Pushing the Performance Frontier: How Expert Knowledge Reshapes Scaling Laws in Enterprise AI*

Jie Zheng† Mohammad Rahman‡ Yu Jeffrey Hu§

Abstract

Scaling laws, the foundational laws of AI development, offer a low-risk recipe for allocating capital and have been a primary engine of AI's rapid progress. As massive investment flows into AI deployment, whether these laws govern enterprise AI remains unknown. We partner with a consulting firm and deploy large language models for its quality check task. Under the common recipe of enterprise deployment, which scales the firm's knowledge base and model size, performance reaches a low ceiling quickly. By contrast, expert knowledge shifts the scaling curve itself upward. In particular, experts' reasoning data substantially expand the performance frontier, with performance rising log-linearly in their amount. Scaling reasoning data also significantly raises the profit of the deployment. Mechanism analyses reveal that the common recipe falls short because the raw knowledge base is noisy, whereas reasoning data convey information about the patterns behind experts' judgments. This study provides guidance to firms on how to allocate resources in AI deployment.

Keywords: scaling laws, enterprise AI, expert knowledge, large language models

*This paper benefited from comments from the audiences and discussants at the 2025 AI and the Future of Work Conference, 2025 SCECR Conference, 2025 INFORMS Conference, 2025 WISE Conference, 2026 BIG.AI Conference, 2026 BizAI Conference, and 2026 SCECR Conference. The authors received no external funding for this research. The data were provided by a partner consulting firm under a data-use agreement; the firm did not compensate the authors. The authors declare no relevant or material financial interests that relate to the research described in this paper.

†Purdue University. Email: zheng795@purdue.edu.

‡Purdue University. Email: mrahman@purdue.edu.

§Purdue University. Email: yuhu@purdue.edu.

"The intelligence of an AI model roughly equals the log of the resources used to train and run it. . . . The socioeconomic value of linearly increasing intelligence is super-exponential in nature. A consequence of this is that we see no reason for exponentially increasing investment to stop in the near future."

— Sam Altman, co-founder and CEO of OpenAI (2025).[1]

1 Introduction

The foundational laws governing the development of AI are scaling laws. These laws predict that the intelligence of AI improves with increased resources, including data, parameters, and compute (Kaplan et al., 2020). This predictability turns AI development into a low-risk investment and has attracted massive capital. Goldman Sachs' baseline scenario projects $765 billion in annual AI capital expenditure in 2026 (Lee and Greenbaum, 2026). Firms are now carrying this scaling-law recipe into deployment. They spent $37 billion on generative AI deployments in 2025, a 3.2-fold increase over the prior year (Tully et al., 2025). Yet building AI and deploying it in a firm's specific business context may be fundamentally different, and whether these laws govern enterprise AI remains an open question.

Firms currently follow a common recipe for deploying AI. Because most work within firms relies on domain-specific knowledge that a general-purpose model does not possess (Ovadia et al., 2024), firms scale data by supplying the model with their own knowledge base. They may also deploy larger models to improve performance. However, anecdotal evidence suggests that 95% of generative AI deployments produce no measurable return (Challapally et al., 2025).[2] These deployments seem to reach a performance ceiling quickly, far below what firms can accept.

One missing ingredient in the recipe could be experts' knowledge. A large share of organizational knowledge is uncodified and resides in experts' minds. One effective form of this knowledge that AI models can utilize may be experts' reasoning data: the explanations behind experts' judgments. Human learning theories point the same way. For example, learning to write is more effective when learners receive explanations of why articles are good or poor than when they receive only examples or quality labels (Van der Kleij et al., 2015; Wittwer and Renkl, 2010).

Each of these resources is costly, which confronts firms with an allocation problem. On

[1] Source: https://blog.samaltman.com/three-observations

[2] Even adopters at the technological frontier report paying millions while losing money. As Chamath Palihapitiya, an entrepreneur and venture capitalist, put it in March 2026: "I am paying these models millions of dollars a year. . . and losing money. . . my team is at the leading edge. I suspect these other (Fortune 1000) companies are in a similar situation."

the data side, firms need to decide both what type of data to provide and how much of it to collect. They may need significant effort to label their knowledge base, since a firm's raw knowledge base may be noisy and inaccurate. Meanwhile, collecting experts' reasoning data, which typically do not exist in current business practice, is costly and time-consuming. On the model side, firms need to choose what size of model to deploy. A larger model may deliver better performance, but the growing shortage of energy driven by the heavy computational demands of AI models (Morales, 2025) makes it costly to run and difficult to deploy. Given these costs, scaling laws are crucial for guiding investment in these resources. This paper therefore examines how the performance of enterprise AI changes as firms scale resources and how they can push performance to a new frontier.

Measuring performance in enterprise deployment requires a metric that reflects the firm's own standards. Many critical enterprise tasks lack an objective measure of output quality. The operative standard is set by the firm's key decision makers and reflected in their judgments. Alignment with these judgments matters for three reasons. First, firms remain accountable for decisions and communications made using AI, so outputs that violate legal, regulatory, or internal constraints expose them to liability and operational risk (Martin, 2019; Berente et al., 2021). Second, competitive advantage rests on firms' unique knowledge that is hard to imitate, whereas general-purpose models default to generic outputs that any competitor could obtain (Grant, 1996; Doshi and Hauser, 2024). Third, AI outputs that deviate from organizational standards require extensive review and rework and can weaken trust, thereby reducing system use (Lebovitz et al., 2022; Dietvorst et al., 2015). We therefore measure the performance of an enterprise AI deployment by the alignment between AI outputs and those of the firm's key decision makers.

We study these questions in partnership with a consulting firm whose core service is delivering pricing benchmark and negotiation strategy reports to clients. We deploy large language models (LLMs) for the quality check performed before reports are delivered, a task that relies heavily on expert judgment and is economically important, since poor reports can cost the firm its clients. Because experts rarely record their reasoning and firms seldom retain draft reports or past quality checks, we first designed a randomized field benchmark exercise. We randomly assigned previously delivered reports and LLM-generated reports to the firm's 10 key decision makers and collected 258 valid quality checks, together with experts' reasoning data behind these checks, including ratings and written comments.

We then deploy LLMs on the same task using retrieval-augmented generation (RAG).[3] We

[3]RAG and fine-tuning are the two main approaches for customizing LLMs with organizational knowledge. However, fine-tuning is costly because the model must be retrained whenever the knowledge base is updated. Additionally, each time a new foundation model is released, firms need to fine-tune the new model again to achieve the best performance. As a result, RAG has become a more widely adopted approach in practice

trace scaling curves along the two standard dimensions of scaling, data size and model size. Data size ranges from 2 to 64 data points in the prompt. For model size, we compare large and small models within three open-source families, GPT-OSS (120B and 20B parameters), DeepSeek-R1 (671B and 8B), and Llama 3.1 (70B and 8B). Along these curves, we vary the ingredients supplied to the model: the raw knowledge base of previously delivered reports,[4] a labeled knowledge base carrying experts' quality check labels, and experts' reasoning data.

Our results show that expert knowledge pushes out the performance frontier of enterprise AI. First, under the common recipe, the frontier is low. Scaling the raw knowledge base does not improve alignment, and even scaling it together with model size brings only limited gains. Second, expert knowledge shifts the scaling curve itself upward. Adding experts' quality check labels improves alignment modestly, whereas adding experts' reasoning data raises it substantially. Alignment rises log-linearly with the amount of reasoning data. At its peak, it is 10.8 to 37.2 percentage points higher than with the raw knowledge base alone, depending on the model. Notably, a small model supplied with reasoning data outperforms a large model without it. Third, an economic analysis suggests that these gains are large enough to turn LLM quality checks from a loss into a profitable deployment.

We then examine the mechanisms behind these scaling patterns. The raw knowledge base holds the frontier down because it is noisy: experts would reject 27.5% of the delivered reports it contains. Reasoning data expand the frontier because they tell the model which patterns make a report good or bad. Alignment improves when the report under review shares patterns with those described in the retrieved reasoning. These gains come from experts' written comments rather than their ratings, and from comments about a report's content rather than its style. Finally, LLM-generated reasoning is not a substitute, as its effect is much smaller than that of expert reasoning, and it misses many key points that experts raise while adding many that experts do not.

This paper makes three main contributions. First, to our knowledge, it is the first to study scaling laws in enterprise AI. Existing scaling laws describe the development of AI models (Kaplan et al., 2020; Hestness et al., 2017; Hoffmann et al., 2022) and, more recently, the use of inference-time compute (Snell et al., 2025). We instead trace how performance scales with data and model size when a firm deploys these models in its own business context. These laws can guide how firms allocate resources in AI deployment. Second, we identify expert knowledge as a critical input to enterprise AI. Like a critical input in a production function, it expands the frontier, which suggests one reason why enterprise AI investments have yet to pay off. Third, we reveal the mechanisms behind these scaling patterns: a noisy knowledge

(Lin, 2024; Tully et al., 2025).

[4]In this paper, "knowledge base" refers to the raw knowledge base. We use these terms interchangeably.

base holds the frontier down, and expert reasoning lifts it by providing the patterns behind experts' judgments. These mechanisms also provide guidance to firms on collecting expert knowledge.

The remainder of this paper is organized as follows. Section 2 reviews the related literature. Section 3 describes our research design. Section 4 presents the main results, Section 5 the mechanisms, and Section 6 the economic implications. Section 7 concludes.

2 Literature Review

This paper relates to three streams of research: scaling laws of AI, enterprise AI and expert knowledge, and AI alignment.

2.1 Scaling Laws of AI

Scaling laws describe how the performance of foundation models improves predictably as resources grow. During training, the test loss of language models falls as a power law in the number of parameters, the amount of training data, and compute (Hestness et al., 2017; Kaplan et al., 2020). These laws also tell developers how to split a fixed compute budget between model size and data (Hoffmann et al., 2022). Recent work extends the same logic to inference: performance also rises with the compute or context a model uses when it answers a query (Snell et al., 2025), including in RAG systems (Yue et al., 2025). Performance likewise improves with the number of examples supplied in the prompt (Brown et al., 2020), which is the data-size dimension we vary in deployment.

Most scaling laws treat data as a quantity, but the kind of data also shapes the curve. Selecting more informative training examples can beat power-law scaling (Sorscher et al., 2022). The value of repeated data decays toward zero once unique data run out (Muennighoff et al., 2023). About a thousand carefully curated examples can align a pretrained model as well as far larger instruction datasets (Zhou et al., 2023). The goal of foundation model development is general-purpose capability, and the knowledge it requires is largely public.

Scaling laws found in development may not apply to enterprise AI deployment. First, with respect to model size, a larger model stores more general knowledge, but a firm's internal standards, processes, and rules of judgment appear in no training corpus. Second, regarding data, the test loss and public benchmarks used in development evaluation draw on knowledge that a vast training corpus largely covers. A firm's records, by contrast, are often limited, repetitive, unlabeled, and of mixed quality, so more records may add more noise than information. Human feedback may not close this gap either. Foundation model developers

write down their standards in labeling guidelines and hire labelers at scale to apply them (Ouyang et al., 2022). A firm's standards may never be written down, because experts know more than they can tell (Polanyi, 1966). Eliciting this knowledge is also costly, so firms may collect too little of it to make a difference. Existing studies have not examined scaling laws in enterprise AI, so it remains unknown whether performance scales predictably with model size and data, and which inputs shift the curve.

2.2 Enterprise AI and Expert Knowledge

Many studies examine how investment in earlier AI technologies affects firm growth and other firm outcomes (Babina et al., 2024; Law and Shen, 2025). Generative AI has widened the range of knowledge work that AI can assist, from software engineering (Peng et al., 2023) and consulting tasks (Dell'Acqua et al., 2026) to evaluating strategic decisions (Doshi et al., 2025). However, industry surveys report that most generative AI deployments have not produced measurable returns (Challapally et al., 2025). This gap echoes the productivity paradox of information technology (Brynjolfsson, 1993) and of AI (Brynjolfsson et al., 2019). One explanation is that the gains from a general-purpose technology arrive only after firms build intangible assets around it (Brynjolfsson et al., 2021; McElheran et al., 2025). Prior research and surveys offer little direct evidence on which specific inputs are missing from AI deployments, in part because proprietary enterprise data are hard to obtain. We identify one such input, expert knowledge, and show that it pushes out the performance frontier of an enterprise deployment.

Around 70% of firms (Databricks, 2024) adapt a general-purpose model to their own context by supplying their knowledge base through RAG (Lewis et al., 2020) or fine-tuning. Yet a recent survey finds that deployed systems still rarely adapt to the firm's context (Challapally et al., 2025). One reason is that the knowledge base accumulates through ongoing organizational activities (Argote and Ingram, 2000), so it records what employees produced, not why experts judged that work to be good or bad.

Much of the knowledge that firms need resides in experts' minds. Experts' judgments draw on years of experience and encode managerial preferences, organizational standards, and "side information" that algorithms do not observe (Alur et al., 2024). Part of this knowledge is tacit and hard to codify into rules or data (Autor, 2014). Given the high cost of eliciting this knowledge, firms need to know which form of it is worth collecting.

Research on human learning suggests an answer. People learn concepts more effectively when positive examples are paired with negative ones than from positive examples alone (Tennyson et al., 1972; Durkin and Rittle-Johnson, 2012), and explanations of why an example

is right or wrong improve learning further (Van der Kleij et al., 2015; Wittwer and Renkl, 2010). Because LLMs rely on pattern learning, much as humans do, the same findings may hold for them. These categories map onto the ingredients we study: the raw knowledge base supplies examples, the labeled knowledge base adds experts' evaluations, and reasoning data add experts' explanations. We also contribute by examining the mechanism through which expert knowledge shapes the performance of enterprise AI, which gives firms guidance on what expert knowledge to collect.

2.3 Alignment as a Performance Metric

The performance of AI models is typically measured as test loss or as accuracy on public benchmarks with objective answers (Kaplan et al., 2020; Wei et al., 2022). Many enterprise tasks are subjective and are instead judged by the firm's key decision makers. We therefore draw on the literature on AI alignment, which studies how to make AI systems act in accordance with human values and intentions (Gabriel, 2020; Ji et al., 2026). For LLMs, alignment methods train models to follow instructions using human feedback (Ouyang et al., 2022; Bai et al., 2022) so that their outputs are reliable, safe, and consistent with human expectations (Liu et al., 2024; Wang et al., 2023). Economics and management studies examine how to align LLMs with human economic preferences, often treating LLMs as simulated economic agents (Horton, 2023; Goli and Singh, 2024; Lu et al., 2025). We apply this idea and use alignment with the judgments of firms' key decision makers as the performance metric for enterprise AI.

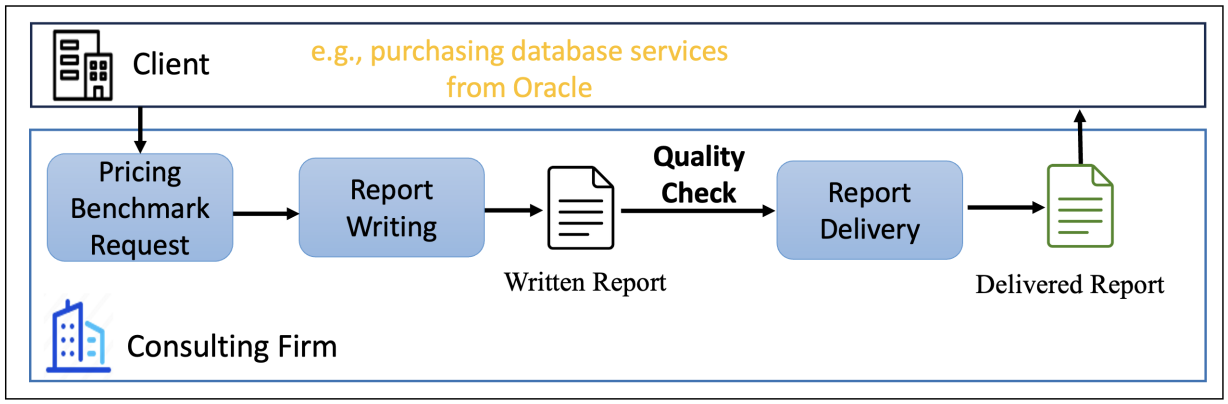
3 Research Design

Our research design proceeds in two stages. In the first stage, we design a randomized field benchmark exercise to collect quality checks and reasoning data from key decision makers. In the second stage, we deploy LLMs for the same quality check task and trace how alignment scales with data size and model size when different types of data are supplied. We first introduce the institutional background, and then present these two stages separately.

