# SerialDistill: Retrieval-Conditioned On-Policy Distillation for Data-Scarce Serial Subtitle Translation

**Yiran Zhang**[1] **Ziqi Zhang**[2]
[1]Tianjin University of Finance & Economics, Tianjin, China
[2]Harbin Institute of Technology, Harbin, China
[1]shiro027@stu.tjufe.edu.cn [2]2023110966@stu.hit.edu.cn

## Abstract

Serial subtitle translation requires work-specific terminology and translation conventions to be maintained across episodes, despite scarce bilingual supervision. Previously finalized subtitles provide both adaptation data and translation memory, but our experiments show that the benefit of combining supervised adaptation and retrieval varies across series. We introduce **SerialDistill**, which combines LoRA-based domain adaptation with retrieval-conditioned on-policy distributional distillation (OPD). A Qwen3-8B student generates translation trajectories, while a frozen Qwen3-14B teacher provides token-level supervision under the same bilingual context retrieved exclusively from previously finalized episodes. On Japanese-to-Chinese translation across two anime series, SerialDistill improves over the non-distilled LoRA-8B+RAG baseline by up to **3.80 chrF and 4.15 BLEU**, measured on pooled held-out episodes. The student also outperforms the larger retrieval-augmented teacher, which has not undergone domain adaptation. These results support retrieval-conditioned distillation as an additional adaptation stage for data-scarce serial subtitle translation.

## 1 Introduction

Serial subtitle translation is a data-scarce domain-adaptation task: only a few thousand work-specific bilingual subtitle pairs are available in our setting, yet translation must remain consistent in character names, terminology, and recurring expressions across episodes. As episodes are finalized, their subtitles form a growing translation memory that is limited in quantity but closely matched to subsequent episodes. We therefore ask how a compact model can effectively exploit this scarce supervision and historical bilingual context for Japanese-to-Chinese subtitle translation.

Supervised adaptation can encode work-specific patterns through low-rank adaptation (LoRA) (Hu et al., 2022), while retrieval-augmented generation (RAG) makes earlier translations explicitly available during generation. However, the gains from their combination vary across series. Combining LoRA and RAG improves translation on the primary series, but does not consistently outperform either component alone on a second series. This motivates investigating whether teacher supervision can further improve a domain-adapted student operating with historical bilingual context.

To address this question, we develop **SerialDistill**, which adapts on-policy distributional distillation (Agarwal et al., 2024) to serial subtitle translation. As illustrated in Figure 1, SerialDistill conditions both student generation and teacher supervision on episode-causal bilingual memory retrieved exclusively from previously finalized episodes. A LoRA-adapted student generates translation trajectories under this historical context, and a frozen teacher provides next-token distributions along the same student-generated prefixes under identical context. This couples the distillation process to the retrieval conditions used at inference time.

We evaluate SerialDistill on two anime series using a Qwen3-8B student and a Qwen3-14B teacher (Yang et al., 2025). On the pooled held-out sets, SerialDistill improves over LoRA-8B+RAG and achieves the highest chrF, BLEU, and COMET scores among the evaluated systems on both series. These results support the additional value of retrieval-conditioned distillation.

Our contributions are threefold:

- We study data-scarce serial subtitle translation with bilingual memory restricted to previously finalized episodes.
- We introduce SerialDistill, combining domain adaptation with retrieval-conditioned on-policy distributional distillation.
- We demonstrate improvements over adapta-



*Proceedings of the 13th Workshop on Asian Translation (WAT 2026)*, pages 71–83
November 9, 2026 ©2026 Association for Computational Linguistics

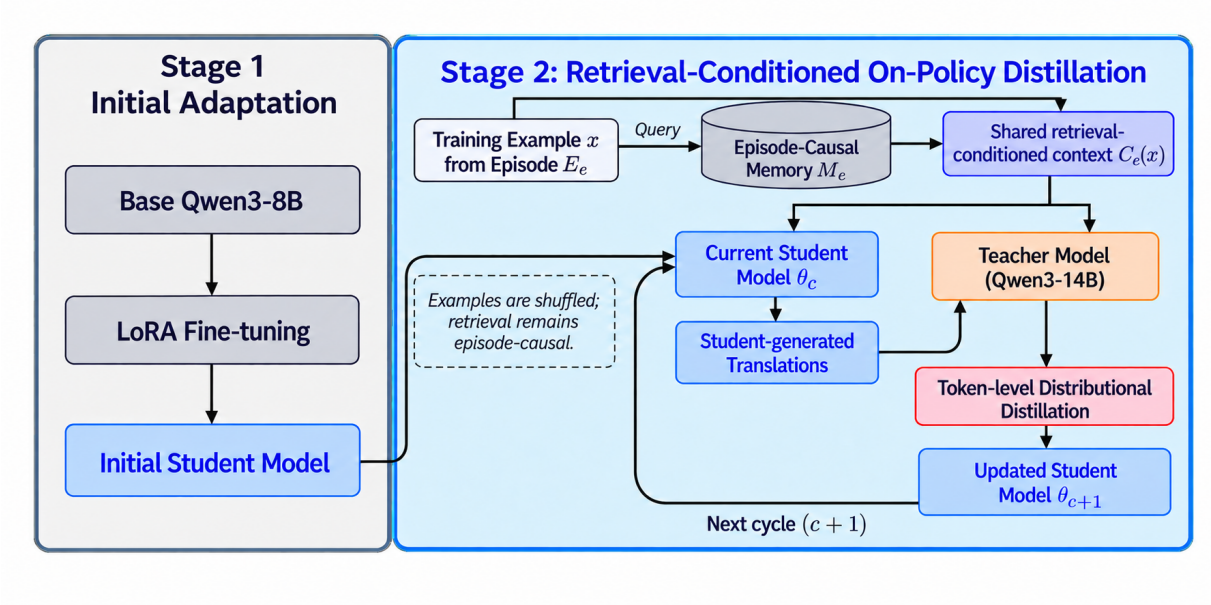



Figure 1: Overview of SerialDistill. Qwen3-8B is first adapted with LoRA, then refined through retrieval-conditioned on-policy distributional distillation. Training examples are shuffled, while each instance accesses only episode-causal memory from previously finalized episodes.

tion, retrieval, and their direct combination on the pooled held-out sets of two anime series.

