WORLDOPD: WORLD-KNOWLEDGE AGENTIC ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Agentic on-policy distillation (OPD) trains a student to match a teacher's action distribution at states the student visits. Yet action distributions alone do not tell the student how the environment changes, how tools respond, or what consequences an action is likely to have. We call this information *world knowledge* (WK). Such knowledge may be especially hard for a smaller student to infer from multi-turn trajectories. We find that teacher-generated world knowledge reduces teacher–student divergence at most rollout states, but can increase it at some states. World knowledge has a state-dependent contextual effect, motivating selective auxiliary supervision. We introduce WorldOPD, which teaches students both how to act and what their actions may lead to. Alongside standard action OPD, WorldOPD distills the teacher's environmental knowledge through student-generated action–state pairs, supervising both candidate actions and their anticipated consequences. Since the benefit of world knowledge varies across interaction states, a lightweight predictor learns from sparse probes to select when to apply this additional supervision. Extensive experiments across function-calling and multi-turn search tasks show that WorldOPD consistently improves student performance.

1 INTRODUCTION

On-policy distillation (OPD) transfers capabilities from a strong teacher to a smaller student by supervising student-generated outputs (Agarwal et al., 2024; Tan et al., 2023). Unlike distillation from fixed teacher demonstrations (Kim & Rush, 2016), OPD provides feedback on the student's own generations, reducing the mismatch between training and inference. In agentic settings, this means supervising actions at states reached through the student's interactions with an environment (Wang et al., 2026a; Liu et al., 2026b; Liang et al., 2026).

Effective interaction requires understanding what actions may lead to. A tool call can succeed, fail, or change the options available at the next step; a search query can reveal evidence that determines what to investigate next. We refer to knowledge of tool behavior, environment transitions, and action consequences as *world knowledge* (WK). Standard action OPD does not explicitly convey this knowledge: it supervises which action to take, while leaving the student to infer the environmental understanding behind that decision. For example, knowing that a delivered order cannot be canceled but may be returned helps explain which tool call is appropriate. A smaller student may struggle to acquire such knowledge from action supervision and multi-turn trajectories alone. Research on world modeling for language agents also highlights the value of understanding action outcomes for planning and interaction (Chen et al., 2025a; Chae et al., 2025; Yu et al., 2026; Lu et al., 2026).

We first examine whether making the teacher's WK explicit helps align the student's action distribution with the teacher's (Section 3). Holding rollout states and student-generated action prefixes fixed, we compare teacher–student divergence with and without teacher-generated WK in the student's context. Our analysis reveals two complementary findings. First, WK reduces divergence on average across the evaluated model pairs and tasks (Figure 2). On function calling, providing more candidate action–state pairs can further reduce divergence, although the most effective pair count varies across model pairs. Second, this aggregate benefit masks substantial variation across states (Figure 3): WK reduces divergence at 57–84% of states but increases it at the remaining 16–43%. Together, these findings motivate transferring WK as an additional source of supervision and selecting where to apply it based on the interaction state.

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

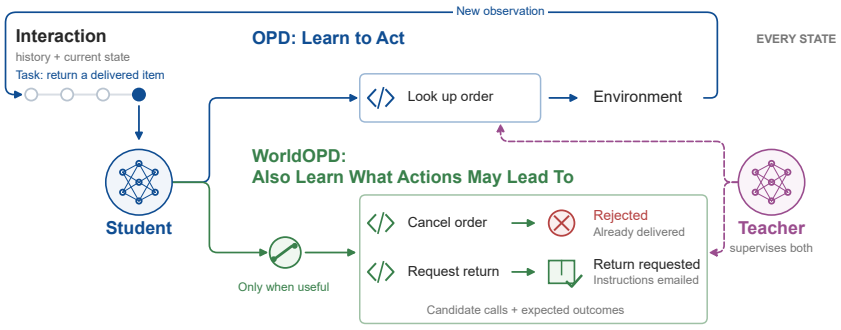


Figure 1: Overview of WorldOPD. Standard action OPD is augmented with state-conditioned OPD over student-generated action–state pairs, transferring the teacher's world knowledge.

We introduce WorldOPD, which teaches students both how to act and what their actions may lead to (Figure 1). WorldOPD retains standard action OPD and adds an auxiliary OPD objective over student-generated action–state pairs. Each pair describes a candidate action and its anticipated consequences. The teacher supervises both components along the student's generated prefixes, transferring its environmental knowledge through feedback on the student's own outputs.

To select where this additional supervision is useful, WorldOPD learns a lightweight predictor conditioned on the interaction history and current observation. At a small random subset of rollout states, we measure how teacher-provided WK context changes action divergence and use this signal as a proxy for WK usefulness. The predictor learns from these sparse probes and selects states before auxiliary WK generation and distillation. Thus, contextual probes guide when to teach WK, while supervision over student-generated pairs determines how that knowledge is transferred.

Experiments across function-calling and multi-turn search benchmarks show that WorldOPD consistently improves over standard OPD. It also generally outperforms additional action supervision under a matched distillation-token budget and achieves the highest average search score for every evaluated student–teacher pair among the compared methods. Ablations show that selecting states for WK supervision improves over applying it at every visited state.

Our contributions are: (1) We show that teacher-provided WK reduces teacher–student action divergence on average, while revealing variation in its effect across rollout states. (2) We introduce WorldOPD, which combines action OPD with joint distillation of student-generated action–state pairs and uses a predictor trained on sparse probes to select states for WK supervision. (3) We demonstrate gains across function-calling and multi-turn search benchmarks, with budget-matched comparisons and ablations supporting the value of WK supervision and state-conditioned selection.

2 Related Work

2.1 On-Policy Distillation

Knowledge distillation (KD) transfers knowledge from a stronger teacher to a student model. Early work studied sequence-level distillation, where the student is trained on teacher-generated outputs (Kim & Rush, 2016). Recent LLM distillation methods move beyond off-policy teacher data: Agarwal et al. (2024) and Tan et al. (2023) introduce on-policy/generalized KD objectives where the teacher supervises student-sampled outputs, reducing distribution mismatch between training and inference. This paradigm has been adopted for reasoning-model post-training (DeepSeek-AI et al., 2026) and surveyed in recent OPD literature (Song & Zheng, 2026). Follow-up work analyzes OPD failure modes (Fu et al., 2026; Li et al., 2026b), stabilizes reverse-KL or entropy-aware objectives (Jang et al., 2026; Jin et al., 2026), and improves OPD through privileged information, advantage redesign, dual-perspective optimization, relaxed objectives, reward extrapolation, and efficient offline training (Penaloza et al., 2026; Zhao et al., 2026a; Hou et al., 2026; Ko et al., 2026; Yang et al., 2026; Wu et al., 2026). Recent variants further reduce reliance on teacher logits via speculative verification or rubric-based supervision (Zhou et al., 2026a; Fang et al., 2026), connect OPD

with self-distillation, RLVR, and binary-reward supervision (Zhao et al., 2026b; Li et al., 2026a; He et al., 2026b), adapt OPD/self-distillation to agents, multilingual reasoning, long-context modeling, and search-augmented reasoning (Zhong et al., 2026; Liu et al., 2026a; Zhang et al., 2026b; Liang et al., 2026), and study when token-level distillation is helpful or harmful (Wang et al., 2026c; Jiang et al., 2026; Shen et al., 2026; Ke et al., 2026).

