT}}[U] = \mathbb{E}_{\mathbf{P}}[U^+] + \frac{\mathrm{Var}_{\mathbf{P}}(U^+)}{\mathbb{E}_{\mathbf{P}}[U^+]}. \tag{104}$$

On the other hand, the identity

$$U = U^+ - U^- \tag{105}$$

implies:

$$\mathbb{E}_{\mathrm{PE}}[U] = \mathbb{E}_{\mathbf{P}}[U^+] - \mathbb{E}_{\mathbf{P}}[U^-]. \tag{106}$$

Subtracting Eq. 106 from Eq. 104 gives:

$$\begin{aligned}&\mathbb{E}_{\mathrm{DAT}}[U] - \mathbb{E}_{\mathrm{PE}}[U] \\ &= \frac{\mathrm{Var}_{\mathbf{P}}(U^+)}{\mathbb{E}_{\mathbf{P}}[U^+]} + \mathbb{E}_{\mathbf{P}}[U^-]. \end{aligned} \tag{107}$$

Because $Z = \mathbb{E}_{\mathbf{P}}[U^+] > 0$, the denominator is positive, while variance and U^- are nonnegative. Hence the right-hand side is nonnegative, proving Eqs. 99-101.

If $Z = 0$, then $U_i^+ = 0$ for every pair with $P_i > 0$. By the fallback rule, $\mathbf{Q} = \mathbf{P}$, and therefore:

$$\mathbb{E}_{\mathrm{DAT}}[U] = \mathbb{E}_{\mathrm{PE}}[U]. \tag{108}$$

□

The exact decomposition in Eq. 99 reveals two sources of utility improvement. The term $\mathbb{E}_{\mathbf{P}}[U^-]$ comes from removing negatively aligned transport paths, while $\mathrm{Var}_{\mathbf{P}}(U^+)/\mathbb{E}_{\mathbf{P}}[U^+]$ comes from assigning larger mass to higher-utility positive paths. The improvement is strict whenever either negative-utility mass exists or the positive utilities are nonconstant on the support of PEOT.

Geometric meaning of positive deficiency utility. The utility improvement above is defined directly in terms of cosine similarity. To connect this quantity to the prediction deficiency in Eq. 11, the following theorem analyzes the local effect of moving a student prediction state along a teacher evolution direction. The result provides a local geometric interpretation of the sign criterion used by DAT in the logit space.

Theorem 5 (Positive utility gives a local logit-space deficiency-descent direction)**.** *Fix an evolution pair (r, s). For each sample-token instance, denote:*

$$\mathbf{e}_t = \Delta\mathbf{z}_{t,r}^{\mathcal{T}}, \qquad \mathbf{d}_t = \mathbf{d}_{t,s}^{\mathcal{S}}. \tag{109}$$

Assume $\mathbf{e}_t \neq \mathbf{0}$ and $\mathbf{d}_t \neq \mathbf{0}$ for all instances considered in the empirical average. Define the normalized teacher evolution:

$$\widehat{\mathbf{e}}_t = \frac{\mathbf{e}_t}{\|\mathbf{e}_t\|_2} \tag{110}$$

and the average deficiency after an infinitesimal movement of size $\gamma \geq 0$ along this direction:

$$G_{rs}(\gamma) = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\left[\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|}\|\mathbf{d}_t - \gamma\widehat{\mathbf{e}}_t\|_2\right]. \tag{111}$$

Then

$$G'_{rs}(0) = -U_{rs}. \tag{112}$$

Therefore, if

$$U_{rs} > 0, \tag{113}$$

there exists $\gamma_0 > 0$ such that:

$$G_{rs}(\gamma) < G_{rs}(0), \qquad \forall\gamma \in (0, \gamma_0). \tag{114}$$

Proof. For a fixed sample-token instance, define:

$$g_t(\gamma) = \|\mathbf{d}_t - \gamma\widehat{\mathbf{e}}_t\|_2. \tag{115}$$

Because $\mathbf{d}_t \neq \mathbf{0}$, g_t is differentiable in a neighborhood of $\gamma = 0$. Using

$$\frac{d}{d\gamma}\|\mathbf{a}(\gamma)\|_2 = \frac{\mathbf{a}(\gamma)^\top\mathbf{a}'(\gamma)}{\|\mathbf{a}(\gamma)\|_2}, \tag{116}$$

with

$$\mathbf{a}(\gamma) = \mathbf{d}_t - \gamma\widehat{\mathbf{e}}_t, \qquad \mathbf{a}'(\gamma) = -\widehat{\mathbf{e}}_t, \tag{117}$$

we obtain:

$$g'_t(\gamma) = -\frac{(\mathbf{d}_t - \gamma\widehat{\mathbf{e}}_t)^\top\widehat{\mathbf{e}}_t}{\|\mathbf{d}_t - \gamma\widehat{\mathbf{e}}_t\|_2}. \tag{118}$$

Evaluating at $\gamma = 0$ gives:

$$\begin{aligned} g'_t(0) &= -\frac{\mathbf{d}_t^\top\widehat{\mathbf{e}}_t}{\|\mathbf{d}_t\|_2} \\ &= -\frac{\mathbf{d}_t^\top\mathbf{e}_t}{\|\mathbf{d}_t\|_2\|\mathbf{e}_t\|_2} \\ &= -\cos(\mathbf{e}_t, \mathbf{d}_t). \end{aligned} \tag{119}$$

Since the distillation dataset and token sequences form a finite empirical average, differentiation commutes with the finite sums in Eq. 111. Therefore,

$$\begin{aligned} G'_{rs}(0) &= \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\left[\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|}g'_t(0)\right] \\ &= -\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\left[\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|}\cos(\Delta\mathbf{z}^{\mathcal{T}}_{t,r}, \mathbf{d}^{\mathcal{S}}_{t,s})\right] \\ &= -U_{rs}, \end{aligned} \tag{120}$$

where the final equality follows directly from Eq. 12. This proves Eq. 112.

If $U_{rs} > 0$, then

$$G'_{rs}(0) = -U_{rs} < 0. \tag{121}$$

By the definition of the right derivative,

$$\lim_{\gamma\to 0^+}\frac{G_{rs}(\gamma) - G_{rs}(0)}{\gamma} = -U_{rs} < 0. \tag{122}$$

Hence there exists $\gamma_0 > 0$ such that for every $\gamma \in (0, \gamma_0)$,

$$\frac{G_{rs}(\gamma) - G_{rs}(0)}{\gamma} < 0. \tag{123}$$

Since $\gamma > 0$, this implies:

$$G_{rs}(\gamma) < G_{rs}(0), \tag{124}$$

which proves Eq. 114. □

Theorem 5 gives a direct geometric interpretation of the DAT criterion. For every retained pair,

$$\Gamma_{rs}^{\mathrm{DAT}} > 0 \Longrightarrow U_{rs} > 0, \tag{125}$$

and therefore its teacher evolution direction decreases the average student prediction deficiency to first order under the local construction above. This result characterizes directional compatibility in logit space and does not imply a descent guarantee for the KL distillation objective. Conversely, pairs with $U_{rs} \leq 0$ provide no such first-order descent guarantee and are removed by the positive-part truncation.

Summary. The above analysis establishes three properties of DAT. First, its multiplicative reweighting is the unique solution of a KL-regularized utility-aware optimization problem, providing a principled interpretation of combining PEOT correspondence with student deficiency. Second, DAT provably does not decrease the transport-weighted deficiency utility, with the exact gain determined by the removal of negative-utility mass and the reweighting of heterogeneous positive utilities. Third, a positive deficiency utility has a local geometric meaning: directly moving along the corresponding teacher evolution direction decreases the average student prediction deficiency to first order. Together, these results formalize why DAT retains transport paths that are both evolution-consistent and relevant to the remaining deficiency of the student.

