Communicative On-Policy Distillation: Distilling Black-Box Teacher Capabilities through Communication for Long-Horizon Tasks

Lirui Luo[1], **Kelong Mao**[2], **Xinwei Yang**[3], **Rongqing Li**[4], **Luyu Chen**[5], **Kieran Wong**[2], **Yudong Guo**[2], **Xinrui Wang**[2], **Jiayin Zhu**[6], **Gu Simiu**[2], **Sulong Xu**[2], **Cong Fang**[1]

[1]Peking University, [2]JD.com, [3]Sichuan University, [4]Beijing Institute of Technology, [5]Renmin University of China, [6]The Hong Kong University of Science and Technology

Abstract

On-policy distillation (OPD) is a powerful paradigm for improving language-model agents on long-horizon tasks, yet its conventional formulation relies on teacher logits that text-only black-box models do not expose. To address this limitation, we introduce Communicative On-Policy Distillation (COPD), which uses natural-language communication as a medium for distilling a stronger model's capabilities into a smaller student. During task execution, the student autonomously decides when to consult a black-box teacher and formulates open-ended natural-language requests without predefined question templates or intervention schedules. Successful student–teacher interactions are retained as privileged references to supervise fresh adviser-free student rollouts through on-policy self-distillation, jointly with task-reward reinforcement learning. On CAMG-II, COPD achieves 2.46 times the gain of the strongest black-box OPD baseline. The capabilities it learns through communication generalize to Terminal-Bench 2.1 and SWE-Bench Pro V2, improving task success without adviser access.

Project website: https://liruiluo.github.io/communicative-opsd/

1 Introduction

On-policy distillation (OPD) is a powerful paradigm for improving language-model agents on long-horizon tasks [1, 19]. Yet conventional OPD relies on teacher logits, which text-only black-box models do not expose. Human learning never needed logits: we consult those more experienced and internalize their guidance to solve problems independently. Figure 1 shows a concrete exchange: a student building a website asks a stronger adviser how to organize its page layout. Can a student agent learn to autonomously communicate with a black-box teacher during task execution, and internalize the acquired knowledge so that the teacher is no longer needed?

Acquiring useful guidance is itself nontrivial: in long-horizon tasks, neither when the student will benefit from external expertise nor what information it should communicate can be fully specified in advance. The student must therefore learn to identify when to seek guidance and formulate task-specific messages as execution unfolds, rather than relying on predefined intervention points or question templates.

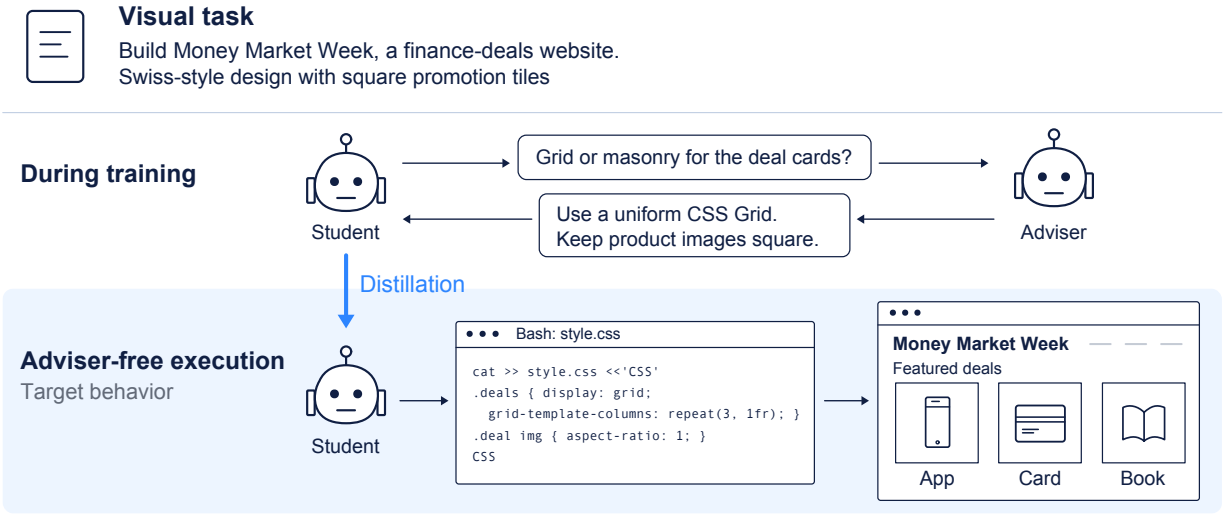


Figure 1 Natural-language communication as a medium for distillation. In a Visual task, the student asks a stronger black-box adviser whether to use a grid or masonry layout for a website's product cards. The adviser recommends a uniform CSS Grid with square product images. Through COPD, the student learns from successful consulted experience to act independently. The lower row illustrates the target adviser-free behavior.

However, completing a task with advice does not ensure independent execution once the adviser is removed. To bridge this gap, we introduce **Communicative On-Policy Distillation (COPD)**, which uses natural-language communication as a medium for distilling a stronger model's capabilities into a smaller student. During task execution, the student autonomously decides when to consult a black-box teacher and formulates open-ended natural-language requests without predefined question templates or intervention schedules. Successful student–teacher interactions are retained as privileged references to supervise fresh adviser-free student rollouts through on-policy self-distillation, jointly with task-reward reinforcement learning.