2.2 World Modeling for Language Agents

Recent work studies agentic reinforcement learning, where language agents improve through multi-step interaction with external environments and delayed feedback rather than only supervised demonstrations. (Zhang et al., 2025; Chen et al., 2025a). Beyond direct policy optimization, world modeling has emerged as a complementary direction: agents learn to predict future states, observations, rewards, or execution outcomes, enabling lookahead planning and denser supervision from sparse feedback (Li et al., 2026c; Chu et al., 2026; Yang, 2026; Levy et al., 2025). Recent work builds world models for web navigation (Chae et al., 2025; Xiao et al., 2026), GUI/mobile control (Zheng et al., 2026; Cao et al., 2026; Koh et al., 2026), software and terminal agents (Sun et al., 2026; Shrivastava et al., 2026), game or symbolic environments (Lehrach et al., 2026; Hu et al., 2025a;b), and general agents (Zuo et al., 2026). Other methods integrate world modeling with RL through transition modeling, explicit state-prediction rewards, policy–world-model co-training, or latent hierarchical planning (Yu et al., 2026; Wang et al., 2026d; Lu et al., 2026; Zhang et al., 2026a). While these methods suggest that world models can improve planning and sample efficiency, recent work shows that current agents may struggle to use them effectively for foresight (Qian et al., 2026).

3 Motivating Analysis: World Knowledge Helps, but Not Uniformly

We first examine whether explicit world knowledge (WK) brings a student's action distribution closer to its teacher's, and how this effect varies across states. We measure the effect of adding teacher-generated WK to the student's context, without updating either model.

Setup and measurement. We use Qwen3-0.6B, Qwen3-1.7B, and Qwen3-4B students with Qwen3-14B or Qwen3-32B teachers (Yang et al., 2025), on BFCL-v4 function calling (Patil et al., 2025) and HotpotQA search (Yang et al., 2018). Throughout the paper, the interaction state $s_t = (h_t, o_t)$ includes both the history h_t (the task and previous action–observation exchanges) and the current observation o_t. We omit the step index when it is clear from context. We use a for a student-generated environment action and w for an explicit description of candidate actions and their predicted consequences. Superscripts T and S distinguish teacher- and student-generated WK; π_T and π_θ denote the teacher and student policies.

At each collected state, the student greedily generates a without WK. The teacher independently generates the complete WK context w^T, including both candidate actions and their predicted next states for function calling (and the full planning context for search). We add this complete context to the student and evaluate divergence along the same fixed action prefixes. For optional student context c, define

$$D(s, a; c) = \frac{1}{|a|} \sum_{j=1}^{|a|} D_{\mathrm{KL}}\Big(\pi_T(\cdot \mid s, a_{<j}) \Big\| \pi_\theta(\cdot \mid s, c, a_{<j})\Big). \tag{1}$$

We write $D_{\mathrm{base}}(s) = D(s, a; \varnothing)$ and $D_{\mathrm{ctx}}(s) = D(s, a; w^T)$ for the baseline and WK-conditioned divergences, respectively, suppressing their dependence on the fixed action a and teacher-generated WK w^T. Here, $w^T \sim \pi_T^{\mathrm{wk}}(\cdot \mid s)$, where the superscript wk denotes the WK-generation prompt applied to the same model. Both divergences use identical action prefixes and a fixed teacher distribution; neither model is updated.

We summarize the contextual effect of WK in two ways. Figure 2 reports the mean relative KL reduction, expressed as a percentage:

$$\mathrm{Reduction}(s, a, w^T) = \frac{D(s, a; \varnothing) - D(s, a; w^T)}{D(s, a; \varnothing)}. \tag{2}$$

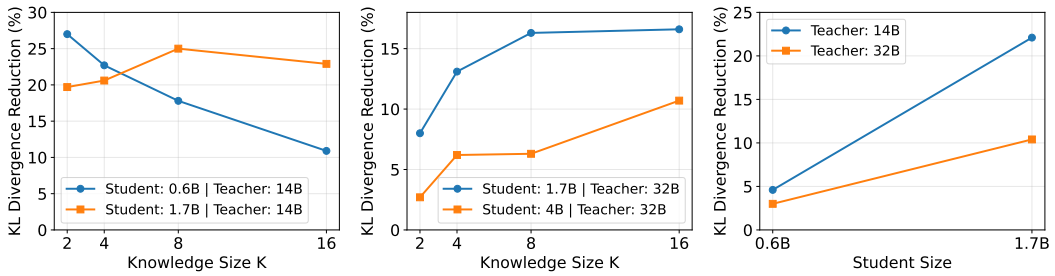


Figure 2: **World knowledge reduces teacher–student divergence on average.** The first two panels show BFCL results as the fixed number of action–state pairs K varies; the final panel shows HotpotQA results. Values are mean relative KL reductions from Eq. (2), expressed as percentages;

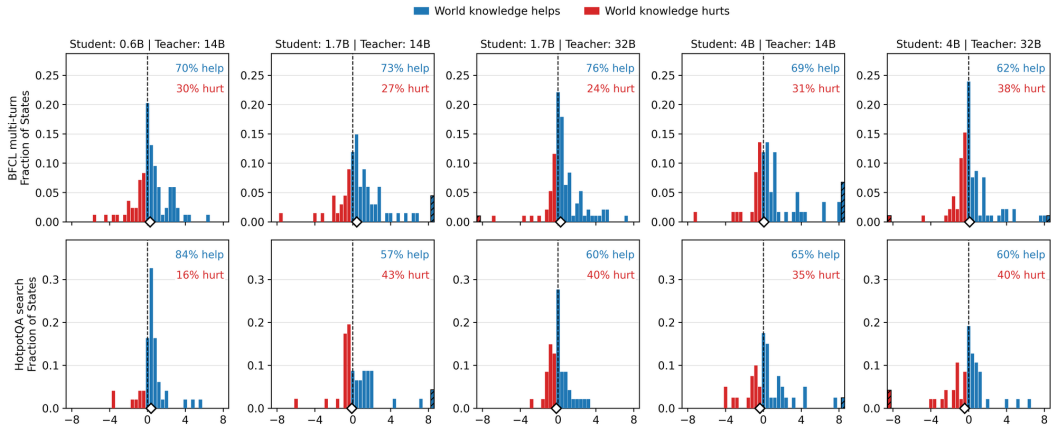


Figure 3: **World-knowledge utility varies across rollout states.** Histograms of the per-state log-ratio $R(s) = \log_2(D_{\text{base}}(s)/D_{\text{ctx}}(s))$ for BFCL multi-turn function calling (top) and HotpotQA search (bottom), across five student–teacher pairs. Positive values (blue) indicate that WK reduces divergence; negative values (red) indicate that WK increases divergence.

The plotted values are averaged over states and multiplied by 100. Figure 3 shows the per-state log-ratio $R(s) = \log_2[D_{\text{base}}(s)/D_{\text{ctx}}(s)]$: positive values indicate reduced divergence, and negative values indicate increased divergence. For the utility predictor in Section 4, we use the unnormalized reduction $U(s) = D_{\text{base}}(s) - D_{\text{ctx}}(s)$ as the training target. Full protocols appear in Appendix A.4.

World knowledge reduces divergence on average. Figure 2 reports the aggregate diagnostic. For BFCL, we vary the fixed number of teacher-provided action–state pairs over $K \in \{2, 4, 8, 16\}$. WK reduces teacher–student KL across model pairs, although the best fixed size depends on student capacity and teacher quality. For example, the Qwen3-1.7B student with the 32B teacher reaches roughly 16% relative reduction, while the 4B student improves steadily as more pairs are provided. On HotpotQA, process-level WK also reduces divergence for both 0.6B and 1.7B students. These results establish that externalized WK can move the student distribution toward the teacher, but the pair-size analysis only characterizes an aggregate effect under a fixed WK configuration.

WK benefits most states, with state-dependent gains. Figure 3 reports the distribution of the per-state log-ratio $R(s)$. WK reduces teacher–student divergence for a majority of states in every displayed student–teacher pair: $70\%, 73\%, 76\%, 69\%$, and 62% on BFCL, and $84\%, 57\%, 60\%, 65\%$, and 60% on HotpotQA. These results show that teacher-provided WK is broadly useful across tasks and model pairs, while the variation in its per-state effect motivates adapting WK distillation to the expected benefit at each state.