B Methodological and Experimental Details

B.1 Details of Optimization of Prediction-Evolution Optimal Transport

In this section, we provide the optimization details of the prediction-evolution optimal transport problem introduced in Sec. 3.4. Recall that the teacher and student contain $N^{\mathcal{T}}$ and $N^{\mathcal{S}}$ evolution steps, respectively, and the prediction-evolution cost matrix is $\mathbf{C}^{\mathrm{PE}} \in \mathbb{R}^{N^{\mathcal{T}} \times N^{\mathcal{S}}}$. Given the teacher marginal $\mathbf{a}^{\mathcal{T}} \in \mathbb{R}^{N^{\mathcal{T}}}$ and the student marginal $\mathbf{b}^{\mathcal{S}} \in \mathbb{R}^{N^{\mathcal{S}}}$, the entropically regularized optimal transport problem is

$$\mathbf{\Gamma}^{\mathrm{PE}} = \underset{\mathbf{\Pi} \geq 0, \mathbf{\Pi} \mathbf{1}_{N^{\mathcal{S}}} = \mathbf{a}^{\mathcal{T}}, \mathbf{\Pi}^{\top} \mathbf{1}_{N^{\mathcal{T}}} = \mathbf{b}^{\mathcal{S}}}{\arg\min} \left[\langle \mathbf{\Pi}, \mathbf{C}^{\mathrm{PE}} \rangle + \varepsilon \sum_{r=1}^{N^{\mathcal{T}}} \sum_{s=1}^{N^{\mathcal{S}}} \Pi_{rs} (\log \Pi_{rs} - 1) \right] \tag{126}$$

where $\mathbf{\Pi}$ denotes a candidate transport plan, $\varepsilon > 0$ is the entropy regularization coefficient, and $\mathbf{1}_n$ denotes an n-dimensional all-ones vector. The marginal constraints ensure that the transport mass assigned to each teacher and student evolution step follows $\mathbf{a}^{\mathcal{T}}$ and $\mathbf{b}^{\mathcal{S}}$, respectively. In our experiments, we use uniform marginals, $a_r^{\mathcal{T}} = 1/N^{\mathcal{T}}$ and $b_s^{\mathcal{S}} = 1/N^{\mathcal{S}}$.

Sinkhorn Factorization. The entropy term in Eq. 126 makes the optimization strictly convex and allows the optimal transport plan to be expressed through diagonal scaling. Define the Gibbs kernel $\mathbf{R} \in \mathbb{R}_+^{N^{\mathcal{T}} \times N^{\mathcal{S}}}$ as:

$$R_{rs} = \exp\left(-\frac{C_{rs}^{\mathrm{PE}}}{\varepsilon} \right) \tag{127}$$

The optimal transport plan admits the following scaling form:

$$\mathbf{\Gamma}^{\mathrm{PE}} = \mathrm{diag}(\mathbf{u}) \mathbf{R} \, \mathrm{diag}(\mathbf{v}) \tag{128}$$

where $\mathbf{u} \in \mathbb{R}_+^{N^{\mathcal{T}}}$ and $\mathbf{v} \in \mathbb{R}_+^{N^{\mathcal{S}}}$ are positive scaling vectors. Substituting Eq. 128 into the marginal constraints yields:

$$\mathbf{u} = \mathbf{a}^{\mathcal{T}} \oslash (\mathbf{R}\mathbf{v}), \qquad \mathbf{v} = \mathbf{b}^{\mathcal{S}} \oslash (\mathbf{R}^{\top}\mathbf{u}) \tag{129}$$

where $\oslash$ denotes element-wise division. Therefore, Sinkhorn optimization alternates between updating $\mathbf{u}$ and $\mathbf{v}$ until the corresponding row and column marginals approach the prescribed distributions.

Algorithm 1 Sinkhorn Optimization for Prediction-Evolution Optimal Transport

Require: $\mathbf{C}^{\mathrm{PE}}$, $\mathbf{a}^{\mathcal{T}}$, $\mathbf{b}^{\mathcal{S}}$, entropy coefficient ε, iteration number M
Ensure: $\mathbf{\Gamma}^{\mathrm{PE}}$

$\mathbf{G} \leftarrow -\mathbf{C}^{\mathrm{PE}}/\varepsilon$
$\boldsymbol{\alpha} \leftarrow \mathbf{0}_{N^{\mathcal{T}}}, \quad \boldsymbol{\beta} \leftarrow \mathbf{0}_{N^{\mathcal{S}}}$
for $m = 1, \ldots, M$ **do**
  for $r = 1, \ldots, N^{\mathcal{T}}$ **do**
    $\alpha_r \leftarrow \log a_r^{\mathcal{T}} - \mathrm{LSE}_{s=1}^{N^{\mathcal{S}}}(G_{rs} + \beta_s)$
  end for
  for $s = 1, \ldots, N^{\mathcal{S}}$ **do**
    $\beta_s \leftarrow \log b_s^{\mathcal{S}} - \mathrm{LSE}_{r=1}^{N^{\mathcal{T}}}(G_{rs} + \alpha_r)$
  end for
end for
for $r = 1, \ldots, N^{\mathcal{T}}$ **do**
  for $s = 1, \ldots, N^{\mathcal{S}}$ **do**
    $\Gamma_{rs}^{\mathrm{PE}} \leftarrow \exp(G_{rs} + \alpha_r + \beta_s)$
  end for
end for
return $\mathbf{\Gamma}^{\mathrm{PE}}$

Log-Domain Optimization. Directly constructing $\mathbf{R}$ in Eq. 127 can suffer from numerical underflow when $C_{rs}^{\mathrm{PE}}/\varepsilon$ becomes large. We therefore perform the Sinkhorn iterations in the log domain. Let

$$G_{rs} = -\frac{C_{rs}^{\mathrm{PE}}}{\varepsilon}, \qquad \alpha_r = \log u_r, \qquad \beta_s = \log v_s \tag{130}$$

and define the log-sum-exp operator as:

$$\mathrm{LSE}(x_1, \ldots, x_n) = \log \sum_{i=1}^{n} \exp(x_i) \tag{131}$$

The teacher-side scaling factors are updated by enforcing the row marginals:

$$\alpha_r = \log a_r^{\mathcal{T}} - \mathrm{LSE}_{s=1}^{N^{\mathcal{S}}}(G_{rs} + \beta_s), \qquad r = 1, \ldots, N^{\mathcal{T}} \tag{132}$$

Similarly, the student-side factors are updated by enforcing the column marginals:

$$\beta_s = \log b_s^{\mathcal{S}} - \mathrm{LSE}_{r=1}^{N^{\mathcal{T}}}(G_{rs} + \alpha_r), \qquad s = 1, \ldots, N^{\mathcal{S}} \tag{133}$$

After M iterations, the transport matrix is recovered as:

$$\Gamma_{rs}^{\mathrm{PE}} = \exp(G_{rs} + \alpha_r + \beta_s) \tag{134}$$

The resulting matrix provides a soft many-to-many correspondence between teacher and student evolution steps. A large $\Gamma_{rs}^{\mathrm{PE}}$ indicates that teacher evolution step r and student evolution step s have a relatively low prediction-evolution transport cost while remaining compatible with the global marginal constraints.

Computational Complexity. Each Sinkhorn iteration evaluates the scaling factors over all $N^{\mathcal{T}}N^{\mathcal{S}}$ teacher–student evolution pairs. Therefore, the optimization requires $\mathcal{O}(MN^{\mathcal{T}}N^{\mathcal{S}})$ time and $\mathcal{O}(N^{\mathcal{T}}N^{\mathcal{S}})$ memory for the transport cost and coupling matrices. Since the number of evolution steps is determined by model depth rather than vocabulary size, this optimization introduces only a small overhead compared with the forward and backward computation of LLM distillation. In our experiments, we use $\varepsilon = 0.1$ and $M = 100$ for all settings.

