Bridging the Tokenizer Gap: Semantics and Distribution-aware Knowledge Transfer for Unbiased Cross-Tokenizer Distillation

Huazheng Wang[1 2*], Yongcheng Jing[2†], Haifeng Sun[1†], Jingyu Wang[1], Jianxin Liao[1], Leszek Rutkowski[3], Dacheng Tao[2†]

[1] State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]Generative AI Lab, College of Computing and Data Science Nanyang Technological University, Singapore 639798
[3]Systems Research Institute of the Polish Academy of Sciences, AGH University of Krakow,
30-059 Kraków, and the SAN University, 90-113, Łódź, Poland,
{wanghz;hfsun;wangjingyu;liaojx}@bupt.edu.cn, yongcheng.jing@ntu.edu.sg,
leszek.rutkowski@ibspan.waw.pl, dacheng.tao@gmail.com

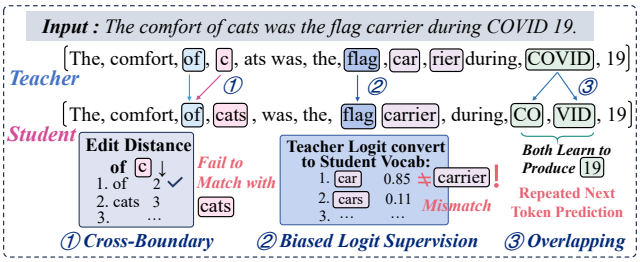
Abstract

Cross-tokenizer knowledge distillation, where the teacher and student employ different tokenizers, is becoming increasingly prevalent, yet it poses underexplored challenges: existing methods fail to capture the rich knowledge encoded in teacher logits, as evidenced by the neglect of semantic information, inaccurate and biased logit alignment, and discarding distributional structure—ultimately leading to unfavorable distillation. To address these issues, we propose SEDI, a semantics and distribution-aware knowledge transfer framework tailored for cross-tokenizer distillation. To preserve factual knowledge, SEDI employs bipartite graph-based alignment at the tokenization level and a sliding window re-encoding strategy at the vocabulary level, enabling unbiased transfer of the teacher's next-token predictions into the student's vocabulary space. To further retain distributional information, we align the student's entropy with that of the teacher by incorporating the student's own logits during training, which helps to mitigate the exposure bias problem. Experiments on ten datasets across three task domains and five different teacher-student model pairs with varying vocabulary sizes demonstrate that SEDI delivers substantial improvements, with gains of up to 19.8%.

Code — https://github.com/MaybeLizzy/SEDI

Introduction

Large language models (LLMs) (OpenAI 2023) have demonstrated remarkable capabilities across a wide range of tasks, largely due to their substantial model capacity—yet this comes with significant computational overhead. To mitigate this, knowledge distillation (Agarwal et al. 2024) has emerged as a key technique for compressing large teacher LLMs into smaller, more efficient student LLMs. Traditional knowledge distillation methods for LLMs typically assume



Methods	Token Semantics	Boundary Consistency	Unbiased Logit Supervision	Non-Overlapping	Overall Confidence
MinED	✓	✗	✗	✗	✗
CDM	✓	✓	✗	✗	✗
ULD	✗	-	-	-	✓
MultiOT	✗	-	-	-	✓
Ours	✓	✓	✓	✓	✓

Figure 1: We identify several key limitations in existing cross-tokenization methods, including *semantic deficiency*, lack of distributional confidence due to *distributional oversimplification*, and various forms of *logit mismatch*, such as cross-boundary alignment, biased supervision, and overlapping predictions—all of which hinder effective distillation.

that the teacher and student models share the same tokenizer (Gu et al. 2024). In practice, however, distillation across different tokenizers is becoming increasingly prevalent and introduces unique difficulties (Boizard et al. 2025). These difficulties primarily arise in two aspects. First, at the *tokenization level*, the teacher and student segment sentences into different sub-tokens, resulting in the sequences of varying lengths. Second, at the *vocabulary level*, the teacher and student employ distinct vocabularies, leading to next-token distributions that differ significantly in both size and semantics. As a result, directly applying KL minimization for distillation becomes considerably more difficult.

To address these challenges, existing cross-tokenization methods generally fall into two categories: (i) Optimal transport-based approaches (Boizard et al. 2025) align the output distributions of the teacher and student models using the Wasserstein distance or its variants (Cui et al. 2025), without accounting for token-level correspondence; (ii) Another

*This work is completed during Huazheng Wang's research attachment at NTU.
†Corresponding Authors.

line of work aims to establish explicit token-to-token mappings using dynamic programming (Wan et al. 2024), based on the minimum edit distance cost between tokens (Chen et al. 2025). Additionally, a learnable projection module (Zhang et al. 2024) can be incorporated to map the hidden states of the two models into a shared space for alignment.

However, these existing approaches face notable limitations in transferring teacher knowledge (Fig. 1): (i) ***Semantic Deficiency*:** Optimal transport-based methods align logits by their numerical distributions but ignore token semantics, causing the student to merely mimic probability rankings rather than truly learn the factual knowledge; (ii) ***Logit Mismatches*:** Explicit token-to-token mapping methods typically result in severe misalignment, as they focus on token similarity while ignoring character order and meaning. This leads to cross-boundary and overlapping mismatches and introduces teacher tokenizer bias, ultimately degrading the student model's language modeling ability and generation quality; (iii) ***Distributional Oversimplification*:** Conventional mapping methods use one-hot logits to avoid mismatches, thereby discarding distributional information and compromising generalization.

To address these limitations by fully leveraging the rich information in the teacher's logits without introducing teacher tokenization bias, we focus on two key aspects of the teacher's next-token predictions: (i) **semantic knowledge** as reflected by the high ranked candidates; (ii) **distributional information** as reflected by the overall confidence. To retain both, in this paper, we propose SEDI, a **Se**mantics and **Di**stribution-aware cross-tokenizer distillation framework.

For one thing, to preserve semantics, we construct pseudo-logits to align the teacher's predicted tokens at two levels:
▷ *At the tokenization level*, we propose a bipartite graph-based alignment, where nodes represent student and teacher token indices obtained by extracting character-level spans for each token using offset information. An undirected edge is added between each pair of tokens whose character spans overlap. We then identify all connected components within this graph to achieve precise tokenization alignment, thereby avoiding any ambiguous or overlapping matches.
▷ *At the vocabulary level*, we introduce sliding window re-encoding strategy that considers both current and subsequent tokens from the teacher and student to determine which teacher logits should be aligned. The top-ranked teacher tokens are decoded and then re-encoded to the student vocabulary space, thereby preventing teacher tokenizer bias.

