AVOIDING WHAT YOUR TEACHER ALREADY KNOWS: ENTITY FAMILIARITY FOR DEEP-SEARCH DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep-search agents are commonly trained by distilling tool-use trajectories from stronger teacher models. However, structurally complex questions may remain relatively easy for teachers familiar with key entities in the underlying entity graph, resulting in lower search demand and weaker supervision. We refer to this teacher-specific variation in question difficulty as *teacher-relative difficulty* and investigate entity familiarity as an early predictor. We first introduce Evidence-Verified Entity Familiarity (**EVF**), which estimates familiarity by verifying a teacher-generated factual brief against online evidence. Across multiple teachers, EVF is significantly correlated with teacher-relative difficulty with AUCs of 0.60–0.70. However, its high cost makes it impractical for constructing entity graphs at scale. To address this limitation, we propose Self-Reported Entity Familiarity (**SEF**), a lightweight gating signal that directly asks a teacher whether it is familiar with an entity before admitting that entity into the graph. Across multiple teachers, trajectories from SEF-gated synthesis exhibit greater observed search demand and yield higher downstream student accuracy than those from ungated synthesis. Because familiarity is teacher specific, we further study cross-teacher reuse. We screen graphs collected for other teachers using Kimi-K3's SEF judgments and use the retained graphs to produce 5,411 additional trajectories. Combined with 2,200 trajectories from newly constructed, SEF-gated graphs, these data improve Qwen3.8-27B's accuracy from 44.10% to 66.41% on a BrowseComp subset. Overall, SEF provides a low-cost way to synthesize more informative supervision for deep-search distillation.

1 INTRODUCTION

Deep search requires a language model to interact with the open web, retrieving, reading, and integrating evidence before committing to an answer. Since its introduction in 2025, BrowseComp, a challenging benchmark for deep search, has rapidly become a standard evaluation for frontier models with web-search capabilities, with the state-of-the-art Kimi-K3 achieving 91.2% accuracy (Wei et al., 2025; Tongyi DeepResearch Team et al., 2025; DeepSeek-AI et al., 2025; Kimi Team et al., 2026). Two common settings call for transferring this capability: distillation into a much smaller student, which reduces inference cost and enables local deployment (Xu et al., 2024), and the synthesis of training data for subsequent model generations (Abdin et al., 2024; Grattafiori et al., 2024). In both settings, a synthesis system constructs question–answer pairs, and the teacher solves the questions with search tools to produce trajectories as supervision (Li et al., 2025; Xia et al., 2025). For this supervision to be effective, the trajectories should exhibit substantive information acquisition and search reasoning.

A natural way to elicit stronger search behavior is to synthesize more difficult questions. Commonly used graph-based question–answer synthesis can increase the number of hops, constraints, and compositional steps, but these properties do not necessarily imply high search demand for a particular teacher. If the teacher already knows the relevant entities, relations, or likely answers, it may narrow the search space and recognize supporting evidence with little exploration. The resulting trajectories can therefore contain fewer meaningful search decisions and less information acquisition, providing weaker supervision for a student that may not share the teacher's prior knowledge. Indeed, recent

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.

work reports that models can answer some BrowseComp questions without tool access (Fan et al., 2026). We use *teacher-relative difficulty* to describe how much information a particular teacher must still obtain externally to solve a question. We use question-level closed-book correctness as a post-hoc measure of this quantity: if a teacher can answer a completed question without search tools, the question has low teacher-relative difficulty for that teacher. However, this signal is available only after the costly question synthesis, thus motivating a low-cost predictor that can be applied during question synthesis.

We hypothesize that *entities less familiar to the teacher are more likely to yield questions with high teacher-relative difficulty*. We first test this hypothesis with *Evidence-Verified Entity Familiarity* (EVF), which estimates familiarity by asking the teacher to produce a factual brief about an entity and verifying its claims against online evidence. Across three teacher-specific question pools, EVF measured on seed entities predicts the teacher's closed-book correctness with AUCs of 0.60–0.70; lower scores correspond to questions less likely to be answered correctly without tools. EVF, however, requires a factual brief, web search, and claim verification for each entity, making repeated screening during graph construction costly. We observe that within the low-EVF region, teachers often report being unfamiliar with the corresponding seed entity. This motivates examining whether a direct familiarity judgment can serve as a cheaper screening signal.

We therefore introduce *Self-Reported Entity Familiarity* (SEF), which reduces familiarity estimation to a single tool-free binary query asking whether the teacher is familiar with an entity. Unlike EVF, SEF requires neither search nor evidence verification, making it inexpensive enough to apply during graph construction. We apply SEF at two entity-admission points of the graph-based synthesis pipeline: seed selection and next-hop expansion. *SEF-gated synthesis* admits only entities that the teacher reports as unfamiliar, whereas *ungated synthesis* omits this screening; all other stages are held fixed. Relative to ungated synthesis, trajectories produced by SEF-gated synthesis for GLM-5.1 (Z.ai, 2026), DeepSeek-V4-Flash (DeepSeek-AI et al., 2026), and Kimi-K3 contain more search turns, reasoning tokens, and tool calls, together with lower tool-assisted solve accuracy, a pattern consistent with higher observed search demand. In the corresponding equal-sized distillation comparisons, students trained on SEF-gated trajectories show gains ranging from 6.15 to 8.46 percentage points over those trained on ungated trajectories.

Because SEF is teacher-specific, a natural question is whether data collected for one teacher can still benefit another. On a shared pool of 5,995 entities, pairwise SEF agreement across the three teachers is 84.4–89.9%. Thus, different teachers share substantial overlap in entity familiarity, but their judgments are not identical. We therefore reuse the collected entity graphs and obtain new SEF judgments for the target teacher. We then resynthesize the question–answer pairs and rerun the target teacher to collect new search trajectories. With Kimi-K3 as the target, this process yields 5,411 trajectories from graphs collected for the other teachers. Combined with 2,200 trajectories generated through Kimi-K3 SEF-gated synthesis, the resulting 7,611 trajectories improve the accuracy of Qwen3.8-27B (Qwen Team, 2026) from 44.10% to 66.41% on the evaluated BrowseComp subset.

Our contributions are threefold:

- We formulate deep-search data difficulty in teacher-relative terms and show that entity familiarity is predictive of whether a teacher can answer a question without search. We study this relationship through both evidence-verified familiarity (EVF) and the much cheaper self-reported entity familiarity (SEF).
- We introduce SEF-gated entity admission as an early intervention in graph-based question–answer synthesis. Across three teacher settings, trajectories from SEF-gated synthesis show higher observed search demand than those from ungated synthesis, and the corresponding students score higher by $+8.20$, $+6.15$, and $+8.46$ percentage points in the evaluated comparisons.
- We study how teacher-specific familiarity affects data reuse across teachers. By reusing previously collected entity graphs to expand Kimi-K3-specific training data, we train a Qwen3.8-27B student that achieves 66.41% accuracy on the BrowseComp subset.

2 Problem Setup and Study Design

We study teacher-specific entity familiarity in the setting of deep-search distillation. Our experiments build on an existing graph-based question–answer synthesis pipeline. This section introduces

the distillation setting, describes the graph-based synthesis pipeline and our intervention points, and outlines the research questions investigated in the following sections.

