Scaling Laws of AI Alignment: The Impact of Human Expertise *

Jie Zheng[†] Mohammad Rahman[‡] Yu Jeffrey Hu[§]

Purdue University

Abstract

Despite substantial investment and resource scaling in LLM deployments in the workflow, many organizations are struggling to generate value. This paper investigates how to address this bottleneck by exploring the key resources in LLM deployments and how effectively scaling these resources can enable companies to generate value. Generating value depends on alignment with organizational standards, especially those of key decision makers, since each organization has unique strategic goals and misalignment can lead to extensive human review and rework. Therefore, we measure the performance of LLMs using the alignment between LLMs' outputs and those of top experts. We partnered with a consulting company and deployed LLMs for the quality check task. To obtain valid comparisons between top experts' outputs and LLMs' outputs, we designed a randomized field benchmark exercise to collect quality checks and reasoning data from top experts. We then deployed the LLM using RAG and compared its performance under different data types, data sizes, and model sizes. Our results show that scaling reasoning data significantly elevates performance to a new height, resulting in

*This paper benefits from comments from the audience and discussants at the 2025 AI and the Future of Work Conference, 2025 WISE Conference, and 2025 INFORMS Conference.

[†]Email: zheng795@purdue.edu

[‡]Email: mrahman@purdue.edu

[§]Email: yuhu@purdue.edu

an improvement of approximately 10.8 to 37.2 percentage points at its peak compared to adding only the raw knowledge base. There is a power-law relationship between the number of reasoning data points and performance improvement. Small models, when supplied with reasoning data, can achieve higher performance than large models. Additionally, we show that reasoning data work by providing information about patterns that distinguish between good and bad reports. The information about these patterns can help LLMs learn quickly when the new task contains similar patterns. This paper provides important implications for organizations to generate value from LLM deployments.

1 Introduction

Large language models (LLMs) have attracted significant attention for their potential to profoundly affect the organizations and economic activities. Companies are rapidly deploying LLMs in their workflows. Tully et al. (2025), for instance, reports that companies spent $37 billion on generative AI in 2025, with a 3.2x year-over-year increase. Organizations typically scale data (i.e., organizational knowledge bases) using techniques like finetuning or RAG (Retrieval-Augmented Generation), and scale model size to improve performance (Ovadia et al. 2024, Tully et al. 2025). Despite this substantial investment and resource scaling, many organizations are struggling to generate value from their LLM deployments. For example, Challapally et al. (2025) shows there is a 95% failure rate in enterprise LLM deployments. This issue remains underexplored in the literature because of the difficulty in accessing comprehensive proprietary enterprise data. This paper investigates the key resources in LLM deployments and how effectively scaling these resources can enable companies to generate value.

AI progressed from a "reasoning period" to a "knowledge period," and then to today's "learning period" (Russell and Norvig 2020, Zhou 2021). LLMs represent one of the pinnacles of this data-driven learning era. However, they may have a critical flaw in reasoning (Dziri et al. 2023, Berglund et al. 2024, Zečević et al. 2023). As Turing Award winner Yann LeCun noted, current LLMs "don't really understand the physical world... They can't really reason and they certainly can't plan"[1]. This motivates us to revisit the roots of early AI: we may need to reconfigure LLMs with human reasoning data in deployments. Reasoning data have also historically been important for human learning (Chi et al. 1989, Hovland and Weiss 1953, Smoke 1933, Chi et al. 1994). For example, learning to write by reading articles without knowing their quality is often less effective than reading articles with knowing their quality (Schwartz and Bransford 1998). Learning is further improved when feedback includes not only quality labels but also explanations of why the articles are good or poor (Hattie and Timperley 2007).

LLM deployments rely on scaling costly resources. Companies may need significant effort to

[1] See https://lexfridman.com/yann-lecun-3-transcript/

label their knowledge base, since a raw organizational knowledge base may be noisy and inaccurate. Meanwhile, collecting experts' reasoning data, which typically does not exist in current business practices, is costly and time-consuming. Organizations need to decide not only what types of data (knowledge base and reasoning data) to provide, but also how much data to collect and what model size to deploy. Larger models may provide better performance. However, with the growing shortage of GPUs and energy caused by the substantial computational demands of LLMs (Morales 2025), larger models can incur substantial costs and are difficult to deploy. This creates trade-offs between scaling the labeled knowledge base, scaling reasoning data, and scaling model size in LLM deployments.