We train COPD-4B and COPD-9B from Qwen3.5 models of matching size on CAMG-II [8]. We evaluate independent execution without adviser access. On CAMG-II, COPD achieves 2.46 times the gain of the strongest black-box OPD baseline. The capabilities it learns through communication generalize to Terminal-Bench 2.1 and SWE-Bench Pro V2, with improved success rates.

Our contributions are as follows:

1. **An algorithm for black-box distillation through natural language.** We introduce COPD, which trains a student to communicate with a stronger black-box adviser and internalize successful consulted experience through on-policy self-distillation for independent task execution (§4).
2. **An analysis of what communication contributes.** We show that self-distillation internalizes knowledge conveyed through adviser replies, improving independent execution beyond communication-only training (§5.2).
3. **An extensive empirical evaluation across five domains.** We demonstrate that COPD improves independent task execution across diverse long-horizon tasks and generalizes beyond the training environments (§5.1, §5.2).

2 Related Work

Learning to seek assistance. These methods train an agent to use an adviser that is present while it acts. It learns what to ask through predicate questions [10] or free-form clarification [12], and whether to request as-

sistance at all [14]. Guidance can also follow failure, with discounted rewards for assisted successes [6]. These methods study help-seeking within specific assistance settings, including oracle questions, task clarification, and failure-triggered guidance. In contrast, COPD allows the student to initiate open-ended natural-language communication at self-selected points during task execution and distills the resulting successful interactions into an adviser-free policy.

Black-box on-policy distillation. Black-box on-policy distillation keeps OPD's student-generated prefixes [1] and queries the teacher only for text. Supervision must therefore come from its generations: chunk-level verification against sampled teacher continuations [19], action-level rewards for semantic agreement with teacher decisions [17], rubric scores induced from teacher–student contrasts [3], an adversarial discriminator converted into an on-policy reward [16], and the teacher's completion of a student-generated prefix as the training target [15]. These methods derive supervision from black-box teacher outputs without training the student to initiate communication during task execution. COPD instead enables the student to autonomously seek guidance through free-form dialogue and uses successful consulted experience to supervise its own fresh adviser-free trajectories.

On-policy self-distillation. On-policy self-distillation (OPSD) supervises student-generated prefixes with a self-teacher conditioned on training-only information. The information may be reference solutions [18], retrospective execution feedback or successful rollouts [5], or complete student trajectories used to adapt the self-teacher [4]. These methods construct privileged supervision from reference solutions or the student's own experience. COPD instead conditions its self-teacher on task-verified successful trajectories collected through student-initiated, free-form communication with a black-box adviser. This allows guidance acquired through dialogue to supervise fresh adviser-free student responses without requiring teacher logits.

3 Preliminaries

Multi-turn agentic reinforcement learning. For task x in environment d, policy π_θ samples a response $a_t = (y_{t,1}, \ldots, y_{t,L_t})$ of L_t tokens from bounded visible history h_t. Executing the response returns an observation and native reward r_t. An episode τ contains T responses and has return $G(\tau) = \sum_{t=1}^{T} \gamma^{t-1} r_t$, where $\gamma \in [0, 1]$ is the discount factor. The environment also determines success $s(\tau) \in \{0, 1\}$. With task distribution μ and trajectory distribution p_θ induced by the policy, task-reward RL maximizes

$$J(\theta) = \mathbb{E}_{(d,x)\sim\mu,\, \tau\sim p_\theta(\cdot|d,x)}[G(\tau)]. \tag{1}$$

On-policy distillation. OPD provides token-level teacher supervision on student-generated responses [1]. At context $c = (h_t, y_{t,<j})$, the student samples $y \sim \pi_\theta(\cdot \mid c)$, and a teacher distribution $q(\cdot \mid c)$ scores the same token. For fixed c and q, minimizing the reverse KL $D_{\mathrm{KL}}(\pi_\theta \| q)$ gives a teacher–student log-ratio signal. The corresponding sampled-token policy-gradient surrogate is

$$\begin{aligned} A_\theta^{\mathrm{OPD}}(c, y) &= \mathrm{sg}[\log q(y \mid c) - \log \pi_\theta(y \mid c)], \\ \ell_{\mathrm{PG}}(\theta) &= -A_\theta^{\mathrm{OPD}}(c, y)\, \log \pi_\theta(y \mid c). \end{aligned} \tag{2}$$

During differentiation, the sampled (c, y) and teacher distribution q are held fixed, and sg stops gradients through the feedback. On-policy self-distillation (OPSD) forms a teacher view of the student conditioned on additional training-time information, such as a reference solution or feedback [5, 18].

4 Method

In this section, we present COPD for distilling a stronger black-box model's capabilities into a smaller student through language (Figure 2). Inspired by human learning through communication, we first collect successful consulted experience (§4.1), then use it to supervise fresh adviser-free responses (§4.2), and finally combine distillation with task reward to train independent execution (§4.3). These components are jointly optimized throughout training rather than implemented as separate training stages.