B.2 Details of Datasets

We report results on the datasets spanning three benchmark series: domain-agnostic instruction-following, domain-specific instruction-following, and reasoning-oriented evaluation. Unless otherwise specified, domain-agnostic distillation is conducted on Databricks-Dolly-15k and evaluation is performed on both the training-domain benchmark and additional held-out instruction benchmarks to assess cross-dataset generalization.

Domain-Agnostic Instruction Benchmarks. We use Databricks-Dolly-15k (**Dolly**) (Conover et al., 2023) as the primary distillation corpus, and evaluate on **Dolly** together with four widely-used instruction-following benchmarks, namely Self-Instruct (**SelfInst**) (Wang et al., 2023), Vicuna Evaluation (**Vicuna**) (Chiang et al., 2023), Super-Natural Instructions (**S-NI**) (Wang et al., 2022), and Unnatural Instructions (**UnNI**) (Honovich et al., 2023). Following common practice in LLM distillation, we split **Dolly** into training, validation, and test sets, and use the remaining benchmarks exclusively for evaluation to measure transferability beyond the distillation distribution. A brief description of each dataset is given below.

- **Databricks-Dolly-15k** (Conover et al., 2023): A public instruction-following dataset with approximately fifteen thousand instruction–response examples. The data cover diverse instruction categories, including information extraction, closed-form and open-ended question answering, summarization, classification, and various generation tasks. The instructions are curated to represent human-facing assistant behaviors and are commonly used as a lightweight distillation and alignment corpus.
- **Self-Instruct** (Wang et al., 2023): A large-scale instruction dataset created via a bootstrapping procedure, where a language model is prompted to produce candidate tasks and corresponding instances, and the resulting pool is filtered and expanded to increase diversity. The benchmark contains a broad spectrum of instructions and input–output pairs and is often used to evaluate whether a distilled student generalizes to novel instruction styles beyond the training set.
- **Vicuna Evaluation** (Chiang et al., 2023): An evaluation suite derived from user-shared conversational prompts. In many LLM distillation studies, a compact subset of challenging prompts is used for evaluation to reduce inference cost while maintaining difficulty. We follow the standard evaluation protocol used in prior work when reporting Vicuna results.
- **Super-Natural Instructions** (Wang et al., 2022): A comprehensive instruction benchmark consisting of a large collection of NLP tasks paired with explicit task descriptions and expert-written instructions. It spans many task families and provides a structured test bed for instruction following under heterogeneous task definitions.
- **Unnatural Instructions** (Honovich et al., 2023): A large synthetic instruction dataset generated primarily by language models with minimal direct human annotation. Compared with human-authored instruction corpora, it emphasizes distribution shift in style and wording, making it suitable for stress-testing robustness and out-of-distribution generalization.

Domain-Specific Instruction Benchmarks. We consider two specialized tracks, medical and legal, where distillation data are domain-restricted and evaluation focuses on domain knowledge measured by MMLU subsets (Hendrycks et al., 2021). This setting tests whether EvaKD remains effective when the teacher and student must transfer fine-grained, domain-specific knowledge.

- **Medical track**: We distill on **GenMedGPT** (Li et al., 2023), a medical instruction-following dataset containing clinical-style questions and responses. We evaluate on three medical subsets of **MMLU** (Hendrycks et al., 2021): clinical knowledge (**CK**), professional medicine (**PM**), and college medicine (**CM**). These subsets are multiple-choice exams covering clinical facts, diagnosis and treatment knowledge, and medicine-related academic concepts.
- **Legal track**: We distill on **Lawyer-Instruct** (Alignment-Lab-AI, 2024), an instruction dataset tailored to legal tasks, and **USTerms** (Chalkidis et al., 2023), which focuses on legal terminology and related classification labels. We evaluate on two law-related **MMLU** (Hendrycks et al., 2021) subsets: international law (**IL**) and professional law

(**PL**). These subsets measure legal knowledge across jurisdictions and professional-level legal reasoning in a standardized multiple-choice format.

Reasoning and Code Benchmarks. In addition to instruction-following evaluation, we assess reasoning and programming capability on three widely adopted benchmarks: **GSM8K** (Cobbe et al., 2021), **MBPP** (Austin et al., 2021), and **HumanEval** (Chen et al., 2021). These datasets represent complementary capabilities that are often sensitive to distillation design.

- **GSM8K** (Cobbe et al., 2021): A grade-school math benchmark consisting of multi-step arithmetic and word problems, used to evaluate numerical reasoning and stepwise problem solving.
- **MBPP** (Austin et al., 2021): A programming benchmark composed of short Python tasks with natural language descriptions and unit tests, measuring basic code synthesis and functional correctness.
- **HumanEval** (Chen et al., 2021): A standard code-generation benchmark with Python function signatures and associated tests, designed to assess the ability to generate correct implementations from docstrings.

B.3 Details of Baselines

We compare EvaKD with representative knowledge distillation methods for LLMs. For clarity, we group the baselines into two categories. *Output-layer distillation* refers to methods that transfer teacher knowledge without explicitly aligning intermediate teacher and student layers, while *intermediate-layer distillation* additionally introduces supervision from internal representations.

Output-Layer Distillation. This category transfers teacher knowledge primarily through output distributions, generated sequences, or other signals derived from the final predictive behavior of the teacher. These methods do not explicitly align intermediate teacher and student layers.

- **KD** (Hinton et al., 2015): The conventional knowledge distillation method that minimizes the discrepancy between the predictive distributions of the teacher and student, typically using forward KL divergence.
- **SeqKD** (Kim & Rush, 2016): Sequence-level knowledge distillation uses sequences generated by the teacher as supervision for training the student. It transfers teacher behavior at the sequence level rather than relying only on token-wise distribution matching.
- **ImitKD** (Lin et al., 2020): ImitKD formulates autoregressive distillation from an imitation-learning perspective and trains the student using prefixes induced by its own generation. This design reduces the exposure mismatch between training and inference.
- **GKD** (Agarwal et al., 2024): Generalized Knowledge Distillation performs on-policy distillation by querying the teacher on student-generated sequences. It allows the student to learn from teacher feedback on states that the student is likely to encounter during generation.
- **MiniLLM** (Gu et al., 2024): MiniLLM replaces conventional forward KL divergence with reverse KL divergence for generative LLM distillation. It is designed to alleviate the tendency of the student to overestimate low-probability regions of the teacher distribution.
- **Distillm** (Ko et al., 2024): Distillm introduces a skew KL divergence objective together with adaptive off-policy distillation. It aims to improve both the effectiveness and computational efficiency of distillation with student-generated outputs.
- **Distillm-2** (Ko et al., 2025): Distillm-2 adopts a contrastive distillation strategy that treats teacher- and student-generated responses differently. It simultaneously encourages teacher responses while suppressing undesirable student responses during distillation.
- **ABKD** (Wang et al., 2025): ABKD introduces an α-β divergence to control probability-mass allocation during knowledge distillation. It provides a flexible trade-off between the concentration behaviors induced by forward and reverse KL divergence.

Intermediate-Layer Distillation. This category further exploits internal representations or layer-wise dynamics of the teacher to provide additional supervision beyond final outputs. Compared with output-layer methods, these approaches explicitly model intermediate knowledge during distillation.

- **TED** (Liang et al., 2023): Task-aware layEr-wise Distillation (TED) introduces task-aware filters to extract task-relevant knowledge from intermediate teacher representations before aligning them with the student. This reduces redundant intermediate supervision and narrows the representation gap between teacher and student.
- **FDD** (Gong et al., 2025): Feature Dynamics Distillation (FDD) models intermediate representations as a depthwise feature trajectory. It aligns both layer-wise features and their first-order changes across adjacent layers to transfer feature dynamics from the teacher to the student.