Given the significant investment in these resources, it is important for organizations to understand the scaling of which resource provides the greatest return and how to combine them optimally. Therefore, this paper examines how these resources (data types, data size, and model size) contribute to the value generated from LLM deployments. As companies increasingly embed LLMs into organizational workflows, generating value depends on aligning AI outputs with organizational standards—especially those of key decision makers. Firms have their own specific knowledge, standards, and strategic goals. If LLM outputs do not align with these standards, they may require extensive human review and rework. This could also lead to a loss of trust. Accordingly, we measure the performance using the alignment between LLMs' outputs and those of top experts who are key decision makers and represent best practices within the organization.[2]

We partnered with a consulting firm whose core service is delivering pricing benchmark and negotiation strategy reports to clients. When the company receives clients' requests for pricing benchmarks, analysts conduct a pricing analysis and generate reports that include negotiation strategies. These reports then undergo an internal quality check to ensure high quality. The quality check is a critical task, because poor-quality reports can lead to the loss of clients who constitute the firm's primary revenue source. This task also requires a high level of expertise and is typically performed by the firm's top experts. This provides an ideal setting to investigate our research

[2]Since LLMs are typically used to directly finish knowledge work in the workflow, higher performance is associated with greater value generated by LLM deployment.

question.

Knowledge work is dynamic: knowledge workers continuously update their understanding as new information emerges. In this setting, repeated finetuning can be costly and inflexible. As a result, RAG has become a widely adopted approach in practice (Lin 2024, Tully et al. 2025). We adopt RAG for the quality check task, which relies on a knowledge base composed of previously delivered consulting reports in this setting. As discussed above, incorporating reasoning data from experts is likely important, but such data need to be collected. We also need to collect quality check results from top experts. This process, however, presents several challenges: experts rarely provide reasoning data in current business because it is time-consuming; most firms do not retain records of draft reports or historical quality check results; and collecting valid and unbiased quality checks and reasoning data requires randomization.

To address this, we designed a randomized field benchmark exercise. We first generated a set of reports using top-performing LLMs and then randomly assigned these reports to 10 top experts for quality checks. We collected a total of 258 completed and valid quality checks and the corresponding expert reasoning data.

We use the alignment between LLMs' outputs and top experts' outputs to measure the performance of LLMs.[3] To answer our research question, we vary data types, data size, and model size when deploying LLMs for the same quality check task. Specifically, for the types of data, we prompt the LLM under five conditions: no additional data; addition of the knowledge base; addition of the knowledge base and labeled knowledge base; addition of the knowledge base, labeled knowledge base, and reasoning data; and addition of reasoning data only. To assess the impact of data size, we vary the number of data points included in the prompt from 2 to 64. Regarding model size, we employ state-of-the-art open-source LLMs and select both large and small models from each family: GPT-OSS (120B and 20B parameters), DeepSeek-R1 (671B and 8B), and Llama 3.1 (70B and 8B). Finally, we use linear probability models to estimate the effects of these different data types, data sizes, and model sizes on the performance of LLMs.

[3] In this paper, since we use alignment as the performance metric, we use the terms "alignment" and "performance" of LLMs interchangeably.

Our findings show scaling a raw knowledge base does not improve performance and scaling a labeled knowledge base leads to better performance. Scaling the knowledge base and model size can bring only limited improvements in performance. Scaling reasoning data can significantly improve the performance of LLMs, resulting in an improvement of approximately 10.8 to 37.2 percentage points at its peak compared to adding only the raw knowledge base across different models. Reasoning data push LLMs' performance to a new height and there is a power-law relationship between the number of added reasoning data points and performance. With reasoning data, smaller models are able to achieve better performance than larger models.

We also demonstrate that reasoning data are effective because they include information about which patterns constitute a good or bad report. When conducting a quality check, if the report contains similar patterns to those found in the reasoning data, the LLM's performance improves. We next show that comments, rather than ratings, drive performance improvement, and that comments about the content of the report, rather than comments about the style of the report, are the main drivers. We also use LLMs to generate reasoning data using the report, quality check label, and ratings as input. However, we find that the effect of adding LLM-generated reasoning data is much smaller than that of adding expert-generated reasoning data. Upon further analysis, we observe that many key points present in expert-generated reasoning are missing from the LLM-generated reasoning. Conversely, LLM-generated reasoning often contains numerous points that are not reflected in expert-generated reasoning.