3.1 Institutional Background

We partnered with a consulting firm whose core service involves producing pricing benchmark and negotiation strategy reports for enterprise clients. These reports help clients evaluate whether their current or proposed contracts are priced above market standards and identify

Figure 1. Workflow of the Pricing Benchmark



potential opportunities for cost reductions. Figure 1 illustrates the firm's pricing benchmark workflow and the quality check task studied in this paper.

When the firm receives a pricing benchmark request, an analyst initiates the process by examining the client's contract details and comparable transactions from the firm's internal database. The analyst then produces a draft written report that includes pricing benchmarks and recommended negotiation strategies. Before the deployment of LLMs, the firm did not have a formal quality check process prior to delivering a report to the client. Most delivered consulting reports are subsequently archived in the firm's internal knowledge base.

This paper deploys LLMs for the quality check task. This task provides an ideal setting for studying the performance of enterprise AI for two reasons. First, pricing benchmark and negotiation strategy reports constitute the firm's core service and primary source of value: clients pay for these reports, so their quality directly affects client retention and revenue. Second, evaluating these reports requires substantial domain knowledge and judgment from key decision makers.

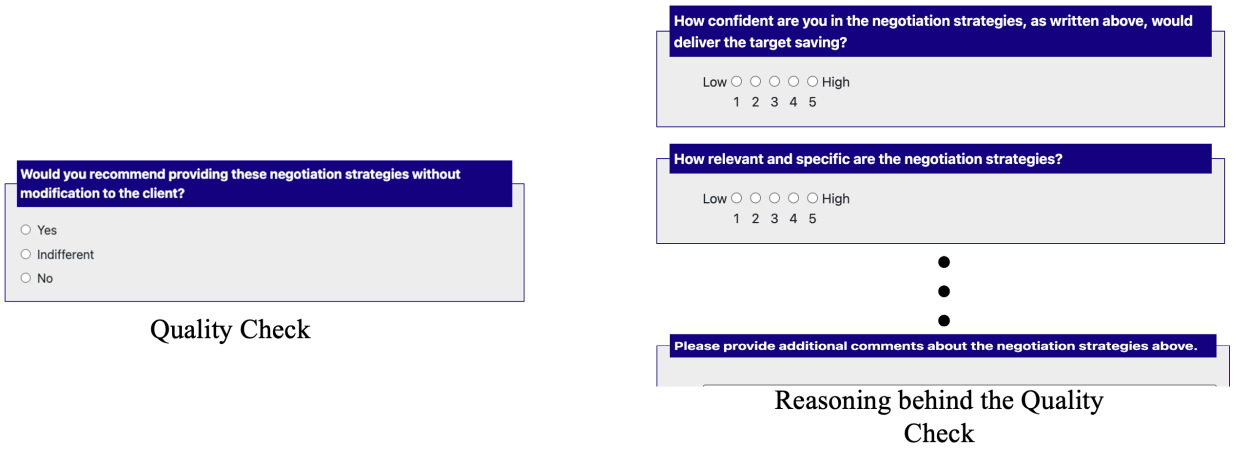
Because this task requires deep domain knowledge, an LLM used out of the box is unlikely to be effective. We therefore examine whether supplying the LLMs with a knowledge base consisting of previously delivered consulting reports, as in the common recipe, improves alignment. We also need to collect quality check outputs from key decision makers as a benchmark for alignment, together with reasoning data that explain their judgments. Unlike the knowledge base, these data are not readily available and must be collected.

There are a number of challenges in collecting these data. Most firms do not retain records of draft reports or historical quality check results, as drafts are typically revised multiple times before delivery and quality checks are often conducted informally, such as through conversations. Furthermore, experts rarely provide reasoning data in current business practice (including during quality checks), as this process is generally time-consuming. In addition,

randomization is needed in the quality check process to mitigate potential biases arising from report assignment and evaluation order. To address these challenges and to enable credible comparisons between the outputs of key decision makers and LLMs, we designed a randomized field benchmark exercise that elicits both quality checks and associated reasoning data from key decision makers.

3.2 Quality Check Benchmark Exercise in the Field

Figure 2. Quality Check and Reasoning



To collect valid quality checks and reasoning data from experts, we first needed to assemble a set of reports for evaluation and then randomly assign them to key decision makers.

3.2.1 Generating Reports

Since the delivered reports archived in the database are mostly positive samples, evaluating them alone would produce little variation in quality, and hence little variation in experts' quality checks. We therefore used LLMs to generate additional reports that experts would evaluate. The firm's knowledge base consists of reports from 1,782 completed deals. Each deal in this sample represents a pricing benchmark request for which the firm had previously delivered a report to a client. We began by drawing a random sample of 100 historical deals from the firm's knowledge base. Using these deals as the underlying cases, we then produced alternative versions of consulting reports with LLMs.

For each of the 100 deals, we first used an LLM with direct prompting. The prompts included detailed deal information such as the client's request, the vendor, the current proposed price of the products, and the fair market value as estimated by analysts. Applying

three distinct prompt strategies to each deal yielded a total of 300 LLM-generated reports.[5] We then fine-tuned an LLM on the remaining 1,682 historical reports in the knowledge base. We used the same deal information and applied the same three prompt strategies to generate another 300 reports. Combined with the 100 original delivered consulting reports, the full set for the benchmark exercise consisted of 700 reports, with 600 generated by LLMs and 100 written historically by analysts. According to feedback from the firm's experts, reports generated by LLMs, particularly those produced using fine-tuned models, were difficult for experts to distinguish from human-generated reports.

3.2.2 Quality Checks

The participant group performing quality checks consisted of five senior analysts and five executives, who are the 10 most senior and highest-ranking experts and leaders of the consulting firm and serve as its key decision makers, each with extensive experience in pricing benchmarks and negotiation strategy evaluation.

To mitigate potential evaluation biases in our field benchmark exercise, we implemented a randomization procedure. Each of the 700 reports was randomly assigned to one of the key decision makers participating in the study, subject to the following restrictions. During assignment, we deliberately avoided cases in which a report would be evaluated by a key decision maker who had previously analyzed or written the report for the underlying deal, as such familiarity could unintentionally bias the quality check. An expert who remembers the original context or their own earlier work might evaluate the report more favorably, thereby distorting both the quality check and the reasoning data collected. We also ensured that each key decision maker received a balanced mix of report types, including historical reports, direct-prompt LLM reports, and fine-tuned LLM reports, so that no expert's evaluation set disproportionately reflected any particular report type. In addition, for each expert, the order in which reports appeared was randomized to avoid potential sequencing effects such as fatigue, anchoring, or learning across evaluations.

To facilitate data collection, we developed an evaluation interface to capture the quality checks and reasoning data. When a key decision maker logged into the interface, for each report assigned to them, they were first shown the complete information for the deal, including the client name, vendor, the client's request, the date of the request, and the relevant information used for pricing benchmarking. Immediately below this information, the interface displayed the report to be evaluated. After reading the report, the expert proceeded to the quality check section and reasoning section.

[5]We used the open-source LLM ranked highest in the benchmark at the time the reports were generated.

The quality check question is "Would you recommend providing these negotiation strategies without modification to the client?" with response options of Yes, Indifferent, or No (see Figure 2). The reasoning section included a set of ratings that asked experts to evaluate specific dimensions of the report, such as the percentage of content that required modification and the estimated time required to modify the report. Experts also rated the report on its financial outcome, relevance and specificity, feasibility, innovativeness, clarity and comprehensibility, comprehensiveness, and the overall quality of the report (see Appendix A). Following these ratings, the interface asked experts to provide written comments explaining the reasoning behind their evaluation.

We collected a total of 258 completed and valid quality checks out of the 700 assigned reports.[6] Because the reports were randomly assigned and each expert evaluated them in a randomized order, the completed quality checks form a random subset of the assignments, which mitigates concerns that selection into the analysis sample is related to report type or quality.[7] Table 1 shows the distribution of quality check results in our field benchmark exercise.

Table 1. Distribution of Quality Check Results

	Yes	Indifferent	No	N
All	79 (30.6%)	58 (22.5%)	121 (46.9%)	258
Human-Generated Reports	26 (51.0%)	11 (21.6%)	14 (27.5%)	51
LLM-Generated Reports	53 (25.6%)	47 (22.7%)	107 (51.7%)	207

3.3 Deploying LLMs for Quality Check

After collecting the 258 quality checks and accompanying reasoning data from key decision makers, the next step is to examine how the performance of the LLM deployment scales with data size and model size, and how this scaling changes with the type of data supplied. We measure performance by the alignment between the LLM's outputs and those of key decision makers for the same quality check task, because these decision makers set and enforce the firm's quality standard. The overall architecture is shown in Figure 3. Following the recent literature on LLM deployment for knowledge-intensive tasks, we adopt RAG (Lewis et al., 2020).

[6]The collected samples represent a sizable dataset, particularly given the high wages of these key decision makers, which makes the cost of collecting these samples substantial. See Section 6 for a detailed accounting of these costs.

[7]We deliberately assigned a large number of reports so that experts would complete as many quality checks as possible. Given how scarce and costly their time is, we ultimately collected 258 of these.

Figure 3. Deploying LLMs for the Quality Check Task Using RAG

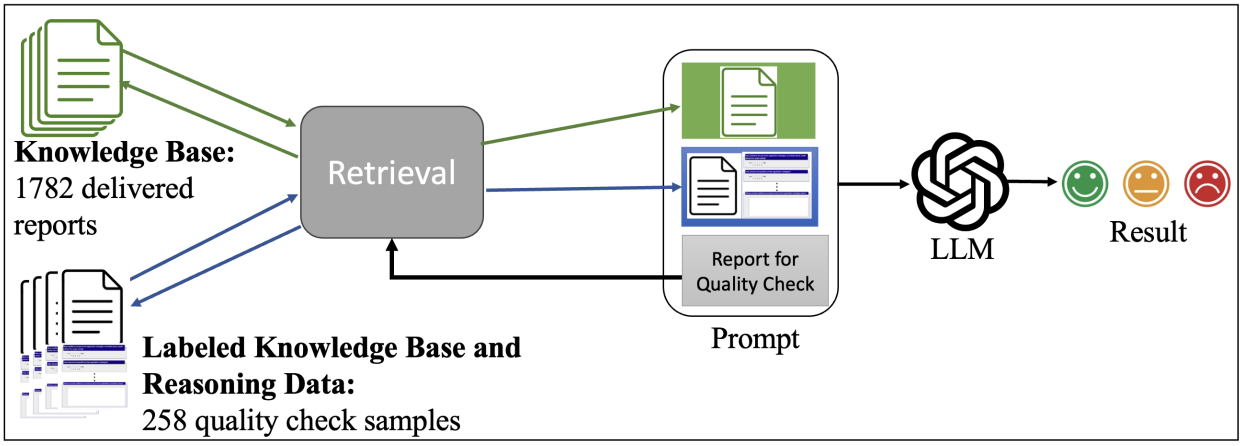
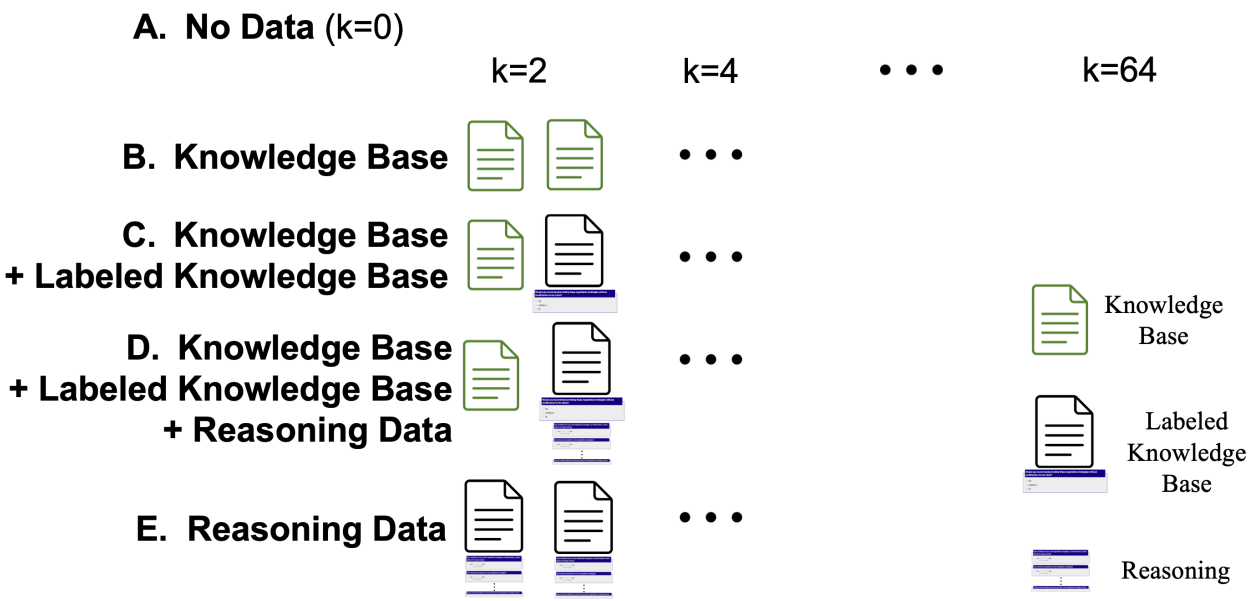


Figure 4. Comparison Groups



In our design, the LLM receives the report to be quality-checked, together with the deal information shown to key decision makers, and is then prompted with the same quality check question that key decision makers answered. We vary the information provided to the LLM in order to evaluate how different types of data contribute to the deployment of the LLM. We consider three types of data, or ingredients, that a firm could supply to the LLM. The *knowledge base* consists of previously delivered consulting reports archived by the firm. It contains report content only, with delivery serving as an implicit positive signal. The *labeled knowledge base* consists of reports paired with a quality-check label (Yes, Indifferent, or No). *Reasoning data* are the ratings and written comments that experts provided to explain each quality check. Because only the reports evaluated in our benchmark exercise carry labels and reasoning, the labeled knowledge base and reasoning data are retrieved from the 258 expert-evaluated reports, whereas the raw knowledge base is retrieved from the firm's full archive. This difference in pools mirrors the choice firms actually face: they hold a large archive of unlabeled reports but can label and elicit reasoning for only a limited number.

Motivated by the research on human learning discussed in Section 2, we expect that incorporating the labeled knowledge base may make learning more efficient than relying on the knowledge base alone, which consists mostly of positive examples. Incorporating reasoning data may further enhance learning efficiency. Therefore, the LLM is prompted under five conditions (shown in Figure 4): no additional data; the addition of the knowledge base; the addition of the knowledge base and labeled knowledge base; the addition of the knowledge base, labeled knowledge base, and reasoning data; and the addition of reasoning data. The final condition adds retrieved reports with their reasoning data but without quality-check labels, which lets us test whether reasoning data are effective without these labels.