Implication. Providing the teacher's complete WK context reduces teacher–student action divergence on average, with variation across states. These observations motivate using world knowledge as additional supervision and adapting its use to the current state.

4 WORLDOPD: SELECTIVE WORLD-KNOWLEDGE DISTILLATION

WorldOPD augments action OPD with distillation of the teacher's understanding of the environment: which candidate actions are relevant and what consequences they may have. Action OPD supervises the decision executed in the environment; WK OPD transfers this broader knowledge through complete action–state pairs. A learned gate selects states for additional WK supervision. We use the notation and diagnostic utility $U(s)$ introduced in Section 3.

4.1 WORLD-KNOWLEDGE DISTILLATION

At a student-visited state s, the student samples an environment action $a \sim \pi_\theta(\cdot \mid s)$. Following the reverse-KL formulation of OPD (Li et al., 2026b), both models are evaluated on the student's own prefixes. We write the per-action loss with token-length normalization:

$$\mathcal{L}_{\text{act}}(s) = \frac{1}{|a|} \sum_{j=1}^{|a|} D_{\text{KL}}\Big(\pi_\theta(\cdot \mid s, a_{<j}) \Big\| \, \pi_T(\cdot \mid s, a_{<j})\Big). \tag{3}$$

Here the KL direction is student-to-teacher. The forward-KL diagnostic D_{base} in Section 3 measures contextual alignment and is distinct from this training loss.

At a state selected for WK distillation, the student generates $w^S \sim \pi_\theta^{\text{wk}}(\cdot \mid s)$, a serialized sequence of candidate action–state pairs (u_k, z_k), where u_k is a candidate action and z_k describes its anticipated consequences. Together, these pairs express an understanding of the environment beyond the single action chosen for execution. The number of pairs is fixed by the task configuration.

We apply the same reverse-KL OPD objective to the complete WK sequence:

$$\mathcal{L}_{\text{wk}}(s) = \frac{1}{|w^S|} \sum_{j=1}^{|w^S|} D_{\text{KL}}\Big(\pi_\theta^{\text{wk}}(\cdot \mid s, w^S_{<j}) \Big\| \, \pi_T^{\text{wk}}(\cdot \mid s, w^S_{<j})\Big). \tag{4}$$

At each position, teacher and student distributions are evaluated on the same student-generated prefix, so supervision remains on-policy across the full pair sequence. Section 3 measures the immediate effect of supplying complete teacher-generated WK w^T as context. Here, we transfer that form of knowledge into the student through supervision on its own WK outputs w^S. Task-specific WK formats appear in Appendix A.5.

4.2 STATE-CONDITIONED ADAPTIVE DISTILLATION

The diagnostic in Section 3 suggests that WK supervision should depend on the interaction context. As defined in Section 3, $s = (h, o)$ contains the prior interaction history and current observation. A separate predictor $\hat{U}_\phi(s)$ uses this full context to estimate WK utility. We train it from sparse probes and invoke WK OPD when its prediction exceeds a threshold τ. We initialize the predictor from Qwen3-0.6B with a prediction head; initialization and training details appear in Appendix A.2.

During training, each visited state is independently selected for probing with probability ρ (5% by default). At a probe state, the teacher generates w^T and we evaluate $D_{\text{ctx}}(s)$ on the same student action used for $D_{\text{base}}(s)$. Their difference gives $U(s)$ from Section 3, reusing the preliminary measurement without changing the rollout action. For the collection of probe states $\mathcal{P}$, we train the predictor and define the gate as

$$\begin{aligned} \mathcal{L}_{\text{utility}}(\phi) &= \frac{1}{|\mathcal{P}|} \sum_{s \in \mathcal{P}} \big(\hat{U}_\phi(s) - \text{sg}[U(s)]\big)^2, \\ g_\phi(s) &= \mathbb{1}[\hat{U}_\phi(s) > \tau], \end{aligned} \tag{5}$$

where sg stops gradients through the probe target. This regression updates only the predictor parameters ϕ. The predictor is evaluated before auxiliary WK generation, using only the history and

Algorithm 1 WorldOPD training

Inputs: student π_θ, frozen teacher π_T, predictor $\hat{U}_\phi$; probe rate ρ, threshold τ, WK weight α; pair count K
for each training batch **do**
  Initialize student loss $\mathcal{L}_{\text{student}} \leftarrow 0$ and probe collection $\mathcal{P} \leftarrow \varnothing$
  for each task **do**
    Initialize h with the task and obtain o; set $s \leftarrow (h, o)$
    while the rollout is not finished **do**
      Evaluate $\hat{U}_\phi(s)$ and set $g_\phi(s)$ using Eq. (5)
      Sample $a \sim \pi_\theta(\cdot \mid s)$; accumulate $\mathcal{L}_{\text{act}}(s)$ in $\mathcal{L}_{\text{student}}$
      if $g_\phi(s) = 1$ **then**
        Sample K complete action–state pairs $w^S \sim \pi_\theta^{\text{wk}}(\cdot \mid s)$
        Evaluate teacher logits on the student WK prefixes
        Accumulate $\alpha\mathcal{L}_{\text{wk}}(s)$ in $\mathcal{L}_{\text{student}}$ over both action and state tokens (Eq. (4))
      end if
      Sample $b \sim \text{Bernoulli}(\rho)$ independently of $g_\phi(s)$
      if $b = 1$ **then**
        Sample teacher WK $w^T \sim \pi_T^{\text{wk}}(\cdot \mid s)$
        Evaluate $D_{\text{base}}(s)$ and $D_{\text{ctx}}(s)$ on the same prefixes of a (Eq. (1))
        Append $(s, \text{sg}[D_{\text{base}}(s) - D_{\text{ctx}}(s)])$ to $\mathcal{P}$
      end if
      Execute a and receive the next observation o_{new}
      Append (o, a) to h; set $o \leftarrow o_{\text{new}}$ and $s \leftarrow (h, o)$
    end while
  end for
  Update student parameters θ using $\mathcal{L}_{\text{student}}$
  if $\mathcal{P}$ is nonempty **then**
    Update predictor parameters ϕ using $\mathcal{L}_{\text{utility}}(\phi)$
  end if
end for

current observation in s. Only selected states trigger student WK generation, teacher evaluation of its prefixes, and computation of $\mathcal{L}_{\text{wk}}(s)$. Sparse probes are sampled independently of this gate to train the predictor; they do not compute the WK training loss. Contextual utility thus guides selection, while WK OPD supervises the complete pair sequence in Eq. (4).

For rollout states $\mathcal{B}$, we first select the WK states and then optimize the student:

$$\mathcal{B}_{\text{wk}} = \{s \in \mathcal{B} : g_\phi(s) = 1\},$$
$$\mathcal{L}_{\text{student}}(\theta) = \sum_{s \in \mathcal{B}} \mathcal{L}_{\text{act}}(s) + \alpha \sum_{s \in \mathcal{B}_{\text{wk}}} \mathcal{L}_{\text{wk}}(s). \tag{6}$$

Here α weights WK supervision. Selection is held fixed during the student update: $\mathcal{L}_{\text{student}}$ updates θ, while $\mathcal{L}_{\text{utility}}$ updates ϕ independently. WorldOPD preserves action OPD at every state and computes additional WK supervision only where selected.

5 Experiments

5.1 Experimental Setup

Datasets and benchmarks. We evaluate WorldOPD on function calling and multi-turn search. For function calling, we train on ToolACE (Liu et al., 2025), follow the setting of MAD-OPD (Wang et al., 2026a), and evaluate on BFCL-v4 (Patil et al., 2025), τ^2-Bench (Barres et al., 2025), and VitaBench (He et al., 2026a). For search, we train on HotpotQA (Yang et al., 2018) and evaluate on HotpotQA, 2Wiki (Ho et al., 2020), Bamboogle (Press et al., 2023), MuSiQue (Trivedi et al., 2022), and BrowseComp-Plus (Chen et al., 2025b).