For another, to further capture model uncertainty and overall confidence, we align the distributional properties by minimizing the entropy (Gao et al. 2025) between the constructed pseudo-logits and the teacher's logits. Notably, since KL-based learning can lead to a distribution mismatch between training and inference (Gu et al. 2024), known as *exposure bias* (Arora et al. 2022), we propose incorporating the student's own logits into the pseudo-logits during training to reduce this distributional shift. This approach helps the student recover from its own mistakes and avoid overfitting.

In sum, our contribution is a novel SEDI framework that bridges the tokenizer gap by enabling the unbiased transfer of comprehensive knowledge from teacher to student. We evaluate the effectiveness of SEDI on three task domains, including instruction following, math reasoning and code generation, using ten datasets and five teacher-student model pairs that cover a range of model and vocabulary sizes. Experimental results show improvements of up to 19.8% on unseen datasets, demonstrating the superiority of SEDI in generalization, while maintaining higher generation quality and lower exposure bias.

Related Work

We provide an overview of recent advances in cross-tokenizer knowledge distillation for LLMs, which can be broadly categorized into two paradigms. One line of work is optimal transport-based methods, which directly aligns the output distributions of teacher and student models using Wasserstein distance (Boizard et al. 2025). To incorporate global information, some works extend optimal transport to sequence-level (Le et al. 2025) via diverse cost matrices, such as Sinkhorn distance (Cui et al. 2025). Another line of work focuses on token-to-token mapping, which explicitly aligns the vocabularies of the teacher and student models using dynamic programming (Fu et al. 2023) to minimize the total edit cost between two token sequences (Wan et al. 2024). To address semantic misalignment, there emerges weighted dynamic time warping by incorporating contextual information to encourage align with lower entropy tokens (Chen et al. 2025). In parallel, other works introduce learnable modules to project the hidden representations of both models into a unified space for alignment. In contrast to previous approaches, our method overcomes the limitations of existing methods from both semantic and distributional perspectives, offering an unbiased and generalized strategy for favorable cross-tokenizer distillation.

Pilot Study

In this section, we provide a detailed analysis of the limitations inherent in existing cross-tokenizer distillation methods.

At the *tokenization level*, as illustrated in Fig. 1, existing methods align the split sub-tokens based on minimum edit distance or token similarity, which suffer from ambiguous or erroneous alignments reflected in two key issues:
▷ ***Cross-Boundary Misalignment.*** When multiple teacher sub-tokens are aligned to a single student token, semantic inconsistencies or conflicting assignments may arise. For example, in the sequence "of cats", the student tokenizer produces `[of, cats]`, while the teacher yields `[of, c, ats]`. Due to the lower edit distance, the teacher token "`c`" may be incorrectly aligned with "`of`" instead of the more appropriate "`cats`", leading to cross-boundary misalignment.
▷ ***Overlapping Misalignment.*** When multiple student sub-tokens are aligned to a single teacher token, they are forced to learn redundant next-token distributions. For instance, in the sequence "COVID 19", the teacher tokenizes "`COVID`" as a single token, while the student splits it into `[CO, VID]`. If both sub-tokens are aligned to "`COVID`", they redundantly predict "`19`", skipping the sub-token "`VID`".

At the *vocabulary level*, discrepancies in vocabulary space introduce further challenges for existing methods:

▷ ***Biased Logit Supervision.*** The teacher's logits are defined over its own vocabulary, which is incompatible with that of the student. Directly converting tokens from the teacher's next-token distribution into the student's vocabulary space can introduce *teacher tokenization bias*, leading to semantic distortions. For instance, in the phrase "the flag carrier", the teacher tokenizes "`carrier`" into `[car, rier]`, while the student treats it as a single token. If the prior token "`flag`" is an exact match, the top candidate in the teacher logit, "`car`", is incorrectly assigned for the student, resulting in a semantically deficient output like "`the flag car`".

▷ ***Lazy Learning.*** In response to the challenge of many-to-many mappings, conventional methods take a shortcut by using one-hot logits, discarding the rich supervision provided by the teacher's distribution. As shown in Tab. 1, a considerable portion of logits are one-hot, reaching as high as 35.12% on the Dolly dataset. We refer to this as *lazy learning*, which leads to overfitting and hinders the generalization ability.

	Dolly	Self-inst	Vicuna	S-NI	Unist
MinED	35.12	38.30	29.11	45.25	44.89

Table 1: Proportion of one-hot logits when using `LLaMA2-7B` as the teacher and `GPT2-124M` as the student.

For a more intuitive evaluation, we report the Top-1 Accuracy by measuring whether the converted top-predicted teacher token matches the ground-truth student token. As shown in Tab. 2, MinEdit achieves the highest top-1 accuracy of only 84.94%, suggesting that a considerable number of tokens remain misaligned. Although CDM avoids using one-hot logits, it averages the logits of many-to-many aligned tokens, thereby still lacking precise token alignment.

Dataset	Dolly	Self-inst	Vicuna	S-NI	Unist
MinED	84.94	63.47	62.37	70.80	76.20
CDM	88.01	65.83	63.03	72.87	76.84
SEDI	95.27	97.32	96.80	98.01	97.62

Table 2: Top-1 accuracy when using `LLaMA2-7B` as the teacher and `GPT2-124M` as the student.

Although optimal transport-based methods avoid explicit token mappings, they align distributions by sorting the teacher and student logits in descending order and simply padding the shorter distribution with zeros, without considering semantic correspondence. As a result, the student fails to truly capture the teacher's factual knowledge.

Proposed Methodology

Motivated by the limitations observed in our pilot study, we model the alignment task as an *unsupervised process* that constructs pseudo-logits, capturing the rich information of the teacher model in a form the student can effectively learn. Specifically, we leverage two key aspects of the teacher's logits: (1) the top candidate next tokens, which reflect factual knowledge, and (2) the overall confidence, which captures distributional information. To preserve both, we propose a novel cross-tokenizer distillation framework, termed SEDI, which encompasses semantics-preserving logit transfer and distribution-aware entropy alignment, as elaborated below.

Semantic-Preserving Logit Transfer

We hypothesize that minimum edit distance-based methods struggle with accurate alignment because they rely only on superficial token similarity, overlooking character order and semantics. To overcome these issues, we introduce a bipartite graph-based alignment at the tokenization level and a sliding window re-encoding strategy at the vocabulary level, allowing for precise transfer of teacher logits to the student space.