To incorporate the knowledge base and reasoning data, we follow the inference-time scaling literature in RAG (Yue et al., 2025) and vary the amount of data retrieved and added to the prompt. We begin by computing embeddings using the Qwen3-Embedding-0.6B model for all 258 reports that were evaluated by key decision makers.[8] We also compute embeddings for every report in the knowledge base. To retrieve knowledge base data, we calculate the cosine similarity between the report being evaluated and all reports in the knowledge base, and then select the most similar reports to include in the prompt. To retrieve labeled knowledge base data or reasoning data, we compute cosine similarity between the report being evaluated and the set of other reports for which key decision makers provided quality checks, and then select the most similar reports. We include their quality-check labels, reasoning data, or both, depending on the comparison group. The report being evaluated, together with the other

[8]Qwen3-Embedding-0.6B is one of the top embedding models at the time we deployed the LLMs. See also its ranking on the MTEB leaderboard at https://huggingface.co/spaces/mteb/leaderboard.

versions of the same deal, is always excluded from the retrieval pool, a leave-one-deal-out procedure that prevents any label or reasoning from the focal deal from ever entering the prompt.

For all comparison groups other than the baseline (**NoData**) group, we vary the amount of additional data included in the prompt by setting k to 2, 4, 8, 16, 32, or 64. Across all non-baseline groups, the total number of retrieved reports is held fixed at k. What differs is their composition and the information attached to each report. For each k, the addition of data for each comparison group is as follows:

- **Knowledge Base:** Add the top k retrieved knowledge base data points.
- **Knowledge Base and Labeled Knowledge Base:** Add the top $k/2$ retrieved knowledge base data points and $k/2$ retrieved labeled knowledge base data points.
- **Knowledge Base, Labeled Knowledge Base, and Reasoning Data:** Add the top $k/2$ retrieved knowledge base data points and $k/2$ retrieved labeled knowledge base data points together with their reasoning data.
- **Reasoning Data:** Add the top k retrieved reports together with their reasoning data, without quality-check labels.

To examine the role of model size, we also select appropriate families of LLMs. Because all consulting reports, deal information, and reasoning are sensitive proprietary data, all LLMs are deployed locally. We therefore use state-of-the-art open-source LLMs that rank highly on public benchmarks. To study how model size contributes to the alignment of LLMs, we include both large and small models from each LLM family. Specifically, we use GPT-OSS in 120B and 20B parameter versions, DeepSeek-R1 in 671B and 8B versions, and Llama 3.1 in 70B and 8B versions.[9]

After retrieving data and constructing the prompts, we apply the LLM to perform the quality check for each of the 258 expert-evaluated reports. We then evaluate the performance of the LLM by measuring the alignment between key decision makers' quality check results and the LLM's quality check results for each report. For each request, we define alignment as a binary indicator equal to one if the LLM selects the same response as the key decision maker (Yes, Indifferent, or No) for that report, and zero otherwise.[10]

[9]We used the open-source LLMs ranked highest in the benchmark at the time when we deployed them.

[10]Each report was evaluated by one key decision maker rather than by several. We treat the key decision maker's judgment as the benchmark for two reasons. First, these experts, the firm's most senior analysts and executives, set and enforce the firm's quality standard, so aligning with them is exactly what the firm wants from a deployed LLM. Second, like many high-level knowledge tasks, the quality check has no objective ground truth. Key decision makers' judgment is the operative standard against which output quality is assessed, both within the firm and, more broadly, in how subjective tasks are evaluated in practice.

To analyze how data types, data sizes, and model sizes affect LLM alignment, we estimate linear probability models (LPMs) in which the dependent variable is the LLM's alignment with key decision makers in the quality check task. We use LPMs rather than nonlinear models such as logit because the coefficients from LPMs can be interpreted directly as percentage-point changes in alignment. We cluster standard errors at the report level, since each report contributes multiple observations across comparison groups, retrieval sizes, and model sizes.

We estimate the following linear probability model with report fixed effects for each model family:

$$\begin{aligned} Alignment_{mrk} = \beta_0 & \\ & + \beta_1 \, KB_{mrk} \\ & + \beta_2 \, KB_LabeledKB_{mrk} \\ & + \beta_3 \, KB_LabeledKB_RD_{mrk} \\ & + \beta_4 \, RD_{mrk} \\ & + \beta_L \, LargeModel_{mrk} \\ & + \alpha_r + \varepsilon_{mrk}. \end{aligned} \tag{1}$$

where the omitted category ($NoData$) corresponds to the baseline group in which no additional data are included in the prompt. Each indicator variable reflects the type of data provided to the LLM for model size m (the large or small version within the family), report r, and retrieval size k. Specifically, KB_{mrk} indicates that only the knowledge base is included; $KB_LabeledKB_{mrk}$ indicates that both the knowledge base and the labeled knowledge base are included; $KB_LabeledKB_RD_{mrk}$ indicates that the knowledge base, labeled knowledge base, and reasoning data are included; and RD_{mrk} indicates that only reasoning data are included.

For all non-baseline groups, $k \in \{2, 4, 8, 16, 32, 64\}$ denotes the total number of retrieved reports included in the prompt. When the raw and labeled knowledge bases are used together, $k/2$ reports are retrieved from each pool. In the $KB_LabeledKB_RD$ group, reasoning data are attached to the labeled reports. Because each indicator pools all six values of k, the coefficients β_1 through β_4 measure the average effect of each comparison group relative to NoData across retrieval sizes. We examine how the effect varies with k in Section 4. $Alignment_{mrk}$ is a binary indicator equal to one if the LLM's quality check agrees with the key decision maker's quality check on report r. $LargeModel_{mrk}$ indicates whether the large

version of the model family is used. α_r denotes report fixed effects, and ε_{mrk} is the error term.[11]

4 Results

4.1 Knowledge Base vs. Reasoning Data

Table 2. The Effects of Adding Knowledge Base and Reasoning Data

Variable	GPT-OSS Family	DeepSeek-R1 Family	Llama 3.1 Family
KB	0.018	0.010	-0.033
	(0.011)	(0.012)	(0.036)
KB_LabeledKB	0.031*	0.113***	0.097***
	(0.016)	(0.010)	(0.033)
KB_LabeledKB_RD	0.116***	0.179***	0.250***
	(0.020)	(0.013)	(0.029)
RD	0.015***	0.115***	0.198***
	(0.004)	(0.010)	(0.022)
LargeModel	0.021***	0.038	0.025**
	(0.003)	(0.031)	(0.010)
Report FEs	Yes	Yes	Yes
R-squared (Overall)	0.812	0.401	0.369
Observations	12,900	12,900	12,900

Note: NoData is the reference group. The dependent variable is Alignment. The reported R^2 is the overall R^2, which includes the variation absorbed by the report fixed effects. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$.

Table 2 reports the effects of adding different types of data on the LLM's alignment across model families. The results show that adding the knowledge base, which mainly consists of positive samples, does not significantly improve alignment compared to providing no additional data. However, relative to NoData, alignment increases by 3.1 percentage points (weakly significant) in the GPT-OSS family, 11.3 percentage points in the DeepSeek-R1 family, and 9.7 percentage points in the Llama 3.1 family when both the knowledge base and labeled knowledge base are included. This suggests that providing labels helps LLMs achieve higher alignment. When the knowledge base, labeled knowledge base, and reasoning

[11] Each observation corresponds to a unique combination of report, model size, comparison group, and retrieval size. For each model family, the 258 reports, two model sizes, and 25 prompt configurations (the *NoData* baseline plus six values of k for each of the four non-baseline comparison groups) yield $258 \times 2 \times 25 = 12{,}900$ observations. To account for the stochasticity of LLM outputs, we run each quality-check request 10 times, which yields 129,000 LLM quality checks per model family, and use the mean alignment across the 10 repetitions as the dependent variable in the regressions.

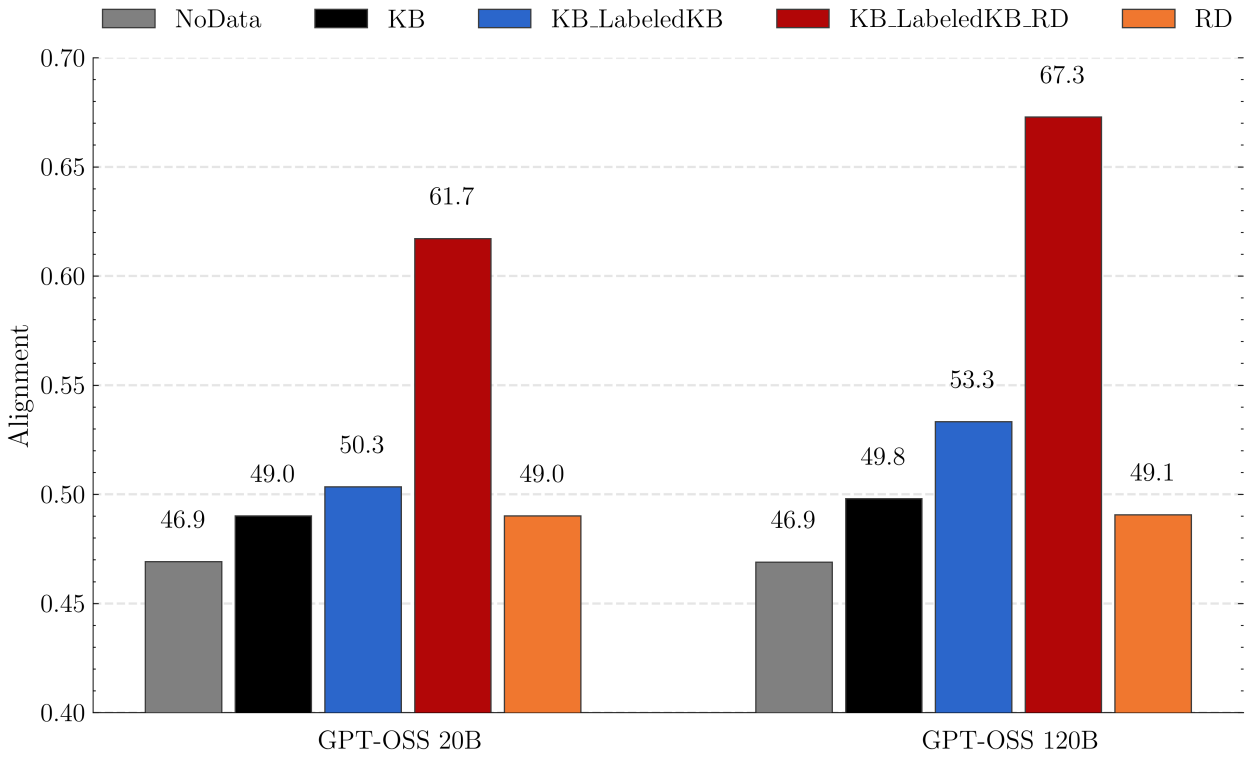
data are all provided, alignment improves even further, resulting in gains of 11.6 percentage points, 17.9 percentage points, and 25.0 percentage points in the GPT-OSS, DeepSeek-R1, and Llama 3.1 families, respectively. Table 12 uses the addition of the knowledge base and labeled knowledge base as the reference group to show the incremental effect of reasoning data, revealing that adding reasoning data further boosts alignment by 8.6 percentage points in GPT-OSS, 6.6 percentage points in DeepSeek-R1, and 15.3 percentage points in Llama 3.1. These improvements are substantial, especially since the regressions pool cases with both small k, where improvements are typically modest, and large k, where the gains are much greater. Finally, when reasoning data are added without quality check labels (the RD group), the effect remains positive and significant but is smaller than that of the KB_LabeledKB_RD group and varies more across models (1.5 percentage points in GPT-OSS, 11.5 in DeepSeek-R1, and 19.8 in Llama 3.1). Reasoning data thus work on their own, but they work best when paired with quality check labels.

The overall R^2 differs across columns because it is driven by the report fixed effects.[12] GPT-OSS answers "No" in every *NoData* run and repeats the same quality check results for 84.7% of the observations of a given report even after data are added (versus 65.3% for DeepSeek-R1 and 61.7% for Llama 3.1), so its alignment is largely determined by the report and absorbed by the fixed effects, whereas the other two families shift their quality check results across comparison groups and therefore have a lower overall R^2.

To compare the best alignment each comparison group can reach across k, we present Figure 5, which reports the highest alignment over the six values of k for each group. This figure shows that the KB_LabeledKB_RD setup achieves a peak alignment of 61.7% for the 20B model and 67.3% for the 120B model, markedly surpassing both the baseline (46.9%) and the raw knowledge base group (49.0%–49.8%). Figures 9 and 10 present the peak alignment for the DeepSeek-R1 and Llama 3.1 families, respectively, revealing results consistent with those of the GPT-OSS family. Because we take the maximum over six values of k, these peak figures should be read as best-case comparisons. The regression estimates above, which pool all k, give the average effect. Across models, peak alignment in the KB_LabeledKB_RD group exceeds peak alignment in the raw knowledge base group by approximately 10.8 to 37.2 percentage points. Expert knowledge thus pushes the performance frontier well beyond the level that the common recipe reaches.

[12] Conditioning on the report fixed effects, the within-R^2 is 0.049 for GPT-OSS, 0.038 for DeepSeek-R1, and 0.092 for Llama 3.1.

Figure 5. Peak Alignment of the GPT-OSS Family for Each Comparison Group



Note: For each comparison group in each model, we report the highest alignment achieved across all values of k.

4.2 Scaling Model Size and Data Size

We first examine the impact of model size. Table 2 shows that large models achieve slightly higher alignment than small models, with increases of 2.1 percentage points for the GPT-OSS family and 2.5 percentage points for the Llama 3.1 family, but the effect is not significant for the DeepSeek-R1 family.

Table 3. The Effects of Number of Added Data Points on Alignment (GPT-OSS 120B)

Variable	KB	KB_LabeledKB	KB_LabeledKB_RD	RD
log(k + 1)	0.005	0.011**	0.052***	0.004***
	(0.004)	(0.005)	(0.006)	(0.001)
Report FEs	Yes	Yes	Yes	Yes
R-squared (Overall)	0.946	0.852	0.731	0.987
Observations	1806	1806	1806	1806

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$.

We then examine the impact of data size. In the literature on scaling laws, model performance often improves with data size following a power-law relationship. Because alignment is a bounded accuracy measure rather than a loss, we test for the natural analog: whether alignment rises linearly in the log of the data count. We take the logarithm of the number of added data points plus one, $\log(\mathrm{k}+1)$, and regress the level of alignment on this transformed variable, a linear-log specification. Below is the regression equation:

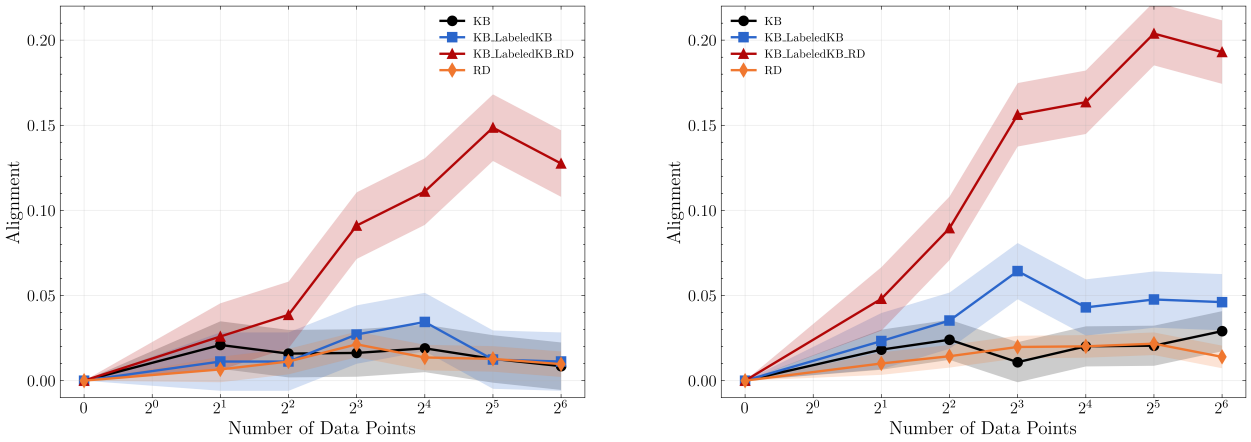
$$\text{Alignment}_r = \beta_0 + \beta_1 \log(k_r + 1) + \alpha_r + \varepsilon_r \tag{2}$$

We estimate the regression for each comparison group and model, where $k = 0$ (the NoData baseline) is included in every regression. Here, $\log(k_r + 1)$ is the log-transformed count of added data points.