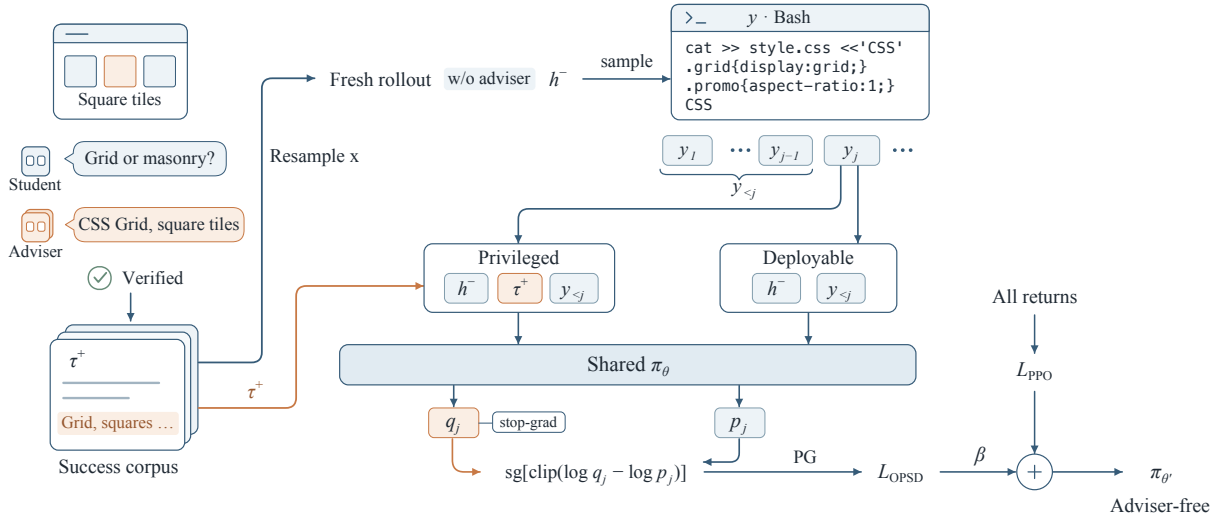


Figure 2 Overview of COPD. The student acquires guidance through communication with a stronger black-box adviser. Successful consulted trajectories enter a corpus, from which tasks are resampled for fresh adviser-free rollouts. Two views of the current student score the same sampled tokens, with only the privileged view receiving a successful trajectory τ^+ from the same task. Their log-probability difference provides distillation feedback, which is combined with task-reward PPO to train independent execution.

4.1 Collecting Experience through Communication

To access the stronger adviser's task-solving knowledge, we first let the student communicate with it during task execution. The collection policy π_b is a snapshot of the student π_θ and generates episodes with adviser access (w. adviser) and without adviser access (w/o adviser). Since useful guidance depends on the task state, communication is itself a policy action. At every step at which interaction is permitted, the student either acts in the environment or sends free text to the adviser,

$$a_t \in \mathcal{A}_{\text{env}} \cup \mathcal{A}_{\text{comm}}, \qquad a_t^{\text{comm}} = \text{Ask}(u_t), \quad u_t \sim \pi_\theta(\cdot \mid h_t), \tag{3}$$

where u_t is text the student generates from its own history, not a template question, a predefined query type, or a choice among fixed options. The student submits it through an ordinary `bash` command and receives the reply as a text observation that guides subsequent actions. The adviser answers in text and never acts in the environment, and the student may communicate again at any later step. Native task rewards train the request tokens together with the policy responses that follow them, so the communication policy and the task policy are one model with one parameter set π_θ, learned jointly from environment returns rather than through a separately trained query selector.

To turn these interactions into distillation data, we retain an execution only if it passes the native success check $s(\tau) = 1$ and actually used the adviser, denoted by consulted(τ). Each reference τ^+ contains the complete trajectory of requests, replies, actions, and observations, preserving how guidance was used during successful execution. We store these references in a rolling corpus $\mathcal{C}$ indexed by environment and task (d, x). They provide the context for supervising fresh adviser-free attempts.

4.2 Constructing Privileged Supervision

To internalize the guidance for independent execution, we apply on-policy self-distillation (OPSD) to fresh adviser-free responses. In standard OPSD the privileged information may be a reference answer, execution feedback, or other additional context. Here it is a complete, environment-verified trajectory that the student itself produced while communicating during an earlier attempt. It requires neither logits from the black-box adviser nor target actions for the student, and the adviser is never asked to judge whether a particular token

Algorithm 1 Communicative On-Policy Distillation

Input: initial student weights θ_0, training tasks $\mathcal{T}$, and fixed black-box adviser $\mathcal{A}$

$\theta \leftarrow \theta_0$, rolling reference corpus $\mathcal{C} \leftarrow \emptyset$
while training continues **do**
  $\mathcal{X} \leftarrow \textsc{SampleTasks}(\mathcal{T}, \mathcal{C})$ ▷ task batch
  $\mathcal{D} \leftarrow \textsc{Rollout}(\pi_b, \mathcal{X}, \mathcal{A})$ ▷ completed episodes
  $\mathcal{S} \leftarrow \{\tau \in \mathcal{D} : s(\tau) = 1 \wedge \text{consulted}(\tau)\}$ ▷ successes
  $\mathcal{C} \leftarrow \textsc{RetainRecent}(\mathcal{C} \cup \mathcal{S})$ ▷ corpus window
  for each w/o adviser trajectory $\tau_i^- \in \mathcal{D}$ **do**
    $\tau_i^+ \leftarrow \textsc{Retrieve}(\tau_i^-, \mathcal{S}, \mathcal{C})$ ▷ same-task reference
  end for
  for optimizer minibatch $\mathcal{B}$ from $\mathcal{D}$ **do**
    Compute native-task $\mathcal{L}_{\text{PPO}}$ on all eligible rows
    $\mathcal{L}_{\text{OPSD}} \leftarrow 0$
    for each valid paired w/o adviser token y in $\mathcal{B}$ **do**
      Score y with student p, privileged q at θ (Eq. 5)
      Compute feedback A^{OPD} (Eq. 6)
      Compute token loss ℓ^{OPD} (Eq. 7)
    end for
    Aggregate $\mathcal{L}_{\text{OPSD}}$ with the fixed token denominator
    Update θ with $\mathcal{L}_{\text{PPO}} + \beta\mathcal{L}_{\text{OPSD}}$ (Eq. 8)
    Update the critic from native returns
  end for
end while

Output: adviser-free student π_θ

of a fresh rollout is correct. It conversed with the student during a real task, and the student's own privileged view turns that conversation into a token-level learning signal. We therefore condition the current student on a successful consulted execution to form a privileged self-teacher [5, 18]. The recorded choices and outcomes provide task information for assessing unaided responses.