This paper makes several primary contributions to the literature on AI at work. First, we contribute to the body of research on how organizations can generate value from AI investments. Existing reports show that significant LLM investment does not always generate returns. This paper suggests there is a bottleneck in LLM deployment, which cannot be solved by only utilizing the knowledge base, scaling the model, or increasing the amount of data. However, scaling reasoning data can push performance to a new height. We also contribute to understanding the mechanisms of how knowledge bases and reasoning data work in LLM deployments.

Second, we contribute to the literature on scaling laws in computer science by providing early

field evidence on how performance can be scaled in organizational LLM deployments. We show that adding reasoning data exhibits a power-law relationship with alignment, whereas the knowledge base does not. We also demonstrate that smaller models can achieve higher performance than larger models when scaling reasoning data.

Third, we contribute to the literature on AI alignment by extending the focus from aligning AI with human values and intentions to a firm-level perspective: aligning AI with organizational standards and strategic goals. We provide early field evidence on how different design choices can impact LLM's alignment with organizational standards.

These findings have important implications. First, companies are investing heavily in AI, especially LLMs, in an effort to generate value from these technologies (Tully et al. 2025). However, many studies highlight the existence of an AI productivity paradox (Brynjolfsson et al. 2021). Our study provides managers with insights into how they can allocate resources in LLM deployments to generate value. Reasoning data is an especially important resource for LLM deployments. Our research also implies that tacit knowledge within organizations can be scaled by adding reasoning data. Second, reasoning data from experts does not exist in current business practice. While reasoning data significantly enhance the performance of LLMs, organizations must develop processes to collect and integrate such data into their workflows. This may involve training employees to document and formalize their reasoning during decision-making, or designing tools that can capture expert reasoning during routine tasks. Third, our results show that increasing the amount of reasoning data allows smaller models to perform better than larger models. This presents two types of costs: the upfront cost associated with collecting reasoning data, and the continuous marginal cost of using larger models, which require ongoing computational resources. Managers need to consider this trade-off when making decisions about building LLMs.

2 Literature Review

2.1 LLM Deployment at Work

Our paper is related to the literature on AI investment and value creation. Many studies have examined how AI investments impact firm growth and other firm activities (Babina et al. 2024, Brynjolfsson et al. 2021, Law and Shen 2024). With the advancement of generative AI and LLMs, AI has significantly expanded its ability to assist with complex and knowledge-intensive work (Zhao et al. 2024), such as interpreting natural language, summarizing complex technical documents, generating content drafts (Dell'Acqua et al. 2023), answering domain-specific questions, assisting in software engineering (Peng et al. 2023), and even offering strategic suggestions (Doshi et al. 2024). This drives companies to invest heavily in LLM deployment (Tully et al. 2025). Despite these large investments, 95% of organizations are seeing zero return (Challapally et al. 2025). Previous studies on broader AI adoption have called this phenomenon the AI productivity paradox (Brynjolfsson et al. 2021). However, it remains underexplored in the literature how this bottleneck can be addressed in LLM deployments, due to the difficulty in collecting comprehensive proprietary enterprise data. This paper contributes to understanding the key resources that enable companies to generate value from LLM deployments and the trade-offs between them.

A straightforward way to deploy LLMs is to use them out of the box. However, because knowledge work requires deep and context-specific domain expertise, direct prompting of general-purpose models is often insufficient in organizations (Ovadia et al. 2024). Organizations typically use more advanced approaches such as RAG or fine-tuning. These approaches rely on a knowledge base, which includes domain knowledge and contextual information. The knowledge base is shaped by ongoing organizational activities (Argote and Ingram 2000). Recent surveys show that although 60% to 70% of companies are customizing models using their knowledge base (databricks 2024, Tully et al. 2025), companies are still not getting a return on LLM investments. A survey (Challapally et al. 2025) indicates that the core barrier to scaling is not infrastructure, regulation, or talent—it is learning. Therefore, a knowledge base alone may not be sufficient to generate value

from LLM deployments.

We look into the evolution of AI, which has progressed through three distinct paradigms (Russell and Norvig 2020, Zhou 2021). The "reasoning period" (1950s–early 1970s) was based on the hypothesis that machine intelligence could be realized through logical and symbolic reasoning (Newell and Simon 1956). While early researchers successfully developed general-purpose problem solvers for well-defined tasks, they soon encountered the combinatorial explosion and recognized the limitations of pure logic when applied to ambiguous and complex environments.