B.4 Details of Implementation Settings

For each dataset, we first fine-tune the teacher model and then use the fine-tuned teacher to distill the corresponding student model. We evaluate domain-agnostic benchmarks using Rouge-L and domain-specific benchmarks using accuracy. For reasoning benchmarks, we follow the standard evaluation metrics of each dataset. All baselines are reproduced under the same teacher-student pairs, training data, and optimization settings for a fair comparison.

Global Training Settings. Unless otherwise specified, we use the same training configuration across different teacher-student pairs and datasets. For a fair comparison, all experimental results are reported as the average over five random seeds. We train the student for 5 epochs with a learning rate of 5×10^{-5}, a cosine learning-rate scheduler, a warmup ratio of 0.05, and a weight decay of 0.01. The per-device batch size is set to 1 with 8 gradient accumulation steps, and the maximum sequence length is 512. We use BFloat16 precision and gradient checkpointing to reduce memory consumption. For parameter-efficient training, we apply LoRA to the student model with rank 8, scaling factor 16, and dropout rate 0.1, covering the attention and feed-forward projection modules.

Method-Specific Settings. For EvaKD, the distillation temperature τ is set to 1.0. In sparse prediction projection, the candidate vocabulary size is controlled by $K = 32$, where the top-K tokens are selected from both teacher and student predictions before adding the ground-truth token. For prediction-evolution optimal transport, we use four representative evolution stages for both the teacher and student. The entropy regularization coefficient ε in Eq. 10 is set to 0.1, and the Sinkhorn solver runs for $M = 100$ iterations. The coefficient λ for the proposed intermediate distillation objective in Eq. 17 is set to 1.0. For computational efficiency, token-level OT computation is processed in chunks of 64 tokens. Unless explicitly varied in the analysis experiments, these method-specific hyperparameters in EvaKD are fixed across all experimental settings.

B.5 Details of Analysis Experiments

Common Settings. Unless otherwise specified, all analysis experiments are conducted on the Dolly validation set. To avoid overlap with the distillation data, we perform exact deduplication against the training set using the concatenated `prompt + output` field. After removing two overlapping examples, 498 validation instances are retained, containing 42,221 and 42,847 valid response tokens for the Llama3 and Qwen3 settings, respectively. We analyze the final trained EvaKD checkpoints under two teacher–student pairs: Llama3.1-8B $\rightarrow$ Llama3.2-1B and Qwen3-4B $\rightarrow$ Qwen3-0.6B. The maximum sequence length is 512, and the sparse candidate vocabulary at token position t is constructed as:

$$\mathcal{C}_t = \mathrm{TopK}(\mathbf{z}_t^{\mathcal{T}}, K) \cup \mathrm{TopK}(\mathbf{z}_t^{\mathcal{S}}, K) \cup \{y_t\} \tag{135}$$

with $K = 32$ unless otherwise specified. For evolution-level analysis, we use four teacher and four student evolution steps. Thus, every model yields four consecutive evolution steps and $4 \times 4 = 16$ teacher-student evolution pairs. The PEOT configuration follows training, using entropy coefficient $\varepsilon = 0.1$ and $M = 100$ Sinkhorn iterations.

Deficiency-Aware Knowledge Selection. This analysis corresponds to Figure 4 and examines how DAT selects useful teacher evolution according to student deficiency. For each teacher evolution step r and student evolution step s, we compute the deficiency utility:

$$U_{rs} = \frac{1}{N_{\text{tok}}} \sum_{(\mathbf{x},\mathbf{y})} \sum_{t=1}^{|\mathbf{y}|} \cos\left(\Delta \mathbf{z}^{\mathcal{T}}_{t,r}, \mathbf{d}^{\mathcal{S}}_{t,s}\right) \tag{136}$$

where N_{tok} denotes the total number of valid response tokens, $\Delta \mathbf{z}^{\mathcal{T}}_{t,r}$ is the teacher prediction evolution at step r, and $\mathbf{d}^{\mathcal{S}}_{t,s}$ is the remaining prediction deficiency at student step s. Thus, each model pair produces a 4×4 utility matrix, where a larger U_{rs} indicates stronger agreement between teacher evolution and student deficiency. Based on the PEOT transport plan $\mathbf{\Gamma}^{\text{PE}}$, DAT truncates each teacher-student evolution pair according to its deficiency utility:

$$\widetilde{\Gamma}_{rs} = \Gamma^{\text{PE}}_{rs}[U_{rs}]_+ \tag{137}$$

where $[x]_+ = \max(0, x)$. The retained transport mass is then normalized as:

$$\Gamma^{\text{DAT}}_{rs} = \frac{\widetilde{\Gamma}_{rs}}{\sum_{r',s'} \widetilde{\Gamma}_{r's'}} \tag{138}$$

when positive utility exists; otherwise, the original PEOT plan is retained. The left heatmaps in Figure 4 visualize the deficiency utility U_{rs}, while the right heatmaps show the normalized deficiency-aware transport plan Γ^{DAT}_{rs}. For both matrices, rows correspond to teacher evolution steps and columns correspond to student evolution steps. Larger Γ^{DAT}_{rs} indicates that a larger proportion of the retained transport mass is assigned to the corresponding evolution pair.

Utility and Deficiency Reduction. This analysis corresponds to Figure 5 and empirically examines whether the deficiency utility U_{rs} is associated with KL improvement under a direct prediction-space perturbation along the teacher evolution direction. We use the same 498 validation examples, sparse Top-32 candidate vocabulary, and 4×4 evolution pairs as in the common setting. For a response token t and an evolution pair (r, s), let

$$\mathbf{e}_{t,r} = \Delta \mathbf{z}^{\mathcal{T}}_{t,r}, \qquad \mathbf{d}^{\mathcal{S}}_{t,s} = \mathbf{z}^{\mathcal{T}}_{t,L^{\mathcal{T}}} - \mathbf{z}^{\mathcal{S}}_{t,s+1} \tag{139}$$

denote the teacher evolution direction and the remaining student deficiency, respectively. The sample-level deficiency utility for pair (r, s) is computed by averaging cosine similarity over all valid response tokens:

$$U^{(i)}_{rs} = \frac{1}{|\mathbf{y}_i|} \sum_{t=1}^{|\mathbf{y}_i|} \cos\left(\mathbf{e}_{t,r}, \mathbf{d}^{\mathcal{S}}_{t,s}\right) \tag{140}$$

To examine the finite-step behavior of this evolution direction, we perform a diagnostic prediction-space perturbation with a dimensionless step size $\eta = 0.05$:

$$\mathbf{z}^{\mathcal{S}\prime}_{t,s+1} = \mathbf{z}^{\mathcal{S}}_{t,s+1} + \eta \|\mathbf{d}^{\mathcal{S}}_{t,s}\|_2 \frac{\mathbf{e}_{t,r}}{\|\mathbf{e}_{t,r}\|_2} \tag{141}$$