Table 1: Tool-calling results across Qwen3 student–teacher pairs. "–" indicates results not reported because the 0.6B student performs poorly on VitaBench.

Benchmark	Student	0.6B		1.7B		4B	
	Teacher	14B	32B	14B	32B	14B	32B
BFCL-v4	OPD	26.3	32.4	51.8	50.7	50.3	56.4
	OPD$^+$	25.2	35.5	52.1	51.8	50.6	59.6
	SAGE-OPD	25.5	32.0	**56.5**	51.4	52.1	57.9
	TCOD	25.7	33.2	54.8	51.7	54.2	57.4
	WorldOPD	**38.4**	**41.4**	54.1	**55.2**	**57.2**	**60.2**
τ^2-Bench	OPD	6.3	6.6	8.9	11.6	12.2	18.2
	OPD$^+$	4.7	7.1	9.3	10.7	12.1	19.4
	SAGE-OPD	8.7	9.2	11.3	11.6	14.4	20.6
	TCOD	9.7	9.1	13.0	11.6	16.1	20.8
	WorldOPD	**9.8**	**11.2**	**20.0**	**16.8**	**22.7**	**27.3**
VitaBench	OPD	–	–	6.1	1.4	11.6	12.6
	OPD$^+$	–	–	**8.2**	2.6	13.9	15.2
	SAGE-OPD	–	–	6.1	5.9	14.3	14.0
	TCOD	–	–	5.8	5.8	11.5	14.4
	WorldOPD	–	–	7.4	**8.1**	**14.7**	**19.2**

Models. We use Qwen3 (Yang et al., 2025) students at 0.6B, 1.7B, and 4B scales, with Qwen3-14B and Qwen3-32B as teachers. To assess generalization beyond Qwen3, we also evaluate a Llama-3.2-1B-Instruct student with a Llama-3.1-8B-Instruct-based teacher; results appear in Appendix B.

Baselines. We compare WorldOPD with **OPD**, which performs standard action-level on-policy distillation, and **OPD$^+$**, which matches WorldOPD's distillation-token budget by randomly sampling additional on-policy rollouts (Appendix A.3). We also compare with **SAGE-OPD** (Zhou et al., 2026b), which selectively applies teacher supervision using turn-level intervention and teacher-confidence weighting, and **TCOD** (Wang et al., 2026b), which gradually increases rollout depth through a temporal curriculum. Section 5.3 isolates the contribution of state-conditioned selection. Additional details are provided in Appendix A.1.

Evaluation metrics. For tool calling, we report BFCL-v4 accuracy and task success rates on τ^2-Bench and VitaBench following the evaluation metrics used by MAD-OPD (Wang et al., 2026a). For search, we report normalized exact match (EM) on HotpotQA, 2Wiki, Bamboogle, and MuSiQue. For BrowseComp-Plus, we use answer accuracy from the following (Chen et al., 2025b).

5.2 Effect of World-Knowledge Distillation

In this section, we evaluate whether WorldOPD improves downstream agent performance over action-only distillation. Tables 1 and 2 report tool-calling and multi-turn search performance, respectively. We make three observations.

(1) *WK supervision improves tool-calling performance.* In Table 1, WorldOPD outperforms OPD and OPD$^+$ across the reported model pairs on BFCL and τ^2-Bench. It also outperforms SAGE-OPD and TCOD on τ^2-Bench and most BFCL settings. For the 1.7B student with the 14B teacher, however, SAGE-OPD and TCOD score 56.5 and 54.8, respectively, versus WorldOPD's 54.1. On VitaBench, WorldOPD improves over OPD, SAGE-OPD, and TCOD across the reported pairs, although it trails OPD$^+$ for the 1.7B student with the 14B teacher (7.4 vs. 8.2).

(2) *The gains extend to multi-turn search.* In Table 2, WorldOPD achieves the highest average score for every evaluated model pair, improving over SAGE-OPD by 3.4–8.2 points and TCOD by 2.9–7.5 points. It also outperforms OPD and OPD$^+$ on each search benchmark. SAGE-OPD leads in two individual settings: MuSiQue for the 1.7B student with the 14B teacher (20.6 vs. 19.9), and BrowseComp-Plus for the 4B student with the 14B teacher (12.9 vs. 11.7). TCOD also leads on BrowseComp-Plus for the 1.7B student with the 14B teacher (9.1 vs. 8.5).

Table 2: Multi-turn search results for OPD, OPD$^+$, SAGE-OPD, TCOD, and WorldOPD.

Method	HotpotQA	2Wiki	Bamboogle	MuSiQue	BrowseComp	Avg.
Qwen3: 0.6B ← 14B						
OPD	29.8	50.9	47.6	10.8	2.5	28.3
OPD$^+$	28.6	51.5	47.8	13.8	4.9	29.3
SAGE-OPD	31.9	50.1	47.2	12.3	2.7	28.8
TCOD	28.9	53.9	50.3	10.0	4.1	29.4
WorldOPD	**37.4**	**55.6**	**54.9**	**16.3**	**6.6**	**34.2**
Qwen3: 0.6B ← 32B						
OPD	14.0	21.5	21.1	5.2	1.1	12.6
OPD$^+$	14.1	21.6	23.0	6.2	3.1	13.6
SAGE-OPD	16.6	25.1	25.5	4.0	4.7	15.2
TCOD	17.5	23.2	24.0	7.6	3.4	15.1
WorldOPD	**24.0**	**37.4**	**32.8**	**13.8**	**5.4**	**22.7**
Qwen3: 1.7B ← 14B						
OPD	40.2	51.5	55.8	16.1	5.2	33.8
OPD$^+$	41.7	54.6	56.8	18.4	5.4	35.4
SAGE-OPD	45.3	53.1	56.3	**20.6**	4.8	36.0
TCOD	45.1	54.3	58.5	16.2	**9.1**	36.6
WorldOPD	**47.4**	**59.4**	**62.8**	19.9	8.5	**39.6**
Qwen3: 1.7B ← 32B						
OPD	35.1	40.2	39.2	9.9	3.9	25.7
OPD$^+$	38.4	43.5	41.7	11.5	2.8	27.6
SAGE-OPD	34.9	40.1	41.1	13.6	7.5	27.4
TCOD	36.2	43.8	41.7	12.0	7.6	28.3
WorldOPD	**42.4**	**48.0**	**58.6**	**20.4**	**8.7**	**35.6**
Qwen3: 4B ← 14B						
OPD	41.0	55.9	56.6	17.1	7.6	35.6
OPD$^+$	43.8	58.4	57.4	16.2	9.2	37.0
SAGE-OPD	39.8	56.8	60.5	21.3	**12.9**	38.3
TCOD	44.2	55.3	61.5	18.5	9.0	37.7
WorldOPD	**49.3**	**60.6**	**64.2**	**22.4**	11.7	**41.6**
Qwen3: 4B ← 32B						
OPD	49.5	64.9	59.6	24.9	9.2	41.6
OPD$^+$	52.3	63.7	61.8	23.7	10.1	42.3
SAGE-OPD	52.0	65.1	58.7	27.9	11.0	42.9
TCOD	49.8	68.9	61.5	29.6	11.8	44.3
WorldOPD	**54.1**	**69.4**	**69.7**	**30.6**	**12.2**	**47.2**

(3) *Additional action supervision does not explain the gains.* OPD$^+$ matches the WK condition's supervised-token budget, yet generally underperforms WK-based distillation in both tables. For example, with the 4B student and 32B teacher, OPD$^+$ raises the average search score from 41.6 to 42.3, whereas WorldOPD reaches 47.2. This comparison suggests that the content of WK supervision contributes beyond simply increasing the number of supervised tokens.