This led to the transition into the "knowledge period" (mid-1970s–1980s). Researchers explicitly embedded systems with specialized, domain-specific information (Feigenbaum et al. 1977). Although this era culminated in the widespread deployment based expert systems, the paradigm was ultimately constrained by the large cost and difficulty of manually codifying tacit human expertise into rules. Consequently, the field moves toward the current "learning period" characterized by algorithms designed to learn probabilistic patterns and representations from vast datasets. This era includes statistical machine learning (Mitchell 1997) and deep learning architectures (LeCun et al. 2015).

Modern LLMs represent the one of the pinnacles of this data-driven learning era. Nevertheless, a growing body of literature highlights a critical flaw: these models predominantly rely on statistical correlations and next-token prediction rather than causal comprehension or robust logical deduction (Dziri et al. 2023, Berglund et al. 2024, Zečević et al. 2023). Overcoming this bottleneck necessitates revisiting the roots of early AI's emphasis on reasoning and operationalizing it through modern learning architectures. We may need to reconfigure LLMs with human reasoning data in deployments to address this gap.

Also, drawing on the psychology literature, prior studies show that learning from both positive and negative examples is more effective than learning from positive examples alone or from examples without labels (Schwartz and Bransford 1998, Hovland and Weiss 1953, Große and Renkl 2007, Smoke 1933) Providing explanations of why examples are correct or incorrect further improves learning efficiency (Hattie and Timperley 2007, Chi et al. 1989, 1994). These insights also

motivate the use of reasoning data to complement the knowledge base. Experts' reasoning is often shaped over years of experience through interactions with colleagues and the work environment. Their reasoning includes managerial preferences, organizational standards, and "side information" (Alur et al. 2025). As a result, reasoning within organizations is rarely codified and is costly to collect. This underscores the need to understand how reasoning data contribute—relative to the knowledge base—to the value generated from LLM deployments, as well as the mechanisms behind this.

2.2 LLM Alignment

AI alignment is a critical area of research that focuses on ensuring AI systems operate in accordance with human values and intentions (Gabriel 2020, Ji et al. 2025). With the rapid advancement of LLMs, alignment has become a highly active research topic. Early foundational studies focused on training models to follow human instructions using human feedback (Ouyang et al. 2022, Bai et al. 2022). In this context, alignment refers to the process of modifying an LLM so that its predictions and outputs are reliable, safe, and consistent with human expectations (Liu et al. 2024, Köpf et al. 2023, Wang et al. 2023, Shen et al. 2023). Furthermore, recent economic and management literature explores how to align LLMs with human economic preferences. Some studies treat LLMs as simulated economic agents (Lu et al. 2025, Goli and Singh 2024, Kim et al. 2024, Brand et al. 2023, Arora et al. 2025, Horton 2023, Gao et al. 2025).

This paper extends the AI alignment literature by shifting the focus to alignment with organizational stardards and expectations. There are several key reasons why this is important. First, organizations have unique strategic goals and standards, which form the basis of their competitive advantage. LLMs are increasingly embedded into more tasks, workflows, and even make decisions autonomously. If companies use the same generic LLM without alignment, they may lose their competitive advantage. Second, if LLM outputs deviate from organizational standards, it will require significant review and rework. This may increase costs and reduce productivity. Third, human experts typically work with LLMs. If their outputs are not aligned with organizational

standards, it may reduce their trust in and adoption of LLMs.

2.3 LLMs and Scaling Law

LLMs are typically pre-trained on massive text corpora using self-supervised learning objectives. A central theme in the development of LLMs is the scaling law—a set of empirical evidence that describes how model performance improves predictably with increased scale in model parameters, training data, and compute resources (Kaplan et al. 2020, Hestness et al. 2017, Hoffmann et al. 2022). Studies have shown that performance often improves following power-law relationships.

Recent attention has shifted to inference-time scaling, which focuses on enhancing performance by increasing resources or context during model inference rather than during training (Snell et al. 2024). This field investigates how strategic allocation of compute at inference can yield significant gains in areas such as reward modeling, RAG, and problem-solving (Liu et al. 2025b, Yue et al. 2024, Snell et al. 2024).