To construct these training pairs, we mix scheduled tasks with tasks resampled from $\mathcal{C}$ and execute each selected task anew in a fresh workspace. Each resulting w/o adviser attempt can be paired with an available consulted success of the same task. Corpus sampling thus creates pairing opportunities even when tasks occur only once in the source schedule.

Since a fresh attempt may reach different states from the reference, we score its responses along its own unassisted history. For example, the successful CSS Grid implementation in Figure 2 provides context for scoring a fresh response that sets the layout and aspect ratio. For episode i, response t, and token j, let $c_{i,t,j} = (h_{i,t}^-, y_{i,t,<j})$ be the unassisted history and sampled response prefix. Given the paired success τ_i^+, the two views are

$$p_{i,t,j} = \pi_\theta(\cdot \mid c_{i,t,j}), \tag{4}$$

$$q_{i,t,j} = \text{sg}\left[\pi_\theta(\cdot \mid c_{i,t,j}, \tau_i^+)\right], \tag{5}$$

where p is the deployable view and q is the detached privileged view. Both share current student weights and the same unassisted prefix, while only q receives τ_i^+. The privileged view scores the student's sampled response without generating another action [1]. The external adviser participates only during collection.

4.3 Joint Reinforcement Learning and Distillation

Finally, we train the student with native task rewards and feedback from the privileged view. For each sampled token $y = y_{i,t,j}$, the distillation feedback is the clipped log-probability gap

$$A_{i,t,j}^{\text{OPD}} = \text{sg}[\text{clip}(\log q_{i,t,j}(y) - \log p_{i,t,j}(y), -C, C)], \tag{6}$$

where $C > 0$ bounds the feedback. A positive value encourages a token that the privileged view finds more likely, while a negative value discourages it.

Because asynchronous collection can use older student weights, we retain the actual behavior probability $b_{i,t,j}(y)$ recorded at generation and form $\rho = p_{i,t,j}(y)/b_{i,t,j}(y)$. This ratio reweights sampled tokens, while the two current-weight views supply the feedback. The clipped surrogate corresponding to Eq. 2, with token indices omitted, is

$$\ell^{\mathrm{OPD}} = -\min\big(\rho A^{\mathrm{OPD}},\ \mathrm{clip}(\rho, 1-\epsilon, 1+\epsilon)A^{\mathrm{OPD}}\big)\,. \tag{7}$$

Here ϵ is the ratio-clip width.

Aggregating valid paired w/o adviser token losses into $\mathcal{L}_{\mathrm{OPSD}}$, we jointly optimize

$$\mathcal{L}(\theta) = \mathcal{L}_{\mathrm{PPO}}(\theta) + \beta\,\mathcal{L}_{\mathrm{OPSD}}(\theta), \tag{8}$$

where $\beta \geq 0$ is the distillation weight. Task PPO [13] trains execution and communication from native returns on both w. adviser and w/o adviser episodes, including failures. OPSD uses paired w/o adviser responses to internalize consulted experience for independent execution.

Algorithm 1 summarizes the update, with asynchronous collection.

5 Experiments

We evaluate whether COPD improves the student's ability to execute tasks without an adviser. We first compare task performance with task-reward controls and black-box distillation methods. We then examine what communication contributes, using component ablations, matched reference sources, and reply-content interventions. Communication dynamics, test-time adviser access, and the training communication budget connect the acquisition of guidance to independent execution. External benchmarks assess transfer beyond the training environments.

5.1 Experimental Setup

Task environments. We use Coding Agent Memory Gym II (CAMG-II) [8], which spans Shop, Coding, AutoResearch, General, and Visual tasks.

Training. We train COPD from fresh `Qwen3.5-4B` and `Qwen3.5-9B` students [11] for 100 training updates with 64 episodes per update, totaling 6,400 episodes, under an equal mixture over the five domains. Training is fully asynchronous, combining native task-reward PPO with OPSD and using a fixed `DeepSeek-V4.1-Flash` adviser.

Baselines. As the task-reward reference, we use CAMP-I [7], which learns with the same file memory and no adviser access. We also compare five black-box OPD methods: learned discriminator rewards in GAD [16], rubric rewards in ROPD [3], supervision on teacher continuations of student prefixes in OLIVE [15], rewards for semantic agreement with teacher actions in ActionOPD [17], and token weighting through speculative verification in OmniOPD [19]. For GAD, we include variants without and with teacher-response warm-up. The main comparison uses the same 4B initialization and evaluation protocol.

Ablations. We compare COPD with w/o OPSD, w/o communication, and w/ SFT. The w/o OPSD variant removes the distillation update while retaining communication and task-reward PPO. The w/o communication variant disables the adviser during training and uses the student's verified unassisted successes as references for OPSD on fresh attempts at the same tasks. The w/ SFT variant replaces OPSD with supervised fine-tuning (SFT) on the student's responses in successful consulted trajectories, retaining communication and task-reward PPO.