5.3 Ablation Study

We examine the contributions of WK supervision and state-conditioned selection by comparing OPD, which uses action distillation alone; WorldOPD$\backslash G$, which applies WK distillation at every visited state; and WorldOPD, which selects states using the learned utility predictor. All three retain action distillation throughout the rollout. Figure 4 yields two observations. (1) *Uniform WK supervision is not always beneficial.* WorldOPD$\backslash G$ often improves over OPD but introduces regressions in some configurations. This complements the preliminary observation that the benefit

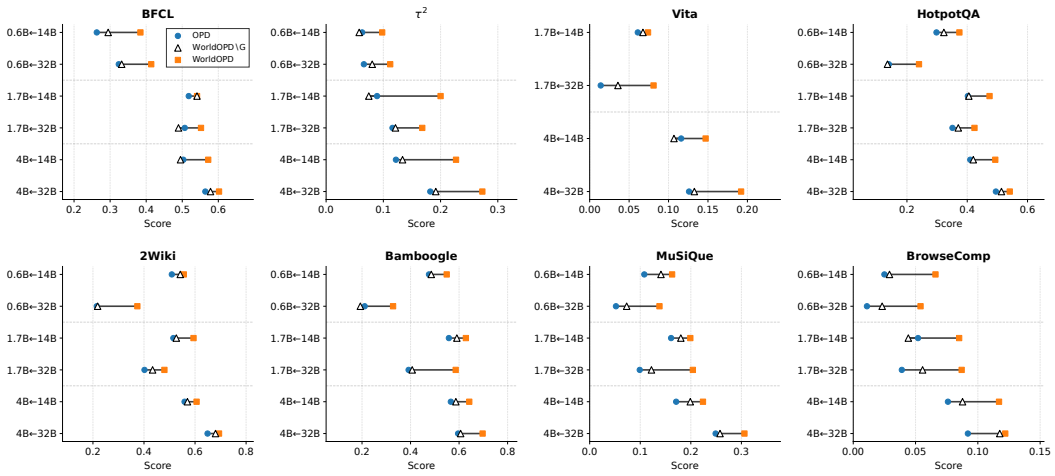


Figure 4: Ablation of WK supervision and state selection.

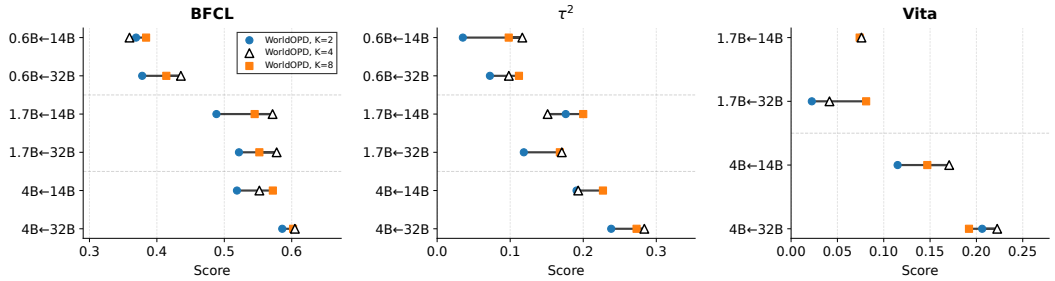


Figure 5: Effect of the WK pair count. WorldOPD with $K \in \{2, 4, 8\}$ candidate action–state pairs on three tool-calling benchmarks.

of WK varies across states (Section 3), suggesting that applying WK supervision indiscriminately can hurt downstream performance. (2) *State-conditioned selection improves WK distillation.* WorldOPD consistently outperforms both alternatives across the evaluated benchmarks and model pairs, including settings where uniform WK already improves over OPD.

5.4 Hyperparameter Analysis

Following the setup in Section 5.1, we vary the number of candidate action–state pairs per WK invocation, $K \in \{2, 4, 8\}$, on BFCL, τ^2-Bench, and VitaBench. While the gate selects which states receive WK supervision, K controls the amount of supervision at each selected state. Figure 5 yields two observations. (1) *Additional WK generally improves performance.* Using $K = 4$ or $K = 8$ generally outperforms $K = 2$, consistent with the preliminary finding that additional WK can further reduce distributional mismatch (Figure 2). (2) *Increasing the pair count yields diminishing returns.* Neither $K = 4$ nor $K = 8$ consistently dominates, suggesting limited benefit from further expansion. Additional pairs may introduce redundant or less reliable supervision. Overall, a moderate pair count appears sufficient to capture most of the benefit of WK supervision.

6 Conclusion

We introduce WorldOPD, which augments action OPD with selective distillation of student-generated action–state pairs. Experiments on function calling and multi-turn search show gains over standard OPD. Supervised-token-matched comparisons and ablations support the value of both WK supervision and state-conditioned selection.

AI USE STATEMENT

Generative AI tools were used for language editing, improving the clarity and presentation of the manuscript, and limited code editing. They were not used to design the methodology or experiments, develop hypotheses, or interpret the results. The authors reviewed and verified all AI-assisted work and take full responsibility for the final manuscript.

ETHICS STATEMENT

This work studies distillation for language agents using existing datasets and benchmarks. More capable agents may inherit teacher biases or produce unreliable tool calls and outputs. Deployment therefore requires appropriate safeguards and human oversight.

REPRODUCIBILITY STATEMENT

We provide the training procedure in Algorithm 1 and the experimental setup in Section 5.1. The appendix details the preliminary analysis protocols (Appendix A.4), WK formats (Appendix A.5), utility predictor (Appendix A.2), and budget-matched baseline (Appendix A.3).

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. τ^2-bench: Evaluating conversational agents in a dual-control environment. *arXiv preprint arXiv:2506.07982*, 2025.

Yilin Cao, Yufeng Zhong, Zhixiong Zeng, Siran Dai, Liming Zheng, Jing Huang, Haibo Qiu, Peng Shi, and Wenji Mao. MobileDreamer: Generative sketch world model for GUI agent. *arXiv preprint arXiv:2601.04035*, 2026.

Hyungjoo Chae, Namyoung Kim, Kai Ong, Minju Gwak, Gwanwoo Song, Jihoon Kim, Sunghwan Kim, Dongha Lee, and Jinyoung Yeo. Web agents with world models: Learning and leveraging environment dynamics in web navigation. In *International Conference on Learning Representations*, volume 2025, pp. 63707–63738, 2025.

Shiqi Chen, Tongyao Zhu, Zian Wang, Jinghan Zhang, Kangrui Wang, Siyang Gao, Teng Xiao, Yee Whye Teh, Junxian He, and Manling Li. Internalizing world models via self-play finetuning for agentic rl. *arXiv preprint arXiv:2510.15047*, 2025a.

Zijian Chen, Xueguang Ma, Shengyao Zhuang, Ping Nie, Kai Zou, Andrew Liu, Joshua Green, Kshama Patel, Ruoxi Meng, Mingyi Su, et al. Browsecomp-plus: A more fair and transparent evaluation benchmark of deep-research agent. *arXiv preprint arXiv:2508.06600*, 2025b.

Meng Chu, Xuan Billy Zhang, Kevin Qinghong Lin, Lingdong Kong, Jize Zhang, Teng Tu, Weijian Ma, Ziqi Huang, Senqiao Yang, Wei Huang, et al. Agentic world modeling: Foundations, capabilities, laws, and beyond. *arXiv preprint arXiv:2604.22748*, 2026.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

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.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Jiacai Liu, Zhuo Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. *arXiv preprint arXiv:2603.25562*, 2026.

Wei He, Yueqing Sun, Hongyan Hao, Xueyuan Hao, Zhikang Xia, Qi Gu, Hui Su, and Xunliang Cai. VitaBench: Benchmarking LLM agents with versatile interactive tasks in real-world applications. In *International Conference on Learning Representations*, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/93fab021315170101c92e8330a56fbdb-Abstract-Conference.html.

Yinghui He, Simran Kaur, Adithya Bhaskar, Yongjin Yang, Jiarui Liu, Narutatsu Ri, Liam Fowl, Abhishek Panigrahi, Danqi Chen, and Sanjeev Arora. Self-distillation zero: Self-revision turns binary rewards into dense supervision. *arXiv preprint arXiv:2604.12002*, 2026b.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6609–6625, 2020.