2.1 Deep-Search Distillation Setting

We consider distillation from a search-enabled teacher model M_T to a student model M_S. Following ReAct-style search agents such as OpenSeeker (Du et al., 2026), the teacher solves a question q by producing a trajectory

$$\tau_T(q) = \big[(r_1, a_1, o_1), \ldots, (r_m, a_m, o_m), \hat{y}\big], \tag{1}$$

where r_t, a_t, and o_t denote reasoning, tool action, and observation at step t, respectively, and $\hat{y}$ is the final answer. A synthesis pipeline produces

$$\mathcal{D}_T = \{(q_i, \tau_T(q_i))\}_{i=1}^{N}, \tag{2}$$

which is used to train the student with supervised fine-tuning on the teacher trajectories.

We distinguish structural complexity from teacher-relative difficulty. Structural complexity concerns properties such as the number of hops or constraints in a question, whereas teacher-relative difficulty concerns how much information a particular teacher must still acquire externally to solve it. A structurally complex question may therefore remain easy for a teacher already familiar with its key entities. In this work, teacher-relative difficulty is treated as a latent task property and studied through observable proxies such as closed-book answerability and search behavior. Note that we define closed-book correctness as whether the model answers a question correctly without tools.

2.2 Graph-Based Synthesis Pipeline

We use a graph-based synthesis pipeline adapted from OffSeeker (Zhou et al., 2026). The synthesis process begins by retrieving webpages from diverse topics and extracting seed entities. Let e denote an entity encountered during synthesis. Starting from a seed entity e_0, a tool-augmented agent searches the web, opens relevant pages, and collects factual properties of the current entity together with evidence-supported relations to other entities. The agent proposes related entities as candidates for the next hop, selects one, and repeats the exploration for multiple hops. This process constructs an evidence graph and a corresponding transition path,

$$G = (V, R), \qquad P = (e_0, \ldots, e_h), \quad e_j \in V, \quad (e_{j-1}, e_j) \in R, \tag{3}$$

where V is the set of explored entities, R is the set of relations supported by retrieved web evidence, and h is the number of transitions in the exploration path. The resulting graph records the properties and relations collected for the entities along this process. It is then provided to a question-generation model, which constructs a multi-hop question–answer pair (q, y^*) from the collected evidence. A subsequent difficulty-enhancement step removes overly explicit clues while preserving the reference answer. The resulting question is then solved by the teacher with search tools, producing a search trajectory whose final answer is judged before conversion to the training format.

Hereafter, e_0 is the *seed entity*. A related entity considered during expansion is a *candidate next-hop entity*; once selected, it becomes a *next-hop entity* on P. These are the two entity-admission points at which SEF can intervene. We refer to synthesis without familiarity screening as *ungated synthesis*, and to the variant that admits only entities reported as unfamiliar as *SEF-gated synthesis*. The two variants are otherwise identical.

2.3 Study Design

Building on the two entity-selection points described above, we organize our empirical study around three research questions (RQs):

- **RQ1: Signal.** Is familiarity with an entity e predictive of teacher-relative difficulty, as reflected in question-level closed-book answerability?
- **RQ2: Intervention.** Does excluding entities that the teacher reports as familiar increase observed search demand and improve downstream student performance?
- **RQ3: Reuse and scaling.** Can previously collected evidence graphs be reused with a new teacher to expand the deep-search training data generated by that teacher?

3 ENTITY FAMILIARITY AS A SIGNAL OF TEACHER-RELATIVE DIFFICULTY

This section investigates whether teacher-specific entity familiarity can provide an early signal of teacher-relative difficulty, addressing RQ1. We first introduce Evidence-Verified Entity Familiarity (EVF), an evidence-backed continuous estimator, and test whether questions with more familiar seed entities are easier for the teacher to answer without search. Based on the EVF analysis, we further introduce Self-Reported Entity Familiarity (SEF), a simpler binary self-report designed to serve a similar entity-screening role at substantially lower cost. Finally, we compare the operational costs of EVF and SEF to assess their suitability for repeated use during data synthesis.

3.1 EVF TRACKS CLOSED-BOOK ANSWERABILITY

We test whether a model's familiarity with a seed entity is associated with its teacher-relative difficulty on the corresponding question. We operationalize entity familiarity using Evidence-Verified Entity Familiarity (EVF) and teacher-relative difficulty using closed-book correctness. Closed-book accuracy is the mean correctness over a set of questions. Our analysis covers GLM-5.1, DeepSeek-V4-Flash, and Kimi-K3, with 300 questions evaluated for each model.

For each question, let e_0 denote its *seed entity*, the first entity on the exploration path. Model M_T first produces a factual brief about e_0 without tools. The same model, now operating with search access, retrieves external evidence and verifies the claims in the brief. Let $\mathcal{C}_{M_T}(e_0)$ denote the set of generated factual claims, and let $z(c) \in \{0, 1\}$ indicate whether claim c is verified by the retrieved evidence. We define EVF as

$$\mathrm{EVF}_{M_T}(e_0) = \frac{1}{|\mathcal{C}_{M_T}(e_0)|} \sum_{c \in \mathcal{C}_{M_T}(e_0)} z(c). \tag{4}$$

EVF measures the fraction of the model-generated claims that can be verified against external evidence, with higher values indicating greater estimated familiarity. To measure teacher-relative difficulty, we ask the model to answer the corresponding question without tools and record whether its final answer is correct. Lower closed-book correctness indicates greater teacher-relative difficulty. Kimi-K3 judges final-answer correctness in all three model settings.

Table 1: Association between EVF and closed-book (CB) correctness. Failures are counted as incorrect. Low and high EVF denote $\mathrm{EVF} < 0.25$ and $\mathrm{EVF} \in [0.75, 1]$, respectively.

Teacher	EVF AUC	ρ (p)	Low-EVF CB acc.	High-EVF CB acc.
GLM-5.1	.600	.174 (.003)	28.41%	45.63%
DeepSeek-V4-Flash	.620	.204 (<.001)	26.67%	46.40%
Kimi-K3	.706	.351 (<.001)	46.05%	80.98%

Table 1 shows that EVF is positively associated with closed-book correctness across all three teachers, with AUCs of .600–.706 and significant positive correlations. Questions with low-EVF seed entities are also consistently less answerable without tools than those with high-EVF seed entities. We use $\mathrm{EVF} < 0.25$ as a fixed operational definition of low familiarity; because EVF depends on both generated claims and retrieved evidence, we treat it as a diagnostic proxy rather than ground-truth model knowledge.

3.2 SEF PROVIDES A HIGH-RECALL LOW-FAMILIARITY GATE

EVF requires factual generation, web search, and evidence verification, making it too expensive for repeated use during graph construction. We therefore ask whether a simpler and less costly signal can retain most low-EVF entities while reducing the candidate pool before synthesis.

Our experiments reveal a useful pattern: within the low-EVF region, teachers often report being unfamiliar with the corresponding seed entity. This observation motivates a lightweight binary gate for graph construction, which we call Self-Reported Entity Familiarity (SEF):

$$\mathrm{SEF}_{M_T}(e) \in \{\textsc{yes}, \textsc{no}\}, \tag{5}$$

where YES and NO indicate that the teacher reports being familiar or unfamiliar with entity e, respectively. SEF requires one tool-free teacher call per entity. We compare closed-book accuracy by the seed entity's SEF judgment, then evaluate its screening behavior through low-EVF retention, high-EVF exclusion, and changes in pool composition.