Bipartite Graph-Based Tokenization-Level Alignment. Given a sentence x, let $\mathcal{T}_S$ and $\mathcal{T}_T$ denote the student and teacher tokenizers with vocabulary sizes M and N, respectively. We tokenize x to obtain two token sequences: $\mathcal{T}_S(x) = [w_0^S, w_1^S, \ldots, w_{L_S-1}^S]$ and $\mathcal{T}_T(x) = [w_0^T, w_1^T, \ldots, w_{L_T-1}^T]$, where L_S and L_T are the respective sequence lengths. We formulate tokenization alignment as a three-step bipartite graph construction and grouping procedure:

1) Node Construction We define two disjoint sets of nodes corresponding to the student and teacher tokens: $V_s = \{0, 1, \cdots, L_S - 1\}$, $V_t = \{0, 1, \cdots, L_T - 1\}$, where node $i \in V_S$ represents student token w_i^S and node $j \in V_T$ represents teacher token w_j^T. For each node, we extract the character-level span using the tokenizer's offset mappings: $\mathcal{S} = (a_i, b_i)_{i=0}^{L_S-1}$ for student tokens and $\mathcal{T} = (a'_j, b'_j)_{j=0}^{L_T-1}$ for teacher tokens.

2) Edge Construction We then construct the bipartite graph $\mathcal{G}$ by adding an edge (i, j) between student node i and teacher node j if their spans overlap, i.e.,

$$E = \{(i, j) \mid [a_i, b_i) \cap [a'_j, b'_j) \neq \emptyset\}. \tag{1}$$

The span overlap condition is defined as:

$$[a_i, b_i) \cap [a'_j, b'_j) \neq \emptyset \iff \neg(b_i \leq a'_j \vee b'_j \leq a_i). \tag{2}$$

3) Connected Component Grouping Finally, we find all connected components in $\mathcal{G}$. Each connected component corresponds to an alignment group containing a set of student token indices S_g and a set of teacher tokens indices T_g, where $S_g = \{s_{(g,0)}, \ldots, s_{(g,R_g-1)}\}$ of length R_g, and $T_g = \{t_{(g,0)}, \ldots, t_{(g,Q_g-1)}\}$ of length Q_g. The final set of alignment groups is given by:

$$\mathcal{A}^* = \{(S_g, T_g)\}_{g=1}^{G}, \tag{3}$$

where G is the number of connected components in $\mathcal{G}$.

Compared to edit-distance-based alignment, this bipartite-graph approach yields fine-grained, semantically faithful alignment, with each group anchored to precise character-level spans, avoiding fuzzy or overlapping matches.

Sliding Window Re-Encoding for Vocabulary-Level Alignment. At vocabulary level, we incorporate information from both the current and subsequent alignment groups to determine which and how teacher logits should be transferred.

Specifically, consider the current alignment group g, our goal is to construct the pseudo-logits $\tilde{\mathbf{z}}_{\text{teacher-top}k} \in \mathbb{R}^M$ by converting the top-ranked teacher-predicted tokens into the student vocabulary. Let $\mathbf{z}_T^{(t)} \in \mathbb{R}^N$ denote the teacher's logit

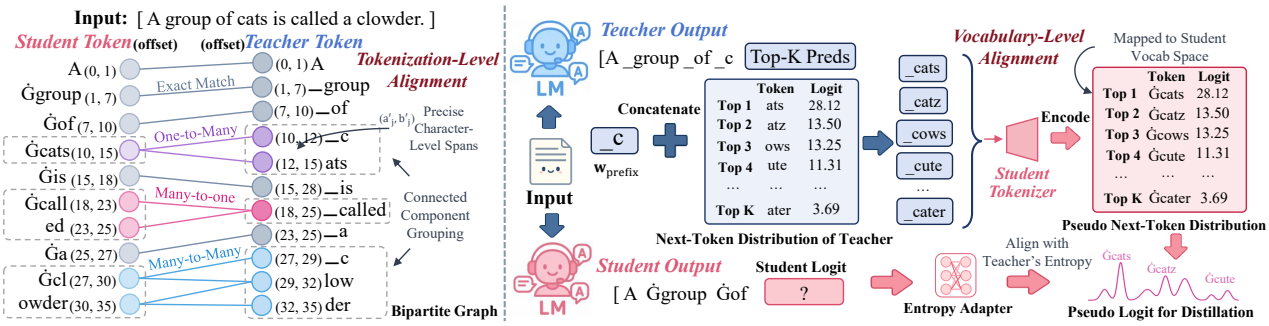


Figure 2: Overview of the SEDI framework, which comprises bipartite graph-based tokenization-level alignment (left), a sliding window re-encoding strategy for vocabulary-level alignment, and entropy alignment that integrates student logits (right).

at position t. We begin by determining whether the next alignment group constitutes a one-to-many or many-to-many mapping, i.e., $Q_{g+1} > 1$. If so, we concatenate the first $Q_{g+1} - 1$ teacher tokens to form a prefix w_{prefix}; otherwise, w_{prefix} is set to the empty string:

$$w_{\text{prefix}} = \texttt{concat}([w^T_{(t_{g+1,0})}, \ldots, w^T_{(t_{g+1,Q_{g+1}-2})}]), \quad (4)$$

where w^T_t denotes the t-th token in $\mathcal{T}_T(x)$. Let $q^\star$ denote the target indices for the teacher logit to be projected:

$$q^* = \begin{cases} t_{(g,Q_g-1)}, & \text{if } Q_{g+1} = 1, \\ t_{(g+1,Q_{g+1}-2)}, & \text{if } Q_{g+1} > 1. \end{cases} \quad (5)$$

We then extract the top-K predictions from the teacher's logit at position $q^\star$, where v_j denotes a teacher vocabulary token ID and p_j is its corresponding score:

$$(v_j, p_j)_{j=1}^K = \text{TopK}(\mathbf{z}_T^{(q^*)}, K). \quad (6)$$

Each v_j is first decoded into its string form and concatenated with the prefix w_{prefix} to form a candidate subword unit $w_j = w_{\text{prefix}} + \texttt{decode}_T(v_j)$, which is then re-encoded using the student tokenizer as $\tilde{v}_j = \texttt{encode}_S(w_j)$. The score p_j is assigned to the first student token ID in $\tilde{v}_j$:

$$\tilde{\mathbf{z}}^{(s_{g,R_g-1})}_{\text{teacher-top}k}[\tilde{v}_j[0]] \leftarrow p_j. \quad (7)$$

If the next student group contains multiple tokens, i.e., $R_{g+1} > 1$, we further assign the score p_j to the leading tokens in the next alignment group S_{g+1} as follows:

$$\tilde{\mathbf{z}}^{(s_{g+1,l-1})}_{\text{teacher-top}k}[\tilde{v}_j[l]] \leftarrow p_j, l = 1, \ldots, \min\{R_{g+1} - 1, |\tilde{v}_j| - 1\}. \quad (8)$$

This approach ensures that top candidate tokens from the teacher are faithfully converted to the student's vocabulary without introducing teacher tokenization bias, thereby avoiding semantic distortion and preserving the student's modeling preferences. Moreover, it prevents the use of one-hot logits and mitigates the risk of lazy learning.