Wenjin Hou, Shangpin Peng, Weinong Wang, Zheng Ruan, Yue Zhang, Zhenglin Zhou, Mingqi Gao, Yifei Chen, Kaiqi Wang, Hongming Yang, et al. Uni-opd: Unifying on-policy distillation with a dual-perspective recipe. *arXiv preprint arXiv:2605.03677*, 2026.

Mengkang Hu, Tianxing Chen, Yude Zou, Yuheng Lei, Qiguang Chen, Ming Li, Yao Mu, Hongyuan Zhang, Wenqi Shao, and Ping Luo. Text2world: Benchmarking large language models for symbolic world model generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 26043–26066, 2025a.

Mengkang Hu, Bowei Xia, Yuran Wu, Ailing Yu, Yude Zou, Qiguang Chen, Shijian Wang, Jiarui Jin, Kexin Li, Wenxiang Jiao, et al. Agent2world: Learning to generate symbolic world models via adaptive multi-agent feedback. *arXiv preprint arXiv:2512.22336*, 2025b.

Ijun Jang, Jewon Yeom, Juan Yeo, Hyunggyu Lim, and Taesup Kim. Stable on-policy distillation through adaptive target reformulation. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 42217–42227. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.2094. URL https://aclanthology.org/2026.findings-acl.2094/.

Yuxuan Jiang, Runchao Li, Shubhashis Roy Dipta, Dawei Li, and Zhao Yang. Cornerstones or stumbling blocks? deciphering the rock tokens in on-policy distillation. *arXiv preprint arXiv:2605.09253*, 2026.

Woogyeol Jin, Taywon Min, Yongjin Yang, Dennis Wei, Yi Zhou, Swanand Ravindra Kadhe, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. *arXiv preprint arXiv:2603.07079*, 2026.

Junlong Ke, Zichen Wen, Weijia Li, Conghui He, and Linfeng Zhang. Respecting self-uncertainty in on-policy self-distillation for efficient llm reasoning. *arXiv preprint arXiv:2605.13255*, 2026.

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

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

Woosung Koh, Sungjun Han, Segyu Lee, Se-Young Yun, and Jamin Shin. Generative visual code mobile world models. *arXiv preprint arXiv:2602.01576*, 2026.

Rikuto Kotoge, Mai Nishimura, and Jiaxin Ma. Can compact language models search like agents? distillation-guided policy optimization for preserving agentic RAG capabilities. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 37733–37746. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1751. URL https://aclanthology.org/2026.acl-long.1751/.

Wolfgang Lehrach, Daniel Hennes, Miguel Lazaro-Gredilla, Xinghua Lou, Carter Wendelken, Zun Li, Antoine Dedieu, Marc Lanctot, Atil Iscen, John Schultz, Marcus Chiam, Ian Gemp, Piotr Zielinski, Satinder Singh, and Kevin Murphy. Code world models for general game playing. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/d8a12fde9e72444e1b356e8c37e53753-Abstract-Conference.html.

Guillaume Levy, Cédric Colas, Pierre-Yves Oudeyer, Thomas Carta, and Clément Romac. Worldllm: Improving llms' world modeling using curiosity-driven theory-making. *arXiv preprint arXiv:2506.06725*, 2025.

Gengsheng Li, Tianyu Yang, Junfeng Fang, Mingyang Song, Mao Zheng, Haiyun Guo, Dan Zhang, Jinqiao Wang, and Tat-Seng Chua. Unifying group-relative and self-distillation policy optimization via sample routing. *arXiv preprint arXiv:2604.02288*, 2026a.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang Gao, Wenkai Yang, Zhiyuan Liu, et al. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026b.

Yixia Li, Hongru Wang, Jiahao Qiu, Zhenfei Yin, Dongdong Zhang, Cheng Qian, Zeping Li, Xiaoteng Ma, Guanhua Chen, and Heng Ji. From word to world: Can large language models be implicit text-based world models? In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8084–8111, 2026c.

Zihan Liang, Yufei Ma, Ben Chen, Zhipeng Qian, Xuxin Zhang, Huangyu Dai, and Lingtao Mao. Search-e1: Self-distillation drives self-evolution in search-augmented reasoning. *arXiv preprint arXiv:2605.22511*, 2026.

Weiwen Liu, Xu Huang, Xingshan Zeng, Xinlong Hao, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, Zhengying Liu, Yuanqing Yu, Zezhong Wang, et al. Toolace: Winning the points of llm function calling. In *International Conference on Learning Representations*, volume 2025, pp. 41359–41381, 2025.

Yihong Liu, Raoyuan Zhao, Michael A Hedderich, and Hinrich Schütze. Crosslingual on-policy self-distillation for multilingual reasoning. *arXiv preprint arXiv:2605.09548*, 2026a.

Yizhou Liu, Qi Sun, Yulin Chen, Siyue Zhang, and Chen Zhao. Search, do not guess: Teaching small language models to be effective search agents. *arXiv preprint arXiv:2604.04651*, 2026b.

Ning Lu, Baijiong Lin, Shengcai Liu, Jiahao Wu, Haoze Lv, Yanbin Wei, Lingting Zhu, Shengju Qian, Xin Wang, Ying-Cong Chen, et al. Policy and world modeling co-training for language agents. *arXiv preprint arXiv:2606.02388*, 2026.

Shishir G Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E Gonzalez. The berkeley function calling leaderboard (bfcl): From tool use to agentic evaluation of large language models. In *Forty-second International Conference on Machine Learning*, 2025.

Emiliano Penaloza, Dheeraj Vattikonda, Nicolas Gontier, Alexandre Lacoste, Laurent Charlin, and Massimo Caccia. Privileged information distillation for language models. *arXiv preprint arXiv:2602.04942*, 2026.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 5687–5711, 2023.

Cheng Qian, Emre Can Acikgoz, Bingxuan Li, Xiusi Chen, Yuji Zhang, Bingxiang He, Qinyu Luo, Gokhan Tur, Dilek Hakkani-Tür, Yunzhu Li, et al. Current agents fail to leverage world model as tool for foresight. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13686–13723, 2026.

Guobin Shen, Xiang Cheng, Chenxiao Zhao, Lei Huang, Jindong Li, Dongcheng Zhao, and Xing Yu. Anti-self-distillation for reasoning rl via pointwise mutual information. *arXiv preprint arXiv:2605.11609*, 2026.

Vaishnavi Shrivastava, Piero Kauffmann, Ahmed Awadallah, and Dimitris Papailiopoulos. Echo: Terminal agents learn world models for free. *arXiv preprint arXiv:2605.24517*, 2026.

Mingyang Song and Mao Zheng. A survey of on-policy distillation for large language models. *arXiv preprint arXiv:2604.00626*, 2026.

Shuang Sun, Huatong Song, Lisheng Huang, Jinhao Jiang, Ran Le, Zhihao Lv, Zongchao Chen, Yiwen Hu, Wenyang Luo, Wayne Xin Zhao, et al. Swe-world: Building software engineering agents in docker-free environments. *arXiv preprint arXiv:2602.03419*, 2026.

Shicheng Tan, Weng Lam Tam, Yuanchun Wang, Wenwen Gong, Shu Zhao, Peng Zhang, and Jie Tang. Gkd: A general knowledge distillation framework for large-scale pre-trained language model. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pp. 134–148, 2023.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

Jianze Wang, Ying Liu, Jinlong Chen, Xuchun Hu, Qilong Zhang, Yu Cao, Jun Wang, Hua Yang, Yong Xie, and Qianglong Chen. Mad-opd: Breaking the ceiling in on-policy distillation via multi-agent debate. *arXiv preprint arXiv:2605.01347*, 2026a.