Table 3 summarizes these results for GPT-OSS 120B. The KB_LabeledKB_RD group has by far the steepest slope: a one-unit increase in $\log(k+1)$ is associated with a 5.2-percentage-point increase in alignment. Scaling the raw knowledge base (KB) has no significant effect (0.005), so we find no evidence that adding more unlabeled reports improves alignment. Scaling reasoning data alone (RD) yields a significant but much flatter slope (0.004), again showing that reasoning data scale best when paired with labels. Table 13 shows similar results for GPT-OSS 20B. The slope for the labeled knowledge base (KB_LabeledKB) is significant in the 120B model but not in the 20B model.

To investigate whether alignment improvements plateau as more data are added, we

Figure 6. The Effect of Adding k Data Points on Alignment of LLMs (GPT-OSS Family)



(a) GPT-OSS 20B (b) GPT-OSS 120B

replace the continuous measure of $\log(k+1)$ with a series of dummy variables corresponding to each value of k. Below is the regression equation:

$$\text{Alignment}_r = \beta_0 + \sum_k \beta_k D_{k,r} + \alpha_r + \varepsilon_r \tag{3}$$

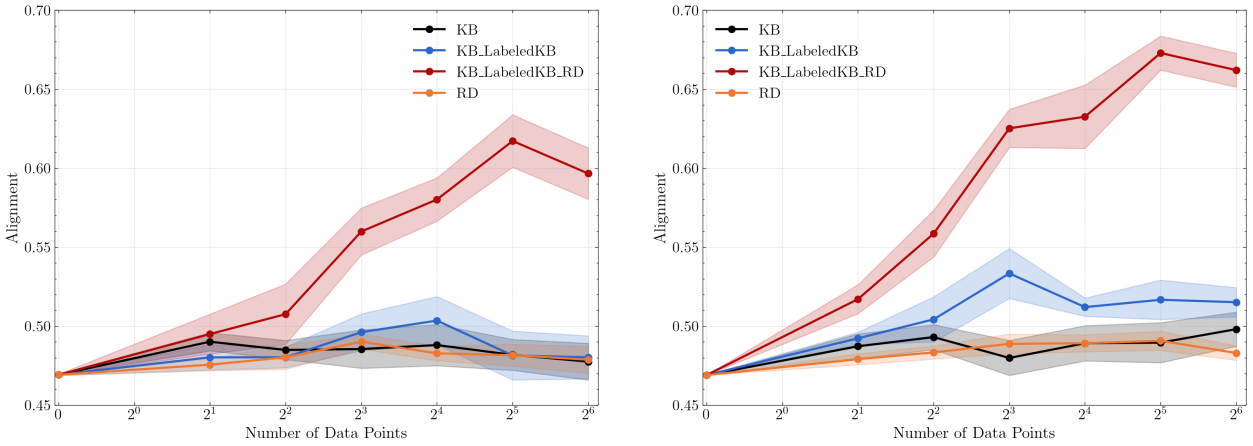
We run the regression for each comparison group and model, with NoData always used as the reference group. Here, $D_{k,r}$ are dummy variables indicating each value of k (the number of data points added).

We then plot the coefficients of the dummies. In this analysis, the x-axis represents k (in log scale) and the y-axis represents the coefficients (the estimated alignment gains relative to No-Data). As illustrated in Figure 6 and Figure 11, adding reasoning data (KB_LabeledKB_RD group) enables LLMs to achieve substantially higher alignment compared to adding only the knowledge base and labeled knowledge base (KB_LabeledKB group). The figures also show that alignment rises with k up to a peak between 16 and 32 added data points and then declines.

This decline is likely attributable to the long-context limitations of LLMs, as the existing literature documents that model performance deteriorates when the prompt becomes excessively long (Liu et al., 2025). Since the quality check task is inherently complex and the consulting reports are typically lengthy, adding too many data points rapidly pushes the length of the prompt toward the model's context limit, thereby reducing alignment. Consistent with this interpretation, in Appendix J we replicate the exercise on GLM-5.2, a more recent and more capable model with a substantially longer context window, and find that alignment keeps improving through 2^6 data points and turns down only slightly at 2^7.

A model with stronger long-context ability thus pushes the turning point outward, indicating that where the curve plateaus is governed by the model's capacity to handle long prompts rather than by the exhaustion of the value of reasoning data.

Figure 7. Average Alignment of LLMs When Adding k Data Points (GPT-OSS Family)



(a) GPT-OSS 20B

(b) GPT-OSS 120B

Our results also suggest that reasoning data can substitute for model scale. Figure 7 and Figure 12 plot the average alignment at each value of k for both large and small models. A small model supplied with reasoning data (KB_LabeledKB_RD group) reaches higher alignment than a large model without it (KB_LabeledKB group), and scaling reasoning data then shifts the scaling curve upward and pushes alignment to new heights. For a firm, investing in reasoning data dominates investing in a larger model in terms of benefits.

To examine the relationship between alignment and inference compute, we also plot alignment against the average number of tokens for each comparison group in each model. Figure 13 shows a similar linear-log scaling relationship when scaling reasoning data together with a labeled knowledge base (KB_LabeledKB_RD).

5 Mechanisms

The previous section showed that the raw knowledge base holds the performance frontier down, whereas expert reasoning data push it out. In this section, we examine the mechanisms behind these results: why the knowledge base does not improve alignment, and why reasoning data do. We focus the mechanism analysis on the GPT-OSS family.

5.1 Why Does the Knowledge Base Not Work?

In Table 1, the human-generated reports evaluated in the benchmark exercise were randomly sampled from past delivered reports—the same population that fills the knowledge base. Even among these delivered reports, experts reject 27.5% and rate another 21.6% as indifferent. The knowledge base therefore contains a sizable share of low-quality reports, which introduces considerable noise for the LLM to learn from. This is an important reason why the knowledge base alone does not substantially improve alignment.

To obtain more robust results, we investigate whether varying the composition of the knowledge base leads to significant differences in alignment. First, we randomly sample half of the reports in the knowledge base five separate times, using each subset as the knowledge base to deploy the LLM via RAG as before. We then compare the alignment across these different subsets.

Next, we test whether incorporating more recent reports into the knowledge base can enhance the LLM's alignment, under the assumption that newer reports may better reflect the current standards of the firm. To do this, we divide the knowledge base into the newest 50% of reports and the oldest 50%, use each as the knowledge base to deploy the LLM with RAG in the same way, and compare their respective alignments.

Table 4. The Effect of Adding Knowledge Base Samples (GPT-OSS Family)

Variable	Sample_1	Sample_2	Sample_3	Sample_4	Sample_5
AddingData	0.010	0.019*	0.012	0.016	0.010
	(0.011)	(0.011)	(0.011)	(0.011)	(0.012)
LargeModel	0.000	0.003	-0.001	0.015*	0.005
	(0.006)	(0.007)	(0.008)	(0.008)	(0.008)
Report FEs	Yes	Yes	Yes	Yes	Yes
R-squared (Overall)	0.818	0.830	0.814	0.817	0.815
Observations	3612	3612	3612	3612	3612

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. The regression specification for each column is: $\text{Alignment}_{mrk} = \beta_0 + \beta_1 \text{AddingData}_{mrk} + \beta_L \text{LargeModel}_{mrk} + \alpha_r + \varepsilon_{mrk}$. For each column, the reference group is NoData. AddingData = 0 if it is NoData, and 1 if data are added as indicated in the column.

Table 4 shows that different samplings of the knowledge base do not significantly affect the alignment of the LLM. Similarly, as seen in Table 5, using a knowledge base consisting of either older or newer reports does not result in significant differences in alignment. These two tests rule out two explanations for the null result: that it is an artifact of which reports we happen to retrieve, or that the archive is simply outdated. As the 27.5% rejection rate shows, even delivered reports include many that experts would reject, and this noise is present across

Table 5. The Effect of Adding Knowledge Base Samples from Different Time Periods (GPT-OSS Family)

Variable	Older Knowledge Base	Newer Knowledge Base
AddingData	0.011	0.013
	(0.010)	(0.013)
LargeModel	-0.001	0.011
	(0.007)	(0.007)
Report FEs	Yes	Yes
R-squared (Overall)	0.850	0.807
Observations	3612	3612

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. The regression specification for each column is: $\text{Alignment}_{mrk} = \beta_0 + \beta_1 \text{AddingData}_{mrk} + \beta_L \text{LargeModel}_{mrk} + \alpha_r + \varepsilon_{mrk}$. For each column, the reference group is NoData. AddingData = 0 if it is NoData and 1 if data are added as indicated in the column.

samples and time periods, which makes it hard for the LLM to learn a clean standard from the knowledge base alone.

5.2 Why Do Reasoning Data Work?

We then investigate why reasoning data work. We show that the central mechanism is that the expert comments tell the LLM directly which patterns make a report good or bad, information that the knowledge base leaves the LLM to infer on its own. The rest of this section builds this case step by step: the improvement comes from the comments rather than the ratings, it is largest when the comments describe patterns that also appear in the report under review, it is driven by the content rather than the style of the comments, and LLM-generated comments do not match the gains from expert comments.

5.2.1 Rating vs. Comment

We first examine which component of reasoning data contributes more to alignment. When we provide the LLM with reasoning data, it receives two components. The first is a set of ratings on different dimensions of the report. The second is a text comment, which explains in detail why the expert made that particular quality check. To isolate the contribution of each component, we provide the LLM with the knowledge base and labeled knowledge base plus only ratings or only comments, and then compare alignment. One difference between ratings and comments is that ratings are relatively generic. For example, ratings might assess feasibility but do not explain why or in which respects a strategy is feasible or not. In contrast, comments are more detailed and explain why each strategy is good or bad.

Table 6. The Effect of Adding Different Types of Reasoning Data (GPT-OSS Family)

Variable	KB_LabeledKB	KB_LabeledKB _Ratings	KB_LabeledKB _Comments	KB_LabeledKB _RD
AddingData	0.031*	0.047***	0.083***	0.116***
	(0.016)	(0.017)	(0.017)	(0.020)
LargeModel	0.022***	0.051***	0.046***	0.044***
	(0.005)	(0.009)	(0.008)	(0.005)
Report FEs	Yes	Yes	Yes	Yes
R-squared (Overall)	0.845	0.723	0.740	0.717
Observations	3612	3612	3612	3612

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. The regression specification for each column is: $\text{Alignment}_{mrk} = \beta_0 + \beta_1 \text{AddingData}_{mrk} + \beta_L \text{LargeModel}_{mrk} + \alpha_r + \varepsilon_{mrk}$. For each column, the reference group is NoData. AddingData = 0 if it is NoData and 1 if data are added as indicated in the column. We also run a regression using KB_LabeledKB_Ratings as the reference group, which shows that KB_LabeledKB_Comments can significantly improve alignment compared to KB_LabeledKB_Ratings.

In Table 6, each column reports the effect of adding the indicated data relative to NoData. The first column, which adds only the knowledge base and labeled knowledge base, provides a benchmark for assessing the incremental contribution of ratings and comments. Building on this, the second column incorporates ratings in addition to the baseline, resulting in only a small improvement in alignment (from 3.1 percentage points to 4.7 percentage points). In contrast, the third column adds comments to the baseline, leading to a much larger improvement (from 3.1 to 8.3 percentage points), which captures most of the gain from adding full reasoning data (11.6 percentage points). These results show that the detailed comments, not the ratings, are the primary factor driving the effect of reasoning data.

5.2.2 Similar Patterns

Next, we investigate why these comments improve the LLM's alignment. A potential mechanism is that these comments directly provide information about patterns of good and bad reports, unlike the knowledge base, where LLMs must identify patterns themselves. Under this mechanism, reasoning data should be more effective if the patterns described in the comments resemble those present in the testing report (the report the LLM is quality checking). We use the comment on the testing report as an indicator of these patterns (since the comment mentions which patterns make the report good or bad). For each testing report, we calculate the average similarity (ranging from 0 to 1) between the comment on the testing report and each of the comments from the reasoning data, and we use this average similarity to predict the LLM's alignment. Table 7 shows that average similarity strongly predicts the LLM's alignment: the coefficient implies that moving across the full similarity range from 0

Table 7. Regression of Alignment on the Average Similarity Between the Comment of the Current Report and the Comments of the Added Reports (GPT-OSS Family)

Variable	Alignment
AvgSimilarity	0.753^{***}
	(0.153)
Large Model	0.052^{***}
	(0.006)
Report FEs	Yes
R-squared (Overall)	0.747
Observations	3096

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. The regression specification is: $\text{Alignment}_{mrk} = \beta_0 + \beta_1 \text{AvgSimilarity}_{mrk} + \beta_L \, \text{LargeModel}_{mrk} + \alpha_r + \varepsilon_{mrk}$. The sample used in the regression is the KB_LabeledKB_RD group.

Table 8. More Similar Comments vs. Less Similar Comments (GPT-OSS Family)

Variable	KB_LabeledKB	KB_LabeledKB _LessSimilar Comments	KB_LabeledKB _MoreSimilar Comments	KB_LabeledKB _Comments
AddingData	0.031^{*}	0.043^{**}	0.058^{***}	0.083^{***}
	(0.016)	(0.018)	(0.015)	(0.017)
LargeModel	0.022^{***}	0.046^{***}	0.049^{***}	0.046^{***}
	(0.005)	(0.009)	(0.008)	(0.008)
Report FEs	Yes	Yes	Yes	Yes
R-squared (Overall)	0.845	0.673	0.767	0.740
Observations	3612	3612	3612	3612

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. The regression specification for each column is: $\text{Alignment}_{mrk} = \beta_0 + \beta_1 \text{AddingData}_{mrk} + \beta_L \, \text{LargeModel}_{mrk} + \alpha_r + \varepsilon_{mrk}$. For each column, the reference group is NoData. AddingData = 0 if it is NoData and 1 if data are added as indicated in the column. We also run a regression using KB_LabeledKB_LessSimilar as the reference group, which shows that KB_LabeledKB_MoreSimilar can significantly improve alignment compared to KB_LabeledKB_LessSimilar.

to 1 raises alignment by 75.3 percentage points. Observed similarity varies over a narrower band, so the realized differences are smaller, but the association is strong and points the same way: when the added reasoning data describe patterns found in the report under review, the LLM's alignment improves.

We further verify this mechanism. For each testing report, we calculate the similarity between the comment on the testing report and each of the comments from the added reasoning data. We then divide the added comments into two groups: "MoreSimilarComments," which includes only the 50% of comments most similar to the testing report's comment, and "LessSimilarComments," which includes the remaining 50% least similar comments. For each group, we compare the impact of adding the data on alignment. As shown in Table 8, the LessSimilarComments group yields only a small improvement compared to not adding comments, whereas the MoreSimilarComments group achieves a much larger improvement, closer to the effect of adding all comments. This result indicates that reasoning data benefit LLMs mainly by providing information about patterns of good or bad reports, and that the largest gains occur when these patterns are relevant to the report under review. In Appendix F, we provide additional evidence supporting this mechanism.