Together, with bipartite graph-based tokenization-level alignment, the flexible mapping in SEDI supports variable-length re-encoding and enhances compatibility across different tokenizers (Fig. 2). As shown in Tab. 2, SEDI achieves a top-1 accuracy of 95.27% on the Dolly dataset, representing an absolute improvement of up to 7.26% over CDM and demonstrating a substantial gain in alignment accuracy.

Distribution-Aware Entropy Alignment

Another important property of the teacher logits is the overall confidence or uncertainty, as quantified by entropy (Agarwal et al. 2025). To preserve this high-level information, we align the entropy of the pseudo-logits with that of the teacher logits, thereby maintaining the teacher's distributional characteristic.

Moreover, depending solely on teacher logits introduces a distribution gap between training and inference, known as *exposure bias* (Gu et al. 2024), which can lead to hallucinations as errors accumulate (Agarwal et al. 2024). To address this, we observe that the student's logits reflect its own generation habits, such as grammar and contextual understanding, even if they lack certain knowledge compared to the teacher. By merging the student's logits into the pseudo-logits during training, we can reduce distribution shift, help the model recover from its own mistakes, and alleviate exposure bias.

Formally, let $\mathbf{z}_{\text{student}} \in \mathbb{R}^M$ denote the student logits. We introduce an entropy adapter $\text{Ada}(\cdot)$, which applies a learnable, element-wise scaling to the student logits. The final pseudo-logits for distillation are given by:

$$\tilde{\mathbf{z}}_{\text{pseudo}} = \underbrace{\tilde{\mathbf{z}}_{\text{teacher-top}k}}_{\text{from teacher logits}} + \underbrace{\text{Ada}(\mathbf{z}_{\text{student}})}_{\text{from student logits}}. \quad (9)$$

To further ensure entropy consistency, we introduce an entropy alignment loss:

$$\mathcal{L}_{\text{entropy}} = \frac{1}{L_S} \sum_{i=1}^{L_S} \left(H\left(\mathbf{z}^{(i)}_{\text{pseudo}}\right) - H\left(\tilde{\mathbf{z}}^{(i)}_{\text{T}}\right) \right)^2, \quad (10)$$

where $\tilde{\mathbf{z}}_{\text{T}}$ denotes the teacher logits after tokenization-level alignment, having the same sequence length L_S as the student. $H(\cdot)$ represents the entropy of the probability distribution induced by the logit $\mathbf{z} \in \mathbb{R}^M$:

$$H(\mathbf{z}) = -\sum_{k=1}^{M} p_k \log p_k, \quad p_k = \frac{\exp(z_k)}{\sum_j \exp(z_j)}. \quad (11)$$

We jointly optimize the student model and entropy adapter using a combination of forward KL divergence loss and entropy alignment loss, where $\mathbf{z}_{\text{student}}$ denotes the student logits:

$$\mathcal{L}_{\text{KD}} = \mathbb{E}_i \left[\text{KL}\left(\mathbf{z}^{(i)}_{\text{pseudo}}, \mathbf{z}^{(i)}_{\text{student}}\right) \right] + \mathcal{L}_{\text{entropy}}. \quad (12)$$

Following prior work (Wan et al. 2024), we use a hyperparameter λ to combine the distillation loss with the standard cross-entropy loss: $\mathcal{L}_{\text{total}} = \lambda \cdot \mathcal{L}_{\text{CE}} + (1 - \lambda) \cdot \mathcal{L}_{\text{KD}}$.

Model	Dataset	*Dolly*		*Self-Inst*		*Vicuna*		*S-NI*		*UnNI*	
	Metric	RougeL	F1	RougeL	F1	RougeL	F1	RougeL	F1	RougeL	F1
Teacher: `Dolly-Pythia-3B`; ***Student***: `OPT-125M`	**Teacher**	30.78	26.19	16.99	12.08	16.62	13.71	25.16	14.74	26.94	18.42
	SFT	19.18	15.74	8.02	6.24	13.34	10.87	11.74	5.32	13.83	8.67
	MinEdit	20.86	16.88	9.49	6.92	14.00	11.56	15.26	6.84	16.57	10.24
	ULD	21.10	17.24	9.72	7.13	13.85	10.72	15.32	7.47	16.88	10.53
	MultiLevelOT	22.32	17.53	9.98	7.30	14.23	11.50	16.11	7.42	16.94	10.43
	CDM	22.56	17.88	10.02	7.14	14.50	11.60	16.35	7.54	16.99	10.89
	DSKD	23.18	18.42	10.44	7.46	15.21	11.66	16.49	7.29	17.36	10.48
	SEDI	**23.92**	**19.54**	**11.27**	**8.17**	**16.16**	**12.68**	**18.18**	**8.01**	**18.51**	**11.76**
Model	Dataset	*Dolly*		*Self-Inst*		*Vicuna*		*S-NI*		*UnNI*	
	Metric	RougeL	F1	RougeL	F1	RougeL	F1	RougeL	F1	RougeL	F1
Teacher: `LLaMA2-7B`; ***Student***: `GPT2-124M`	**Teacher**	30.24	24.31	20.89	14.61	19.71	15.42	36.89	18.39	33.49	22.26
	SFT	20.34	16.80	9.09	6.44	13.96	12.07	17.15	7.74	18.34	11.39
	MinEdit	22.81	17.41	10.38	7.20	14.78	10.40	19.24	9.10	19.45	12.52
	ULD	22.94	17.63	10.68	7.53	14.03	10.15	19.14	9.13	20.16	12.80
	MultiLevelOT	23.11	18.38	11.10	7.75	14.47	12.00	20.25	9.39	20.52	13.15
	CDM	23.17	18.35	11.17	7.90	14.94	11.48	19.97	9.38	20.40	13.33
	DSKD	23.33	18.60	11.68	**8.13**	15.77	12.10	20.70	10.11	21.10	13.59
	SEDI	**24.38**	**19.67**	**12.48**	8.08	**16.54**	**12.12**	**21.92**	**10.70**	**21.77**	**14.20**

Table 3: Experimental results on the Instruction Following dataset, evaluated across two pairs of teacher-student models. SEDI demonstrates superior performance to all baselines, with improvements of up to 12.21% (10.48 → 11.76) on UnNI.