Evaluation. We report per-domain scores and their equal-weight average, with the native metric of each domain. All methods use the same CAMG-II test set of 128 tasks per domain, giving 640 tasks in total, with adviser access disabled in the main comparison. We measure external transfer on the network-free task subsets of Terminal-Bench 2.1 and SWE-Bench Pro V2 [2, 9].

	Task success (%)					
Method	Shop	Coding	AutoResearch	General	Visual	Avg.↑
Frozen control						
Qwen3.5-4B [11]	77.3	27.3	46.1	39.8	36.0	45.3
Qwen3.5-9B [11]	90.6	26.6	56.2	60.9	36.5	54.2
CAMP-I [7]	85.2	32.0	38.3	**68.8**	46.5	54.2
GAD [16]	0.0	18.8	7.0	35.2	16.6	15.5
GAD (w/ warm-up) [16]	85.2	22.7	39.8	57.8	39.0	48.9
ROPD [3]	73.4	27.3	43.0	63.3	40.8	49.6
OLIVE [15]	79.7	24.2	39.8	53.1	39.0	47.2
ActionOPD [17]	**93.0**	24.2	41.4	58.6	39.0	51.2
OmniOPD [19]	80.5	25.8	43.0	55.5	40.8	49.1
COPD (ours)	92.2	**36.7**	**55.5**	66.4	**48.2**	**59.8**
COPD-9B	**93.0**	**38.3**	**57.0**	**68.8**	**49.4**	**61.3**
Teacher	100.0	78.1	83.6	91.4	50.4	80.7

Table 1 COPD attains the highest average success rate among the trained policies. Among the 4B systems it outperforms CAMP-I, every black-box OPD baseline, and the frozen student. The largest gain lands in General, 26.6 points over the frozen student, and Shop and General are the two cells where a baseline stays ahead. Scaling the same recipe to a 9B student lifts the average to 61.3. Visual reports the mean judged grade.

5.2 Results

Table 1 reports adviser-free task performance. The following analyses take that gain apart: which components produce it, what the student keeps when the adviser is gone, and how its communication changes during training.

COPD improves independent execution across domains. To test whether communicating with a stronger adviser improves what the student can do on its own, we evaluate every policy on the same CAMG-II panels with adviser access disabled (Table 1). COPD-4B raises the mean adviser-free task score from 45.3 to 59.8, against 54.2 for CAMP-I. The 14.5-point gain is 2.46 times what ActionOPD, the strongest of the five black-box OPD baselines, obtains at 51.2. Applied unchanged to the 9B student, the recipe reaches 61.3, seven points above its 54.2 starting point.

Two properties of the update distinguish COPD from the other black-box distillation baselines and may account for the lead: the trajectory is the student's own, and the signal arrives at every token. The teacher contributes language and never an action, so every supervised token is sampled from the current policy and the update stays on-policy, whereas OLIVE and OmniOPD supervise segments the teacher wrote and those segments lie off the student's policy. Two views of the same tokens give a signal at every position, whereas GAD and ROPD reduce an episode to one discriminator or rubric score and ActionOPD pays out only at the turns it groups.

COPD internalizes the adviser's guidance. To test whether what the student gains is internalized guidance rather than a dependence, we score each policy on the same tasks twice, with the adviser reachable and with it removed (Figure 3). Removing it costs the initial student a fifth of its score, w/o OPSD nearly a quarter, and COPD-4B a fortieth. The ordering flips with access: w/o OPSD passes COPD-4B at 62.4 against 61.4 while the adviser is there, and falls 12.3 points behind it without.

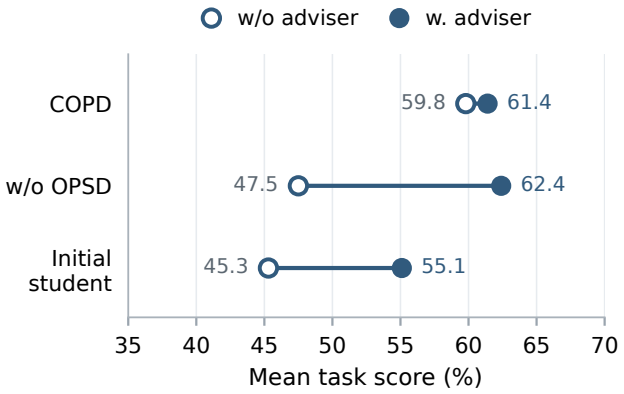


Figure 3 COPD internalizes most of the help it receives. Each line spans one policy's two conditions, adviser removed and adviser available, on the same tasks. Access is worth 9.8 points to the untrained student and 14.9 to w/o OPSD, and 1.6 to COPD-4B, which is the only policy whose level barely moves with the adviser.

A consulted success teaches more than an unassisted one. The capability survives without the adviser, so what put it there is the material the student distilled. To separate a reference's content from the number of successes it supplies, we compare consulted and unassisted references on tasks where both succeed, holding the student, the fresh attempt, and the update fixed. The consulted success moves the score 0.6 points on a test set, three times the 0.2 of the student's own unassisted success (Figure 4). A consulted success carried over from another task moves it the other way, by -0.1, and no reference leaves the score at 0.0. The guidance therefore carries the update, and it has to fit the task: a success from another task is worth less than none.