5.2.3 Comments About the Content of the Report vs. Comments About the Style of the Report

To further investigate what type of content in comments contributes to the alignment improvement, we examine whether comments about the style of the report or the content of the report are more effective. We split each comment into sentences and use a top open-source model (GPT-OSS 120B) to classify each sentence as relating to style, content, or both. We then construct two groups: for the CommentsAboutStyle group, we add only those sentences classified as about style to the reasoning data prompt; for the CommentsAboutContent group, we add only the sentences classified as about content. As shown in Table 9, adding comments about the content of the report results in a much larger improvement in alignment than adding comments about the style of the report (and is close to the effect of adding all comments). This suggests that comments about the content of the report are the primary factor driving the effect of adding comments.

5.2.4 Expert-Generated Reasoning vs. LLM-Generated Reasoning

LLMs can also be used to generate reasoning data, raising the question of whether LLM-generated reasoning data can be as effective as expert-generated reasoning data. To investigate this, we first use the LLM to generate the reasoning data: for all 258 quality-checked samples,

Table 9. Comments About the Style of the Report vs. Comments About the Content of the Report (GPT-OSS Family)

Variable	KB_LabeledKB	KB_LabeledKB _CommentsAbout Style	KB_LabeledKB _CommentsAbout Content	KB_LabeledKB _Comments
AddingData	0.031*	0.044**	0.072***	0.083***
	(0.016)	(0.018)	(0.019)	(0.017)
LargeModel	0.022***	0.039***	0.045***	0.046***
	(0.005)	(0.011)	(0.011)	(0.008)
Report FEs	Yes	Yes	Yes	Yes
R-squared (Overall)	0.845	0.587	0.562	0.740
Observations	3612	3612	3612	3612

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. The regression specification for each column is: $\text{Alignment}_{mrk} = \beta_0 + \beta_1 \text{AddingData}_{mrk} + \beta_L \text{LargeModel}_{mrk} + \alpha_r + \varepsilon_{mrk}$. For each column, the reference group is NoData. AddingData = 0 if it is NoData and 1 if data are added as indicated in the column. We also run a regression using KB_LabeledKB_CommentsAboutStyle as the reference group, which shows that KB_LabeledKB_CommentsAboutContent can significantly improve alignment compared to KB_LabeledKB_CommentsAboutStyle.

we provide the report, the quality check result, and the ratings to the LLM and prompt it to generate a comment explaining each quality check. We then add these LLM-generated comments to the knowledge base and labeled knowledge base and compare alignment with the condition using expert-generated comments.

Because the expert-generated comments are mostly one paragraph in length, we consider two conditions when generating LLM-based comments: one where the response length is explicitly limited to a single paragraph, and another where there is no length restriction. For both conditions, we use the following prompt: "You are provided with a negotiation strategy report, the quality check result from a key decision maker, and their ratings on different dimensions of the report. You need to generate the reasoning behind the expert's quality check result." For the one-paragraph condition, we modify the prompt to specify: "You need to generate the reasoning behind the expert's quality check result in one paragraph." We use a top open-source model (GPT-OSS 120B) to generate the LLM-generated reasoning data.

Table 10 shows that, relative to NoData, LLM-generated reasoning data improve alignment (5.1 percentage points without a length limit and 4.7 with a one-paragraph limit, both significant), but far less than expert-generated reasoning data (8.3 percentage points). LLM-generated reasoning thus captures only part of what makes expert reasoning effective. For robustness, in Appendix H, we use a top commercial model (Gemini 3 Pro) to generate the reasoning data and find consistent results.

To understand why, we use an LLM (GPT-OSS 120B) to decompose both expert-generated

Table 10. The Effect of LLM-Generated Reasoning (GPT-OSS Family)

Variable	KB_LabeledKB	KB_LabeledKB _LLMReasoning	KB_LabeledKB _LLMReasoning OneParagraph	KB_LabeledKB _Comments
AddingData	0.031*	0.051***	0.047***	0.083***
	(0.016)	(0.019)	(0.018)	(0.017)
LargeModel	0.022***	0.044***	0.023**	0.046***
	(0.005)	(0.011)	(0.011)	(0.008)
Report FEs	Yes	Yes	Yes	Yes
R-squared (Overall)	0.845	0.600	0.591	0.740
Observations	3612	3612	3612	3612

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. The regression specification for each column is: $\text{Alignment}_{mrk} = \beta_0 + \beta_1 \text{AddingData}_{mrk} + \beta_L \text{LargeModel}_{mrk} + \alpha_r + \varepsilon_{mrk}$. For each column, the reference group is NoData. AddingData = 0 if it is NoData and 1 if data are added as indicated in the column. We also run a regression using KB_LabeledKB_LLMReasoning as the reference group, which shows that KB_LabeledKB_Comments can significantly improve alignment compared to KB_LabeledKB_LLMReasoning.

and LLM-generated reasoning into individual points, and then count the points that appear in one but not the other. Two patterns emerge from Figure 15 and Figure 16. First, the expert-generated reasoning is short and selective, with an average of 3.21 points, and many of these points are missing from the LLM-generated reasoning, so the LLM fails to surface patterns the experts judged important. Second, the LLM-generated reasoning is far longer, with an average of 10.79 points without a length limit and 9.37 points with a one-paragraph limit, and most of these extra points do not appear in the expert reasoning. One interpretation is that these extra points add noise that crowds out the patterns that matter, whereas experts report only the few points that drive their judgment.

6 Economic Implications

Reasoning data raise alignment, but collecting them is costly. We now use a benefit-cost analysis to ask whether the alignment gains are large enough to justify that cost. In practice, complex and high-stakes tasks typically require humans in the loop. Experts need to review the LLM's output. In our case, without LLM deployment, quality checks are performed by the firm's key decision makers. To our knowledge, a reasonable estimate of their hourly wage is around 500 USD, and completing a quality check takes approximately 20 minutes. Table 11 shows the benefit-cost matrix for LLM quality check and expert review outcomes. Producing a label requires an expert to perform a quality check on the report, so labeling one knowledge base report costs the same as one quality check (20 minutes). Based on

estimates provided by the partner firm, collecting reasoning data requires approximately 5 additional minutes per check (for a total of 25 minutes) and reviewing the LLM's quality check result takes approximately 3 minutes. There is also a token cost that varies across models: GPT-OSS 120B costs 0.15 USD per 1M input tokens and 0.6 USD per 1M output tokens, while GPT-OSS 20B costs 0.05 USD per 1M input tokens and 0.2 USD per 1M output tokens.[13]

Table 11. Benefit-Cost Matrix for LLM Quality Check and Expert Review Outcomes

LLM QC Result	**Expert Reviewer Decision**	
	Approve	**Reject**
Correct	Labor savings	0 (Redo by expert)
Incorrect	$-2\times$ Labor cost (Low quality report delivered or unnecessary rewrite)	0 (Redo by expert)

Note: An LLM quality check is considered correct when it agrees with the key decision maker's benchmark judgment. Delivering a low-quality report or performing an unnecessary rewrite incurs a substantial cost. We assume this cost is twice the labor cost required to perform a quality check. The payoffs shown exclude the review, token, and data-collection costs, which are deducted separately.

We assume that the reviewer approves a correct LLM quality check and rejects an incorrect one with the same probability, which we refer to as review accuracy. The matrix maps each combination of the LLM's result and the reviewer's decision to a payoff. When the LLM is correct and the reviewer approves, the firm saves labor, because it pays only for a 3-minute review instead of a full 20-minute quality check. When the reviewer rejects the LLM's result, the quality check is redone by an expert, leaving no labor savings relative to expert-only quality checks before the additional review, token, and data-collection costs are deducted. When the LLM is incorrect and the reviewer mistakenly approves, we assume a cost equal to twice the labor cost of a quality check, representing losses from delivering a low-quality report or performing an unnecessary rewrite. Net value therefore rises with the LLM's alignment: higher alignment produces more correct results for reviewers to approve and fewer incorrect results that slip through.

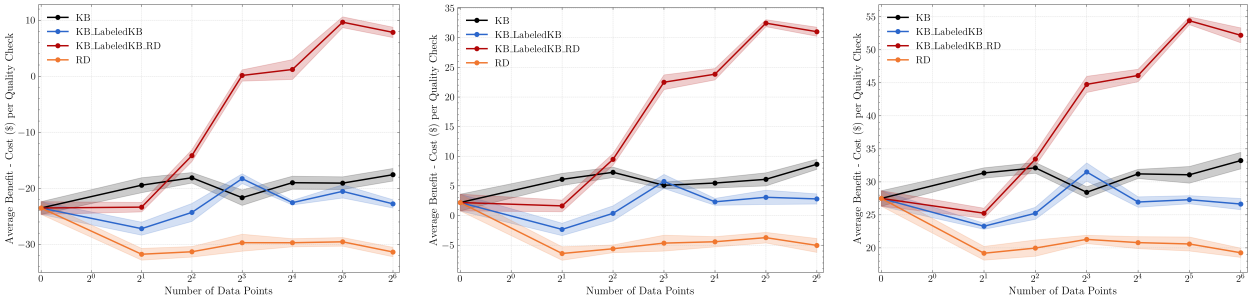
The cost of collecting reasoning data is incurred once, but the data can be reused across many quality checks. We assume the 258 reasoning data points can be reused for at least 5,000 quality checks before they need to be updated.[14] The corresponding one-time costs

[13]We obtained the price information from the cloud LLM marketplace Together AI.

[14]The reasoning data encode the experts' standards for what distinguishes a good report from a bad one,

of collecting labels and reasoning data are spread over these 5,000 quality checks. We then simulate 5,000 quality checks by resampling with replacement from the collected data, and plot the average net value (benefit minus cost) per quality check against the number of added data points for each comparison group and model, at different levels of expert reviewer accuracy. Figure 8 shows that using an LLM without additional data to perform quality checks results in a loss when the review accuracy is 0.7, and adding knowledge base data does not substantially improve this outcome. In contrast, scaling reasoning data can create substantial value and generate profits for the firm. When the review accuracy is higher (0.8 or 0.9), scaling reasoning data generates substantially greater profits. Figure 14 shows similar results for the GPT-OSS 20B model.

Figure 8. Net Value per Quality Check (GPT-OSS 120B)



(a) Review Accuracy: 0.7 (b) Review Accuracy: 0.8 (c) Review Accuracy: 0.9

Note: Each curve plots the average benefit minus cost (in USD) per quality check for one comparison group as the number of added data points k increases from 0 to 64 (log scale on the horizontal axis), at expert review accuracy of 0.7, 0.8, and 0.9. A curve above zero means the LLM creates value relative to expert-only quality checks. Shaded bands show the confidence interval across the resampled quality checks.

7 Conclusion

"The big breakthrough of pre-training is the realization that this recipe [i.e., scaling laws] is good. . . . Companies love this because it gives you a very low-risk way of investing your resources."

— Ilya Sutskever, co-founder of OpenAI and Safe Superintelligence Inc. and one of the most highly cited computer scientists (2025).[15]

and these standards are stable over a multi-year horizon, so the data can be reused until the firm's standards or market conditions shift materially. Our partner firm completes on the order of 5,000 deals every two to three years, each requiring a quality check, so one collection of 258 reasoning data points serves roughly this many checks before it needs refreshing. The figure of 5,000 is a conservative estimate: it assumes the standards are refreshed every two to three years, and a longer useful life would only raise the net value of reasoning data by spreading the one-time collection cost over more checks.

[15]Source: https://www.youtube.com/watch?v=aR20FWCCjAs

This paper asks whether scaling laws govern enterprise AI. We find that performance can scale predictably, but what a firm scales determines how far performance can go. Under the common recipe, which scales the firm's raw knowledge base and model size, performance reaches a low ceiling quickly: adding more knowledge base reports does not improve alignment with the firm's key decision makers, and larger models bring only limited gains. Expert knowledge shifts the scaling curve itself upward. Experts' quality check labels raise alignment modestly, whereas experts' reasoning data expand the performance frontier substantially, with alignment rising log-linearly in the amount of reasoning data. With labels and reasoning data combined, peak alignment is 10.8 to 37.2 percentage points higher than with the raw knowledge base alone, and a small model supplied with reasoning data outperforms a large model without it. Our benefit-cost analysis shows that these gains are large enough to turn LLM quality checks from a loss into a profitable deployment.

Scaling laws have been powerful in driving the development of LLMs over the last several years, as they inform not just researchers but also managers and investors about how to allocate resources. But previous scaling laws on training do not tell firms how to raise the low performance frontier in deployment: they tell firms to add data and buy larger models, yet failure rates remain high. Our results suggest that two factors hold this frontier down. First, the knowledge base is noisy, which is common in many firms. Second, it is difficult for firms to scale expert knowledge when deploying LLMs, especially the knowledge that resides only in experts' minds. Reasoning data are an effective form of this expert knowledge because they tell the LLM directly which patterns make a report good or bad, information that the knowledge base leaves the model to infer on its own. This is why even a small amount of reasoning data can significantly improve LLMs.

These findings give managers concrete guidance on how to allocate resources in LLM deployment. To create value, firms should invest in scaling expert knowledge rather than in larger models or more historical data alone. Managers also face a trade-off between the one-time cost of collecting reasoning data and the ongoing marginal cost of running larger models, and our results suggest that a smaller model with reasoning data is often the better investment.

Reasoning data rarely exist in current business practice, and collecting them is costly. It's also challenging because some expert knowledge is tacit. One way to lower this cost is to integrate the collection process into the workflow, which raises the question of how to incentivize employees to provide data that improve AI models. Another is to design AI tools that capture reasoning data automatically. For example, when employees collaborate with an LLM chatbot and ask it to refine answers they dislike, the tool can learn the reasoning behind those actions. Beyond collection costs, whether the gains we document transfer across

related tasks, such as writing a report and checking its quality, is an open question for future work.

Finally, scaling expert knowledge may change the nature of organizations. When experts leave, their knowledge is typically lost. By recording their reasoning for use with LLMs or by encoding it in LLMs, firms can retain it, and the moat of a firm may shift from individual experts to models that incorporate the knowledge of previous experts. How this shift reshapes the cost of training new employees and the power structure of organizations is a promising avenue for future research.

References

Alur R, Raghavan M, Shah D (2024) Human expertise in algorithmic prediction. *Advances in Neural Information Processing Systems* 37:138088–138129.

Argote L, Ingram P (2000) Knowledge Transfer: A Basis for Competitive Advantage in Firms. *Organizational Behavior and Human Decision Processes* 82(1):150–169, ISSN 0749-5978, URL http://dx.doi.org/10.1006/obhd.2000.2893.

Autor D (2014) Polanyi's Paradox and the Shape of Employment Growth. URL http://dx.doi.org/10.3386/w20485.

Babina T, Fedyk A, He A, Hodson J (2024) Artificial intelligence, firm growth, and product innovation. *Journal of Financial Economics* 151:103745, ISSN 0304-405X, URL http://dx.doi.org/10.1016/j.jfineco.2023.103745.

Bai Y, Jones A, Ndousse K, Askell A, Chen A, DasSarma N, Drain D, Fort S, Ganguli D, Henighan T, Joseph N, Kadavath S, Kernion J, Conerly T, El-Showk S, Elhage N, Hatfield-Dodds Z, Hernandez D, Hume T, Johnston S, Kravec S, Lovitt L, Nanda N, Olsson C, Amodei D, Brown T, Clark J, McCandlish S, Olah C, Mann B, Kaplan J (2022) Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback. URL http://dx.doi.org/10.48550/arXiv.2204.05862.

Berente N, Gu B, Recker J, Santhanam R (2021) Managing Artificial Intelligence. *MIS Quarterly* 45(3):1433–1450, ISSN 02767783, URL http://dx.doi.org/10.25300/MISQ/2021/16274.

Brown T, Mann B, Ryder N, Subbiah M, Kaplan JD, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, Agarwal S, Herbert-Voss A, Krueger G, Henighan T, Child R, Ramesh A, Ziegler D, Wu J, Winter C, Hesse C, Chen M, Sigler E, Litwin M, Gray S, Chess B, Clark J, Berner C, McCandlish S, Radford A, Sutskever I, Amodei D (2020) Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*, volume 33, 1877–1901 (Curran Associates, Inc.).