Table 2: Closed-book (CB) accuracy by seed-entity SEF judgment and SEF screening of seed entities. Low and high EVF denote $\mathrm{EVF} < 0.25$ and $\mathrm{EVF} \in [0.75, 1]$, respectively. Retention is the fraction of low-EVF entities judged NO; exclusion is the fraction of high-EVF entities judged YES. "Before" refers to all evaluated seed entities and "after" to the subset judged NO.

Teacher	CB accuracy by seed SEF		SEF screening relative to EVF			
	YES	NO	Low-EVF retention (%)	High-EVF exclusion (%)	Low-EVF share (%) Before → after	High-EVF share (%) Before → after
GLM-5.1	44.06%	35.81%	96.59%	71.70%	30.14% → 55.19%	54.45% → 29.22%
DeepSeek-V4-Flash	44.29%	27.04%	97.14%	81.60%	35.84% → 65.38%	42.66% → 14.74%
Kimi-K3	84.24%	57.04%	94.74%	71.71%	26.21% → 53.73%	70.69% → 43.28%

Table 2 shows that questions with NO seed entities have lower closed-book accuracy across all three teachers. SEF also retains nearly all low-EVF seed entities while excluding most high-EVF entities. The retained pool has a higher low-EVF share and a lower high-EVF share, although a substantial high-EVF fraction remains for Kimi-K3. Although SEF is not a precise measure of teacher knowledge, it provides an informative signal of teacher-specific entity familiarity.

3.3 SEF Enables Low-Cost Early Screening

Table 3 compares the measured decision costs of SEF and EVF. SEF requires one tool-free LLM call per entity, whereas EVF must generate a factual brief, search, and verify evidence.

Table 3: Decision cost of familiarity assessment. Calls are mean LLM/tool calls per assessed entity, and wall-clock times are measured within each deployment setting.

Teacher	Method	LLM calls	Tool calls	Mean wall time
GLM-5.1	SEF	1.00	0.00	5.28 s
GLM-5.1	EVF	10.86	14.27	107.83 s
Kimi-K3	SEF	1.00	0.00	24.32 s
Kimi-K3	EVF	8.35	10.45	224.07 s

Table 3 shows that SEF requires far fewer LLM and tool calls than EVF, with mean wall-clock time lower by roughly 9–20$\times$ in the measured deployment settings. This low per-decision cost makes repeated screening at entity-admission points practical during graph construction.

Answer to RQ1 (Signal). Entity familiarity is an informative early signal of teacher-relative difficulty: questions built from less familiar entities are less likely to be answered without search. A single direct familiarity judgment provides a practical gate for selecting entities during entity-graph construction.

4 SEF-Gated Synthesis and Distillation

We evaluate whether using SEF to admit entities changes the teacher's search behavior and produces more useful training data. Figure 1 summarizes the pipeline: the target teacher screens candidate seeds before graph construction and candidate next-hop entities during graph expansion, while the downstream question-generation, difficulty-enhancement, and teacher-search stages remain unchanged from ungated synthesis. We then measure search demand in teacher trajectories and compare student performance under equal-sized training sets.

4.1 SEF-Gated Entity Admission

Section 3 evaluates EVF and SEF only on seed entities e_0. Nevertheless, a multi-hop question also draws on later entities along its exploration path. An unfamiliar seed entity alone therefore does not

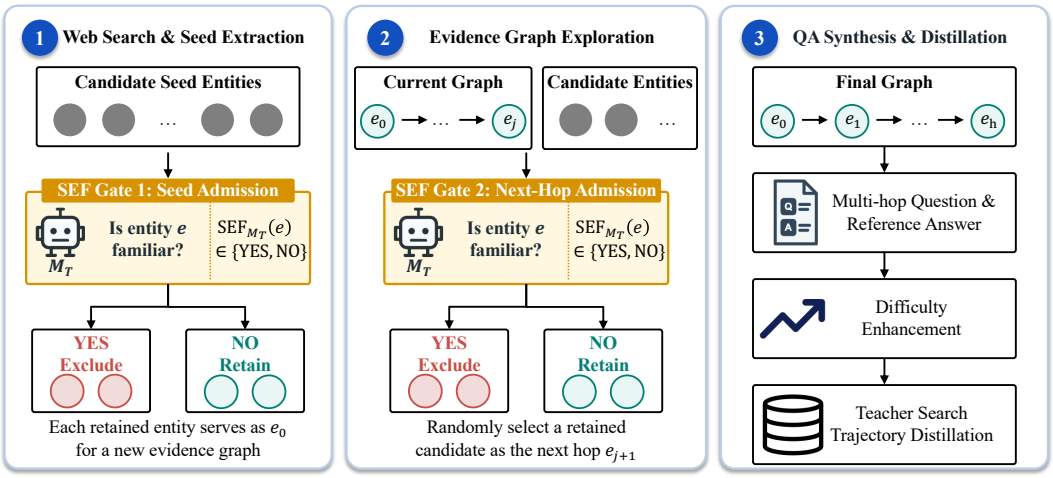

Figure 1: SEF-gated graph-based synthesis and trajectory collection. The target teacher M_T screens candidate seed and next-hop entities, retaining only those it reports as unfamiliar ($\text{SEF}_{M_T}(e) = \text{NO}$). Retained seeds initialize evidence graphs, and next hops are sampled from retained candidates. The completed graph supports multi-hop question–answer generation and difficulty enhancement, followed by tool-assisted teacher search to collect trajectories for distillation.

ensure that all entities supporting the question are unfamiliar to the teacher. Accordingly, SEF-gated synthesis applies SEF at both entity-admission points: before graph construction for seed entities and during expansion for candidate next-hop entities.

At expansion step k, let $\mathcal{C}_k$ be the candidate next-hop entities. Before sampling the next-hop entity, we form an admissible set containing only those that the target teacher M_T reports as unfamiliar:

$$\mathcal{C}_k^- = \{e \in \mathcal{C}_k : \text{SEF}_{M_T}(e) = \text{NO}\}, \qquad e_{k+1} \sim \text{Uniform}(\mathcal{C}_k^-). \tag{6}$$

If $\mathcal{C}_k^-$ is empty, expansion of that branch stops and the partial graph is preserved. A teacher-specific cache reuses earlier SEF judgments whenever an entity is encountered again.

Ungated and SEF-gated synthesis follow the same graph-based pipeline described in Section 2.2; they differ only in entity admission.

4.2 SEF Gating Increases Observed Search Demand

We compare ungated and SEF-gated synthesis for GLM-5.1, DeepSeek-V4-Flash, and Kimi-K3, yielding three teacher-specific comparisons and six trajectory pools. Each pool contains more than 2,000 trajectories. For DeepSeek-V4-Flash and Kimi-K3, whose reasoning effort can be configured, we set `reasoning_effort` to `high`. Kimi-K3 judges whether each trajectory's final answer is correct. For each full trajectory pool, we report mean LLM turns, reasoning tokens, and structured tool calls per trajectory, along with tool-assisted solve accuracy.

Table 4: Teacher trajectory statistics for ungated → SEF-gated synthesis. All per-trajectory means and solve accuracies use the complete raw trajectory pools.