A key technique for deploying LLMs is RAG (Lewis et al. 2020). By retrieving knowledge from external sources, RAG systems can generate outputs that are more accurate and aligned with domain-specific needs. Recent work has started to explore how inference-time scaling operates within RAG systems (Yue et al. 2024). Within organizations, RAG systems have enabled continuous learning for LLMs, which would otherwise be expensive to achieve through methods like fine-tuning.

With the growing shortage of GPUs and power driven by the substantial computational demands of LLMs, it becomes increasingly important to understand the relative gains of scaling models versus scaling data when deploying LLMs in companies. In particular, a key question is whether comparable or even superior performance can be achieved with smaller models. Another important issue is understanding how effectively different types of data—such as the knowledge base and reasoning data—can be scaled. Examining these trade-offs is crucial for understanding how to generate value from LLM deployments in companies, but this topic remains underexplored in the literature.

3 Research Design

3.1 Institutional Background

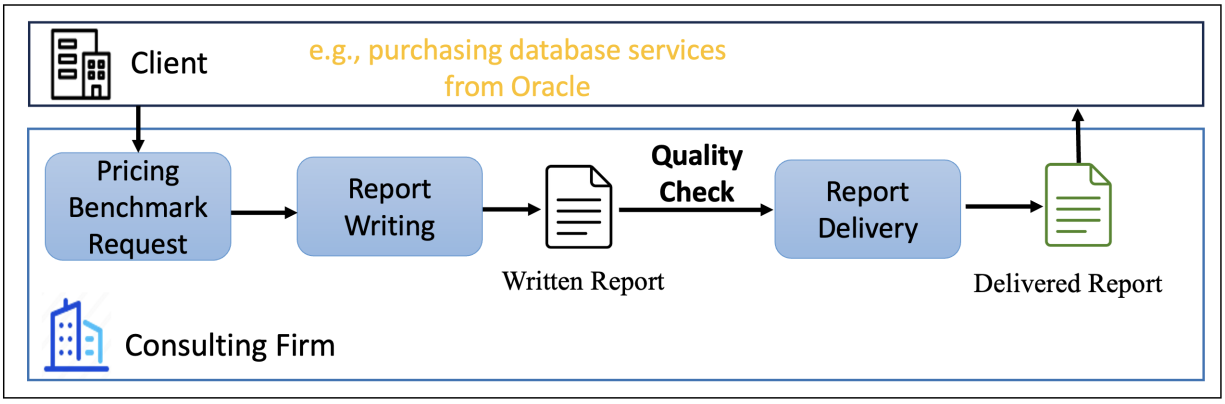


Figure 1: Workflow of the Pricing Benchmark

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 potential opportunities for cost reductions. Figure 1 illustrates the firm's pricing benchmark workflow.

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. To maintain high quality standards, a report is approved for delivery to the client only after it passes this quality check. Most delivered consulting reports are subsequently archived in the firm's internal knowledge.

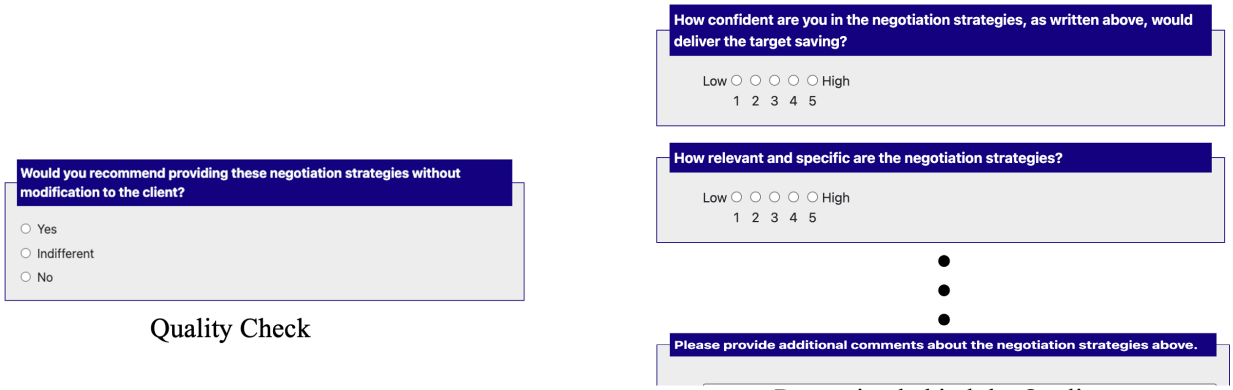
This setting provides an ideal context for our study for two reasons. First, consulting is a highly knowledge-intensive industry in which both domain knowledge and reasoning are central to producing high-quality outputs. Second, the quality-check task is typically performed by the firm's top experts, providing an ideal setting to measure the performance of LLMs by comparing the output of LLMs with that of top experts.