## 2 Related Work

### 2.1 Large Language Models for Machine Translation

Large language models (LLMs) have been evaluated for translation across language pairs and prompting settings (Hendy et al., 2023; Jiao et al., 2023). For few-shot translation, example quality and domain relevance influence performance (Vilar et al., 2023; Agrawal et al., 2023). The use of contextual information has been explored for document-level and subtitle translation (Wang et al., 2023; Pramodya et al., 2025). However, effectively exploiting broader context remains challenging (Jin et al., 2023). Studies of subtitle translation have also examined context-dependent phenomena such as deixis, ellipsis, and lexical cohesion (Voita et al., 2019). Our setting is a domain-specific, low-resource case in which earlier subtitles remain directly relevant to later episodes.

### 2.2 Domain Adaptation and Parameter-Efficient Fine-Tuning

Fine-tuning on in-domain data is a standard approach to domain adaptation in neural machine translation (NMT) (Saunders, 2022). Parameter-efficient alternatives include lightweight adapter modules for pretrained language models and translation systems (Houlsby et al., 2019; Bapna and Firat, 2019b). LoRA represents parameter updates with low-rank matrices (Hu et al., 2022). In subtitle translation, LoRA can internalize recurring terminology, conventions, and style; we pair this parameter adaptation with retrieval to preserve access to historical translations.

### 2.3 Retrieval-Augmented Machine Translation

Retrieval-augmented machine translation supplements a model with relevant bilingual examples or external knowledge at inference time; earlier subtitles can therefore serve as a translation memory. Earlier retrieval-based MT methods reuse translation pieces or retrieve phrase-level context from bilingual memory (Zhang et al., 2018; Bapna and Firat, 2019a). At the token level, kNN-MT augments predictions with a datastore of translation examples and supports domain adaptation through the choice of datastore (Khandelwal et al., 2021). More recent work has studied retrieval for machine translation in semantic, document-level, and external-knowledge settings (Bouthors et al., 2024; Peng et al., 2025; Wang et al., 2025; Lu et al., 2025). Serial subtitles are a natural test bed because finalized episodes often reuse terminology and expressions. Our work uses this translation memory to condition both teacher and student during distillation.

### 2.4 Knowledge Distillation and On-Policy Distillation

Knowledge distillation transfers information from a stronger teacher through generated sequences or predictive distributions (Hinton et al., 2015). Sequence-level distillation trains on teacher translations (Kim and Rush, 2016), whereas token-level methods preserve richer distributional information. Recent LLM distillation methods also study the choice of distribution-matching objectives and the use of student-generated outputs (Gu et al., 2024; Ko et al., 2024). Offline methods typically supervise fixed teacher trajectories; on-policy distillation instead lets the student generate trajectories and obtains supervision on the states it visits (Agarwal et al., 2024). Our work applies on-policy distributional distillation to data-scarce serial subtitle translation. We combine a domain-adapted student with bilingual retrieval restricted to previously finalized episodes, using the same retrieved context for teacher and student.

## 3 Method

### 3.1 Task Definition and Overview

Given a Japanese subtitle sequence

$$X = (x_1, \ldots, x_n), \tag{1}$$

the model generates the corresponding Chinese subtitle sequence

$$Y = (y_1, \ldots, y_n). \tag{2}$$

Here, $x_i$ and $y_i$ denote aligned subtitle segments. Within an individual target segment $y$, $y_t$ denotes its $t$-th token.

We consider a serial translation setting in which each training instance is associated with an episode-causal retrieval memory. When translating an example from episode $e$, the retrieval memory contains only finalized Japanese–Chinese subtitle pairs from episodes $1, \ldots, e-1$. Target subtitles from the current and future episodes are excluded from retrieval. Source-side subtitles from future episodes are also excluded from retrieval in this evaluated setting.

Our framework, **SerialDistill**, consists of two stages. First, Qwen3-8B is adapted to the target series through supervised fine-tuning (SFT) with LoRA. Second, the resulting domain-adapted student is further refined through retrieval-conditioned on-policy distributional distillation from a frozen Qwen3-14B teacher.

This episode-causal construction aligns retrieval-conditioned distillation with the assumed serial translation workflow: the retrieved bilingual evidence excludes the current and future episodes. This restriction applies to retrieval memory; it does not impose an episode-by-episode order on parameter updates.

### 3.2 Domain Adaptation with LoRA

We first adapt Qwen3-8B using the available in-domain bilingual subtitles. LoRA is used for parameter-efficient supervised fine-tuning, while the underlying pretrained model remains frozen.

The student is optimized with the standard autoregressive response-token objective:

$$\mathcal{L}_{\mathrm{SFT}} = -\sum_{t} \log p_\theta(y_t \mid y_{<t}, x). \tag{3}$$

This stage adapts the student to work-specific terminology, translation conventions, and recurring linguistic patterns. The resulting Qwen3-8B LoRA model serves as the initialization for the subsequent distillation stage.

### 3.3 Episode-Causal Subtitle Memory

Let the finalized bilingual subtitle set for episode $j$ be

$$D_j = \{(x_i^{(j)}, y_i^{(j)})\}_{i=1}^{N_j}, \tag{4}$$

where $x_i^{(j)}$ and $y_i^{(j)}$ denote the Japanese source and finalized Chinese translation of subtitle $i$.

When processing episode $e$, the available translation memory is

$$M_e = \bigcup_{j<e} D_j. \tag{5}$$

Only subtitle pairs originating from episodes earlier than $e$ are eligible for retrieval; pairs from the current and future episodes are excluded.