Teacher	LLM turns	Reasoning tokens	Tool calls	Solve acc. (%)
GLM-5.1	24.01 → 32.32	19,587 → 23,169	35.05 → 48.72	76.89 → 64.87
DeepSeek-V4-Flash	39.20 → 40.12	20,846 → 21,857	43.78 → 45.05	76.96 → 71.46
Kimi-K3	25.50 → 31.00	4,700 → 5,211	65.04 → 77.66	96.35 → 95.43

Table 4 reports ungated → SEF-gated results on the full teacher pools. All nine search-demand measures increase: LLM turns by 2.3–34.6%, reasoning tokens by 4.8–18.3%, and tool calls by 2.9–39.0%. Tool-assisted solve accuracy falls by 12.02 and 5.50 percentage points for GLM-5.1 and DeepSeek-V4-Flash, respectively. Kimi-K3 makes 21.6% more turns and 19.4% more tool

calls, while its accuracy changes by only -0.92 points, showing that greater search demand can emerge even when a strong teacher largely maintains its success rate. Together, these differences are consistent with greater teacher-relative difficulty for questions from SEF-gated synthesis.

4.3 SEF-Gated Trajectories Yield Higher Student Accuracy

The teacher-side statistics characterize search demand but do not determine the training value of the resulting data. We therefore evaluate downstream utility separately under equal-sized training sets, holding the number of trajectories fixed within each teacher setting. For GLM-5.1 and DeepSeek-V4-Flash, we train Qwen3-30B-A3B-Thinking on 1,490 trajectories from each synthesis variant per teacher. For Kimi-K3, we train Qwen3.8-27B on 2,200 trajectories from each synthesis variant. We evaluate all students on the 130-question BrowseComp subset selected by QUEST (Xie et al., 2026). Each condition is run independently three times with a maximum context length of 256k, and we report the mean accuracy and sample standard deviation across runs. We use the same tool configuration and tool-call definitions as OpenResearcher and use Kimi-K3 to judge answer correctness. *To reduce benchmark-answer leakage, the search tools filter Hugging Face URLs from search results and block direct access to those pages.*

Table 5: Student accuracy on the 130-question BrowseComp subset selected by QUEST. Except for the approximate Qwen3-30B-A3B-Thinking baseline, values are mean $\pm$ sample standard deviation over three independent runs.

Trajectory teacher	Student	Before SFT	Ungated	SEF-gated	Gain (pp)
GLM-5.1	Qwen3-30B-A3B-Thinking	$\sim$2.5%	$33.08 \pm 0.77\%$	$41.28 \pm 1.94\%$	+8.20
DeepSeek-V4-Flash	Qwen3-30B-A3B-Thinking	$\sim$2.5%	$25.90 \pm 2.91\%$	$32.05 \pm 1.18\%$	+6.15
Kimi-K3	Qwen3.8-27B	$44.10 \pm 2.47\%$	$55.13 \pm 4.37\%$	$63.59 \pm 1.18\%$	+8.46

Table 5 shows that both synthesis variants improve mean accuracy over the corresponding before-SFT student. With the number of training trajectories held fixed, SEF-gated trajectories yield mean accuracy gains over ungated trajectories of 8.20, 6.15, and 8.46 percentage points for GLM-5.1, DeepSeek-V4-Flash, and Kimi-K3, respectively. These results favor SEF gating across three trajectory teachers and two student backbones. Together with the teacher-side results, they indicate greater search demand during synthesis and stronger student performance when trained on the same number of trajectories.

Answer to RQ2 (Intervention). Across the evaluated settings, excluding entities already familiar to the teacher yields higher observed search demand, while equal-sized SEF-gated trajectory sets train students with higher accuracy than the corresponding ungated sets. These parallel outcomes support entity familiarity as a practical control for data synthesis.

5 Cross-Teacher Adaptation through Graph Reuse

SEF judgments are teacher-specific, raising a practical concern: adapting to each new teacher by exploring evidence graphs from scratch with SEF gating would repeat the costly work of entity discovery and evidence collection. In this section, we therefore investigate whether existing evidence graphs can be reused.

5.1 SEF Judgments Overlap but Require Target-Specific Reassessment

Table 6: Pairwise SEF judgments on the shared 5,995-entity pool.

Teacher pair (A / B)	Both unfamiliar	A familiar B unfamiliar	A unfamiliar B familiar	Agreement
GLM-5.1 / DeepSeek-V4-Flash	29.6%	9.6%	6.0%	84.4%
GLM-5.1 / Kimi-K3	32.8%	7.3%	2.8%	89.9%
DeepSeek-V4-Flash / Kimi-K3	32.2%	8.0%	7.2%	84.8%

Table 6 compares SEF judgments by GLM-5.1, DeepSeek-V4-Flash, and Kimi-K3 on entities sampled from three teacher-specific pools (2,000 per pool). Pairwise agreement is high (84.4–89.9%),

but directional disagreements still account for 10.1–15.6% across teacher pairs. These differences mean that a source teacher's SEF decision cannot be transferred directly: the target teacher must reassess candidate paths before reuse.

5.2 Reuse Evidence Graphs, Regenerate Target-Specific Supervision

We reuse only the evidence graph—including its entities, evidence-backed relations, retrieved evidence, and candidate paths—rather than the source teacher's trajectories. For a candidate path $P = (e_0, \ldots, e_h)$ and target teacher M_T, we accept the path only if all of its entities are judged unfamiliar:

$$\mathrm{Accept}_{M_T}(P) = \mathbf{1}[\forall e \in P,\ \mathrm{SEF}_{M_T}(e) = \textsc{no}]\,. \tag{7}$$

Accepted paths are then used to resynthesize question–answer pairs and collect new search trajectories from the target teacher. Graph reuse therefore preserves prior entity-discovery and evidence-collection work, while familiarity screening, question synthesis, and trajectory generation remain target-specific.

5.3 Scaling Target-Specific Supervision with Reused Graphs

We use Kimi-K3 as the target teacher. Starting from the 2,200 SEF-gated trajectories of Kimi-K3 in Section 4, we load 7,143 evidence graphs collected for the other teachers and reapply SEF with Kimi-K3 to their candidate paths. Only paths whose entities pass the Kimi-K3-specific SEF criterion in Eq. 7 are retained for question resynthesis and trajectory generation. This target-specific screening yields 5,411 additional Kimi-K3 trajectories, expanding the training set to 7,611 trajectories.

We evaluate Qwen3.8-27B on the 130-question BrowseComp subset selected by QUEST (Xie et al., 2026) with a 256k context, reporting mean $\pm$ sample standard deviation over three independent runs. With 2,200 native SEF-gated trajectories, the student reaches $63.59 \pm 1.18\%$ accuracy; adding 5,411 trajectories generated from reused graphs raises accuracy to $66.41 \pm 2.35\%$, a 2.82-percentage-point gain. This higher mean suggests downstream value from expanding training data through graph reuse, without isolating reuse from the effect of additional training examples.

Answer to RQ3 (Reuse and Scaling). Evidence graphs can transfer across teachers after target-specific SEF reassessment, expanding training data. In this data-expansion setting, the additional trajectories are associated with higher mean student accuracy.

Table 7 summarizes results on BrowseComp (BC), BrowseComp-ZH (BC-ZH) (Zhou et al., 2025), SEAL-0 (Pham et al., 2026), the text-only subset of Humanity's Last Exam with tools (Phan et al., 2025), and xbench-DeepSearch-2510 (xbench Team, 2025). We evaluate on the full evaluation set of each benchmark, using the text-only subset for HLE. For BC and BC-ZH, we report results both with a ReAct setup and with context management.