Jiaqi Wang, Wenhao Zhang, Weijie Shi, Yaliang Li, and James Cheng. TCOD: Exploring temporal curriculum in on-policy distillation for multi-turn autonomous agents. *arXiv preprint arXiv:2604.24005*, 2026b.

Jiaxuan Wang, Xuan Ouyang, Zhiyu Chen, Yulan Hu, Zheng Pan, Xin Li, and Lan-Zhe Guo. Trace: Distilling where it matters via token-routed self on-policy alignment. *arXiv preprint arXiv:2605.10194*, 2026c.

Kangrui Wang, Pingyue Zhang, Zihan Wang, Yaning Gao, Linjie Li, Qineng Wang, Hanyang Chen, Yiping Lu, Zhengyuan Yang, Lijuan Wang, et al. Vagen: Reinforcing world model reasoning for multi-turn vlm agents. *Advances in Neural Information Processing Systems*, 38:172871–172933, 2026d.

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

Zikai Xiao, Jianhong Tu, Chuhang Zou, Yuxin Zuo, Zhi Li, Peng Wang, Bowen Yu, Fei Huang, Junyang Lin, and Zuozhu Liu. Webworld: A large-scale world model for web agent training. *arXiv preprint arXiv:2602.14721*, 2026.

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

Sherry Yang. World models as an intermediary between agents and the real world. *arXiv preprint arXiv:2602.00785*, 2026.

Wenkai Yang, Weijie Liu, Ruobing Xie, Kai Yang, Saiyong Yang, and Yankai Lin. Learning beyond teacher: Generalized on-policy distillation with reward extrapolation. *arXiv preprint arXiv:2602.12125*, 2026.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 2369–2380, 2018.

Xiao Yu, Baolin Peng, Ruize Xu, Yelong Shen, Pengcheng He, Suman Nath, Nikhil Singh, Jiangfeng Gao, and Zhou Yu. Reinforcement world model learning for llm-based agents. *arXiv preprint arXiv:2602.05842*, 2026.

Kai Zhang, Xiangchao Chen, Bo Liu, Tianci Xue, Zeyi Liao, Zhihan Liu, Xiyao Wang, Yuting Ning, Zhaorun Chen, Xiaohan Fu, et al. Agent learning via early experience. *arXiv preprint arXiv:2510.08558*, 2025.

Wancong Zhang, Basile Terver, Artem Zholus, Soham Chitnis, Harsh Sutaria, Mido Assran, Randall Balestriero, Amir Bar, Adrien Bardes, Yann LeCun, et al. Hierarchical planning with latent world models. *arXiv preprint arXiv:2604.03208*, 2026a.

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

Hanyang Zhao, Haoxian Chen, Han Lin, Genta Indra Winata, David Yao, and Wenpin Tang. Opd+: Rethinking the advantage design for on-policy distillation. *arXiv preprint arXiv:2606.01039*, 2026a.

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

Yuhao Zheng, Li'an Zhong, Yi Wang, Rui Dai, Kaikui Liu, Xiangxiang Chu, Linyuan Lv, Philip Torr, and Kevin Qinghong Lin. Code2world: A gui world model via renderable code generation. *arXiv preprint arXiv:2602.09856*, 2026.

Qiyong Zhong, Mao Zheng, Mingyang Song, Xin Lin, Jie Sun, Houcheng Jiang, Xiang Wang, and Junfeng Fang. Sod: Step-wise on-policy distillation for small language model agents. *arXiv preprint arXiv:2605.07725*, 2026.

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*, 2026a.

Yuhang Zhou, Lizhu Zhang, Yifan Wu, Mingyi Wang, Bo Peng, Jiayi Liu, Xiangjun Fan, and Zhuokai Zhao. SAGE-OPD: Selective agent-guided intervention for multi-turn on-policy distillation. *arXiv preprint arXiv:2606.19659*, 2026b.

Yuxin Zuo, Zikai Xiao, Li Sheng, Fei Huang, Jianhong Tu, Yuxuan Liu, Tianyi Tang, Xiaomeng Hu, Yang Su, Qingfeng Lan, et al. Qwen-agentworld: Language world models for general agents. *arXiv preprint arXiv:2606.24597*, 2026.

A Experimental and World-Knowledge Details

This appendix provides implementation details for the divergence diagnostic in Section 3, the downstream evaluation in Section 5, and the task-specific world-knowledge formats used by WorldOPD. Additional Llama results are reported in Appendix B.

A.1 Downstream Evaluation

Models and tasks. The main experiments use Qwen3-series models (Yang et al., 2025). Unless otherwise specified, the students are Qwen3-0.6B, Qwen3-1.7B, and Qwen3-4B, and the teachers are Qwen3-14B and Qwen3-32B. For function calling, we train on ToolACE (Liu et al., 2025), follow MAD-OPD (Wang et al., 2026a), and evaluate on BFCL-v4, τ^2-Bench, and VitaBench. For multi-turn search, we follow Search-E1 (Liang et al., 2026), train on HotpotQA (Yang et al., 2018), and evaluate on HotpotQA, 2Wiki (Ho et al., 2020), Bamboogle (Press et al., 2023), MuSiQue (Trivedi et al., 2022), and BrowseComp-Plus (Chen et al., 2025b).

Compared variants. OPD uses standard action-level on-policy distillation. OPD$^+$ adds randomly sampled on-policy rollouts to match WorldOPD's distillation-token budget, as detailed in Appendix A.3. SAGE-OPD (Zhou et al., 2026b) uses selective turn-level intervention and teacher-confidence weighting; TCOD (Wang et al., 2026b) gradually increases rollout depth through a temporal curriculum. WorldOPD combines action OPD with state-conditioned WK distillation. The ablation WorldOPD$\backslash G$ applies WK distillation at every visited state, corresponding to $\mathcal{B}_{\mathrm{wk}} = \mathcal{B}$ in Eq. (6). Unless stated otherwise, rollout and optimization settings follow MAD-OPD for function calling and Search-E1 for search.

WorldOPD hyperparameters. Unless otherwise specified, we use $K = 8$ candidate action–state pairs per WK invocation, a WK loss weight of $\alpha = 0.2$, and a utility threshold of $\tau = 0.5$. The pair-count analysis in Section 5.4 varies K over $\{2, 4, 8\}$.

Evaluation metrics. Following Section 5.1, we report BFCL-v4 accuracy and task success rates on τ^2-Bench and VitaBench. For search, we use normalized EM on HotpotQA, 2Wiki, Bamboogle, and MuSiQue, and official-evaluator answer accuracy on BrowseComp-Plus. Scores are percentages.

A.2 Utility Predictor Initialization and Training

We initialize the utility predictor's backbone from Qwen3-0.6B and attach a prediction head that outputs a scalar estimate $\hat{U}_\phi(s)$. The predictor takes the interaction context $s = (h, o)$, including the task, prior action–observation history, and current observation. It is separate from the student policy.

The predictor is trained on the sparse probe targets $U(s)$ using the mean squared error in Eq. (5). Each visited state is independently sampled for probing with probability $\rho = 5\%$. At a sampled state, we measure the change in action divergence after adding teacher-generated WK to the student's context, keeping the student action prefixes fixed. The resulting utility is detached before updating the predictor; action and WK distillation losses update the student separately. Before generating auxiliary WK, we evaluate the predictor and invoke WK distillation only if $\hat{U}_\phi(s) > \tau$. Probe sampling remains independent of this decision. Each probe stores the interaction state and its detached utility target. As in Algorithm 1, the student is updated using $\mathcal{L}_{\mathrm{student}}$, and the predictor is updated separately using $\mathcal{L}_{\mathrm{utility}}$ when the probe collection is nonempty.

A.3 Token-Budget Matching for OPD$^+$

For each experimental configuration, we first run WorldOPD and record its total number of distillation tokens, including both action and WK tokens that receive teacher supervision. We compare this count, N_{WorldOPD}, with the corresponding standard OPD count, N_{OPD}, and use the difference $\Delta N = N_{\mathrm{WorldOPD}} - N_{\mathrm{OPD}}$ as the additional supervision budget for OPD$^+$.