For a current Japanese subtitle $x$, the retriever returns up to $K$ highest-scoring bilingual matches from $M_e$:

$$R_e(x) = \mathrm{TopK}_{(x',y')\in M_e} \mathrm{BM25}(x, x'). \tag{6}$$

The retrieved bilingual examples and the current source subtitle are then combined into a single model context:

$$C_e(x) = \mathrm{Prompt}\left(R_e(x), x\right). \tag{7}$$

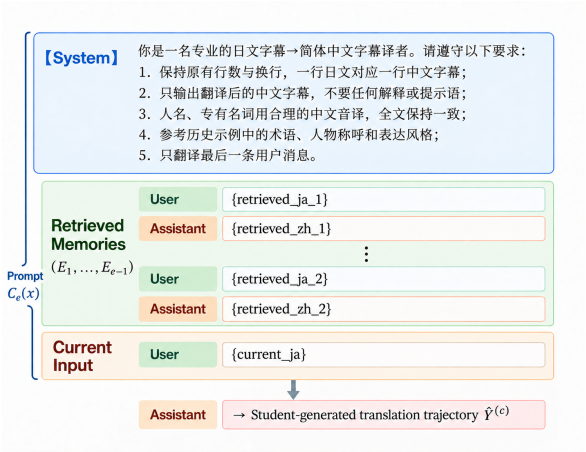



Figure 2: Role-structured construction of the episode-causal retrieval prompt. Retrieved bilingual examples are represented as User–Assistant dialogue turns, followed by the current Japanese input. A fixed bilingual demonstration is omitted for clarity.

Figure 2 illustrates the role-structured prompt construction used to instantiate $C_e(x)$.

This construction makes retrieval strictly causal with respect to episode order while allowing the bilingual memory to grow naturally as more episodes become historical.

Training instances are shuffled for OPD; episode order constrains retrieval availability rather than optimization order.

### 3.4 Retrieval Conditioning

SerialDistill uses a frozen Qwen3-14B model as the teacher and the LoRA-adapted Qwen3-8B model as the student.

During distillation, both models receive the same current source subtitle and the same causal retrieval context. Their predictive distributions are therefore

$$p_T\left(y_t \mid C_e(x), y_{<t}\right) \tag{8}$$

for the teacher, and

$$p_\theta\left(y_t \mid C_e(x), y_{<t}\right) \tag{9}$$

for the student.

The teacher contributes the predictive behavior of the larger model, while the student begins with work-specific knowledge already acquired through SFT. The teacher and student operate under identical retrieval evidence, so teacher supervision is conditioned on information that is also available to the student.

### 3.5 Retrieval-Conditioned Distillation

Figure 3 summarizes the distillation cycle. Let $\theta_c$ denote the student parameters and $B_c$ the batch of training examples at the beginning of cycle $c$. For each example $x \in B_c$, we first construct its episode-causal retrieval context $C_e(x)$. The current student then samples a fresh translation trajectory

$$\hat{Y} = (\hat{y}_1, \ldots, \hat{y}_T) \sim \pi_{\theta_c}^{\text{rollout}}(\cdot \mid C_e(x)). \tag{10}$$

Here, $\pi_{\theta_c}^{\text{rollout}}$ denotes the current student's sampling policy with temperature 0.7 and top-$p$ 0.9. Teacher and student distributions used in the KL objective are computed at temperature 1 over the full vocabulary, without nucleus truncation. Generated token IDs and teacher targets are held fixed during backpropagation.

The frozen teacher evaluates the same student-generated prefixes under the identical retrieval context:

$$p_T^t(v) = p_T(v \mid C_e(x), \hat{y}_{<t}), \tag{11}$$

$$p_{\theta_c}^t(v) = p_{\theta_c}(v \mid C_e(x), \hat{y}_{<t}). \tag{12}$$

At each response position $t$, let $S_t$ denote the teacher's top-$K_T$ token set. Both models are compressed using the same set. For $m \in \{T, \theta_c\}$, we define a distribution over $S_t \cup \{\text{other}\}$:

$$q_m^t(v) = \begin{cases} p_m^t(v), & v \in S_t, \\ 1 - \sum_{u \in S_t} p_m^t(u), & v = \text{other}. \end{cases} \tag{13}$$

The retained probabilities are not renormalized; the remaining probability mass forms the residual category.

The loss for cycle $c$ is computed as the token-weighted mean KL divergence over all valid response tokens in the cycle:

$$\mathcal{L}_c = \frac{\sum_{x \in B_c} \sum_{t \in \mathcal{T}_x} D_{\text{KL}}\left(q_T^{x,t} \,\|\, q_{\theta_c}^{x,t}\right)}{\sum_{x \in B_c} |\mathcal{T}_x|}, \tag{14}$$

where $\mathcal{T}_x$ contains response positions whose generated token IDs are not special tokens. Prompt, padding, and special-token positions, including EOS, are excluded from the loss.

Gradients are accumulated over the cycle and normalized by its total number of valid response tokens. After gradient clipping, we perform one AdamW update of the student's LoRA parameters, yielding $\theta_{c+1}$. The student backbone and teacher

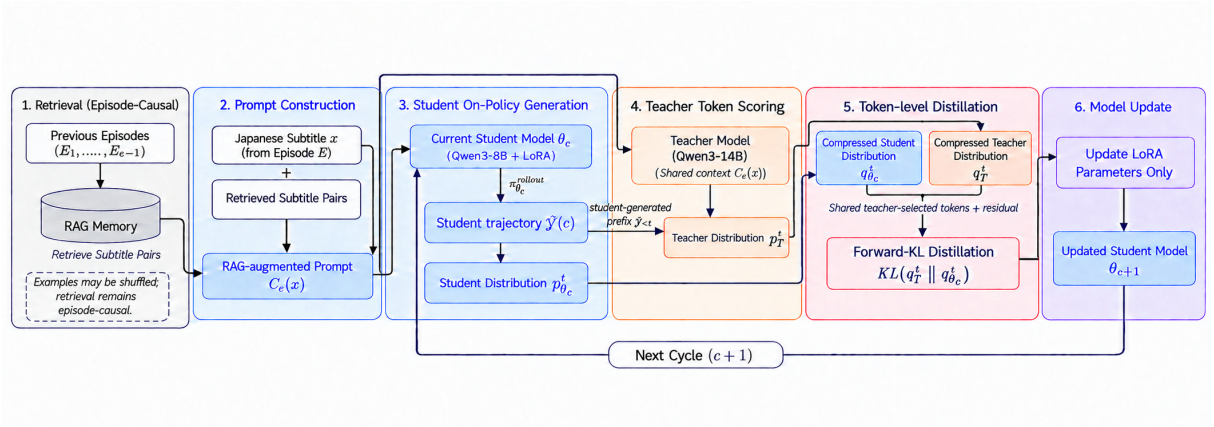



Figure 3: Overview of retrieval-conditioned on-policy distillation. Student and teacher share the same episode-causal RAG context; the teacher provides token-level supervision along student-generated trajectories, and the updated student generates fresh trajectories in the next optimization cycle.

remain frozen, and optimizer state is preserved across cycles.