Table 7: Comparison of deep research agents across five benchmarks.

Method	BC	BC-ZH	SEAL-0	HLE text + tools	xbench 2510
OpenAI Deep Research (Wei et al., 2025; Zhou et al., 2025)	51.5	42.9	–	–	–
Tongyi DeepResearch 30B-A3B (Tongyi DeepResearch Team et al., 2025)	$43.4/58.3^{*}$	$46.7/58.1^{*}$	–	$32.9^{\dagger}/38.3^{*\dagger}$	55.0
OffSeeker-8B-DPO (Zhou et al., 2026)	12.8	26.6	–	13.8	–
OpenResearcher-30B-A3B (Li et al., 2026)	26.3	–	–	19.6^{*}	–
QUEST-35B (Xie et al., 2026)	$45.5^{*\dagger}/64.6^{*}$	–	–	37.2^{*}	–
Apodex-1.0-mini (Apodex Team, 2026)	71.5^{*}	**80.6***	–	46.8	–
FORT-Searcher (Deng et al., 2026)	$55.9/72.2^{*}$	$62.1/75.0^{*}$	$43.7/46.0^{*}$	–	$54.1/57.2^{*}$
DeepSearch-World-9B (Geng et al., 2026)	31.2^{*}	36.4^{*}	–	25.7^{*}	–
AREX-Turbo (Lu et al., 2026)	70.7^{*}	–	–	40.6^{*}	57.0^{*}
Ours (Qwen3.8-27B)	$66.2/$**74.3***	$64.7/69.6^{*}$	**49.5**	**48.6**	**65.0**

* marks a result with explicit context management, † marks an evaluation on fewer examples than the target split, and a dash denotes an unavailable or incompatible result. Our method performs competitively across multiple benchmarks. However, student models and training procedures differ across methods; some use mid-training and reinforcement learning rather than supervised fine-tuning alone. The comparisons should therefore be interpreted as indicative rather than strictly controlled.

6 Related Work

Synthetic Deep-Search Data and Difficulty Control. Two mechanisms are often used to control the difficulty of synthetic deep-search questions, sometimes in combination. One shapes entity paths, reasoning constraints, evidence distribution, or clues (Li et al., 2025; Xia et al., 2025; Wu et al., 2026; Tao et al., 2025b); TaskCraft expands task depth and breadth in a broader agent setting (OPPO AI Agent Team, 2025), while FORT-Searcher selects long-tail graph roots to reduce prior-knowledge shortcuts (Deng et al., 2026). The other uses solver feedback to revise or filter candidates (Xu et al., 2026; Pandit et al., 2025; Yue et al., 2026; Du et al., 2026); OpenSeeker discards questions answered correctly without tools. WebLeaper combines structural task design with trajectory filtering (Tao et al., 2025a). We study a distinct but compatible, teacher-relative dimension: how the intended teacher's familiarity with selected entities relates to the difficulty of the resulting questions. Building on an exsiting graph-based pipeline (Zhou et al., 2026), SEF screens seeds and candidate next-hop entities using the target teacher's self-report before question generation.

Are Search Agents Actually Searching? Closed-book, evidence-blocking, and counterfactual interventions test whether answers persist without usable evidence, exposing memory-backed verification as well as over- and under-searching (Fan et al., 2026; Zhang et al., 2026; Kim, 2026). Separate audits show that web agents can retrieve benchmark copies or answer keys at inference time and thereby inflate reported scores, while dynamic, recent, and counterfactual benchmarks seek to reduce contamination and parametric-memory shortcuts (Han et al., 2025; Wang et al., 2026a; Fan et al., 2026; Wang et al., 2026b; Chen et al., 2026). Distillation evidence also warns that fluent imitation need not transfer factual capability (Gudibande et al., 2024); related agent-distillation pipelines filter correct traces, align supervision with student capacity, or preserve only searches that are counterfactually necessary (Kang et al., 2025; Tang & Zhao, 2026; Cai et al., 2026). Prior work thus treats search versus recall chiefly as an evaluation behavior to diagnose or repair on already-formed questions and trajectories. We instead make it an entity-level data-construction rule, and evaluate consequences for search demand in teacher trajectories and downstream student gains.

7 Conclusion

In this paper, we investigated how teacher-specific entity familiarity can guide the synthesis of deep-search distillation data. We first introduced EVF to assess familiarity against web evidence and found that it is associated with teachers' closed-book answerability, although its verification cost limits its use during graph construction. We therefore used the lighter-weight SEF judgment to screen seed and next-hop entities. Across three teachers, SEF-gated synthesis produced trajectories with greater observed search demand and yielded higher student accuracy than ungated synthesis in equal-sized training comparisons. We further found that target-specific SEF reassessment enables cross-teacher graph reuse: expanding Kimi-K3's training set with reused graphs achieved a further gain in student accuracy. Together, these results support using teacher familiarity as an early data-construction signal rather than relying on graph structure alone to anticipate useful difficulty.

Limitations. EVF and SEF are operational signals of teacher behavior rather than ground-truth or calibrated knowledge boundaries, and validity tests mainly probe root entities. Our empirical scope is limited to three teachers, two student backbones, and one fact-seeking evaluation set. The current method applies only to graph-based QA synthesis in which nodes represent entities; adapting entity-familiarity gating to other QA synthesis paradigms remains future work.

AI USE STATEMENT

We used generative AI tools to assist with searching for relevant literature, writing experimental code, and polishing the manuscript. The authors manually verified and corrected all AI-assisted outputs, including identified references, experimental code, and revised text, and take full responsibility for the final content of this paper.

REPRODUCIBILITY STATEMENT

Sections 2.2, 4.1, and 5.2 describe graph-based question synthesis, SEF screening at seed and next-hop admission, and target-teacher screening for graph reuse; Appendix A gives further synthesis details. Appendix B provides the exact EVF and SEF prompts, message roles, and response-handling rules. Appendix C documents sampling, answer judgment, failure handling, and metrics for the familiarity and teacher-trajectory analyses, while Appendix D reports trajectory filtering and student fine-tuning settings. Sections 4.3 and 5.3 state the compared training-set sizes and student evaluation setups. Student accuracy is evaluated with 256k context over three independent runs on the 130-question BrowseComp subset selected by QUEST; we report the mean and sample standard deviation, count failed evaluations as incorrect, and block Hugging Face sources during web search. Because the experiments use hosted models and live web retrieval, exact outputs may vary with model versions and the state of the web.

REFERENCES

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

Apodex Team. Apodex-1.0: A Verification-Centric Agent Team for Discoverative Intelligence. Technical report, June 2026, 2026. URL `https://framerusercontent.com/images/us2FrK69YXqcWwu2AAUVAVCnK0.pdf`.

Qinnan Cai, Yibo Zhao, and Xiang Li. Think big, search small: Where capacity matters in hierarchical search agents? *arXiv preprint arXiv:2607.07548*, 2026.

Jiawei Chen, Xintian Shen, Lihao Zheng, Lifu Mu, Haoyi Sun, Ning Mao, Hao Ma, Tao Wei, Pan Zhou, and Kun Zhan. Evaluating the search agent in a parallel world. *arXiv preprint arXiv:2603.04751*, 2026.