Let $\mathbf{p}^{\mathcal{T}}_{t,L^{\mathcal{T}}}$, $\mathbf{q}^{\mathcal{S}}_{t,s}$, and $\mathbf{q}^{\mathcal{S}\prime}_{t,s}$ denote the sparse predictive distributions obtained from the final teacher state, the original student state, and the updated student state, respectively. We measure the improvement induced by teacher evolution using the sparse-KL reduction:

$$\Delta D^{(i)}_{rs} = \frac{1}{|\mathbf{y}_i|} \sum_{t=1}^{|\mathbf{y}_i|} \left[D_{\text{KL}}(\mathbf{p}^{\mathcal{T}}_{t,L^{\mathcal{T}}} \| \mathbf{q}^{\mathcal{S}}_{t,s}) - D_{\text{KL}}(\mathbf{p}^{\mathcal{T}}_{t,L^{\mathcal{T}}} \| \mathbf{q}^{\mathcal{S}\prime}_{t,s}) \right] \tag{142}$$

where $\Delta D^{(i)}_{rs} > 0$ indicates that moving the student prediction along the corresponding teacher evolution direction reduces its divergence from the final teacher prediction. Each blue point in Figure 5 corresponds to one validation example and one evolution pair. Since each example contains 16 evolution pairs, each model produces $498 \times 16 = 7{,}968$ sample-stage points. The red diamonds represent the 16 dataset-level evolution-pair aggregates. Pearson correlation measures the linear association between U and ΔD, while Spearman correlation measures their rank association. We additionally report $P(\Delta D > 0 \mid U > 0)$, i.e., the proportion of positive-utility pairs that also produce a positive sparse-KL reduction. This analysis is intended as a prediction-space diagnostic of directional compatibility rather than a simulation of the parameter update induced by Eq. 16.

Sparse Prediction Projection. This analysis corresponds to Figure 6 and evaluates both the approximation fidelity and computational efficiency of sparse prediction projection. We vary the number of top-ranked tokens selected from each model as:

$$K \in \{4, 8, 16, 32, 64, 128\} \tag{143}$$

while keeping the trained teacher and student checkpoints fixed. For every token t, the candidate vocabulary is reconstructed as:

$$\mathcal{C}_t(K) = \mathrm{TopK}(\mathbf{z}_t^{\mathcal{T}}, K) \cup \mathrm{TopK}(\mathbf{z}_t^{\mathcal{S}}, K) \cup \{y_t\} \tag{144}$$

with duplicate tokens removed.

Vocabulary coverage and approximation error. For a full-vocabulary predictive distribution $\mathbf{p}_t$, we define its retained probability mass under $\mathcal{C}_t(K)$ as:

$$\alpha_t = \sum_{v \in \mathcal{C}_t(K)} p_t(v) \tag{145}$$

where the probabilities $p_t(v)$ are computed using the full-vocabulary log-normalizer rather than a softmax restricted to the candidate set. Therefore, α_t measures the exact probability mass preserved by sparse projection. Let $\mathbf{p}_t^{\mathcal{C}}$ denote the distribution obtained by conditioning and renormalizing $\mathbf{p}_t$ on $\mathcal{C}_t(K)$. As established in Appendix A.1, the corresponding total-variation approximation error is:

$$d_{\mathrm{TV}}(\mathbf{p}_t, \mathbf{p}_t^{\mathcal{C}}) = 1 - \alpha_t \tag{146}$$

which is reported as the TV error in Figure 6. For each validation example, coverage and TV error are first averaged over valid response tokens, after which all 498 examples are averaged with equal weight. The uncertainty bands denote 95% confidence intervals over the example-level means.

Projection efficiency. We separately benchmark the actual intermediate LM-head projection routine used by EvaKD. Both sparse and full projection use the original final normalization module and LM head, BFloat16 precision, four intermediate depth stages, and 64 token states. For each configuration, we perform three warm-up runs followed by ten measured runs. Projection latency is measured using CUDA events and reported as the median over the ten runs. Peak memory is measured as the incremental `max_memory_allocated` caused by a single projection call. Latency and peak memory in Figure 6 are normalized by the corresponding full-vocabulary projection. The "projected elements" curve reports the ratio between the number of sparse projection outputs and full-vocabulary projection outputs. This quantity describes the theoretical output scale and should be distinguished from measured peak memory, since the sparse implementation additionally constructs token-specific candidate LM-head buffers.

Depth-wise Deficiency Utility. This analysis corresponds to Figure 7 and investigates whether the usefulness of transported teacher evolution changes across student depth. We use the same 498 validation examples, Top-32 candidate vocabulary, four teacher and student evolution steps, entropy coefficient $\varepsilon = 0.1$, and 100 Sinkhorn iterations. For every validation example, we independently reconstruct its PEOT transport plan $\mathbf{\Gamma}^{\mathrm{PE}}$ before applying deficiency-aware refinement. For student evolution step s, we define the positive transport ratio as:

$$R_s = \frac{\sum_{r=1}^{N^{\mathcal{T}}} \Gamma_{rs}^{\mathrm{PE}} \mathbf{1}(U_{rs} > 0)}{\sum_{r=1}^{N^{\mathcal{T}}} \Gamma_{rs}^{\mathrm{PE}}} \tag{147}$$

where $\mathbf{1}(\cdot)$ is the indicator function. Thus, R_s measures the fraction of PEOT mass arriving at student step s from teacher evolution with positive deficiency utility. Conversely, $1 - R_s$ quantifies the fraction of pre-DAT transport mass associated with non-positive utility and therefore serves as a diagnostic measure of potentially conflicting transfer at this depth. We further compute the PEOT-weighted average deficiency utility:

$$\bar{U}_s = \frac{\sum_{r=1}^{N^{\mathcal{T}}} \Gamma_{rs}^{\mathrm{PE}} U_{rs}}{\sum_{r=1}^{N^{\mathcal{T}}} \Gamma_{rs}^{\mathrm{PE}}} \tag{148}$$

which measures the overall compatibility between transported teacher evolution and the remaining deficiency of student step s. Importantly, both R_s and $\bar{U}_s$ are computed separately for every validation example using its own PEOT plan. We then average the resulting quantities equally over

Table 4: Results on reasoning benchmarks (Llama3.1-8B → Llama3.2-1B). The metric for GSM8K is accuracy, and the metrics for MBPP and HumanEval are pass@1.

Datasets	Teacher	Student	GKD	MiniLLM	Distillm	Distillm-2	ABKD	TED	FDD	EvaKD
GSM8K (Acc)	60.90	12.50	16.10	16.90	17.30	17.60	18.20	18.50	18.90	**19.80**
MBPP (pass@1)	48.33	17.41	21.36	21.42	21.47	21.96	22.36	23.05	22.63	**24.31**
HumanEval (pass@1)	38.40	13.35	15.36	15.58	15.67	15.84	15.98	16.71	17.02	**17.82**

Table 5: Results of distilling from Llama2-13B to TinyLlama-1.1B on domain-agnostic benchmarks.

	Llama2-13B → TinyLlama-1.1B					
Methods	**Dolly**	**SelfInst**	**S-NI**	**UnNI**	**Vicuna**	**Avg.**
KD	21.37	13.82	27.46	25.93	15.24	20.76
EvaKD	**24.31**	**16.47**	**30.82**	**29.58**	**15.92**	**23.42**

the 498 examples. Error bars denote 95% confidence intervals computed as $1.96 \times \mathrm{SEM}$ across example-level measurements. This sample-wise procedure preserves variation in the sign of utility across examples and avoids the information loss that would arise from averaging utility before applying the positive-utility indicator.

C Additional Experimental Results

C.1 Results on Reasoning Benchmarks

To further examine whether EvaKD can generalize beyond instruction-following tasks, we evaluate the distilled Llama3.2-1B model on three reasoning benchmarks under the Llama3.1-8B → Llama3.2-1B setting. Specifically, we use GSM8K for mathematical reasoning and MBPP and HumanEval for code generation, where GSM8K is evaluated by accuracy and MBPP and HumanEval by pass@1. The results are reported in Table 4.