Therefore, this paper deploys LLMs for the quality check task. Because this task requires deep

domain knowledge, an LLM used out of the box is unlikely to be effective. Based on the discussion in the previous section, we adopt RAG rather than fine-tuning. In our setting, the knowledge base consists of previously delivered consulting reports. However, reasoning data are not readily available and must be collected.

To evaluate the performance of LLMs, we compare their quality check outputs to those of top experts. However, there are some challenges. 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, to obtain valid and unbiased quality checks and reasoning data, it is crucial to implement randomization in the quality check process to mitigate selection and expectation effects. To address these challenges and to enable credible comparisons between the outputs of top experts and LLMs, we designed a randomized field benchmark exercise that elicits both quality checks and associated reasoning data from top experts.

3.2 Quality Check Benchmark Exercise in the Field



Quality Check

Reasoning behind the Quality Check

Figure 2: Quality Check and Reasoning

To collect valid quality checks and reasoning data from experts, we first needed to generate valid reports and then randomly assign them to top experts.

3.2.1 Generating Reports

Since the delivered reports archived in the database are mostly positive samples, we needed to use LLMs to generate the set of reports that experts would evaluate. The firm's knowledge base consists of 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 to generate reports. The prompt included detailed deal information such as the client's request, vendor, current proposed price of products, and the fair market value as estimated by analysts. We then applied three distinct prompt strategies to generate a total of 300 LLM-generated reports.[4] We then fine-tuned an LLM on the remaining 1,682 historical reports in the knowledge base. We used the same prompt as before 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—600 generated by LLMs and 100 written historically by analysts. A noteworthy point is that reports generated by LLMs—particularly those produced using fine-tuned models—are 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[5], each with extensive experience in pricing benchmarks and negotiation strategy evaluation.

To ensure that we collected valid and high-quality quality checks and reasoning data in our field benchmark exercise, we implemented a randomization procedure. Each of the 700 reports was randomly assigned to one of the top experts participating in the study. During assignment, we deliberately avoided cases in which a report would be evaluated by a top expert who had previously

[4]We used the open-source LLM ranked highest in the benchmark at the time the report was generated.

[5]They are the most experienced experts in the consulting firm.

analyzed or written 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 top expert received a balanced mix of reports generated through the different strategies—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 top expert 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.

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.[6] Table 1 shows the distribution of quality check results in our field benchmark exercise. It is noteworthy that 27.5% of human-generated reports were rejected in the quality checks. This indicates that although all reports in the knowledge base were delivered in the past, they may still receive a "no" in the quality check

[6]The collected samples represent a sizable dataset, particularly given the high wages of these top experts, which makes the cost of collecting these samples substantial.

process.

Table 1: Distribution of Quality Check Results

	Yes	Indifferent	No	N
All	79 (30.6%)	58 (22.5%)	121 (46.9%)	258
Quality Checks by Senior Analysts	42 (29.2%)	45 (31.2%)	57 (39.6%)	144
Quality Checks by Executives	37 (32.5%)	13 (11.4%)	64 (56.1%)	114
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