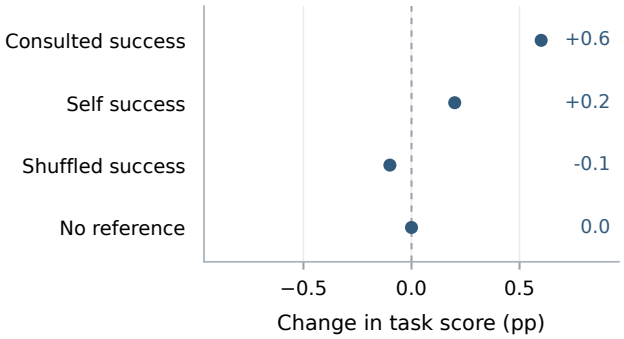


Figure 4 The same success is worth more when it comes from a consultation. We start from one student checkpoint and allow a single OPSD update, varying only the reference: a consulted success of the task, the student's own unassisted success, a consulted success from another task in the same domain, or nothing. Each updated policy is then scored on a test set, and the horizontal axis is the resulting change in adviser-free task score. The unassisted success moves that score, the consulted success of the same task moves it several times further, and a success taken from another task moves it slightly the other way.

Changing the replies changes what the student learns. To test how much of the supervision sits in the replies' content, we retrain the recipe twice more: once with the replies deleted, and once with them moved onto a success from another task, keeping the questions, actions, and observations unchanged (Figure 5). Removing the replies costs 2.9 points and misplacing them 5.2, which takes COPD-4B to 54.6, level with the task-reward reference. The replies therefore add supervision of their own, and they add it only when they fit the task.

The student asks less often and asks much better. To test whether the student's use of the channel changes as the policy improves, we measure its ask rate and the quality of its questions every twenty updates (Figure 6). Communication carries the guidance that COPD distills, so a student that asks less and asks better extracts

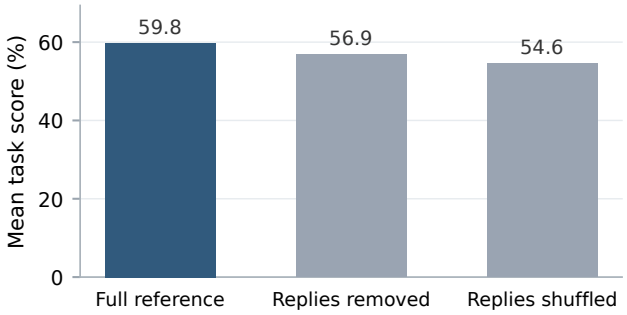


Figure 5 The adviser's words are worth more than the solution they came with. We run the full recipe three times, changing only the replies inside the privileged reference: the complete success, the same success with the words deleted, and those words moved onto a success from another task. Mean adviser-free success falls from 59.8 to 56.9 when the words are deleted and to 54.6 when they come from another task, so the words help only when they describe the task being learned.

more from each exchange. Requests rise from 48.9% at the start to 55.2% at update 20, then fall to 38.9% by update 100. Quality moves the other way: gpt-6-astra's scores climb from 51.2 to 85.8, most of the rise in the first forty updates. The student therefore ends training asking less often and asking far better, spending the channel where the task needs it instead of at every opportunity.

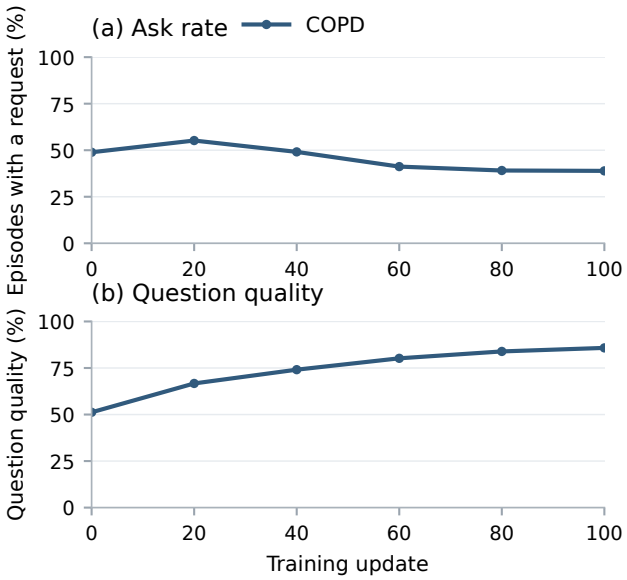


Figure 6 The student ends training asking less often and asking far better. Ask rate and question quality for COPD, measured every twenty updates. Ask rate is the fraction of adviser-enabled episodes containing at least one request. Question quality is a rubric score on task relevance, clarity, and context sufficiency, given by gpt-6-astra from the request and the student's pre-request context, normalized to 0–100. Requests peak at update 20 and then fall, while question quality climbs throughout, so the student leans on the channel less and uses it better.

Ablation study. To test whether each piece of the recipe is necessary, we remove one piece at a time and score every arm without adviser access on the same test set (Figure 7). The w/o OPSD variant keeps communication and task-reward PPO but drops the distillation update, and reaches 47.5, 12.3 points below the full recipe and below CAMP-I at 54.2, so consulted experience does not become capability on its own. The w/ SFT variant imitates the consulted responses instead of scoring the student's own attempts, and reaches 55.4, 4.4 points short, so the target of the update matters as much as its presence. The w/o communication variant keeps distillation but references the student's own verified unassisted successes, and reaches 52.1, 7.7

points short, so the reference has to come from consulted experience. The update carries the largest share of the gain, and neither the channel alone nor imitation of the same trajectories substitutes for it.