Brynjolfsson E (1993) The productivity paradox of information technology. *Communications of the ACM* 36(12):66–77, ISSN 0001-0782, URL http://dx.doi.org/10.1145/163298.163309.

Brynjolfsson E, Rock D, Syverson C (2019) Artificial Intelligence and the Modern Productivity Paradox: A Clash of Expectations and Statistics. Agrawal A, Gans J, Goldfarb A, eds., *The Economics of Artificial Intelligence: An Agenda*, 23–60 (University of Chicago Press), ISBN 978-0-226-61347-5, URL http://dx.doi.org/10.7208/chicago/9780226613475.003.0001.

Brynjolfsson E, Rock D, Syverson C (2021) The Productivity J-Curve: How Intangibles Complement General Purpose Technologies. *American Economic Journal: Macroeconomics* 13(1):333–372, ISSN 1945-7707, URL http://dx.doi.org/10.1257/mac.20180386.

Challapally A, Pease C, Raskar R, Chari P (2025) The GenAI Divide: State of AI in Business 2025. Technical report, MIT NANDA (Project NANDA).

Databricks (2024) State of Data + AI. Technical report, Databricks.

Dell'Acqua F, McFowland E, Mollick E, Lifshitz H, Kellogg KC, Rajendran S, Krayer L, Candelon F, Lakhani KR (2026) Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of Artificial Intelligence on Knowledge Worker Productivity and Quality. *Organization Science* 37(2):403–423, ISSN 1047-7039, URL http://dx.doi.org/10.1287/orsc.2025.21838.

Dietvorst BJ, Simmons JP, Massey C (2015) Algorithm aversion: People erroneously avoid algorithms after seeing them err. *Journal of Experimental Psychology: General* 144(1):114–126, ISSN 1939-2222, URL http://dx.doi.org/10.1037/xge0000033.

Doshi AR, Bell JJ, Mirzayev E, Vanneste BS (2025) Generative artificial intelligence and evaluating strategic decisions. *Strategic Management Journal* 46(3):583–610, ISSN 1097-0266, URL http://dx.doi.org/10.1002/smj.3677.

Doshi AR, Hauser OP (2024) Generative AI enhances individual creativity but reduces the collective diversity of novel content. *Science Advances* 10(28):eadn5290, URL http://dx.doi.org/10.1126/sciadv.adn5290.

Durkin K, Rittle-Johnson B (2012) The effectiveness of using incorrect examples to support learning about decimal magnitude. *Learning and Instruction* 22(3):206–214, ISSN 0959-4752, URL http://dx.doi.org/10.1016/j.learninstruc.2011.11.001.

Gabriel I (2020) Artificial Intelligence, Values, and Alignment. *Minds and Machines* 30(3):411–437, ISSN 1572-8641, URL http://dx.doi.org/10.1007/s11023-020-09539-2.

Goli A, Singh A (2024) Frontiers: Can Large Language Models Capture Human Preferences? *Marketing Science* 43(4):709–722, ISSN 0732-2399, URL http://dx.doi.org/10.1287/mksc.2023.0306.

Grant RM (1996) Toward a knowledge-based theory of the firm. *Strategic Management Journal* 17(S2):109–122, ISSN 1097-0266, URL http://dx.doi.org/10.1002/smj.4250171110.

Hestness J, Narang S, Ardalani N, Diamos G, Jun H, Kianinejad H, Patwary MMA, Yang Y, Zhou Y (2017) Deep Learning Scaling is Predictable, Empirically. URL http://dx.doi.org/10.48550/arXiv.1712.00409.

Hoffmann J, Borgeaud S, Mensch A, Buchatskaya E, Cai T, Rutherford E, de Las Casas D, Hendricks LA, Welbl J, Clark A, Hennigan T, Noland E, Millican K, van den Driessche G, Damoc B, Guy A, Osindero S, Simonyan K, Elsen E, Vinyals O, Rae JW, Sifre L (2022) Training compute-optimal large language models. *Proceedings of the 36th International Conference on Neural Information Processing Systems*, 30016–30030, NIPS '22 (Red Hook, NY, USA: Curran Associates Inc.), ISBN 978-1-7138-7108-8.

Horton JJ (2023) Large Language Models as Simulated Economic Agents: What Can We Learn from Homo Silicus? URL http://dx.doi.org/10.3386/w31122.

Ji J, Qiu T, Chen B, Zhou J, Zhang B, Hong D, Lou H, Wang K, Duan Y, He Z, Vierling L,

Zhang Z, Zeng F, Dai J, Pan X, Xu H, O'Gara A, Ng K, Tse B, Fu J, Mcaleer S, Wang Y, Yang M, Liu Y, Wang Y, Zhu SC, Guo Y, Yang Y, Gao W (2026) AI Alignment: A Contemporary Survey. *ACM Computing Surveys* 58(5):132:1–132:38, ISSN 0360-0300, URL http://dx.doi.org/10.1145/3770749.

Kaplan J, McCandlish S, Henighan T, Brown TB, Chess B, Child R, Gray S, Radford A, Wu J, Amodei D (2020) Scaling Laws for Neural Language Models. URL http://dx.doi.org/10.48550/arXiv.2001.08361.

Law KKF, Shen M (2025) How Does Artificial Intelligence Shape Audit Firms? *Management Science* 71(5):3641–3666, ISSN 0025-1909, URL http://dx.doi.org/10.1287/mnsc.2022.04040.

Lebovitz S, Lifshitz-Assaf H, Levina N (2022) To Engage or Not to Engage with AI for Critical Judgments: How Professionals Deal with Opacity When Using AI for Medical Diagnosis. *Organization Science* 33(1):126–148, ISSN 1047-7039, URL http://dx.doi.org/10.1287/orsc.2021.1549.

Lee G, Greenbaum L (2026) Tracking Trillions: The Assumptions Shaping the Scale of the AI Build-Out. Technical report, Goldman Sachs.

Lewis P, Perez E, Piktus A, Petroni F, Karpukhin V, Goyal N, Küttler H, Lewis M, Yih Wt, Rocktäschel T, Riedel S, Kiela D (2020) Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *Advances in Neural Information Processing Systems*, volume 33, 9459–9474 (Curran Associates, Inc.).

Lin B (2024) From RAGs to Vectors: How Businesses Are Customizing AI Models. *Wall Street Journal* ISSN 0099-9660.

Liu J, Zhu D, Bai Z, He Y, Liao H, Que H, Wang Z, Zhang C, Zhang G, Zhang J, Zhang Y, Chen Z, Guo H, Li S, Liu Z, Shan Y, Song Y, Tian J, Wu W, Zhou Z, Zhu R, Feng J, Gao Y, He S, Li Z, Liu T, Meng F, Su W, Tan Y, Wang Z, Yang J, Ye W, Zheng B, Zhou W, Huang W, Li S, Zhang Z (2025) A Comprehensive Survey on Long Context Language Modeling. URL http://dx.doi.org/10.48550/arXiv.2503.17407.

Liu Y, Yao Y, Ton JF, Zhang X, Guo R, Cheng H, Klochkov Y, Taufiq MF, Li H (2024) Trustworthy LLMs: A Survey and Guideline for Evaluating Large Language Models' Alignment. URL http://dx.doi.org/10.48550/arXiv.2308.05374.

Lu W, Chen DL, Hansen CB (2025) Aligning Large Language Model Agents with Rational and Moral Preferences: A Supervised Fine-Tuning Approach. URL http://dx.doi.org/10.48550/arXiv.2507.20796.

Martin K (2019) Ethical Implications and Accountability of Algorithms. *Journal of Business Ethics* 160(4):835–850, ISSN 1573-0697, URL http://dx.doi.org/10.1007/s10551-018-3921-3.

McElheran K, Yang MJ, Brynjolfsson E, Kroff Z (2025) The Rise of Industrial AI in America: Microfoundations of the Productivity J-curve(s). URL http://dx.doi.org/10.2139/ssrn.5036270.

Morales J (2025) Microsoft CEO says the company doesn't have enough electricity to install all the AI GPUs in its inventory - 'you may actually have a bunch of chips sitting in inventory that I can't plug in'. https://www.tomshardware.com/tech-industry/artificial-intelligence/microsoft-ceo-says-the-company-doesnt-have-enough-electricity-to-install-all-the-ai-gpus-in-its-inventory-you-may-actually-have-a-bunch-of-chips-sitting-in-inventory-that-i-cant-plug-in.

Muennighoff N, Rush A, Barak B, Le Scao T, Tazi N, Piktus A, Pyysalo S, Wolf T, Raffel CA (2023) Scaling Data-Constrained Language Models. *Advances in Neural Information Processing Systems* 36:50358–50376.

Ouyang L, Wu J, Jiang X, Almeida D, Wainwright C, Mishkin P, Zhang C, Agarwal S, Slama K, Ray A, Schulman J, Hilton J, Kelton F, Miller L, Simens M, Askell A, Welinder P, Christiano PF, Leike J, Lowe R (2022) Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems* 35:27730–27744.

Ovadia O, Brief M, Mishaeli M, Elisha O (2024) Fine-Tuning or Retrieval? Comparing Knowledge Injection in LLMs. Al-Onaizan Y, Bansal M, Chen YN, eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 237–250 (Miami, Florida, USA: Association for Computational Linguistics), URL http://dx.doi.org/10.18653/v1/2024.emnlp-main.15.

Peng S, Kalliamvakou E, Cihon P, Demirer M (2023) The Impact of AI on Developer Productivity: Evidence from GitHub Copilot. URL http://dx.doi.org/10.48550/arXiv.2302.06590.

Polanyi M (1966) *The Tacit Dimension* (Garden City, NY: Doubleday).

Snell C, Lee J, Xu K, Kumar A (2025) Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. *International Conference on Learning Representations*, volume 2025, 10131–10165.

Sorscher B, Geirhos R, Shekhar S, Ganguli S, Morcos A (2022) Beyond neural scaling laws: Beating power law scaling via data pruning. *Advances in Neural Information Processing Systems* 35:19523–19536.

Tennyson RD, Woolley FR, Merrill MD (1972) Exemplar and nonexemplar variables which produce correct concept classification behavior and specified classification errors. *Journal of Educational Psychology* 63(2):144–152, ISSN 1939-2176, URL http://dx.doi.org/10.1037/h0032368.

Tully T, Redfern J, Das D, Xiao D (2025) 2025: The State of Generative AI in the Enterprise. Menlo Ventures, https://menlovc.com/perspective/2025-the-state-of-generative-ai-in-the-enterprise/.

Van der Kleij FM, Feskens RCW, Eggen TJHM (2015) Effects of feedback in a computer-based learning environment on students' learning outcomes: A meta-analysis. *Review of Educational Research* 85(4):475–511, ISSN 0034-6543, URL http://dx.doi.org/10.3102/0034654314564881.

Wang Y, Zhong W, Li L, Mi F, Zeng X, Huang W, Shang L, Jiang X, Liu Q (2023) Aligning Large

Language Models with Human: A Survey. URL http://dx.doi.org/10.48550/arXiv.2307.12966.

Wei J, Tay Y, Bommasani R, Raffel C, Zoph B, Borgeaud S, Yogatama D, Bosma M, Zhou D, Metzler D, Chi EH, Hashimoto T, Vinyals O, Liang P, Dean J, Fedus W (2022) Emergent Abilities of Large Language Models. *Transactions on Machine Learning Research* ISSN 2835-8856.

Wittwer J, Renkl A (2010) How Effective are Instructional Explanations in Example-Based Learning? A Meta-Analytic Review. *Educational Psychology Review* 22(4):393–409, ISSN 1040-726X, URL http://dx.doi.org/10.1007/s10648-010-9136-5.

Yue Z, Zhuang H, Bai A, Hui K, Jagerman R, Zeng H, Qin Z, Wang D, Wang X, Bendersky M (2025) Inference Scaling for Long-Context Retrieval Augmented Generation. *International Conference on Learning Representations*, volume 2025, 72914–72938.

Zhou C, Liu P, Xu P, Iyer S, Sun J, Mao Y, Ma X, Efrat A, Yu P, Yu L, Zhang S, Ghosh G, Lewis M, Zettlemoyer L, Levy O (2023) LIMA: Less Is More for Alignment. *Advances in Neural Information Processing Systems* 36:55006–55021, URL http://dx.doi.org/10.52202/075280-2400.

Appendix

A Quality Check and Reasoning Questions

The following questions were used in the experts' quality check benchmark exercise.

1. **Recommendation:** Would you recommend providing these negotiation strategies without modification to the client?

   A. Yes

   B. Indifferent

   C. No

2. **Percentage to be Modified:** What percentage of the content still needs to be modified?

   A. No modifications required (0%)

   B. Minor modifications needed (<10%)

   C. Some modifications needed (10%–30%)

   D. Significant modifications required (30%–60%)

   E. Rewrite (>60%)

3. **Time to Modify:** How much additional time do you expect to spend modifying the content before sharing strategies with the client?

   A. 0

   B. 15m

   C. 30m

   D. 45m

   E. 1h

   F. 2h

4. **Financial Outcome:** How confident are you that the negotiation strategies, as written, would deliver the target savings?
   (Scale: 1 = Low, 5 = High)

5. **Relevance and Specificity:** How relevant and specific are the negotiation strategies?
   (Scale: 1 = Low, 5 = High)

6. **Feasibility:** How would you rate the feasibility of the negotiation strategies?
   (Scale: 1 = Low, 5 = High)

7. **Innovativeness:** How innovative are the negotiation strategies?
   (Scale: 1 = Low, 5 = High)

8. **Clarity and Comprehensibility:** How would you rate the clarity and comprehensibility of the negotiation strategies?
   (Scale: 1 = Low, 5 = High)

9. **Comprehensiveness:** How comprehensive are the negotiation strategies?
   (Scale: 1 = Low, 5 = High)

10. **Overall Quality:** How would you rate the overall quality of the negotiation strategies?
    (Scale: 1 = Low, 5 = High)

11. **Comment:** Please provide additional comments about the negotiation strategies above.

B Prompt Construction

Below we present the prompt text used for the quality check task.

Task: Act as a world-class IT spend management consultant and analyst with more than 20 years of experience. Your specialized focus lies in validating price competitiveness, optimizing licensing agreements, aligning stakeholders within IT procurement teams, quantifying cost savings, and formulating effective negotiation strategies to drive substantial value for businesses. Please complete the task below.

Your team has generated a negotiation strategy report for your company's client who is purchasing from a vendor. Your task is to evaluate the quality of the report.

(Conditional Additional Information, depending on the experimental setting):

- **Knowledge Base:** You are provided with the related past reports, which were recommended and sent out to clients in the past. Please learn from these to better evaluate the current report.
- **Labeled Knowledge Base or Reasoning Data:** You are provided with the past evaluation results for these reports. Please learn from these to better evaluate the current report.

The report to be evaluated is shown below.

`[Text of the report to be evaluated]`

Please evaluate it by answering the below question.