DeepSeek-AI, Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, et al. Deepseek-v3.2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*, 2025.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, et al. DeepSeek-V4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

Jia Deng, Yimeng Chen, Xiaoqing Xiang, Ziyang Zeng, Shuo Tang, et al. FORT-Searcher: Synthesizing shortcut-resistant search tasks for training deep search agents. *arXiv preprint arXiv:2606.12087*, 2026.

Yuwen Du, Rui Ye, Shuo Tang, Xinyu Zhu, Yijun Lu, Yuzhu Cai, and Siheng Chen. OpenSeeker: Democratizing frontier search agents by fully open-sourcing training data. *arXiv preprint arXiv:2603.15594*, 2026.

HuiMing Fan, Xiao Wang, Zheng Chu, Qianyu Wang, Zhuoyao Wang, Ming Liu, Bing Qin, and XingYu. LiveBrowseComp: Are search agents searching, or just verifying what they already know? *arXiv preprint arXiv:2605.28721*, 2026.

Xinyu Geng, Xuanhua He, Sixiang Chen, Yanjing Xiao, Fan Zhang, et al. DeepSearch-World: Self-distillation for deep search agents in a verifiable environment. *arXiv preprint arXiv:2607.07820*, 2026.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Arnav Gudibande, Eric Wallace, Charlie Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. The false promise of imitating proprietary language models. In *International Conference on Learning Representations*, 2024.

Ziwen Han, Meher Mankikar, Julian Michael, and Zifan Wang. Search-time data contamination. *arXiv preprint arXiv:2508.13180*, 2025.

Minki Kang, Jongwon Jeong, Seanie Lee, Jaewoong Cho, and Sung Ju Hwang. Distilling LLM agent into small models with retrieval and code tools. In *Advances in Neural Information Processing Systems*, 2025.

Minho Kim. When should LLMs search? counterfactual supervision for search routing. In *FAGEN Workshop at ICML*, 2026.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026.

Kuan Li, Zhongwang Zhang, Huifeng Yin, Liwen Zhang, Litu Ou, Jialong Wu, Wenbiao Yin, Baixuan Li, Zhengwei Tao, Xinyu Wang, et al. WebSailor: Navigating super-human reasoning for web agent. *arXiv preprint arXiv:2507.02592*, 2025.

Zhuofeng Li, Dongfu Jiang, Xueguang Ma, Haoxiang Zhang, Ping Nie, et al. OpenResearcher: A fully open pipeline for long-horizon deep research trajectory synthesis. *arXiv preprint arXiv:2603.20278*, 2026.

Shuqi Lu, Chaofan Li, Kun Luo, Zhang Zhang, Hui Wang, et al. AREX: Towards a recursively self-improving agent for deep research. *arXiv preprint arXiv:2607.21461*, 2026.

OPPO AI Agent Team. TaskCraft: Automated generation of agentic tasks. *arXiv preprint arXiv:2506.10055*, 2025.

Shrey Pandit, Xuan-Phi Nguyen, Yifei Ming, Austin Xu, Jiayu Wang, Caiming Xiong, and Shafiq Joty. Synthesizing agentic data for web agents with progressive difficulty enhancement mechanisms. *arXiv preprint arXiv:2510.13913*, 2025.

Thinh Pham, Nguyen Nguyen, Pratibha Zunjare, Weiyuan Chen, Yu-Min Tseng, and Tu Vu. SealQA: Raising the bar for reasoning in search-augmented language models. In *International Conference on Learning Representations*, 2026.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, et al. Humanity's last exam. *arXiv preprint arXiv:2501.14249*, 2025.

Qwen Team. Qwen3.8-27B. Hugging Face model card, 2026. URL `https://huggingface.co/Qwen/Qwen3.8-27B`.

Guokai Tang and Feng Zhao. SmartAD: Capacity-aligned agent distillation for small language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 27045–27057, 2026.

Zhengwei Tao, Haiyang Shen, Baixuan Li, Wenbiao Yin, Jialong Wu, Kuan Li, Zhongwang Zhang, Huifeng Yin, Rui Ye, Liwen Zhang, Xinyu Wang, Pengjun Xie, Jingren Zhou, and Yong Jiang. WebLeaper: Empowering efficiency and efficacy in webagent via enabling info-rich seeking. *arXiv preprint arXiv:2510.24697*, 2025a.

Zhengwei Tao, Jialong Wu, Wenbiao Yin, Junkai Zhang, Baixuan Zhang, Haiyang Shen, Kuan Li, Liwen Zhang, Xinyu Wang, Yong Jiang, Pengjun Xie, Fei Huang, and Jingren Zhou. WebShaper: Agentically data synthesizing via information-seeking formalization. *arXiv preprint arXiv:2507.15061*, 2025b.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, et al. Tongyi deepresearch technical report. *arXiv preprint arXiv:2510.24701*, 2025.

Yongjie Wang, Xinyue Zhang, Kunhong Yao, Zhiwei Zeng, Kaisong Song, Jun Lin, and Zhiqi Shen. Search-time contamination in deep research agents: Measuring performance inflation in public benchmark evaluation. *arXiv preprint arXiv:2606.05241*, 2026a.

Yunhan Wang, Jiaan Wang, Lianzhe Huang, Xianfeng Zeng, and Fandong Meng. EvoBrowseComp: Benchmarking search agents on evolving knowledge. *arXiv preprint arXiv:2606.13120*, 2026b.

Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford, Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese. BrowseComp: A simple yet challenging benchmark for browsing agents. *arXiv preprint arXiv:2504.12516*, 2025.

Tongzhou Wu, Yuhao Wang, Xinyu Ma, Xiuqiang He, Shuaiqiang Wang, Dawei Yin, and Xiangyu Zhao. DeepResearch-9K: A challenging benchmark dataset of deep-research agent. *arXiv preprint arXiv:2603.01152*, 2026.

xbench Team. xbench-DeepSearch-2510 dataset. Versioned dataset on Hugging Face, 2025. URL `https://huggingface.co/datasets/xbench/DeepSearch-2510`. Benchmark overview: `https://xbench.org/agi/aisearch`.

Ziyi Xia, Kun Luo, Hongjin Qian, and Zheng Liu. Open data synthesis for deep research. *arXiv preprint arXiv:2509.00375*, 2025.

Jian Xie, Tianhe Lin, Zilu Wang, Yuting Ning, Yuekun Yao, et al. QUEST: Training frontier deep research agents with fully synthetic tasks. *arXiv preprint arXiv:2605.24218*, 2026.

Fangyuan Xu, Rujun Han, Yanfei Chen, Zifeng Wang, I-Hung Hsu, Jun Yan, Vishy Tirumalashetty, Eunsol Choi, Tomas Pfister, and Chen-Yu Lee. Sage: Steerable agentic data generation for deep search with execution feedback. *arXiv preprint arXiv:2601.18202*, 2026.

Jiajun Xu, Zhiyuan Li, Wei Chen, Qun Wang, Xin Gao, Qi Cai, and Ziyuan Ling. On-device language models: A comprehensive review. *arXiv preprint arXiv:2409.00088*, 2024.

Zhenrui Yue, Kartikeya Upasani, Xianjun Yang, Suyu Ge, Shaoliang Nie, Yuning Mao, Zhe Liu, and Dong Wang. Dr. Zero: Self-evolving search agents without training data. *arXiv preprint arXiv:2601.07055*, 2026.