Experimental Setup

In this section, we provide a detailed overview of our experimental setup, including the datasets, evaluation metrics, baselines, and implementation details.

We evaluate SEDI on three tasks enabled by the scaling of model parameters: **instruction following**, **code generation** and **math reasoning** (Tab. 4). For the instruction following task, we use `Dolly-15K` (Ouyang et al. 2022) as training dataset and evaluate on four additional unseen datasets: `Self-Inst` (Wang et al. 2023), `Vicuna` (Chiang et al. 2023), `S-NI` (Wang et al. 2022) and `UnNI` (Honovich et al. 2023). Performance is measured using ROUGE-L and F1 scores. For code generation task, we use `CodeM` (Zan et al. 2024) for training and `HumanEval` (Chen et al. 2021) for evaluation. For math reasoning, we train on `MetaMath` (Yu et al. 2024) and evaluate on three unseen datasets: `GSM8K` (Cobbe et al. 2021), `MATH` (Hendrycks et al. 2021), and `ORCA` (Mitra et al. 2024). We report ROUGE-L and Pass@1, which reflects the percentage of questions correctly passed when generating a single solution per question.

We compare SEDI with standard supervised finetuning (SFT) as well as existing cross-tokenizer distillation methods, including MinED (Wan et al. 2024), ULD (Boizard et al. 2025), MultiLevelOT (Cui et al. 2025), CDM (Chen et al. 2025), and DSKD (Zhang et al. 2024).

Instruction Following Tasks					
Dataset	**Dolly**	**Self-Inst**	**Vicuna**	**S-NI**	**UnNI**
Train	11435	/	/	/	/
Test	500	242	80	731	1000

Math Reasoning & Code Generation Tasks						
Dataset	**MetaMath**	**GSM8K**	**Math**	**ORCA**	**CodeM**	**HumanEval**
Train	10000	/	/	/	9000	/
Test	500	1319	500	107	/	164

Table 4: Statistics and splits of the datasets.

We evaluate four pairs of teacher-student models with varying model scales and vocabulary sizes (Tab. 6). For Dolly, we use `LLaMA2-7B` and `Dolly-Pythia-3B` (Conover et al. 2023) as teachers, with `OpenAI-GPT2-124M` (Radford et al. 2019) and `OPT-125M` (Zhang et al. 2022) as students. For MetaMath, we select `DeepSeek-R1-Distill-Llama-8B` (DeepSeek-AI 2025) and `MetaMath-7B` (Yu et al. 2024) as teachers, and `Pythia-410M` and `GPT2-Medium-355M` (Radford et al. 2019) as the corresponding students. For CodeM, we choose `Deepseek-Coder-6.7B` (Guo et al. 2024) as the teacher, and `Pythia-160M` (Biderman et al. 2023) as student. Except for `MetaMath-7B` and `Deepseek-Coder`, which are already fine-tuned and used directly, all other teacher models are fine-tuned using LoRA with a rank of 256 for 10 epochs. Student models are first fine-tuned for 3 epochs before applying distillation for an additional 7 epochs. For each baseline, we use the default hyperparameters reported in their respective papers. For SEDI, we set $\lambda = 0.5$ and $K = 100$. All evaluation metrics are averaged over five runs.

Experimental Results

In this section, we present both quantitative and qualitative analyses, highlighting key findings from multiple perspectives, including exposure bias, generation fluency, computational overhead and ablation study.

SEDI Achieves Superior Performance across All Tasks and Datasets. As shown in Tab. 3, 5 and 7, SEDI outperforms all baselines by up to 12.21% on instruction following task, 19.80% on math reasoning task and 8.70% on code generation task. The consistent improvements across multiple datasets and task types demonstrate the robustness and generalizability of SEDI in efficiently transferring knowledge.

SEDI Demonstrates Strong Generalization. When evaluated on unseen datasets, SEDI consistently outperforms baseline methods, achieving Pass@1 scores up to 4.38 times higher than SFT on ORCA and exceeding DSKD by 19.8%.

Model	Dataset	*MetaMath*		*GSM8K*		*Math*		*ORCA*	
	Metric	Pass@1	RougeL	Pass@1	RougeL	Pass@1	RougeL	Pass@1	RougeL
Teacher: `DeepSeek-R1-Distill-Llama-8B`; ***Student***: `Pythia-410M`	**Teacher**	44.60	46.29	52.01	35.55	36.22	38.18	45.79	40.69
	SFT	9.41	30.85	4.05	22.84	1.34	20.90	1.93	24.68
	MinEdit	12.63	32.65	5.05	23.43	1.36	19.60	2.27	24.75
	ULD	11.54	32.75	5.12	22.85	1.54	20.22	2.98	25.85
	MultiLevelOT	13.42	33.00	6.91	23.44	1.92	20.35	3.53	26.48
	CDM	13.88	33.29	6.52	23.63	1.95	20.43	3.98	26.44
	DSKD	14.64	33.73	7.23	24.55	2.04	20.96	4.63	27.63
	SEDI	**15.22**	**34.73**	**7.65**	**25.12**	**2.12**	**21.14**	**4.93**	**27.91**
Model	Dataset	*MetaMath*		*GSM8K*		*Math*		*ORCA*	
	Metric	Pass@1	RougeL	Pass@1	RougeL	Pass@1	RougeL	Pass@1	RougeL
Teacher: `MetaMath-7B`; ***Student***: `GPT2-Medium-355M`	**Teacher**	42.40	51.91	40.33	35.04	33.60	34.94	35.97	37.83
	SFT	6.81	33.84	1.41	25.54	1.05	20.80	0.80	28.13
	MinEdit	10.62	36.22	3.56	27.58	1.22	20.93	1.87	29.12
	ULD	9.84	36.24	3.71	28.85	1.20	20.89	1.74	29.81
	MultiLevelOT	10.89	37.02	3.76	29.43	1.40	21.02	2.80	30.74
	CDM	10.74	37.71	3.60	28.84	1.47	21.77	2.87	30.00
	DSKD	11.62	39.04	3.63	29.39	1.73	21.41	2.93	29.74
	SEDI	**12.66**	**39.81**	**3.94**	**29.85**	**1.80**	**22.65**	**3.51**	**31.42**

Table 5: Experimental results on the Math Reasoning dataset, evaluated across two pairs of teacher-student models. SEDI demonstrates superior performance to all baselines, with improvements of up to 19.80% (2.93 $\rightarrow$ 3.51) on UnNI dataset.