1. **Recommendation:** Would you recommend providing these negotiation strategies without modification to the client?
    - A. Yes
    - B. Indifferent
    - C. No

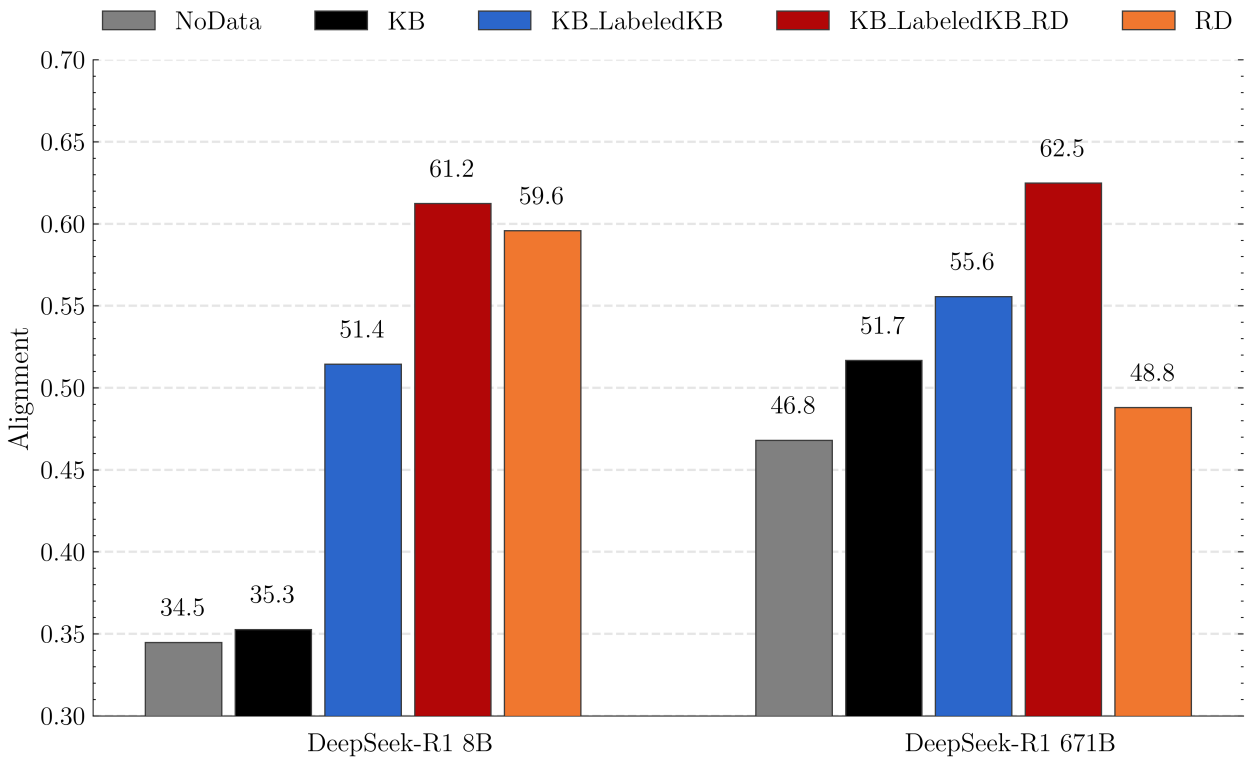
C The Effects of Adding Knowledge Base and Reasoning Data (Supplementary Results)

Table 12. The Effects of Adding Knowledge Base and Reasoning Data (KB_LabeledKB as the Reference Group)

Variable	GPT-OSS Family	DeepSeek-R1 Family	Llama 3.1 Family
NoData	-0.031*	-0.113***	-0.097***
	(0.016)	(0.010)	(0.033)
KB	-0.013	-0.103***	-0.130***
	(0.010)	(0.010)	(0.012)
KB_LabeledKB_RD	0.086***	0.066***	0.153***
	(0.010)	(0.006)	(0.014)
RD	-0.016	0.002	0.101***
	(0.014)	(0.010)	(0.026)
LargeModel	0.021***	0.038	0.025**
	(0.003)	(0.031)	(0.010)
Report FEs	Yes	Yes	Yes
R-squared (Overall)	0.812	0.401	0.369
Observations	12,900	12,900	12,900

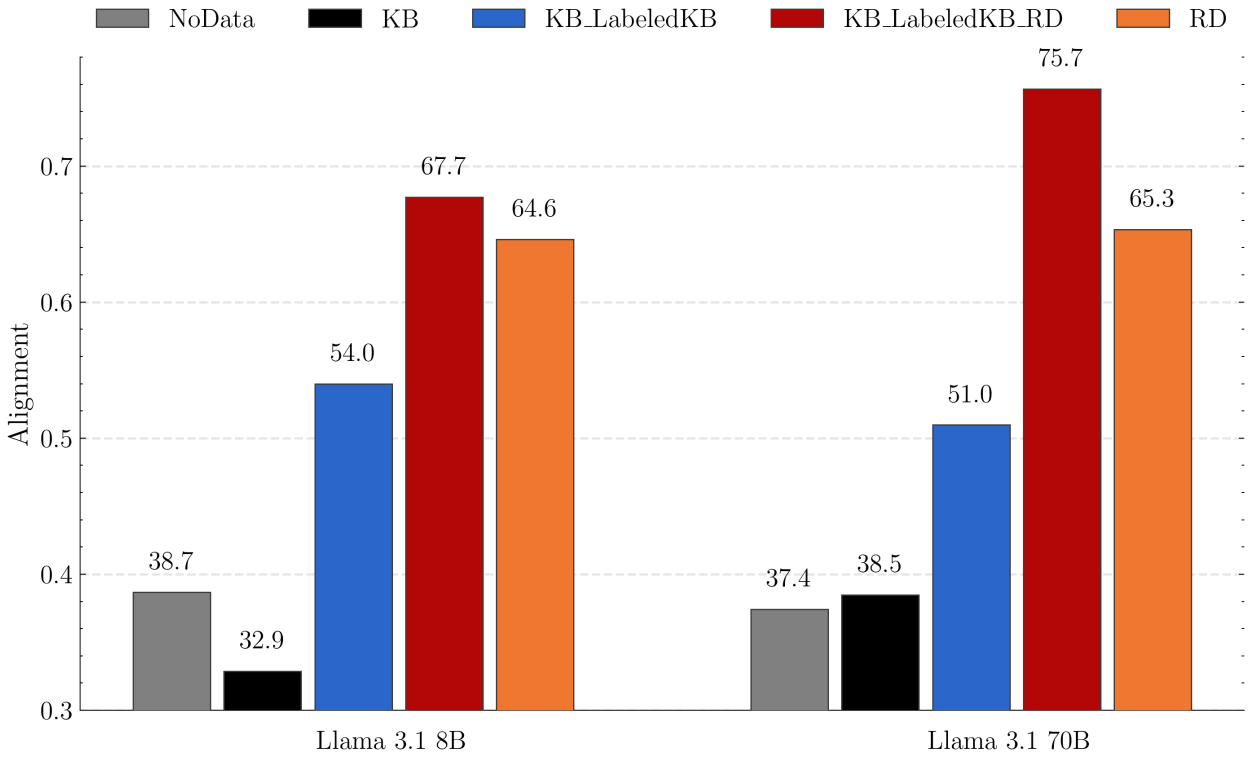
Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$.

Figure 9. Peak Alignment of the DeepSeek-R1 Family for Each Comparison Group



Note: For each comparison group in each model, we report the highest alignment achieved across all values of k.

Figure 10. Peak Alignment of the Llama 3.1 Family for Each Comparison Group



Note: For each comparison group in each model, we report the highest alignment achieved across all values of k.

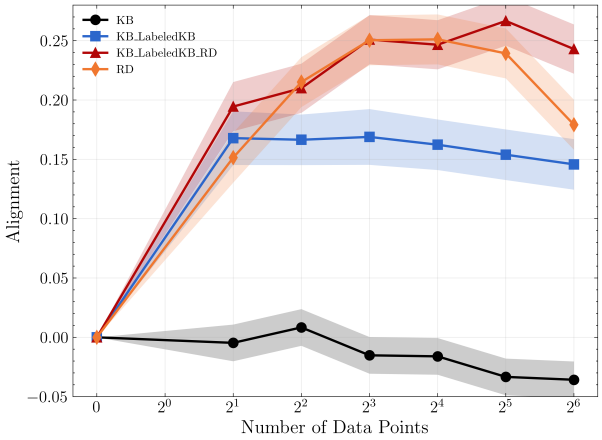
D Scaling Model Size and Data Size (Supplementary Results)

Table 13. The Effects of Number of Added Data Points on Alignment (GPT-OSS 20B)

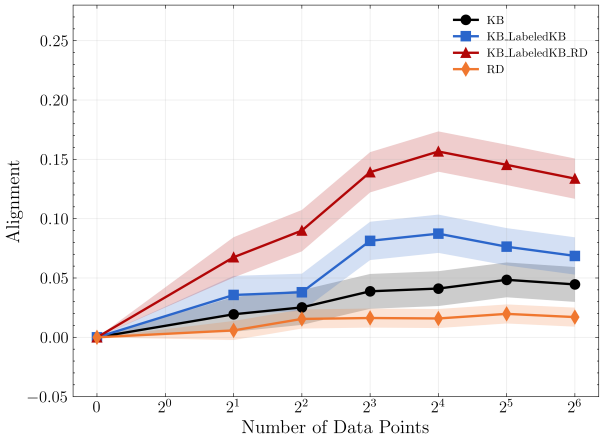
Variable	KB	KB_LabeledKB	KB_LabeledKB_RD	RD
log(k + 1)	0.001	0.003	0.037***	0.002*
	(0.004)	(0.005)	(0.006)	(0.001)
Report FEs	Yes	Yes	Yes	Yes
R-squared (Overall)	0.929	0.854	0.747	0.988
Observations	1806	1806	1806	1806

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$.

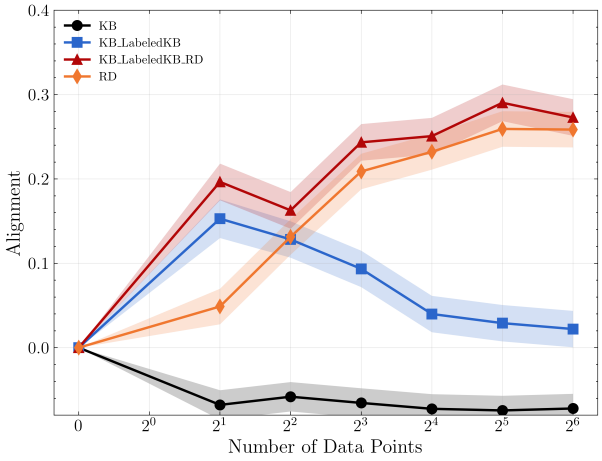
Figure 11. The Effect of Adding k Data Points on Alignment of LLMs (DeepSeek-R1 Family and Llama 3.1 Family)



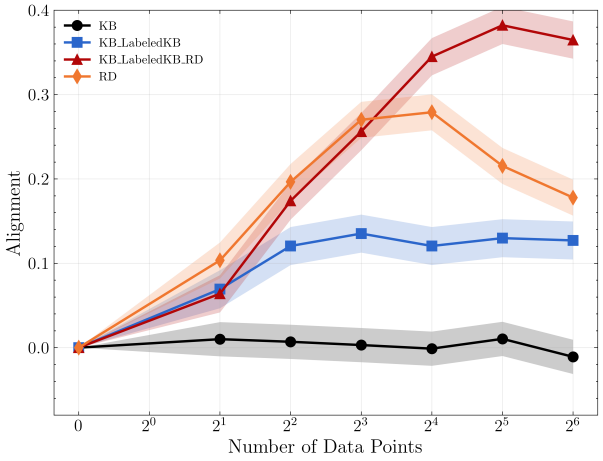
(a) DeepSeek-R1 8B



(b) DeepSeek-R1 671B

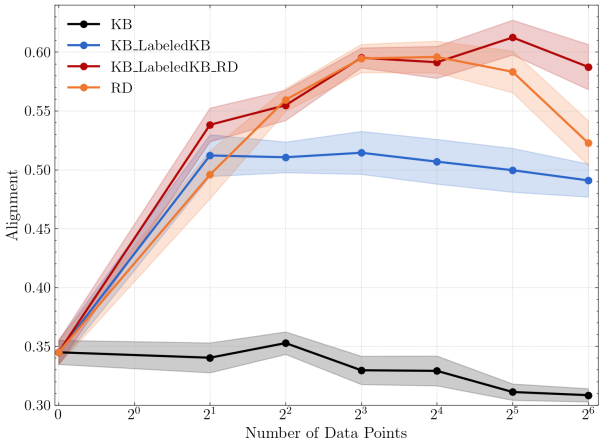


(c) Llama 3.1 8B

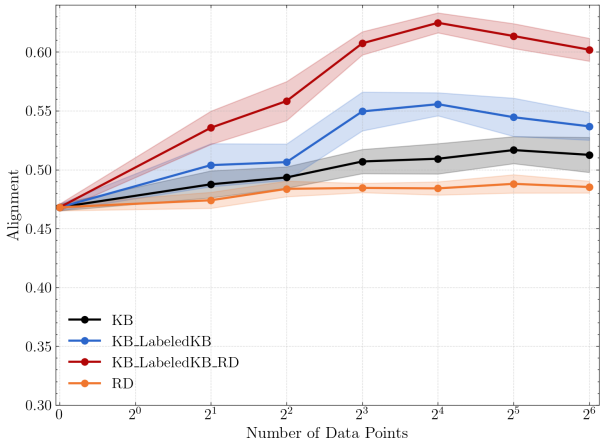


(d) Llama 3.1 70B

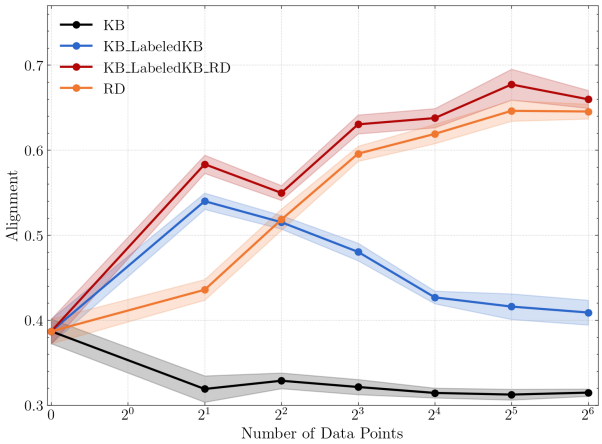
Figure 12. Average Alignment of LLMs When Adding k Data Points (DeepSeek-R1 Family and Llama 3.1 Family)



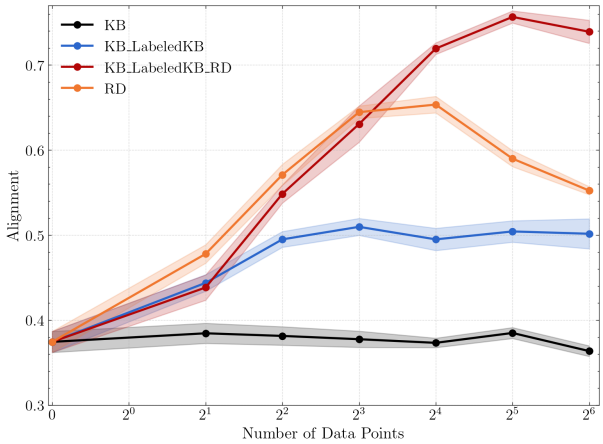
(a) DeepSeek R1 8B



(b) DeepSeek R1 671B

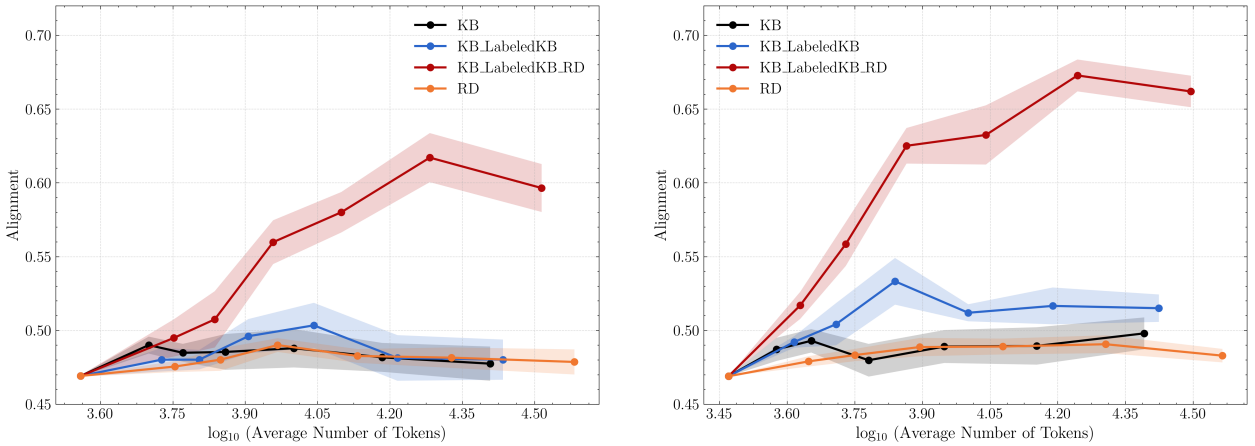


(c) Llama 3.1 8B



(d) Llama 3.1 70B

Figure 13. Alignment vs. Number of Tokens (GPT-OSS Family)