First, EvaKD achieves the best performance on all three reasoning benchmarks. It obtains 19.80 on GSM8K, 24.31 on MBPP, and 17.82 on HumanEval, outperforming the strongest baselines by +0.90↑, +1.26↑, and +0.80↑, respectively. These consistent improvements indicate that evolution-aware intermediate knowledge transfer remains effective for reasoning-oriented tasks beyond general instruction following. Second, EvaKD also outperforms strong intermediate-layer baselines TED and FDD across mathematical and code reasoning tasks. In particular, the gains on both MBPP and HumanEval suggest that flexible evolution correspondence and deficiency-aware selective transfer can better preserve useful intermediate knowledge for program synthesis, demonstrating the general applicability of EvaKD across different reasoning scenarios.

C.2 Results on Larger Teacher Models

To investigate whether EvaKD remains effective when distilling from a larger teacher model, we further conduct experiments with Llama2-13B as the teacher and TinyLlama-1.1B as the student, with results shown in Table 5. First, EvaKD consistently outperforms vanilla KD across all five domain-agnostic benchmarks. Specifically, compared with KD, EvaKD improves Dolly from 21.37 to 24.31, SelfInst from 13.82 to 16.47, S-NI from 27.46 to 30.82, and UnNI from 25.93 to 29.58, yielding an average improvement of 2.66 Rouge-L points (20.76→23.42). These results show that the proposed evolution-aware distillation remains effective even under a substantially larger teacher-student scale gap. Second, the gains are observed not only on Dolly, which is used for distillation, but also on held-out benchmarks such as SelfInst, S-NI, UnNI, and Vicuna. This indicates that EvaKD can transfer useful intermediate evolution knowledge from the larger teachers while maintaining good cross-dataset generalization, demonstrating its scalability to the stronger teacher models.

Table 6: Results of distillation from Llama3.1-8B to Qwen3-0.6B on domain-agnostic benchmarks.

	Llama3.1-8B → Qwen3-0.6B					
Methods	**Dolly**	**SelfInst**	**S-NI**	**UnNI**	**Vicuna**	**Avg.**
KD	27.52	22.61	41.47	43.42	19.13	30.83
EvaKD	**27.85**	**22.75**	**42.23**	**43.65**	**19.36**	**31.17**

C.3 Results on Heterogeneous Distillation

To further investigate the generalization of EvaKD across heterogeneous model families and architectural configurations, we conduct distillation from Llama3.1-8B to Qwen3-0.6B on the domain-agnostic benchmarks, with the results reported in Table 6. We use MultiLevelOT (Cui et al., 2025) to align their vocabularies. First, EvaKD consistently outperforms vanilla KD across all five benchmarks, improving the average Rouge-L from 30.83 to 31.17, with an absolute gain of +0.34↑. Notably, the improvement reaches +0.76↑ on S-NI and +0.33↑ on Dolly, while positive gains are also observed on SelfInst, UnNI, and Vicuna. Second, these consistent improvements indicate that the proposed evolution-aware correspondence and deficiency-aware transfer remain effective even when the teacher and student belong to different model families. This further demonstrates the applicability and robustness of EvaKD beyond homogeneous settings.

C.4 Analysis of Cross-Task Prediction Evolution

To further examine whether prediction evolution follows a fixed depthwise pattern across different tasks, we analyze the eight task categories in the Dolly validation set under both Llama3.1-8B → Llama3.2-1B and Qwen3-4B → Qwen3-0.6B. For each task, we measure prediction changes between adjacent model depths in the sparse prediction space and compare the resulting evolution-rate profile with the overall Dolly distribution.

Task-Specific Evolution-Rate Residuals. Figure 8 shows the relative evolution-rate residual of each task with respect to the overall Dolly profile, where positive and negative values indicate locally enhanced and suppressed prediction evolution, respectively. Although several tasks remain close to the global profile, clear task-dependent variations emerge at specific depths. In particular, Classification exhibits the largest deviations, reaching maximum absolute residuals of 15.6%/14.6% for the Llama teacher/student and 25.3%/24.8% for Qwen. In contrast, Open QA remains much closer to the overall profile. These results indicate that prediction evolution is not determined solely by normalized model depth, but also varies with the task being processed. This further supports using evolution-aware correspondence rather than a fixed layer mapping shared across inputs and tasks.

Decoder-Internal Prediction Evolution. Figure 9 removes the dominant first evolution step and renormalizes the remaining decoder-internal evolution, allowing differences in middle and later layers to be observed more clearly. The resulting trajectories remain non-uniform and exhibit task-dependent local peaks and valleys. For example, the Llama teacher generally shows stronger refinement near the final layers, whereas the Qwen teacher and student exhibit more pronounced fluctuations across intermediate depths. Moreover, the locations and magnitudes of these variations differ across tasks. Therefore, the observed evolution mismatch cannot be explained only by the initial representation transition, but persists throughout the decoder, further motivating flexible alignment based on actual prediction changes.

Task-Specific Evolution Fingerprints. Figure 10 summarizes these residuals as depthwise heatmaps, where red and blue indicate evolution rates above and below the overall Dolly profile, respectively. Distinct task-specific patterns can be observed across both model families. Classification shows the strongest alternating deviations, while Closed QA, Summarization, and Information Extraction exhibit similar suppression in several early-to-middle Qwen depths followed by different later-stage refinements. Meanwhile, Open QA and Brainstorming remain closer to the overall profile. These structured residual fingerprints show that task variation is superimposed on the shared model-level evolution pattern. Overall, Figures 8, 9, and 10 demonstrate that prediction evolution