Model	Teacher (Vocab Size)	Student (Vocab Size)
Dolly	LLama2-7B (32000) Dolly-Pythia-3B (50280)	GPT2-124M (50257) OPT-125M (50272)
MATH	DeepSeek-R1-Distill-Llama-8B (128256) MetaMath-7B (32001)	Pythia-410M (50304) GPT2-Medium-355M (50257)
CodeM	Deepseek-Coder-6.7B (32256)	Pythia-160M (50304)

Table 6: Model Configurations and Vocabulary Sizes.

Metric	Pass@1	RougeL	Fluency
Teacher	72.56	25.56	8.08
SFT	50.27	18.61	7.33
MinEdit	53.05	21.53	7.27
ULD	54.88	21.96	7.08
MultiLevelOT	55.91	21.47	7.30
CDM	55.63	21.85	7.32
DSKD	56.10	23.01	7.34
SEDI	60.98	22.93	7.39

Table 7: Results on the HumanEval Code Generation dataset. SEDI outperforms baselines by up to 8.70%.

We attribute this strong generalization to three key factors. First, unlike edit distance-based methods, SEDI avoids introducing teacher tokenization bias during distillation and ensures accurate token alignment, thereby preserving the student's language modeling ability. Second, compared to optimal transport-based methods that focus solely on distribution matching, SEDI fully leverages the semantic information in the teacher's logits, resulting in more effective knowledge transfer. Finally, entropy alignment enables the student to learn not only the knowledge itself but also the appropriate level of confidence, reducing overfitting and ultimately leading to better generalization.

SEDI Bridges Large Model Gaps across Architectures. We evaluate SEDI on five diverse teacher-student model pairs with varying model sizes and architectures. Even in scenarios with a substantial parameter gap between teacher and student, where the student typically faces significant learning challenges due to limited capacity (Zhong et al. 2024), SEDI still demonstrates impressive improvements. For example, when using `Deepseek-Coder-6.7B` as the teacher and `Pythia-160M` as the student, representing a 41.9$\times$ size gap, SEDI achieves improvements of up to 8.70%.

SEDI is Robust to Vocabulary Size Variation. We evaluate SEDI in scenarios with diverse vocabulary size relationships, including cases where the teacher's vocabulary is much larger, smaller, or comparable to the student's. Across all settings, SEDI consistently achieves substantial improvements. Notably, when using `DeepSeek-R1-Distill-Llama-8B` as teacher and `Pythia-410M` as student, the student's vocabulary is 2.55 times smaller than the teacher's. A smaller vocabulary typically reduces tokenization efficiency and limits the model's representational capacity, making the student's learning task more challenging (Takase et al. 2024). Nevertheless, SEDI still improves the student's math reasoning ability by up to 5.8% on GSM8K, demonstrating its superiority.

Stronger Teachers with Larger Vocabularies lead to Better Students. We observe an interesting phenomenon in math reasoning tasks: when using two teachers with different vocabulary sizes, the student trained with `GPT2-Medium-355M` achieves a higher ROUGE score than that trained with `Pythia-410M`, but a lower Pass@1 score. We suspect this is because the student primarily learns to *imitate the output style of the teacher rather than its actual capabilities* (Gudibande et al. 2023), resulting in output text that more closely resembles the ground truth in form but lacks true reasoning ability. We hypothesize that this phenomenon is related to the significant difference in teacher model vocabulary size. Notably, `DeepSeek-R1-Distill-Llama-8B` has a vocabulary four times larger than that of `MetaMath-7B`, offering greater tokenization fertility and allowing it to capture a broader

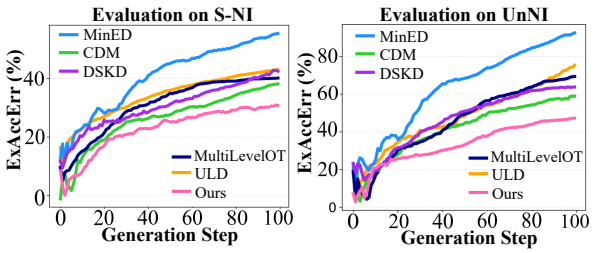


Figure 3: Illustration of ExAccErr, which quantifies the excess error accumulated due to the distribution mismatch between training and inference. Lower values of ExAccErr reflect a reduction in exposure bias.

Metric	Dolly	Self-Inst	Vicuna	S-NI	UnNI
MinEdit	4.86	4.45	5.46	3.65	3.74
ULD	4.70	4.36	6.01	3.79	3.96
MultiLevelOT	4.87	4.66	6.30	3.88	3.98
CDM	4.86	4.68	6.19	3.86	3.92
DSKD	4.81	4.46	6.28	3.75	3.95
SEDI	**4.91**	**4.69**	**6.33**	**3.92**	**3.99**

Table 8: Fluency ratios across instruction-following datasets.

range of concepts and nuances in the corpus (Tao et al. 2024). In such cases, teacher models with richer vocabularies tend to facilitate more comprehensive knowledge transfer, enabling student models to acquire genuine capabilities rather than merely imitating surface-level patterns.

SEDI Mitigates Exposure Bias More Effectively. To quantify exposure bias, we follow Arora et al. (2022); Gu et al. (2024) and evaluate the excess error accumulated over generation length, referred to as the ExAccErr metric. Experiments are conducted on instruction-following tasks using two unseen datasets, S-NI and UnNI, with `OPT-125M` as the student model. As shown in Fig. 3, at the early stages of generation, the ExAccErr curve displays noticeable fluctuations. This may be attributed to the limited context and statistical noise when the model is still generating with near-oracle prefixes. As generation proceeds, exposure bias accumulates, resulting in a steady increase in ExAccErr as the model's outputs increasingly deviate from the ground-truth distribution. Despite this, SEDI consistently exhibits lower exposure bias as generation length grows. We attribute this to incorporating the student's own logits during distillation, thereby reducing the mismatch between training and inference-time generation.

SEDI Enhances Generation Quality. To evaluate the generation quality of existing methods, we adopt the Fluency metric (Meng et al. 2022), calculated as the weighted average of bi- and tri-gram entropies, defined as $-\sum_k f(k)\log_2 f(k)$, where $f(\cdot)$ denotes the n-gram frequency distribution. A higher Fluency score indicates more informative and diverse generation. Experiments are conducted on instruction-following tasks using the LLaMA-GPT teacher-student model pair. As shown in Tab. 8, SEDI consistently achieves the highest fluency ratio. We attribute this to the semantics-preserving logit transfer strategy, which enables SEDI to align teacher tokens to the student vocabulary without introducing teacher tokenization bias. Nevertheless, the absolute fluency on unseen datasets remains suboptimal, requiring further preference learning (Fang et al. 2025; Liu et al. 2025).