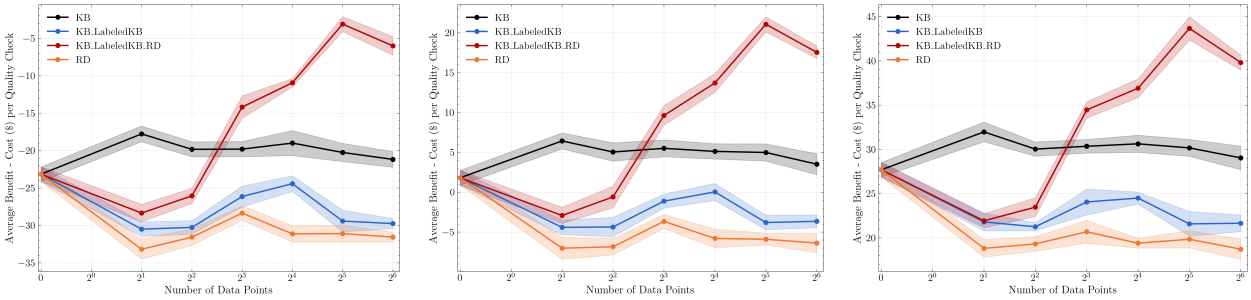


(a) GPT-OSS 20B (b) GPT-OSS 120B

Note: The X-axis indicates the number of tokens in the prompt and LLM output, while the Y-axis indicates the alignment of the LLM. Each point represents the model's alignment on the dataset when adding k data points, and each curve shows the alignment trajectory as k ranges from 0 to 64.

E Economic Implications (Supplementary Results)

Figure 14. Net Value per Quality Check (GPT-OSS 20B)



(a) Review Accuracy: 0.7 (b) Review Accuracy: 0.8 (c) Review Accuracy: 0.9

Note: Each curve plots the average benefit minus cost (in USD) per quality check for one comparison group as the number of added data points k increases from 0 to 64 (log scale on the horizontal axis), at expert review accuracy of 0.7, 0.8, and 0.9. A curve above zero means the LLM creates value relative to expert-only quality checks. Shaded bands show the confidence interval across the resampled quality checks.

F Can Large Models Better Utilize Knowledge Base and Reasoning Data?

Table 14. The Effects of Adding Knowledge Base and Reasoning Data (with LargeModel Interaction) (Including Three Model Families)

Variable	KB	KB_LabeledKB	KB_LabeledKB_RD
AddingData	-0.023	0.085***	0.187***
	(0.018)	(0.014)	(0.015)
LargeModel	0.037**	0.037**	0.037**
	(0.017)	(0.017)	(0.017)
AddingData × LargeModel	0.043***	-0.011	-0.011
	(0.010)	(0.011)	(0.013)
Report FEs	Yes	Yes	Yes
R-squared (Overall)	0.274	0.310	0.367
Observations	10836	10836	10836

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. The regression specification for each column is: $\text{Alignment}_{mrk} = \beta_0 + \beta_1 \text{AddingData}_{mrk} + \beta_L \text{LargeModel}_{mrk} + \beta_2 \text{AddingData}_{mrk} \times \text{LargeModel}_{mrk} + \alpha_r + \varepsilon_{mrk}$. For each column, the reference group is NoData. AddingData = 0 if it is NoData and 1 if data are added as indicated in the column.

We examine whether the effectiveness of adding the knowledge base and reasoning data differs across model sizes. Larger models may have greater capacity to interpret and utilize retrieved data. To test this, we add interaction terms between the LargeModel indicator and the variable AddingData, which equals 1 if any type of data is added. Each column in Table 14 represents the addition of a different type of data.

Table 14 demonstrates that large models are better able to utilize the knowledge base, as indicated by the significant interaction term coefficient (4.3 percentage points). In contrast, the effect of further adding labels and reasoning data is similar for both small and large models. These findings suggest that when provided with a knowledge base, LLMs must still learn from examples and identify patterns of good and bad reports, a process in which larger models may have an advantage due to their greater capacity. However, when reasoning data are added, they directly provide information about patterns of what constitutes a good or bad report, allowing even smaller models to utilize this information effectively.

G Does the Style of the Comments Matter?

Next, we examine whether the improvement in alignment comes solely from the content of the comments or whether the style of the comments also plays a role. While the content of the comments provides information about why a report is good or bad, it is possible that the writing style of the firm's experts could help the LLM learn more effectively. To investigate this, we prompt an LLM to remove the style from the comments while retaining their main content, and then compare alignment with these rewritten comments and with the original comments.

The prompt we use is: "Remove the style from the following text and only keep the main content without changing the meaning of the text."

Table 15. The Effect of Removing the Style of the Comments (GPT-OSS Family)

Variable	KB_LabeledKB	KB_LabeledKB _CommentsWithoutStyle	KB_LabeledKB _Comments
AddingData	0.031*	0.075***	0.083***
	(0.016)	(0.018)	(0.017)
LargeModel	0.022***	0.041***	0.046***
	(0.005)	(0.011)	(0.008)
Report FEs	Yes	Yes	Yes
R-squared (Overall)	0.845	0.612	0.740
Observations	3612	3612	3612

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. The regression specification for each column is: $\text{Alignment}_{mrk} = \beta_0 + \beta_1 \text{AddingData}_{mrk} + \beta_L \text{ LargeModel}_{mrk} + \alpha_r + \varepsilon_{mrk}$. For each column, the reference group is NoData. AddingData = 0 if it is NoData and 1 if data are added as indicated in the column.

The results in Table 15 show that after removing the style, the effect of adding comments remains similar to the effect when comments are added without style removal. This suggests that the content of the comments is the primary factor driving the effect of adding comments.

H LLM-Generated Reasoning by Gemini 3 Pro

For robustness, we also examine the effect of adding LLM-generated reasoning data using Gemini 3 Pro, which is a leading commercial model. Table 16 shows that the effect of adding LLM-generated reasoning data with Gemini 3 Pro is similar to that of adding LLM-generated reasoning data with GPT-OSS 120B, both of which are less effective than expert-generated reasoning data.

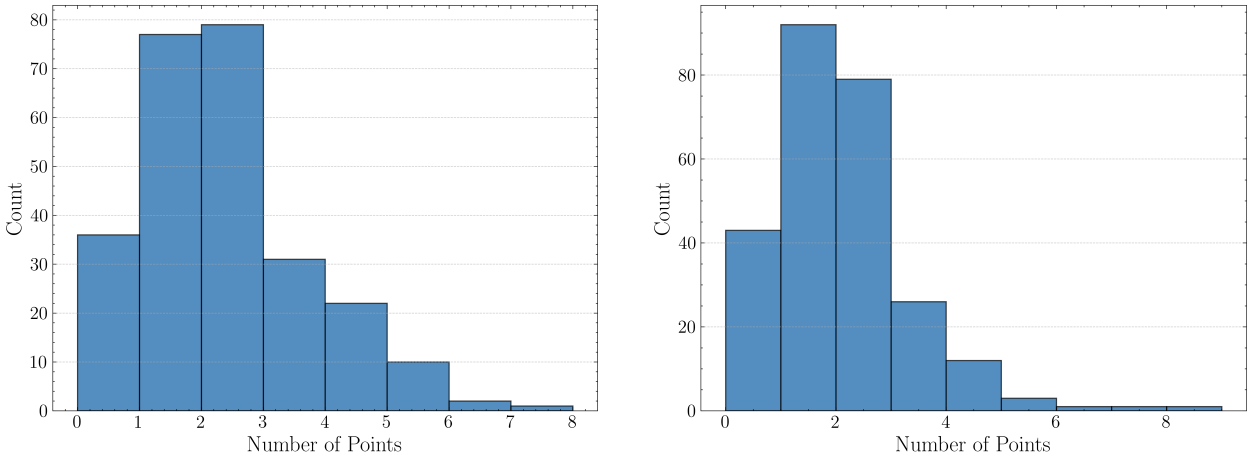
Table 16. The Effect of LLM-Generated Reasoning Using Gemini 3 Pro (GPT-OSS Family)

Variable	KB_LabeledKB	KB_LabeledKB _LLMReasoningGemini	KB_LabeledKB _Comments
AddingData	0.031^{*}	0.059^{***}	0.083^{***}
	(0.016)	(0.019)	(0.017)
LargeModel	0.022^{***}	0.037^{***}	0.046^{***}
	(0.005)	(0.011)	(0.008)
Report FEs	Yes	Yes	Yes
R-squared (Overall)	0.845	0.553	0.740
Observations	3612	3612	3612

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. The regression specification for each column is: $\text{Alignment}_{mrk} = \beta_0 + \beta_1 \text{AddingData}_{mrk} + \beta_L \text{LargeModel}_{mrk} + \alpha_r + \varepsilon_{mrk}$. For each column, the reference group is No-Data. AddingData = 0 if it is NoData and 1 if data are added as indicated in the column. We also run a regression using KB_LabeledKB_LLMReasoningGemini as the reference group, which shows that KB_LabeledKB_Comments can significantly improve alignment compared to KB_LabeledKB_LLMReasoningGemini.

I Distribution of the Number of Points in Expert-Generated Reasoning and LLM-Generated Reasoning

Figure 15. Distribution of the Number of Points Appearing in Expert-Generated Reasoning but Not in LLM-Generated Reasoning

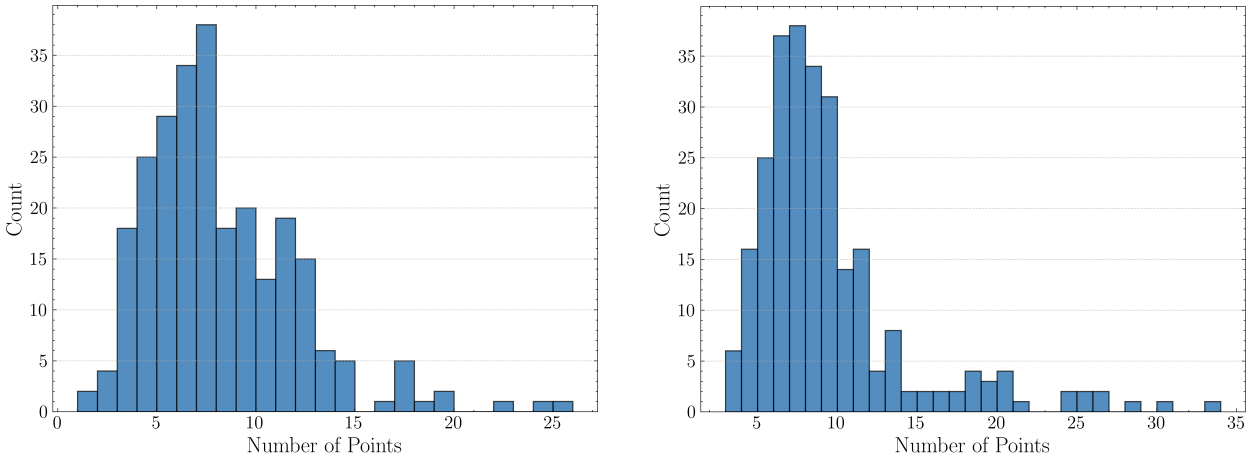


(a) One-Paragraph Restriction

(b) No Length Restriction

Note: On average, the expert-generated reasoning contained 3.21 points, while the LLM-generated reasoning contained 10.79 points with no length restriction and 9.37 points with a one-paragraph restriction.

Figure 16. Distribution of the Number of Points Appearing in LLM-Generated Reasoning but Not in Expert-Generated Reasoning



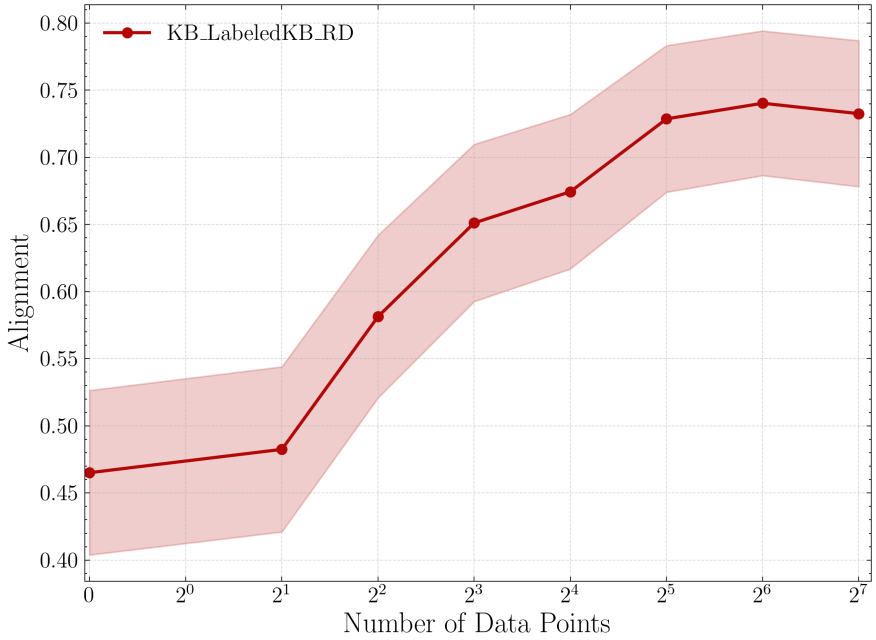
(a) One-Paragraph Restriction (b) No Length Restriction

Note: On average, the expert-generated reasoning contained 3.21 points, while the LLM-generated reasoning contained 10.79 points with no length restriction and 9.37 points with a one-paragraph restriction.

J Scaling Reasoning Data on a Long-Context Advanced Model

A natural concern about the non-monotonic scaling pattern documented in Section 4 is that the eventual decline in alignment may be a feature of scaling reasoning data rather than the long-context limitations of the models. To address this, we repeat the exercise on GLM-5.2, a more recent and more capable model that supports a substantially longer context window and is designed for long-horizon tasks, and we extend the grid from $2^6 = 64$ to $2^7 = 128$ added data points. Figure 17 plots the average alignment of GLM-5.2 as we scale the number of added data points in the KB_LabeledKB_RD group. Alignment rises steadily from 0.47 with NoData to 0.74 at 2^6 data points, where the models examined in the main text have already turned down, and it then edges down only slightly to 0.73 at 2^7. This pattern supports our interpretation: a stronger long-context model extends the range over which scaling reasoning data continues to pay off, so the log-linear gains from reasoning data persist wherever the model can still process the prompt, and the decline observed in the main text reflects the model's context capacity rather than a failure of scaling reasoning data itself. At the same time, the mild reversal at 2^7 indicates that long-context limitations are attenuated but not eliminated.

Figure 17. Average Alignment of GLM-5.2 When Adding k Data Points



Note: The X-axis indicates the number of data points added (in log scale) and the Y-axis indicates the alignment of the LLM. The shaded band shows the confidence interval.