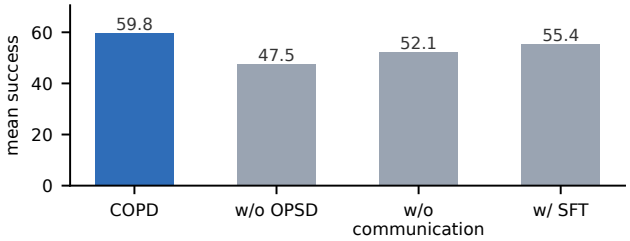


Figure 7 Every component carries part of the gain. Mean adviser-free success over the five domains for COPD and three variants that each remove one piece: the distillation update, the adviser during training, and on-policy targets replaced by supervised fine-tuning on the consulted responses. The w/o communication variant keeps distillation and references the student's own verified unassisted successes. Every variant falls short, and the largest drop comes from the update itself.

The acquired capability transfers beyond CAMG-II. We evaluate the initial students, a larger same-generation model, and both trained policies on Terminal-Bench 2.1 and SWE-Bench Pro V2 without an adviser (Table 2). COPD-4B gains 4.9 points on Terminal-Bench and 5.3 on SWE-Bench Pro, and COPD-9B gains 9.9 and 4.4. Both policies move on both suites, so the gains are not concentrated in one. At 30.3 the 9B policy comes within 0.3 points of Qwen3.5-35B-A3B at 30.6, a larger model of the same generation.

Table 2 Both trained policies improve on both suites, and COPD-9B closes to within 0.3 points of a larger same-generation model. Resolution rate on Terminal-Bench 2.1 and SWE-Bench Pro V2 for the initial students, a larger same-generation model, and the trained policies.

Policy	TB 2.1	SWE-Bench Pro V2	Avg.
Qwen3.5-4B	23.1	11.2	17.1
Qwen3.5-9B	23.1	23.1	23.1
Qwen3.5-35B-A3B	**33.8**	27.4	**30.6**
COPD-4B	28.0	16.5	22.3
COPD-9B	33.0	**27.5**	30.3

Case study. Figure 8 takes one Visual brief, a two-day training-event page, and runs it twice: once with the adviser mounted and advertised, once with the channel removed. Both runs return a finished page with the same sections in a comparable layout, so the behavior established during consultation is available without the channel. This is the qualitative counterpart of the aggregate adviser-free Visual score in Table 1.

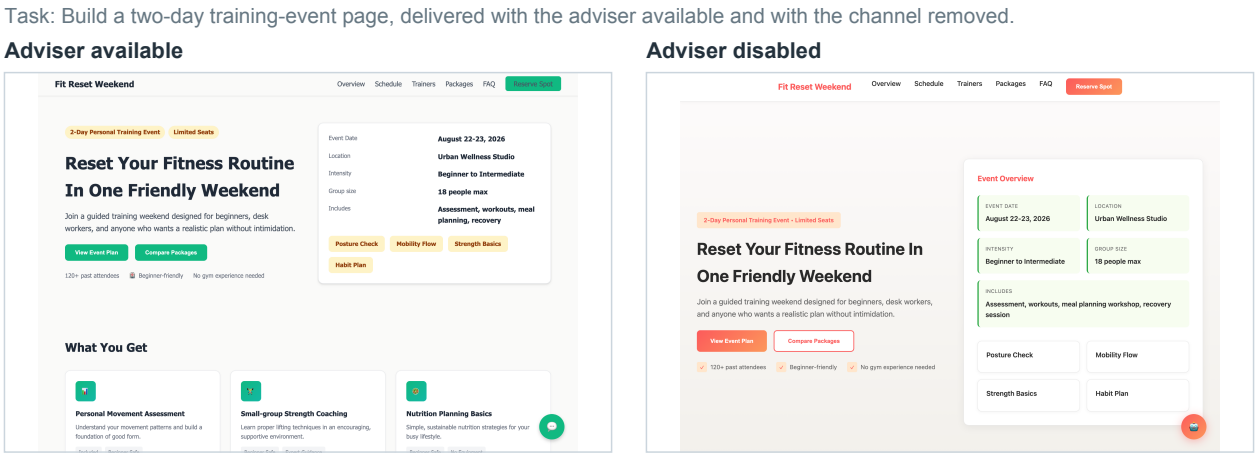


Figure 8 The trained policy reaches a comparable page with or without an adviser. The same visual brief is delivered twice, once with an adviser mounted and advertised and once with the channel removed. The two pages differ in detail but are comparable in structure and finish, so the behavior the consultation established does not depend on reaching an adviser at test time.

6 Conclusion

In this paper, we introduced Communicative On-Policy Distillation (COPD), a framework for transferring black-box teacher capabilities through student-initiated, open-ended natural-language communication. Rather than prescribing when or how the student should seek guidance, COPD allows the student to communicate autonomously during task execution and internalizes successful consulted experience through on-policy self-distillation. Across five long-horizon domains, COPD substantially improves adviser-free task performance over existing black-box distillation methods, with gains transferring to Terminal-Bench 2.1 and SWE-Bench Pro V2. Our ablations and controlled interventions show that successful communication alone does not guarantee independent capability: task-relevant guidance must be internalized into the student's own policy. These findings establish autonomous communication as a viable mechanism for black-box knowledge transfer, enabling students not only to learn with stronger teachers, but also to learn from those interactions to act independently.

Limitations. The study is limited to students up to Qwen3.5-9B and to distillation from a single frozen teacher. Whether the observed gains persist at larger scales or carry over to multi-teacher OPD (MOPD) remains untested.