After this update, the next cycle processes a new batch of training examples and generates fresh trajectories using the updated student $p_{\theta_{c+1}}$. No replay buffer or fixed teacher-generated trajectory set is used.

SerialDistill is therefore on-policy at the cycle level: each batch is generated by the current student, while retrieval remains episode-causal for every training instance.

## 4 Experimental Setup

### 4.1 Data and Splits

We use Japanese–Chinese subtitles from a 12-episode anime series. Lyric cues are excluded from the primary-series experiments. Dataset usage is summarized in Table 1.

OPD examples from EP2–9 are shuffled across episodes, with retrieval restricted to earlier finalized episodes for each example. At evaluation time, the memory grows with episode order: EP10, EP11, and EP12 use finalized subtitles from EP1–9, EP1–10, and EP1–11, respectively. Earlier evaluation episodes enter the bilingual memory only after they are treated as finalized, and no further parameter updates are performed. For each evaluated episode, reference translations from that episode and all future episodes are excluded from its retrieval context. EP11–12 are not used for adaptation, distillation, or configuration selection. These boundaries prevent target-episode reference leakage while permitting reuse of previously finalized translations in the sequential workflow.

| Use | Episodes | Pairs |
|---|---|---|
| SFT / initial memory | 1–9 | 2,687 |
| OPD | 2–9 | 2,448 |
| Development | 10 | 346 |
| Held-out test | 11 / 12 | 277 / 212 |

Table 1: Primary-series episode split and retained non-lyric pairs.

| System | Base model | LoRA | RAG | Distillation |
|---|---|---|---|---|
| Base-8B | Qwen3-8B | – | – | – |
| LoRA-8B | Qwen3-8B | ✓ | – | – |
| RAG-8B | Qwen3-8B | – | ✓ | – |
| LoRA-8B+RAG | Qwen3-8B | ✓ | ✓ | – |
| 14B+RAG Teacher | Qwen3-14B | – | ✓ | – |
| SerialDistill | Qwen3-8B | ✓ | ✓ | OPD |

Table 2: Summary of evaluated systems.

**Cross-series replication.** To examine whether the main finding transfers beyond the primary series, we independently replicate the SerialDistill pipeline on a second 12-episode anime series, *Gimai Seikatsu*, using publicly available Japanese–Chinese subtitles released by Haruhana Funsub. The 12-episode dataset follows the same split protocol as the primary experiment: Episodes 1–9 provide LoRA adaptation and retrieval memory, EP2–9 provide OPD training instances, EP10 is the development set, and EP11–12 are held-out tests. Held-out episodes are evaluated only after training and configuration are fixed.

### 4.2 Models and Systems

The evaluated systems are summarized in Table 2.

The Qwen3-14B teacher remains frozen, and SerialDistill is initialized from the domain-adapted

Qwen3-8B LoRA model. LoRA-8B+RAG uses the same initial adapter without distillation. Both series use this student–teacher configuration.

### 4.3 Training Configuration

The primary-series experiments are run on a single GPU. We first adapt the student to the target series with LoRA and then refine the same adapter through retrieval-conditioned on-policy distillation. During distillation, each optimization cycle uses fresh student-generated trajectories, while the student backbone and teacher remain frozen. Hardware, software versions, and training hyperparameters are reported in Appendix A.3.

### 4.4 Retrieval and Inference

Both series use sentence-level BM25 and the episode-causal memory protocol described above. The number of retrieved examples is specified in Appendix A.3. The shared translation instruction contains one fixed bilingual demonstration in addition to any retrieved examples. We compare dense semantic retrieval and structured word-level memory on the primary-series development set in Section 5.4.

Evaluation uses deterministic decoding with thinking disabled and a maximum of 128 new tokens; OPD training uses sampled trajectories.

### 4.5 Evaluation

We report corpus chrF (Popović, 2015), BLEU (Papineni et al., 2002) computed with SacreBLEU and Chinese tokenization (Post, 2018), and mean segment COMET using `Unbabel/wmt22-comet-da` (Rei et al., 2020). Major comparisons additionally use two-sided paired bootstrap resampling for significance testing (Koehn, 2004). For the primary-series teacher comparison and both second-series comparisons in Table 5, we use 2,000 paired resamples of subtitle segments with random seed 42.

## 5 Results

### 5.1 Main Results

Table 3 summarizes the main results on the EP10 development set and the held-out EP11–12 test sets. On EP10, both LoRA and retrieval improve over the base model, and their combination improves all three metrics over either component alone.

Retrieval-conditioned on-policy distillation further improves over LoRA-8B+RAG by 0.72 chrF and 1.20 BLEU on the pooled held-out set (Table 5). SerialDistill also surpasses the larger 14B+RAG teacher across chrF, BLEU, and COMET on both EP11 and EP12.

The same trend holds on the pooled 489-example test set. SerialDistill exceeds the teacher by 3.60 chrF, 6.80 BLEU, and 0.0067 COMET, with the pooled differences reaching statistical significance (Table 5). This consistency across episodes and metrics indicates that the improvement is not driven by a single test split.

### 5.2 Cross-Series Replication

To test whether the main SerialDistill result is specific to the primary series, we independently replicate the pipeline on *Gimai Seikatsu*. As shown in Table 4, SerialDistill achieves the strongest pooled performance and preserves the student-over-teacher pattern on both held-out episodes.