Metric	Training Time (Second)	Memory (MiB)
MinEdit	1.5714	23212
ULD	1.5311	24672
MultiLevelOT	1.5110	23960
CDM	5.7921	19982
DSKD	1.5276	24836
SEDI	1.8432	26362

Table 9: Computational overhead of existing methods.

Dataset	Dolly	Self-Inst	Vicuna	S-NI	UnNI
K=10	23.89	10.32	15.81	14.97	17.82
K=50	**24.05**	10.65	15.90	17.16	17.96
K=100	23.92	**11.27**	16.16	**18.18**	18.51
K=200	23.88	11.25	16.17	18.17	18.53
K=300	23.86	11.23	**16.19**	18.15	18.52
K=400	23.79	11.19	16.15	18.11	**18.57**

Table 10: Impact of K on instruction-following datasets.

SEDI is Computationally Efficient. We evaluate the computational overhead from two perspectives: *training time per step* and *maximum memory usage*. Experiments are conducted on the Dolly dataset using the LLaMA-GPT teacher-student model pair. As shown in Tab. 9, SEDI incurs minimal additional cost in terms of training time or memory usage, confirming its computational efficiency.

K as the number of teacher tokens to be projected. The choice of top-K tokens from the teacher's logits for mapping to the student vocabulary plays a crucial role in the performance of SEDI. We investigate the impact of K on the instruction-following task using the Pythia-OPT teacher-student model pair. As shown in Tab. 10, SEDI already achieves state-of-the-art results with $K = 10$ compared to the baselines, but its generalization ability remains limited. As K increases, the generalization of SEDI to unseen datasets improves, reaching optimal performance mostly at $K = 100$. However, as K continues to grow, accuracy begins to fluctuate, likely due to the introduction of noisy token mappings. Therefore, we set $K = 100$ in our experiments.

Conclusion

In this paper, we present SEDI, a novel semantic-preserving and distribution aware alignment framework for cross-tokenizer knowledge distillation. SEDI bridges the tokenizer gap by transferring both factual and distributional knowledge from teacher next-token predictions into a form that the student model can effectively learn, without introducing teacher tokenization bias. The proposed approach leads to improved generalization, higher generation quality, and reduced exposure bias. Overall, SEDI offers an intuitive yet effective solution, even in scenarios with large model and vocabulary size gaps, thereby enabling more flexible and scalable deployment of language models in real-world applications.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants (62471055, 62321001, U23B2001, 62201072), the High-Quality Development Project of the MIIT(2440STCZB2584), the Ministry of Education and China Mobile Joint Fund (MCM20200202, MCM20180101), the Fundamental Research Funds for the Central Universities (2024PTB-004), the 2025 Education and Teaching Reform Project Funding at Beijing University of Posts and Telecommunications (2025YZ005), and BUPT Excellent Ph.D. Students Foundation (CX20242009).

This work was also supported by the program "Excellence initiative - research university" for the AGH University of Krakow, as well as the ARTIQ project UMO-2021/01/2/ST6/00004 and ARTIQ/0004/2021, and by Polish Ministry of Science and Higher Education funds assigned to the AGH University of Krakow. Dr Tao's research is supported by NTU RSR and Start Up Grants.

References

Agarwal, R.; Vieillard, N.; Zhou, Y.; Stanczyk, P.; Garea, S. R.; Geist, M.; and Bachem, O. 2024. On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*.

Agarwal, S.; Zhang, Z.; Yuan, L.; Han, J.; and Peng, H. 2025. The Unreasonable Effectiveness of Entropy Minimization in LLM Reasoning. *CoRR*, abs/2505.15134.

Arora, K.; Asri, L. E.; Bahuleyan, H.; and Cheung, J. C. K. 2022. Why Exposure Bias Matters: An Imitation Learning Perspective of Error Accumulation in Language Generation. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, 700–710. Association for Computational Linguistics.

Biderman, S.; Schoelkopf, H.; Anthony, Q. G.; Bradley, H.; O'Brien, K.; Hallahan, E.; Khan, M. A.; Purohit, S.; Prashanth, U. S.; Raff, E.; Skowron, A.; Sutawika, L.; and van der Wal, O. 2023. Pythia: A Suite for Analyzing Large Language Models Across Training and Scaling. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, 2397–2430. PMLR.

Boizard, N.; Haddad, K. E.; Hudelot, C.; and Colombo, P. 2025. Towards Cross-Tokenizer Distillation: the Universal Logit Distillation Loss for LLMs. *Trans. Mach. Learn. Res.*

Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; de Oliveira Pinto, H. P.; Kaplan, J.; Edwards, H.; Burda, Y.; Joseph, N.; Brockman, G.; Ray, A.; Puri, R.; Krueger, G.; Petrov, M.; Khlaaf, H.; Sastry, G.; Mishkin, P.; Chan, B.; Gray, S.; Ryder, N.; Pavlov, M.; Power, A.; Kaiser, L.; Bavarian, M.; Winter, C.; Tillet, P.; Such, F. P.; Cummings, D.; Plappert, M.; Chantzis, F.; Barnes, E.; Herbert-Voss, A.; Guss, W. H.; Nichol, A.; Paino, A.; Tezak, N.; Tang, J.; Babuschkin, I.; Balaji, S.; Jain, S.; Saunders, W.; Hesse, C.; Carr, A. N.; Leike, J.; Achiam, J.; Misra, V.; Morikawa, E.; Radford, A.; Knight, M.; Brundage, M.; Murati, M.; Mayer, K.; Welinder, P.; McGrew, B.; Amodei, D.; McCandlish, S.; Sutskever, I.; and Zaremba, W. 2021. Evaluating Large Language Models Trained on Code. arXiv:2107.03374.

Chen, Y.; Liu, Y.; Meng, F.; Chen, Y.; Xu, J.; and Zhou, J. 2025. Enhancing Cross-Tokenizer Knowledge Distillation with Contextual Dynamical Mapping. *CoRR*, abs/2502.11104.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; Stoica, I.; and Xing, E. P. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; Hesse, C.; and Schulman, J. 2021. Training Verifiers to Solve Math Word Problems. *arXiv preprint arXiv:2110.14168*.

Conover, M.; Hayes, M.; Mathur, A.; Xie, J.; Wan, J.; Shah, S.; Ghodsi, A.; Wendell, P.; Zaharia, M.; and Xin, R. 2023. Free Dolly: Introducing the World's First Truly Open Instruction-Tuned LLM.

Cui, X.; Zhu, M.; Qin, Y.; Xie, L.; Zhou, W.; and Li, H. 2025. Multi-Level Optimal Transport for Universal Cross-Tokenizer Knowledge Distillation on Language Models. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, 23724–23732. AAAI Press.

DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv:2501.12948.