Z.ai. GLM-5.1: Towards long-horizon tasks. Official model release article, 2026. URL `https://z.ai/blog/glm-5.1`.

Wenlin Zhang, Kuicai Dong, Junyi Li, Yingyi Zhang, Xiaopeng Li, Pengyue Jia, Yi Wen, Derong Xu, Maolin Wang, Yichao Wang, Yong Liu, and Xiangyu Zhao. To search or not to search: Aligning the decision boundary of deep search agents via causal intervention. In *Proceedings of the ACM Web Conference 2026*, 2026. doi: 10.1145/3774904.3792235.

Peilin Zhou, Bruce Leon, Xiang Ying, Can Zhang, Yifan Shao, et al. BrowseComp-ZH: Benchmarking web browsing ability of large language models in chinese. *arXiv preprint arXiv:2504.19314*, 2025.

Yuhang Zhou, Kai Zheng, Qiguang Chen, Mengkang Hu, Qingfeng Sun, Can Xu, and Jingjing Chen. OffSeeker: Online reinforcement learning is not all you need for deep research agents. *arXiv preprint arXiv:2601.18467*, 2026.

APPENDIX

A QUESTION SYNTHESIS AND GRAPH-DEPTH DIAGNOSTICS

A.1 DEEPFORGE QUESTION SYNTHESIS

We adapt the graph-based question-synthesis workflow of OffSeeker (Zhou et al., 2026). Topic-diverse web pages are collected and used to extract seed entities. Starting with a seed e_0, the explorer searches and reads web evidence, records factual properties and relations, and proposes related entities. It samples a candidate next-hop entity from the available relations and repeats exploration up to a sampled depth, retaining a partial graph if the process stops early. Thus, the recorded transition path $P = (e_0, \ldots, e_h)$ describes exploration, not necessarily the reasoning steps required to answer a generated question.

The resulting evidence graph is passed to a question generator to produce a question and reference answer. A subsequent enhancement step removes overly explicit clues while keeping the answer fixed. The search-enabled teacher then attempts the enhanced question, and its final answer is judged before the trajectory is converted into student training data. In ungated synthesis, seed and next-hop entities are admitted without a familiarity test. SEF-gated synthesis instead screens both admission points with the target teacher's tool-free self-report and samples the next hop only from entities judged unfamiliar. If none passes, exploration stops without discarding the partial graph. These are entity-admission decisions made *before* the downstream question and trajectory exist; no source-teacher question or trajectory is transferred when an evidence graph is later reused by a different teacher.

A.2 DEPTH VERSUS CLOSED-BOOK ANSWERABILITY

We use the length of the recorded entity transition path as graph depth. Table 8 groups each teacher's closed-book correctness by this depth. Missing final answers, length cutoffs, and timeouts count as incorrect.

Table 8: Closed-book correctness by recorded graph depth. Cells give correct answers / questions and the corresponding accuracy; ρ is the question-level Spearman correlation with its unadjusted p value.

Teacher	Depth 1	Depth 2	Depth 3	Depth 4	ρ (p)
GLM-5.1	9/28 (32.1%)	46/123 (37.4%)	49/120 (40.8%)	15/29 (51.7%)	$+.086$ (.138)
DeepSeek-V4-Flash	9/38 (23.7%)	43/124 (34.7%)	45/110 (40.9%)	8/28 (28.6%)	$+.067$ (.250)
Kimi-K3	28/37 (75.7%)	82/112 (73.2%)	85/122 (69.7%)	21/29 (72.4%)	$-.038$ (.517)

GLM-5.1's binned accuracy rises with depth; DeepSeek-V4-Flash is nonmonotonic, and Kimi-K3 does not show an increasing trend. Hence these data do *not* show that simply increasing exploration depth reliably decreases a teacher's closed-book answerability. They illustrate why graph-structural complexity need not translate into teacher-relative difficulty.

One possible explanation for GLM-5.1's increasing accuracy with depth is that longer paths provide more opportunities to encounter well-known entities. If prominent entities are proposed more often as next-hop candidates, an explorer selecting among the proposed relations may reach familiar entities even when the seed entity is obscure. The present analysis does not measure entity connectivity or familiarity at each hop, so this mechanism remains a hypothesis.

B EXACT EVF AND SEF PROMPT TEMPLATES

We provide the exact prompts used for EVF and SEF. Entity names and factual briefs are inserted into the templates at inference time. SEF and EVF brief generation use *user* messages without an explicit system message; EVF verification uses both a system message and an entity-specific user message. All calls use temperature zero. The EVF verifier is the probed teacher with browser access, whereas Kimi-K3 judges the final answers to synthesized questions.

B.1 SEF SELF-REPORT

The one-call, tool-free self-report uses the following user-message template:

```
Do you have enough specific factual knowledge about the entity named "{name}"
to answer detailed questions about its history, properties, or attributes?

Reply with only one word: YES or NO.
- YES: you can recall multiple specific verifiable facts about this exact entity
- NO: you cannot, or you're unsure which entity this refers to, or your knowledge
  is vague/category-level only

If in doubt, answer NO.
```

During synthesis, judgments are cached by entity and teacher-model identifier; the signal and cost evaluations make uncached calls.

B.2 EVF FACTUAL BRIEF

The teacher produces its factual brief without browser tools, using this user-message template:

```
Tell me everything you confidently know about the entity named: {name}

Write a single paragraph of factual statements. Each statement should be a
verifiable claim such as a date, place, role, relationship, organization,
work, achievement, or attribute.

Rules:
- Write only facts you are confident about. Do not speculate or fill space.
- Do NOT use hedging language like "I think", "possibly", "around",
  "approximately", "may have", "is believed to be". State each fact directly
  or omit it.
- Do NOT add disclaimers about your knowledge cutoff or your uncertainty.
- Plain prose only. No markdown, no bullet lists, no headings.
- Keep it under 150 words. Quality over quantity.
- If you have no reliable information about this entity, output exactly:
  I have no information about this entity.
```

When the teacher provides no factual brief, EVF is set to zero without evidence verification.

B.3 EVF EVIDENCE VERIFICATION

The same teacher then verifies the brief using browser search, open, and find tools. The verification system message is:

```
You are a fact-checking assistant with access to browser tools
(browser.search, browser.open, browser.find). You will be given an entity
name and a candidate brief about that entity, written by another LLM with
no tool access. Your job is to extract atomic factual claims from the brief
and verify each via the browser tools.

Procedure:
1. Read the brief and extract up to 8 atomic factual claims. Each claim must:
   - Be a single self-contained proposition (no compound statements joined
     by "and" / "but" / ";").
   - Use the entity's full name rather than pronouns (resolve "he/she/it"
     to the entity).
   - Be a concrete fact in principle verifiable through web search.
   - Not be a trivial restatement of the entity's name or category.
2. For each claim, decide its verdict by actually issuing browser searches:
   - "true"    : confirmed by at least one reliable source you retrieved
                 in this conversation.
   - "false"   : contradicted by a reliable source you retrieved.
   - "unknown" : you searched but could not find evidence either way, or
                 sources disagree, or you could not pin the claim down to
                 the precision stated in the brief.
3. Be strict. Hedging or partial matches are NOT "true". If the brief says
    "founded in 1850" and your evidence says "founded in the 1850s", that
    is "unknown" — the precise year is not confirmed.


When verification is complete, output your final answer as JSON inside
<result>...</result> tags, with no text after the closing tag:

<result>
{
  "claims": [
```

```
    {"text": "<atomic claim>", "verdict": "true|false|unknown", "evidence": "<short URL or quote>"}
  ]
}
</result>

If the candidate brief is exactly the sentinel
"I have no information about this entity." (or trivially close to it),
skip all searches and output:

<result>{"claims": []}</result>
```

The accompanying user message has the following form:

```
Entity: {entity_name}

Candidate brief:
{brief}
```

The verifier prompt refers to the brief writer as *another LLM*, although the same teacher model fills both roles in these experiments, first without and then with browser access. Each claim labeled `true` contributes one to the verified-claim count; `false` and `unknown` do not. EVF divides the verified-claim count by the number of extracted claims and assigns zero if the list is empty.