On the pooled EP11–12 set, SerialDistill improves over the corresponding non-distilled LoRA-8B+RAG baseline by 3.80 chrF and surpasses the 14B+RAG teacher by 2.46 chrF. All three metric improvements over LoRA-8B+RAG are significant at $p < .001$; against the teacher, chrF and BLEU are significant at $p < .001$ and COMET at $p = .001$. Table 5 summarizes the pooled gains over the non-distilled student and the teacher for both series.

### 5.3 Teacher Retrieval Ablation

We examine teacher-side retrieval on the primary series. The ablated variant removes retrieved examples only from the teacher input during distillation; student retrieval and the episode-causal evaluation protocol are retained in both variants.

As shown in Table 6, teacher retrieval improves chrF by 1.09 and BLEU by 1.12. The chrF gain is statistically significant ($p = .014$; two-sided paired bootstrap, 2,000 resamples, seed 42). These results support the value of providing historical bilingual evidence to the teacher during distillation.

### 5.4 Retrieval Strategy Comparison

To assess the choice of retrieval strategy, Table 7 compares sentence-level BM25, dense semantic retrieval, and word-level structured memory using Qwen3-8B without LoRA on the primary-series development set. The difference between BM25 and dense retrieval is not statistically significant, whereas the structured memory does not improve over the simpler sentence-level approach.

| System | Development: EP10 chrF | BLEU | COMET | Held-out Test: EP11 chrF | BLEU | COMET | Held-out Test: EP12 chrF | BLEU | COMET | Held-out Test: EP11+12 pooled chrF | BLEU | COMET |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Base-8B | 24.60 | 28.43 | .8286 | 23.80 | 28.43 | .8207 | 23.87 | 27.62 | .8153 | 23.83 | 28.07 | .8184 |
| LoRA-8B | 32.19 | 37.18 | .8569 | 36.93 | 42.38 | .8753 | 37.75 | 42.48 | .8708 | 37.29 | 42.43 | .8733 |
| RAG-8B | 33.71 | 38.26 | .8590 | 38.39 | 37.70 | .8735 | 34.60 | 39.32 | .8702 | 36.75 | 38.39 | .8721 |
| LoRA-8B+RAG | 35.59 | 40.58 | .8645 | 43.26 | 47.32 | .8867 | 38.13 | 43.48 | .8785 | 40.98 | 45.61 | .8832 |
| 14B+RAG Teacher | 34.34 | 37.66 | .8630 | 39.41 | 40.64 | .8806 | 36.46 | 39.24 | .8754 | 38.11 | 40.01 | .8784 |
| SerialDistill | **36.94** | **42.48** | **.8723** | **43.42** | **48.00** | **.8883** | **39.55** | **44.78** | **.8809** | **41.70** | **46.81** | **.8851** |

Table 3: Main results on EP10 development and EP11/12 held-out tests. The pooled column is computed on the concatenated 489-example held-out set.

| System | Development: EP10 chrF | BLEU | COMET | Held-out Test: EP11 chrF | BLEU | COMET | Held-out Test: EP12 chrF | BLEU | COMET | Held-out Test: EP11+12 pooled chrF | BLEU | COMET |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Base-8B | 16.98 | 17.99 | .7877 | 19.04 | 20.94 | .8017 | 18.26 | 19.86 | .7945 | 18.69 | 20.45 | .7983 |
| LoRA-8B | 22.03 | 25.39 | **.8162** | 19.87 | 22.93 | .8177 | 22.50 | 26.31 | .8179 | 21.03 | 24.48 | .8178 |
| RAG-8B | 21.61 | 23.16 | .8066 | 21.22 | 23.49 | .8086 | 25.11 | 28.05 | .8215 | 22.94 | 25.55 | .8146 |
| LoRA-8B+RAG | 23.76 | **27.33** | .8117 | 19.56 | 22.76 | .8056 | 24.35 | 28.29 | .8215 | 21.68 | 25.31 | .8130 |
| 14B+RAG Teacher | 22.16 | 23.02 | .8097 | 22.18 | 22.99 | .8115 | 24.07 | 25.54 | .8166 | 23.02 | 24.12 | .8139 |
| **SerialDistill** | **23.85** | 27.08 | .8136 | **22.70** | **25.83** | **.8183** | **28.99** | **33.64** | **.8306** | **25.48** | **29.46** | **.8240** |

Table 4: Cross-series replication on *Gimai Seikatsu*. EP10 is the development set, EP11 and EP12 are held-out test sets, and the pooled column is computed on the concatenated 672-example EP11–12 test set.

These results suggest that increasing retrieval complexity does not necessarily improve translation quality in this setting. Because BM25 achieves comparable downstream performance with a simpler retrieval mechanism, we use sentence-level BM25 in the final SerialDistill system.

### 5.5 Qualitative Analysis

To complement the quantitative comparisons, Figure 4 presents three illustrative translations from LoRA-8B+RAG and SerialDistill. In (a), SerialDistill recovers the omitted description of the project's scale. In (b), it preserves the source statement about insufficient awareness of a crisis rather than recasting it as the addressee's failure to notice the crisis. In (c), it translates the reference to a battlefield smell without adding the unsupported specification of blood.

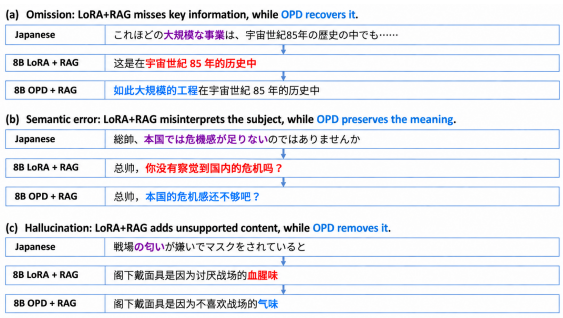



Figure 4: Qualitative examples comparing LoRA-8B+RAG and SerialDistill (labeled OPD in the figure). The examples illustrate omission, semantic error, and unsupported additions in the adapted student output, together with the corresponding SerialDistill corrections.