LLM Usage Statement

Large language model assistants (OpenAI GPT-6.1 Sol in Codex and DeepSeek-V4.1-Flash-1017 in the DeepSeek Harness) were used to polish the writing, write experiment code, and help polish the figures and tables. All model outputs were reviewed by the authors.

References

[1] Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes. In International Conference on Learning Representations (ICLR), 2024.

[2] Soham Dan, Jeff Da, Sanyam Bhutani, Miguel Romero Calvo, et al. SWE-Bench Pro V2: A cleaner, harder-to-game coding benchmark. arXiv preprint arXiv:2610.24206, 2026.

[3] Junfeng Fang, Zhepei Hong, Mao Zheng, Mingyang Song, Gengsheng Li, Houcheng Jiang, Dan Zhang, Haiyun Guo, Xiang Wang, and Tat-Seng Chua. Rubric-based On-policy Distillation. arXiv preprint arXiv:2605.07396, 2026.

[4] Yangyang Feng, Zhuoyan Feng, and Junlan Chen. PAST: Privileged Adaptation from Complete Student Trajectories for On-Policy Self-Distillation. arXiv preprint arXiv:2608.08726, 2026.

[5] Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement Learning via Self-Distillation. In International Conference on Machine Learning (ICML), 2026.

[6] Zirui Liu, Tingyue Pan, Jie Ouyang, Qi Liu, Xianquan Wang, Jiayu Liu, Qingchuan Li, Jing Sha, Zhenya Huang, Shijin Wang, and Enhong Chen. SocraticPO: Policy Optimization via Interactive Guidance. arXiv preprint arXiv:2606.09887, 2026.

[7] Lirui Luo, Kelong Mao, Heming Xia, Rongqing Li, Xinwei Yang, Luyu Chen, Kieran Wong, Yudong Guo, Xinrui Wang, Jiayin Zhu, Simiu Gu, Sulong Xu, and Cong Fang. Coding agent memory post-training: Unlocking the memory potential of pre-trained file operations for long-horizon tasks via reinforcement learning. arXiv preprint arXiv:2609.34422, 2026.

[8] Lirui Luo, Kelong Mao, Xinwei Yang, Rongqing Li, Luyu Chen, Kieran Wong, Yudong Guo, Xinrui Wang, Jiayin Zhu, Gu Simiu, Sulong Xu, and Cong Fang. Coding Agent Memory Post-Training II: Unlocking the memory potential of pre-trained file operations for long-horizon tasks via on-policy distillation. arXiv preprint arXiv:2611.14009, 2026.

[9] Mike A. Merrill, Alexander G. Shaw, Nicholas Carlini, Boxuan Li, Harsh Raj, Ivan Bercovich, et al. Terminal-bench: Benchmarking agents on hard, realistic tasks in command line interfaces. arXiv preprint arXiv:2601.11868, 2026.

[10] Ross Murphy, Sergey Mosesov, Javier Leguina Peral, and Thymo ter Doest. Ask Before You Act: Generalising to Novel Environments by Asking Questions. arXiv preprint arXiv:2209.04665, 2022.

[11] Qwen Team. Qwen3.5: Towards native multimodal agents, 2026. Qwen3.5 family release, February 2026. https://qwen.ai/blog?id=qwen3.5.

[12] Ram Ramrakhya, Matthew Chang, Xavier Puig, Ruta Desai, Zsolt Kira, and Roozbeh Mottaghi. Grounding Multimodal LLMs to Embodied Agents that Ask for Help with Reinforcement Learning. arXiv preprint arXiv:2504.00907, 2025.

[13] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms. arXiv preprint arXiv:1707.06347, 2017.

[14] Tu Trinh, Mohamed Elfeki, Guangze Luo, Kelvin Luu, Nathan Hunt, Ernesto Hernández, Nandan Marwaha, Yannis Yiming He, Charles Wang, Fernando Carabedo, Alessa Castillo, and Bing Liu. HiL-Bench (Human-in-Loop Benchmark): Do Agents Know When to Ask for Help? arXiv preprint arXiv:2604.09408, 2026.

[15] Haojin Wang, Dylan Zhang, Huaibo Chen, Suhao Yu, Yihang Sun, Zhanyang Jin, Jiaying Ye, Dianqi Li, Prasanna Sattigeri, Kamal Youcef-Toumi, and Hao Peng. Learning from Teacher Continuations at Student States. arXiv preprint arXiv:2609.36246, 2026.

[16] Tianzhu Ye, Li Dong, Zewen Chi, Xun Wu, Shaohan Huang, and Furu Wei. Black-Box On-Policy Distillation of Large Language Models. arXiv preprint arXiv:2511.10643, 2025.

[17] Huining Yuan, Xiangmin Yi, Zelai Xu, Zixu Wang, Xinlei Chen, Xiao-Ping Zhang, Yu Wang, and Chao Yu. ActionOPD: Distilling Decisions, Not Tokens. arXiv preprint arXiv:2610.19820, 2026.

[18] Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-Distilled Reasoner: On-Policy Self-Distillation for Large Language Models. In International Conference on Machine Learning (ICML), 2026.

[19] Yuhang Zhou, Lizhu Zhang, Yifan Wu, Mingyi Wang, Bo Peng, Jiayi Liu, Xiangjun Fan, and Zhuokai Zhao. OmniOPD: Logit-Free On-Policy Distillation via Speculative Verification. arXiv preprint arXiv:2606.01476, 2026.