OPD$^+$ uses this budget to randomly sample additional on-policy rollouts and applies standard action OPD to their student-generated actions. Additional rollouts are sampled from the OPD$^+$ student,

rather than reused from WorldOPD, and their supervised action tokens are counted toward ΔN. Thus, OPD$^+$ allocates the extra distillation budget to action supervision instead of WK supervision. The budget counts tokens that contribute to the student distillation loss; conditioning prompts, environment observations, and utility probes are not included. This matches supervised-token counts rather than total training compute, which also includes rollout generation and predictor-related costs.

A.4 Student–Teacher Divergence Diagnostic

The motivating analyses in Section 3 are diagnostic and do not update model parameters. We collect states through teacher-driven rollouts and measure teacher–student divergence along the same student-generated action prefixes, with and without the complete teacher-generated WK context. For function calling, this context includes both teacher-proposed actions and their predicted next states; neither component is removed for the diagnostic. We report two complementary views of these measurements. The aggregate analysis in Figure 2 reports the percentage relative KL reduction, $100\big(D_{\text{base}} - D_{\text{ctx}}\big)/D_{\text{base}}$, under fixed WK configurations. The state-conditioned analysis in Figure 3 instead plots the per-state log-ratio $R(s) = \log_2\big(D_{\text{base}}(s)/D_{\text{ctx}}(s)\big)$. Positive values are helpful and negative values are harmful; this sign agrees with the signed utility $U(s) = D_{\text{base}}(s) - D_{\text{ctx}}(s)$.

The offline diagnostic uses the teacher-driven state collection described below. During WorldOPD training, sparse probes instead operate at student-visited states and reuse the sampled rollout action a (Algorithm 1). Both use the forward-KL diagnostic in Eq. (1); action and WK training use the reverse-KL objectives in Section 4.

Models. For function calling, we evaluate Qwen3-0.6B, Qwen3-1.7B, and Qwen3-4B students with Qwen3-14B and Qwen3-32B teachers, depending on the student–teacher pair. For search, we evaluate Qwen3-0.6B and Qwen3-1.7B students with Qwen3-14B and Qwen3-32B teachers.

Function calling. We randomly sample 50 dialogues from the BFCL-v4 multi-turn base split and execute them in the live BFCL environment. The teacher drives each rollout: it greedily generates a tool call, the environment executes it, and the returned observation is appended to the history. We roll out for up to three steps per turn and six turns per dialogue, producing 216 measurement states for the 14B teacher and 248 for the 32B teacher. At each state, the student greedily generates an action without WK. We then compute both divergences on that fixed action. The diagnostic varies the fixed amount of tool-related WK over $K \in \{2, 4, 8, 16\}$; this is an analysis of WK content, not a learned per-state allocation rule.

Search. The main-text search diagnostic reports HotpotQA. We sample 50 distractor-validation questions, each paired with ten candidate Wikipedia paragraphs. The agent emits either `Search: <query>` or `Answer: <final answer>`; a BM25 retriever returns the top passages for each query. Teacher-driven rollouts yield approximately 123 measurement states per student–teacher pair on HotpotQA. At every state, only the student receives the teacher-generated search plan in the WK condition, while the teacher distribution is evaluated from the original state.

A.5 Prompts for World-Knowledge Generation

We provide the WK formats and task prompts for both the auxiliary distillation branch and the contextual diagnostic. In WK OPD, the prompt is appended to state s and the student generates a WK sequence w^S; the teacher provides next-token distributions along the student's WK prefixes. Both candidate-action and predicted-state tokens receive the auxiliary loss. Teacher and student distributions are evaluated along the same complete student-generated prefixes. In the preliminary diagnostic and sparse probes, the teacher instead generates WK w^T that is inserted into the student's context solely to measure action divergence on fixed action prefixes. This diagnostic context is not used to generate the executed rollout action.

The task configuration fixes the pair count K for each run. The preliminary pair-count analysis varies $K \in \{2, 4, 8, 16\}$, while the downstream hyperparameter analysis uses $K \in \{2, 4, 8\}$; neither uses the gate to change K within a run.

In Eq. (4), the loss is averaged over the generated WK sequence w^S, including both candidate-action and state-description tokens. There is no state-only loss mask. Prompt tokens are conditioning context and are outside the generated sequence. The probe collection $\mathcal{P}$ and rollout collection $\mathcal{B}$ retain repeated visits to a state. Each state includes the task and past interactions together with the current observation. Student and predictor parameters are optimized separately: action and WK losses update the student, and detached probe targets train the utility predictor. The predictor update is skipped when no probes are sampled.

Tool calling. For tool-calling tasks, world knowledge is represented as a set of candidate action–state pairs. In auxiliary WK OPD, the student proposes the actions and predicts their resulting states. The teacher supervises both parts of the generated pairs. In the contextual diagnostic, the teacher generates the complete pairs used as probe context. The following block illustrates the pair format; both Action and Resulting state fields contribute to the auxiliary WK loss:

Tool-calling world knowledge

```
For reference, here are some possible next actions and the
↪  environment
state each would lead to:

1. Action: <action 1>
   Resulting state: <state 1>

2. Action: <action 2>
   Resulting state: <state 2>

...

K. Action: <action K>
   Resulting state: <state K>
```

Search. The generation prompt asks for the information already obtained, what is still missing, and candidate searches with their expected outcomes, without revealing the final answer. It is used for student WK generation and teacher diagnostic context generation. Auxiliary OPD supervises the complete student-generated WK response, including candidate search actions and their expected returns and downstream effects. The prompt below specifies this output structure:

Search world knowledge

```
Output a WORLD MODEL of the task in two parts:
(1) WHAT WE HAVE NOW -- facts already confirmed so far, and
the unknowns still blocking the answer.
(2) PROPOSED ACTIONS AND THEIR DOWNSTREAM IMPACT -- propose K
distinct candidate search actions.
For EACH action, keep the expected return to a brief phrase,
and focus on the IMPACT: how that result would change the
REMAINING search, including which unknown it resolves, what
the NEXT step becomes once the result is available, and what
it enables or rules out.
Use exactly the following format:
Action k:  <search query>
Returns (brief):  <a brief note of what it returns >
Impact on remaining plan:  <how it unblocks or reshapes the
next steps>
```

A.6 Illustration of World Knowledge

Consider a tool-calling task in which a user wants to return a delivered item. Action reasoning may explain why the agent should look up the order. WK explicitly describes candidate actions and

their anticipated consequences: under a policy that disallows canceling delivered orders but permits returns, a cancellation request would be rejected, whereas a return request for an eligible item would initiate the return process. This is an illustrative example, not a recorded rollout. Reasoning can also discuss such consequences; WK supervision makes the candidate action–state structure explicit and distills both components.

B Search Results with Llama Models

Table 3 evaluates WorldOPD with a Llama-3.2-1B-Instruct student and a Llama-3.1-8B-Instruct-based teacher. Student, teacher, PPO, KD, and DGPO scores are taken from Kotoge et al. (2026); WorldOPD scores are from our experiments. DGPO combines KD initialization with PPO and selective teacher KL guidance.

Table 3: Llama search results (1B student, 8B teacher). Scores are exact match percentages; Avg. is the mean over the four reported benchmarks.

Method	TriviaQA	HotpotQA	2Wiki	Bamboogle	Avg.
Student-1B	8.0	2.7	4.2	2.4	4.3
Teacher-8B	64.7	42.7	44.3	44.4	49.0
PPO	49.9	22.2	18.1	6.5	24.2
KD	50.8	36.9	35.5	26.6	37.5
DGPO	55.3	41.2	37.9	33.9	42.1
WorldOPD	56.2	44.2	39.2	36.1	43.9