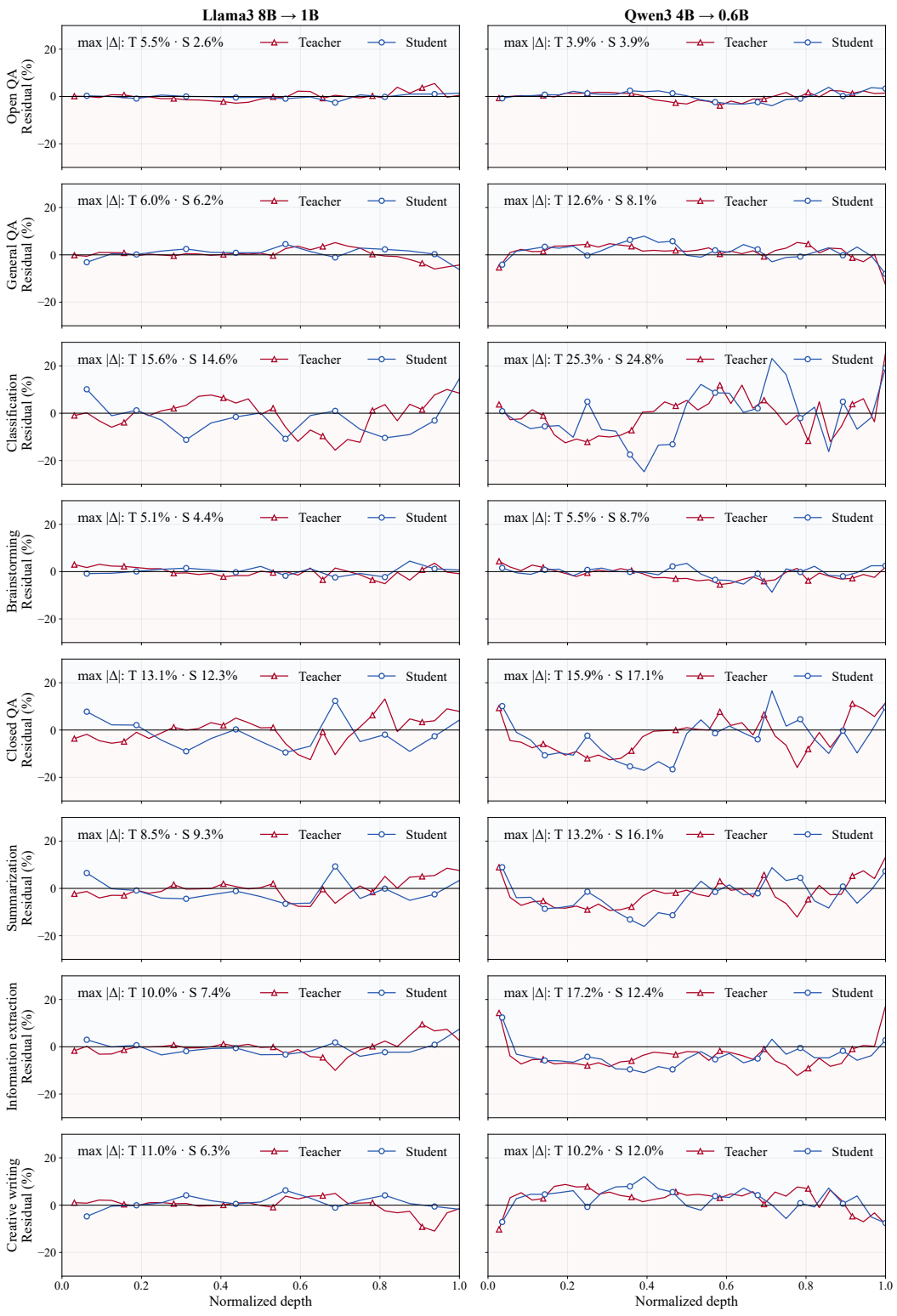


Figure 8: Analysis of task-specific evolution-rate residuals versus all Dolly.

is both model-dependent and task-dependent, providing further evidence that predefined depth cor-

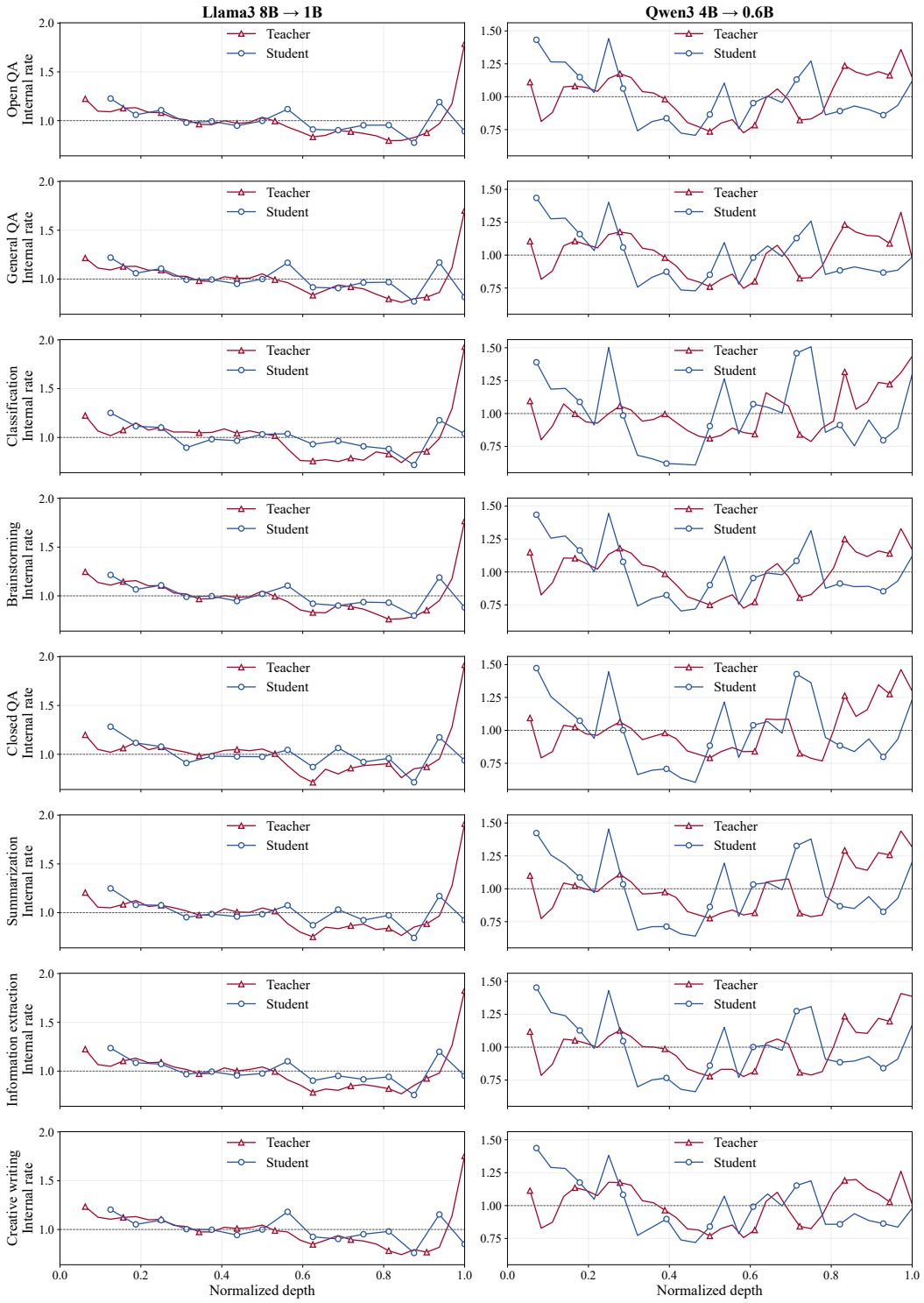


Figure 9: Analysis of decoder-internal evolution after removing the shared first step.

respondence cannot reliably represent teacher-student evolution stages and supporting the adaptive evolution correspondence introduced by PEOT.

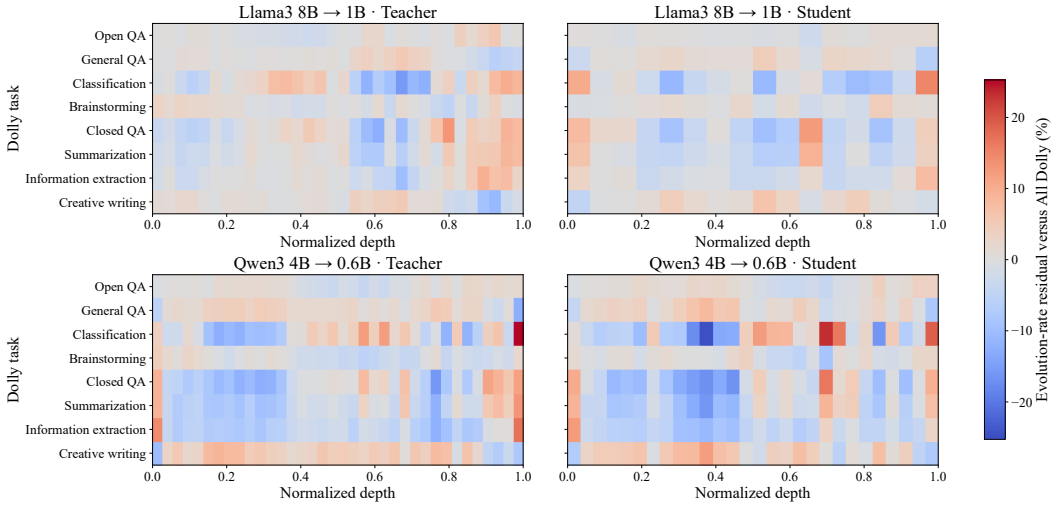


Figure 10: Analysis of task-specific residual fingerprints across depth.