## 6 Discussion

### 6.1 Student–Teacher Performance Comparison

On the pooled held-out sets, SerialDistill's advantage over the 14B+RAG teacher is observed under two different starting conditions. On the primary series, LoRA-8B+RAG already outperforms the teacher before distillation and improves further afterward. On *Gimai Seikatsu*, the same baseline starts below the teacher in chrF and COMET, whereas the distilled student surpasses it on all three metrics. In both cases, the student combines prior adaptation to the target series with supervision from the larger teacher, which has not undergone domain-specific fine-tuning.

This pattern suggests that the teacher's own translation scores do not fully capture its usefulness as a source of supervision. In SerialDistill, the teacher provides next-token guidance along the adapted student's generated prefixes under the same retrieved bilingual context. Such guidance may complement the translation patterns acquired during supervised adaptation, helping the student improve beyond both its initial performance and the teacher's independently generated translations.

The examples in Figure 4 illustrate changes accompanying the aggregate gains: recovery of omitted information, preservation of intended meaning, and avoidance of unsupported details.

| Series | Comparison baseline | chrF Δ | chrF $p$ | BLEU Δ | BLEU $p$ | COMET Δ | COMET $p$ |
|---|---|---|---|---|---|---|---|
| Primary | LoRA-8B+RAG | +0.72 | $< .001$ | +1.20 | $< .001$ | +.0019 | $< .001$ |
| | 14B+RAG Teacher | +3.60 | $< .001$ | +6.80 | $< .001$ | +.0067 | .042 |
| *Gimai Seikatsu* | LoRA-8B+RAG | +3.80 | $< .001$ | +4.15 | $< .001$ | +.0110 | $< .001$ |
| | 14B+RAG Teacher | +2.46 | $< .001$ | +5.34 | $< .001$ | +.0102 | .001 |

Table 5: SerialDistill gains on the pooled EP11–12 test sets (489 and 672 examples for the primary and second series, respectively). Positive Δ indicates improvement over the comparison baseline. The $p$ values are from two-sided paired bootstrap tests. Deltas use unrounded scores where available; the primary LoRA-8B+RAG comparison uses the reported baseline scores.

| Teacher retrieval | chrF | BLEU | COMET |
|---|---|---|---|
| Without | 40.17 | 45.06 | .8811 |
| With | **41.26** | **46.18** | **.8819** |

Table 6: Teacher-retrieval ablation on the pooled primary-series EP11–12 test set. These training runs are independent of the main experiment in Table 3.

| Retrieval | chrF | COMET |
|---|---|---|
| Sentence BM25 | 33.71 | .8590 |
| Dense top-2 | **33.81** | **.8602** |
| Structured word-level | 32.47 | .8575 |

Table 7: Retrieval comparison for Qwen3-8B without LoRA on the primary-series EP10 development set. Bold marks the highest score in each column.

### 6.2 Implications for Serial Translation

Combining adaptation and retrieval improves over either component alone on the primary-series pooled test set, while its benefits vary across metrics on *Gimai Seikatsu*. Distillation adds gains in both settings, yielding the highest pooled scores on all three metrics in each series.

Previously finalized subtitles support adaptation and retrieval, while teacher supervision enables further training without additional parallel examples. The gains observed with a few thousand training pairs in each series support this approach to data-scarce domain adaptation.

At inference time, the 8B student translates with retrieval alone. The final system retains the model size and inference structure of LoRA-8B+RAG, fitting a workflow that reuses completed subtitles for subsequent episodes.

## 7 Conclusion

SerialDistill combines LoRA adaptation with retrieval-conditioned on-policy distillation in a serial subtitle translation workflow. On both evaluated Japanese-to-Chinese anime series, it improves over LoRA-8B+RAG and surpasses the unadapted 14B+RAG teacher on pooled held-out chrF, BLEU, and COMET. Previously finalized subtitles supply adaptation data and episode-causal retrieval memory, while teacher guidance adds supervision without additional parallel examples. The resulting 8B student translates with retrieval alone at inference time. These findings support distillation as an additional adaptation stage under the evaluated bilingual-history setting.

## Limitations

Our experiments evaluate Japanese-to-Chinese subtitle translation in two anime series, using finalized translations of earlier episodes as retrieval memory. Evaluation across additional language pairs, genres, and translation-memory conditions would help establish the broader applicability of SerialDistill.

Due to time constraints, we have not conducted cold-start experiments in which no previously translated episodes are available for training or retrieval. Our results therefore apply to settings with an existing work-specific bilingual history. Evaluating performance from an initially empty memory as translations accumulate is an important direction for future work.

The evaluation combines automatic MT metrics with qualitative analysis of individual translations; a systematic human evaluation remains to be conducted. Such an evaluation would assess terminology consistency and context-sensitive meaning across episodes, alongside translation adequacy and fluency, and quantify the occurrence of the improvements illustrated by the qualitative examples.

## References

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier

Bachem. 2024. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*.

Sweta Agrawal, Chunting Zhou, Mike Lewis, Luke Zettlemoyer, and Marjan Ghazvininejad. 2023. In-context examples selection for machine translation. In *Findings of ACL*, pages 8857–8873.

Ankur Bapna and Orhan Firat. 2019a. Non-parametric adaptation for neural machine translation. In *Proceedings of NAACL-HLT*, pages 1921–1931.

Ankur Bapna and Orhan Firat. 2019b. Simple, scalable adaptation for neural machine translation. In *Proceedings of EMNLP-IJCNLP*, pages 1538–1548.

Maxime Bouthors, Josep Crego, and François Yvon. 2024. Retrieving examples from memory for retrieval augmented neural machine translation: A systematic comparison. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3022–3039.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. 2024. MiniLLM: Knowledge distillation of large language models. In *International Conference on Learning Representations*.

Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. How good are GPT models at machine translation? a comprehensive evaluation. ArXiv:2302.09210.

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

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of ICML*, pages 2790–2799.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Wenxiang Jiao, Wenxuan Wang, Jen tse Huang, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2023. Is ChatGPT a good translator? yes with GPT-4 as the engine. ArXiv:2301.08745.

Linghao Jin, Jacqueline He, Jonathan May, and Xuezhe Ma. 2023. Challenges in context-aware neural machine translation. In *Proceedings of EMNLP*, pages 15246–15263.

Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2021. Nearest neighbor machine translation. In *International Conference on Learning Representations*.

Yoon Kim and Alexander M. Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of EMNLP*, pages 1317–1327.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. 2024. DistiLLM: Towards streamlined distillation for large language models. In *Proceedings of ICML*, pages 24872–24895.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP*, pages 388–395.

Yichen Lu, Wei Dai, Jiaen Liu, Ching Wing Kwok, Zongheng Wu, Xudong Xiao, Ao Sun, Sheng Fu, Jianyuan Zhan, Yian Wang, Takatomo Saito, and Sicheng Lai. 2025. ViDove: A translation agent system with multimodal context and memory-augmented reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 228–243.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318.

Ziqian Peng, Rachel Bawden, and François Yvon. 2025. Self-retrieval from distant contexts for document-level machine translation. In *Proceedings of the Tenth Conference on Machine Translation*, pages 220–240.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation*, pages 186–191.

Ashmari Pramodya, Yusuke Sakai, Justin Vasselli, Hidetaka Kamigaito, and Taro Watanabe. 2025. Translating movie subtitles by large language models using movie-meta information. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 315–330.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of EMNLP*, pages 2685–2702.

Danielle Saunders. 2022. Domain adaptation and multi-domain adaptation for neural machine translation: A survey. ArXiv:2104.06951, version 2.

David Vilar, Markus Freitag, Colin Cherry, Jiaming Luo, Viresh Ratnakar, and George Foster. 2023. Prompting PaLM for translation: Assessing strategies and performance. In *Proceedings of ACL*, pages 15406–15427.

Elena Voita, Rico Sennrich, and Ivan Titov. 2019. When a good translation is wrong in context: Context-aware machine translation improves on deixis, ellipsis, and lexical cohesion. In *Proceedings of ACL*, pages 1198–1212.

Jiaan Wang, Fandong Meng, Yingxue Zhang, and Jie Zhou. 2025. Retrieval-augmented machine translation with unstructured knowledge. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 5858–5871.

Longyue Wang, Chenyang Lyu, Tianbo Ji, Zhirui Zhang, Dian Yu, Shuming Shi, and Zhaopeng Tu. 2023. Document-level machine translation with large language models. In *Proceedings of EMNLP*, pages 16646–16661.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, et al. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Jingyi Zhang, Masao Utiyama, Eiichro Sumita, Graham Neubig, and Satoshi Nakamura. 2018. Guiding neural machine translation with retrieved translation pieces. In *Proceedings of NAACL-HLT*, pages 1325–1335.

# A Appendix

## A.1 Detailed Dataset Statistics

| Episode | Primary series | *Gimai Seikatsu* | Experimental role |
|---|---|---|---|
| 1 | 239 | 282 | SFT and initial retrieval memory |
| 2 | 288 | 423 | SFT, OPD, and retrieval memory |
| 3 | 335 | 470 | SFT, OPD, and retrieval memory |
| 4 | 305 | 335 | SFT, OPD, and retrieval memory |
| 5 | 295 | 313 | SFT, OPD, and retrieval memory |
| 6 | 309 | 300 | SFT, OPD, and retrieval memory |
| 7 | 252 | 453 | SFT, OPD, and retrieval memory |
| 8 | 348 | 354 | SFT, OPD, and retrieval memory |
| 9 | 316 | 349 | SFT, OPD, and retrieval memory |
| 10 | 346 | 344 | Development |
| 11 | 277 | 360 | Held-out test |
| 12 | 212 | 312 | Held-out test |

Table 8: Episode-level dataset statistics. Counts are computed from the dataset files and corresponding audit records.

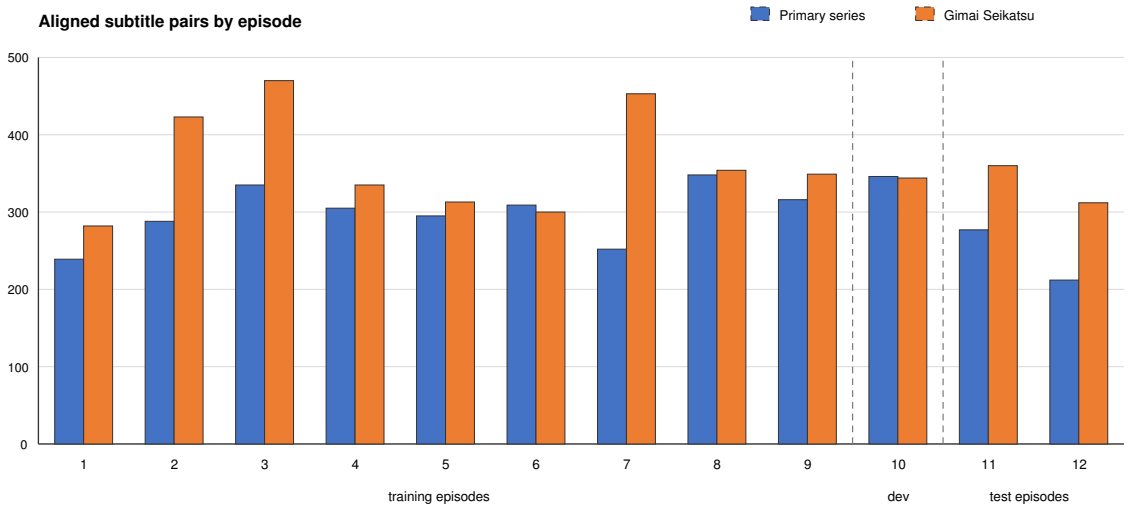



Figure 5: Episode-level distribution of usable aligned subtitle pairs in the two series. Dashed boundaries separate the training, development, and held-out test portions.

This appendix reports the episode-level statistics underlying the two-series experiments. Table 8 gives the exact counts of usable aligned subtitle pairs, and Figure 5 shows how these counts vary across episodes in the two series. For the primary series, the counts exclude lyric cues; for *Gimai Seikatsu*, they correspond to the valid aligned pairs recorded by the dataset audit.