B.4 Verifier Turn Limit

If verification reaches its configured tool-loop limit, the teacher receives the following *user* message and must return a final answer without further tool calls:

```
STOP. Do not use any more tools.  Based only on the searches
you have already performed in this conversation, output the
result JSON now.  Mark any claim you could not confirm as
"unknown".  Output only the <result>...</result> block and
nothing else.
```

The turn limit is configurable. Results obtained after forced finalization are not stored in the EVF cache. Cost measurements were performed without cache reuse.

C Evaluation Details for the Reported Tables

The analyses of teacher answers count length cutoffs, timeouts, empty final answers, and missing judgments as incorrect. Answered-only accuracy is used solely to diagnose generation failures. Kimi-K3 judges the final answers from both the closed-book evaluations and the teachers' tool-assisted search trajectories. EVF claim verification instead uses the probed teacher with browser access (Appendix B.3). The three familiarity samples and the ungated/gated training-data pools are teacher-specific.

C.1 Table 1: EVF and Closed-Book Solving

For GLM-5.1, DeepSeek-V4-Flash, and Kimi-K3, the familiarity analysis samples enhanced questions from teacher-specific QA pools constructed using the same graph-based synthesis procedure. Each question is paired with its graph's seed entity. The teacher writes a tool-free factual brief and verifies its extracted claims with browser access using the prompts in Appendix B. Unsuccessful EVF probes have no score, whereas a valid no-information brief receives zero.

Each teacher independently answers its own question without tools, and Kimi-K3 evaluates the final answer. EVF AUC binarizes each entity's mean closed-book correctness at $\geq .5$, while Spearman's ρ uses the continuous entity-level outcome. The low and high EVF regions use the fixed cutoffs $[0, .25)$ and $[.75, 1]$. Table 8 stratifies the same closed-book outcomes by recorded graph depth.

C.2 Table 2: SEF Screening

Each teacher gives a tool-free YES/NO self-report on the seed entity using Appendix B.1. The analysis joins valid SEF and EVF outcomes by entity identity, with EVF$< .25$ defining the low-EVF region. Low-EVF retention and high-EVF exclusion measure screening within their respective

EVF regions. Before/after shares measure the change in pool composition after retaining only NO judgments; precision and enrichment characterize the low-EVF content of the retained pool. EVF is an operational proxy, not an externally verified knowledge label. The SEF labels themselves receive no external judgment.

C.3 TABLE 3: DECISION COSTS

EVF and SEF are evaluated without cache hits on teacher-specific entity samples. Table 3 reports mean LLM calls, browser tool calls, and wall time per assessed entity. EVF includes the tool-free brief, every verifier turn, and any forced final call; SEF consists of one tool-free call. Wall times are comparable only within a deployment.

C.4 TABLE 4: TEACHER SEARCH BEHAVIOR

For each teacher, ungated and SEF-gated collections follow the same graph-based QA synthesis and tool-assisted solving stages. The gated condition adds SEF screening when admitting seed and next-hop entities. Each condition generates its own graph and question pool, so the two conditions do not contain paired questions or paths.

Turns, reasoning tokens, tool calls, and solve accuracy are measured on every raw teacher trajectory in each ungated or SEF-gated pool. Assistant messages count as turns, and structured browser actions count as tool calls. Nonempty assistant reasoning fields are tokenized with the Qwen3-30B-A3B-Thinking-2507 tokenizer; this proxy excludes tool observations and final answers and is not any teacher's native API token count. Kimi-K3 judges final-answer correctness, with empty predictions counted as incorrect. Because the ungated and gated graph and question pools are not paired, these comparisons do not isolate the effect of entity screening.

C.5 TABLE 5: STUDENT EVALUATION

Students are evaluated on the 130-question BrowseComp subset selected by QUEST (Xie et al., 2026) with a 256k maximum context length. Each condition is run independently three times on all 130 questions. Kimi-K3 judges final-answer correctness, and generation failures, empty answers, and missing judgments count as incorrect. We report mean accuracy and the sample standard deviation across the three runs. The before-SFT Qwen3-30B-A3B-Thinking result is retained as an approximate $\sim$2.5% reference; the other student results use the three-run aggregation.

To reduce benchmark-answer leakage during web evaluation, the browser removes search results whose URLs point to Hugging Face before their titles, snippets, or links are shown to the student, and it rejects direct attempts to open those pages. This filtering is applied in every reported student run.

C.6 TABLE 6: CROSS-TEACHER SEF AGREEMENT

Each teacher contributes 2,000 sampled entities from its historical SEF cache. Deduplication produces a shared pool of 5,995 names, on which GLM-5.1, DeepSeek-V4-Flash, and Kimi-K3 each issue fresh tool-free SEF judgments. Table 6 reports the share judged unfamiliar by both teachers, the two directional disagreement shares, and overall pairwise agreement on valid judgments.

C.7 GRAPH REUSE AND STUDENT EVALUATION

Previously collected non-Kimi evidence graphs supply candidate paths for the Kimi-K3 target, but neither questions nor teacher trajectories. Kimi-K3 applies SEF to the candidate paths; accepted paths are used to resynthesize questions and collect new Kimi-K3 search trajectories. The 2,200 native trajectories and 5,411 trajectories from reused graphs form the 7,611-example training set.

Under the same 256k-context student-evaluation protocol, the model trained on the 2,200 native SEF-gated trajectories achieves $63.59 \pm 1.18\%$ accuracy, while the model trained on all 7,611 trajectories achieves $66.41 \pm 2.35\%$. Each value is the mean and sample standard deviation over three independent runs on the QUEST-selected BrowseComp subset, with the failure and answer-judgment rules described in Appendix C.5.

D STUDENT TRAINING DETAILS

The students are trained by supervised fine-tuning on serialized teacher search trajectories, not on the closed-book familiarity probes. Each six-epoch run uses full-parameter updates in bfloat16 with DeepSpeed ZeRO-3, a 262,144-token maximum sequence length, sequence packing, Adam betas $(.9, .95)$, weight decay $.1$, and gradient clipping at 1.0. The student tokenizer and chat template serialize the complete sequence, including tool interactions. Both students use an effective global batch size of 32 packed sequences and a cosine learning-rate schedule. Qwen3-30B-A3B-Thinking uses a learning rate of 5×10^{-5}; Qwen3.8-27B uses 5×10^{-6}.

We convert judged teacher trajectories into SFT examples after removing records with no assistant messages, no tool use, malformed tool calls, or an unfinished reasoning or tool action at the end. Records that cannot be rendered in the student's conversation format are also removed. Incorrect answers are not excluded.