Fang, W.; Liu, S.; Zhou, Y.; Zhang, K.; Zheng, T.; Chen, K.; Song, M.; and Tao, D. 2025. SeRL: Self-Play Reinforcement Learning for Large Language Models with Limited Data. *arXiv preprint arXiv:2505.20347*.

Fu, Y.; Peng, H.; Ou, L.; Sabharwal, A.; and Khot, T. 2023. Specializing Smaller Language Models towards Multi-Step Reasoning. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, 10421–10430. PMLR.

Gao, Z.; Chen, L.; Zhou, J.; and Dai, B. 2025. One-shot Entropy Minimization. *CoRR*, abs/2505.20282.

Gu, Y.; Dong, L.; Wei, F.; and Huang, M. 2024. MiniLLM: Knowledge Distillation of Large Language Models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Gudibande, A.; Wallace, E.; Snell, C.; Geng, X.; Liu, H.; Abbeel, P.; Levine, S.; and Song, D. 2023. The False Promise of Imitating Proprietary LLMs. *CoRR*, abs/2305.15717.

Guo, D.; Zhu, Q.; Yang, D.; Xie, Z.; Dong, K.; Zhang, W.; Chen, G.; Bi, X.; Wu, Y.; Li, Y.; Luo, F.; Xiong, Y.; and Liang, W. 2024. DeepSeek-Coder: When the Large Language Model Meets Programming – The Rise of Code Intelligence.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring Mathematical Problem Solving With the MATH Dataset. In Vanschoren, J.; and Yeung, S., eds., *Proceedings of the Neural Information Processing Systems Track on Datasets*

and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual.

Honovich, O.; Scialom, T.; Levy, O.; and Schick, T. 2023. Unnatural Instructions: Tuning Language Models with (Almost) No Human Labor. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, 14409–14428. Association for Computational Linguistics.

Le, A. D.; Vu, T.; Hai, N. L.; Diep, N. T. N.; Van, L. N.; Le, T.; and Nguyen, T. H. 2025. CoT2Align: Cross-Chain of Thought Distillation via Optimal Transport Alignment for Language Models with Different Tokenizers. *CoRR*, abs/2502.16806.

Liu, S.; Fang, W.; Hu, Z.; Zhang, J.; Zhou, Y.; Zhang, K.; Tu, R.; Lin, T.-E.; Huang, F.; Song, M.; et al. 2025. A survey of direct preference optimization. *arXiv preprint arXiv:2503.11701*.

Meng, K.; Bau, D.; Andonian, A.; and Belinkov, Y. 2022. Locating and Editing Factual Associations in GPT. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Mitra, A.; Khanpour, H.; Rosset, C.; and Awadallah, A. 2024. Orca-Math: Unlocking the potential of SLMs in Grade School Math. arXiv:2402.14830.

OpenAI. 2023. GPT-4 Technical Report. *CoRR*, abs/2303.08774.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C. L.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P. F.; Leike, J.; and Lowe, R. 2022. Training language models to follow instructions with human feedback. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language Models are Unsupervised Multitask Learners.

Takase, S.; Ri, R.; Kiyono, S.; and Kato, T. 2024. Large Vocabulary Size Improves Large Language Models. *CoRR*, abs/2406.16508.

Tao, C.; Liu, Q.; Dou, L.; Muennighoff, N.; Wan, Z.; Luo, P.; Lin, M.; and Wong, N. 2024. Scaling Laws with Vocabulary: Larger Models Deserve Larger Vocabularies. In Globersons, A.; Mackey, L.; Belgrave, D.; Fan, A.; Paquet, U.; Tomczak, J. M.; and Zhang, C., eds., *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Wan, F.; Huang, X.; Cai, D.; Quan, X.; Bi, W.; and Shi, S. 2024. Knowledge Fusion of Large Language Models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Wang, Y.; Kordi, Y.; Mishra, S.; Liu, A.; Smith, N. A.; Khashabi, D.; and Hajishirzi, H. 2023. Self-Instruct: Aligning Language Models with Self-Generated Instructions. In Rogers, A.; Boyd-Graber, J. L.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, 13484–13508. Association for Computational Linguistics.

Wang, Y.; Mishra, S.; Alipoormolabashi, P.; Kordi, Y.; Mirzaei, A.; Arunkumar, A.; Ashok, A.; Dhanasekaran, A. S.; Naik, A.; Stap, D.; Pathak, E.; Karamanolakis, G.; Lai, H. G.; Purohit, I.; Mondal, I.; Anderson, J.; Kuznia, K.; Doshi, K.; Patel, M.; Pal, K. K.; Moradshahi, M.; Parmar, M.; Purohit, M.; Varshney, N.; Kaza, P. R.; Verma, P.; Puri, R. S.; Karia, R.; Sampat, S. K.; Doshi, S.; Mishra, S.; A, S. R.; Patro, S.; Dixit, T.; Shen, X.; Baral, C.; Choi, Y.; Hajishirzi, H.; Smith, N. A.; and Khashabi, D. 2022. Benchmarking Generalization via In-Context Instructions on 1, 600+ Language Tasks. *CoRR*, abs/2204.07705.

Yu, L.; Jiang, W.; Shi, H.; Yu, J.; Liu, Z.; Zhang, Y.; Kwok, J. T.; Li, Z.; Weller, A.; and Liu, W. 2024. MetaMath: Bootstrap Your Own Mathematical Questions for Large Language Models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Zan, D.; Yu, A.; Liu, W.; Shen, B.; Lin, S.; Gong, Y.; Yao, Y.; Liu, Y.; Guan, B.; Luo, W.; Wang, Y.; Wang, Q.; and Cui, L. 2024. CodeM: Less Data Yields More Versatility via Ability Matrix. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, 714–729.

Zhang, S.; Roller, S.; Goyal, N.; Artetxe, M.; Chen, M.; Chen, S.; Dewan, C.; Diab, M.; Li, X.; Lin, X. V.; Mihaylov, T.; Ott, M.; Shleifer, S.; Shuster, K.; Simig, D.; Koura, P. S.; Sridhar, A.; Wang, T.; and Zettlemoyer, L. 2022. OPT: Open Pre-trained Transformer Language Models. arXiv:2205.01068.

Zhang, S.; Zhang, X.; Sun, Z.; Chen, Y.; and Xu, J. 2024. Dual-Space Knowledge Distillation for Large Language Models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, 18164–18181. Association for Computational Linguistics.

Zhong, Q.; Ding, L.; Shen, L.; Liu, J.; Du, B.; and Tao, D. 2024. Revisiting Knowledge Distillation for Autoregressive Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, 10900–10913. Association for Computational Linguistics.