## A.2 On-Policy Distillation Diagnostics

Table 9 summarizes the primary-series OPD trace in consecutive cycle ranges, while Figure 6 shows the cycle-level fluctuations in KL and gradient norm. Each row is computed from the saved per-cycle log. KL is token-weighted within each cycle; the table reports arithmetic means of the cycle-level KL values and pre-clipping gradient norms. Coverage is the teacher probability mass retained by its top-64 token set, shared with the student; mean coverage averages the cycle means, and minimum coverage is the lowest token-level value in the range. These diagnostics describe optimization behavior and were not used to select a checkpoint on the held-out test episodes.

Mean KL decreases from 1.4092 in Cycles 1–10 to 0.8537 in Cycles 71–77. At the same time, the retained top-64 set accounts for more than 99.99% of the teacher probability mass on average in every range.

| Cycles | Examples | Mean KL | Mean grad. norm | Mean coverage (%) | Min. coverage (%) | Response tokens |
|---|---|---|---|---|---|---|
| 1–10 | 320 | 1.4092 | 2.3147 | 99.9946 | 96.9318 | 2,274 |
| 11–20 | 320 | 1.2840 | 2.1207 | 99.9961 | 97.2676 | 2,181 |
| 21–30 | 320 | 1.1872 | 2.1202 | 99.9954 | 98.2997 | 2,182 |
| 31–40 | 320 | 1.2077 | 2.2150 | 99.9949 | 94.4799 | 2,412 |
| 41–50 | 320 | 1.0819 | 1.9129 | 99.9961 | 98.2002 | 2,131 |
| 51–60 | 320 | 1.0099 | 1.8003 | 99.9967 | 98.8558 | 2,184 |
| 61–70 | 320 | 0.9189 | 1.7747 | 99.9950 | 95.2910 | 2,128 |
| 71–77 | 208 | 0.8537 | 1.6327 | 99.9970 | 98.6739 | 1,363 |

Table 9: Primary-series OPD diagnostics aggregated from the 77 saved optimization cycles. The final range contains 208 examples because the last cycle contains the remaining 16 of the 2,448 training instances.

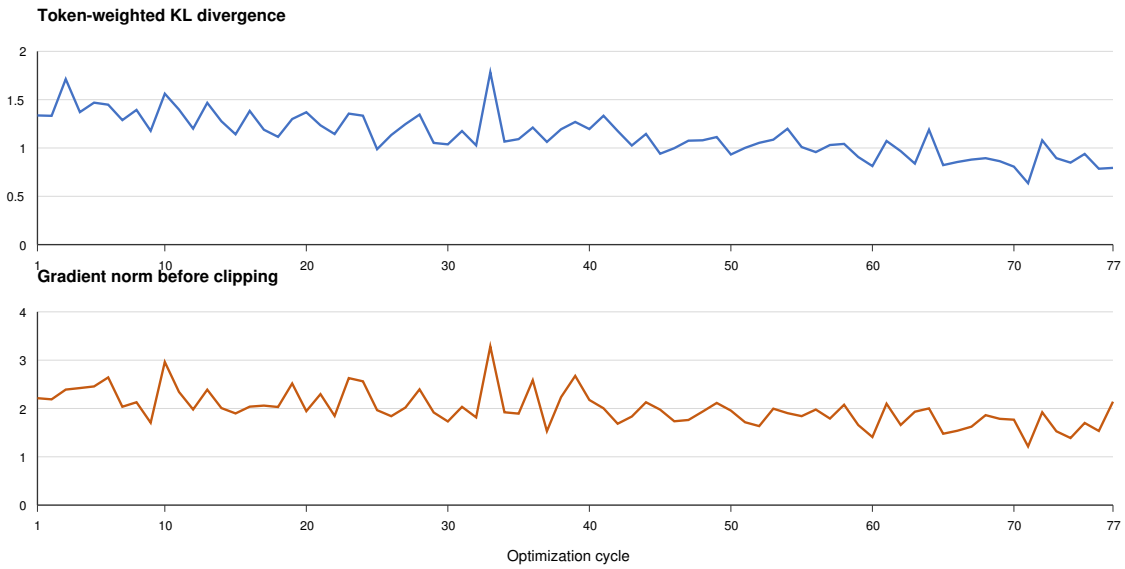



Figure 6: Cycle-level primary-series OPD optimization traces. The upper panel shows the token-weighted KL objective; the lower panel shows the gradient norm before clipping.

### A.3 Training and Retrieval Configuration

Table 10 lists the hardware, software, and training settings for the primary series, together with the retrieval setting shared by both series. Dataset sizes and splits are given in Tables 1 and 8.

| Setting | Value |
|---|---|
| *Primary series: hardware and software* | |
| GPU | One NVIDIA GeForce RTX 5090 (32 GB) |
| Numerical precision | bf16 |
| Gradient checkpointing | Enabled |
| PyTorch | 2.11.0+cu128 |
| Transformers | 5.16.1 |
| PEFT | 0.20.0 |
| *Primary series: supervised adaptation* | |
| LoRA configuration | Rank 16; $\alpha = 32$; dropout 0.05 |
| LoRA target modules | `q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj`, `down_proj` |
| Training epochs | 1 |
| Optimizer | AdamW |
| Learning rate | $2 \times 10^{-4}$ |
| Batch size | 1 |
| Gradient accumulation | 8 steps |
| Maximum sequence length | 512 tokens |
| Learning-rate schedule | Cosine |
| Warm-up steps | 10 |
| Gradient clipping | 1.0 |
| *Primary series: on-policy distillation* | |
| Optimizer | AdamW |
| Learning rate | $5 \times 10^{-6}$ |
| Optimization cycles | 77 |
| Examples per cycle | Up to 32 |
| Updates per cycle | 1 |
| Rollout micro-batch size | 4 |
| Teacher scoring / student update micro-batch size | 1 / 1 |
| Teacher distribution representation | Top $K_T = 64$ tokens plus residual mass |
| Sampling temperature | 0.7 |
| Sampling top-$p$ | 0.9 |
| Maximum new tokens | 128 |
| *Both series: retrieval* | |
| Maximum retrieved bilingual examples | $K = 2$ |

Table 10: Primary-series implementation settings and the retrieval size shared by both series.