C.5 Case Study

To qualitatively examine the generation behavior of different distillation methods, we present three representative examples covering grammar error identification, minimum-step reasoning, and dialogue reconstruction, in the setting of Llama3.1-8B → Llama3.2-1B on domain-agnostic datasets. As shown in Tables 7, 8, and 9, EvaKD consistently produces outputs that closely match the ground-truth responses while following the task instructions. In the grammar identification case, several baselines either incorrectly predict "No Error", return incomplete fragments, or introduce unnecessary corrections, whereas EvaKD accurately identifies and reproduces the erroneous sentence. In the minimum-step reasoning example, EvaKD correctly predicts the required number of steps as three, while several methods produce incorrect numerical conclusions despite generating superficially plausible explanations. This case also illustrates that high lexical overlap does not necessarily imply correct reasoning. In the dialogue reconstruction example, EvaKD successfully recovers the complete dialogue with the correct speaker order and content, whereas several baselines exhibit missing turns, incorrect ordering, role confusion, or irrelevant generation. Overall, these examples suggest that EvaKD provides more reliable supervision for intermediate representations, leading to improved instruction following, semantic consistency, and generation accuracy across heterogeneous tasks.

Table 7: Case study on Grammar and Error Identification.

Instruction: You are given a paragraph with several sentences. Please identify the sentence(s) which contain(s) errors in grammar, punctuation or spelling. If there are no errors, write 'No Error'.	
Input: Sentences: We went to store so that we could buy some food for our picnic; However, they were closed by the time we got there.	
Ground-truth: We went to store so that we could buy some food for our picnic; However, they were closed by the time we got there.	
Method	**Generated Result**
KD	We went to store so that...
SeqKD	No Error Sentences which contain errors: "We went to store so that
RKD	The first sentence is wrong in grammar, the second sentence is wrong in punctuation, and the third sentence is wrong in spelling.
GKD	There are no grammar, punctuation or spelling errors in your input.
ImitKD	There are two errors in the above paragraph: 1. Last sentence should read as Followed by us, to buy some food for our picnics
MiniLLM	The sentence containing errors is: 'They were closed by the timewe got there'. The sentence is correct, but the word 'they' is incorrect as it is a pronoun, not a noun.
Distillm	Error in grammar: We went to store, so that we could buy food for our picnic; however, they were closed by time we got there. Error in punctuation: However, they were closed, by the time we got, there. Error in spelling: We went to store and buy food for our picnic.
Distillm-2	The following sentences contain errors: We went to store so that We went to store so we could buy some food The following sentences do not contain any errors: No Error
ABKD	Paragraph contains errors: Sentences: We could not go to the store lest we not buy food for our picnic. Sentences: Finally, we got to the store just in time to buy our goods.
TED	No Error
FDD	The sentences contain errors in grammar, punctuation, or spelling.
EvaKD	**Sentences: We went to store so that we could buy some food for our picnic; However, they were closed by the time we got there.**

Table 8: Case study on Minimum-step Reasoning.

Instruction: You need to output the minimum number of steps needed to complete a process, given a set of steps describing the process. If two steps can be combined into one step, that means they do not need to be completed as separate steps.	
Input: Steps: [‘The seeds are dispersed by wind, animals, etc’, ‘The seeds reach the ground’, ‘Grow into new trees’, ‘The process repeats itself over and over’, ‘A tree produces seeds’].	
Ground-truth: The minimum number of steps needed to complete the process is 3.	
Method	**Generated Result**
KD	4
SeqKD	The process will take exactly four steps to complete.
RKD	1 step: The seeds are dispersed by wind
GKD	The minimum number of steps needed is 2, because the process repeats itself twice – once to grow the trees and once to produce the seeds.
ImitKD	The minimum number of steps needed is two. The ground + trees + grow into tree + seeds is a repetitive cycle, so there needs to be four steps in the process to achieve a state instead three if two steps were combined.
MiniLLM	The minimum number of steps needed is 3. The seeds are dispersed by wind and animals, and then reach the ground. The ground is then turned into trees, and the trees produce seeds.
Distillm	3
Distillm-2	The minimum number of steps needed is 3. The first step is “The seeds reach the ground”, which can be combined with the second step “The process repeats itself over” into one step.
ABKD	Two steps can be combined into a single step, so you can output steps 2, 4 and 5 as being the minimum number of steps.
TED	The minimum number of steps needed to complete the process is 5.
FDD	The minimum number of steps needed to complete the process is 4.
EvaKD	**The minimum number of steps needed to complete the process is 3.**

Table 9: Case study on Dialogue Reconstruction.

Instruction: You will be given a piece of text that contains two people talking to each other. Each person's dialogue is split up into different lines and you need to put the pieces back in order to make complete sentences. The input consists of an array of strings, where each string represents one line from either Person A or Person B. Note that the strings alternate between A and B – always starting with A. You also have the transcripts for both sides as arrays with the same length as lines, but they are not guaranteed to be in any particular order.

Input: Lines: ['Person A: Where are you going?', 'Person B: I'm going home.', 'Person A: Why are you going there?', 'Person B: Because I live there.']
TranscriptA: ['I'm going home.', 'Why are you going there?']
TranscriptB: ['Where are you going?', 'Because I live there'].

Ground-truth: Person A: Where are you going?
Person B: I'm going home.
Person A: Why are you going there?
Person B: Because I live there.

Method	Generated Result
KD	A going home: B going home: A: Why are you going there? B: Yeah I have to go there.
SeqKD	1. 'Where are you going? I'm going home.' 2. 'I'm going home. Why are you going there?' 3. 'Where are we going? We are going home.'
RKD	A: I'm going home. B: Why are you goingthere? A: Because I live there. B: Where are you going? A: I'm goinghome. B: Why areyou going there? B: Because I live there.
GKD	A: I'm going home. B: Why are you going home? A: Because I live there B: What are you doing there?
ImitKD	The corrected dataset would be the following: Lines: ['Person B: I're home.', 'Person A: I'm going homea.', 'Person B: Because Ib've raised there.'], Transcripts: ['I'm going brother.', 'Reason to be here?'], ['All bells sound dead.', 'Where are you brother?'] 'led home.', ['You drink what?']
MiniLLM	The first line of each person's dialogue is the starting point for the corresponding person's dialogue. The first line of each person's dialogue is the ending point for the corresponding person's dialogue. The second line of each person's Dialogue is the starting point for their dialogue and the first line of the corresponding person's dialogue is the ending point. The third line of each person's Dialogue: Person A: Where areyougoing? Person B: I'mgoinghome. Person A: Whyareyougoingthere? Person B: BecauseIlivethere.
Distillm	Below is a sorted list of the lines, with the lines for Person A and Person B alternating between the two: ['Person A: Where are-you-going?', 'Person B: -I'm-going-home.', 'Person A: -Why-are-you-going-her?', 'Person B: Because-I-live-there.'] ['Person A: -I'm-greeh-home.', 'Person B: I'm-greehy-yeaving-there.'] ['-I'm-going-home', 'Person A: -Where-are-you-going?'] ['-Where-are-you-going', 'Person B: -Because-I-live-there.]
Distillm-2	The following is the expected output: Person A: Where are-you-going? Person B: I-am-going-home. Person A: Why-are-you-going-there? Person B: -Because-I-live-there.
ABKD	In the above example, lines is an array of strings containing the lines of dialogue, while transcriptsA and transcriptsB are arrays of strings containing the transcripts from Person A and Person B respectively. Note that the strings in transcriptsA alternate between Person A and Person B, so both have the same length.
TED	Here is the completed dialogue in the correct order: Person A: Where are you going? Person B: I'm going home.
FDD	Here is the completed dialogue in order: Person A: Where are you going? Person B: I'm going home. Person A: Why are you going there? Person B: Because I live there.
EvaKD	**Person A: Where are you going?** **Person B: I'm going home.** **Person A: Why are you going there?** **Person B: Because I live there.**