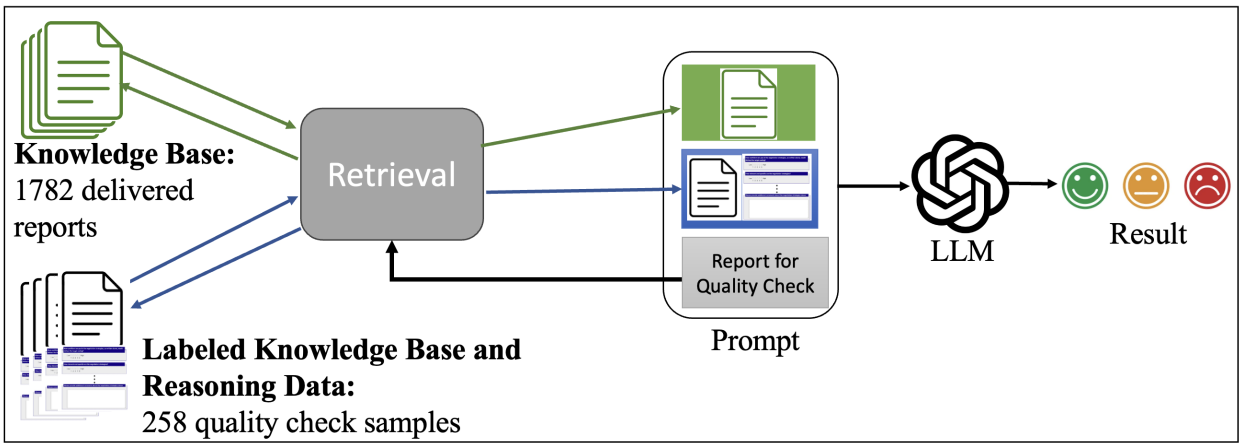


Figure 3: Deploying LLMs for the Quality Check Task Using RAG

After collecting the full set of quality checks and accompanying reasoning data from top experts, the next step is to examine how different types of data, the size of the data, and the size of the model contribute to the performance of an LLM by comparing its outputs to those of top experts for the same quality check task. 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).

In our design, the LLM receives the report to be quality-checked and is then prompted with the same quality check question that top experts answered. We vary the information provided to the LLM in order to evaluate how different types of data contribute to the training of the LLM. Motivated by psychology literature discussed in the previous section, incorporating labeled

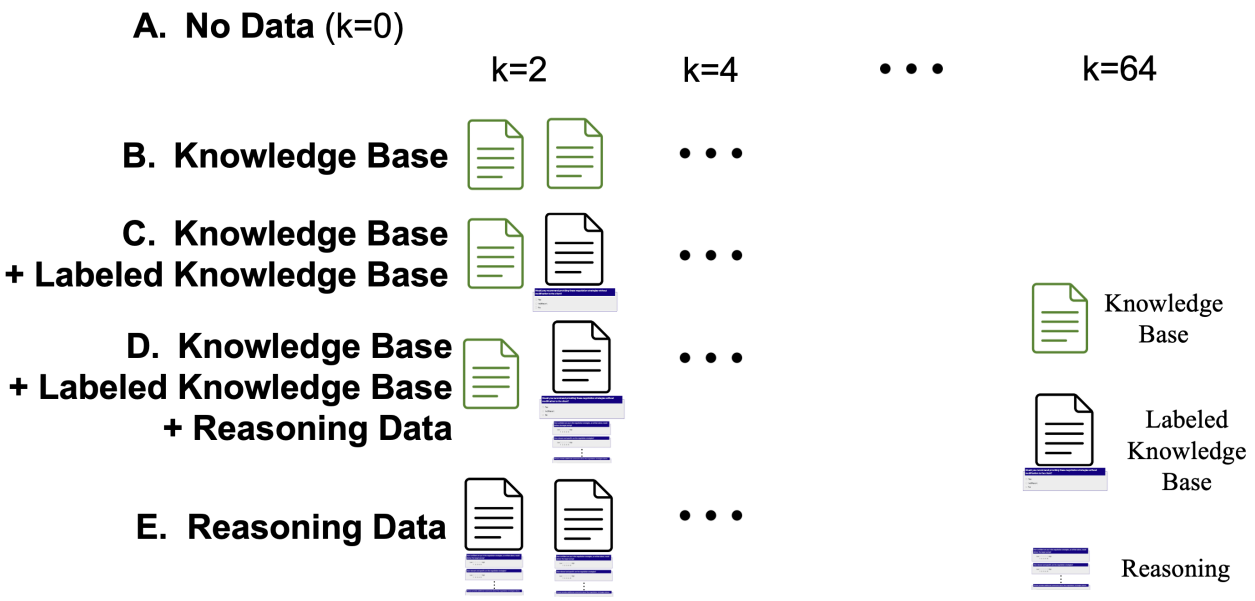


Figure 4: Comparison Groups

knowledge base may make learning more efficient than relying on the knowledge base alone, which consists mostly of positive examples; furthermore, 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.

To incorporate the knowledge base and reasoning data, we follow the inference-time scaling literature in RAG (Yue et al. 2024) and vary the amount of data retrieved and added to the prompt. We begin by computing embeddings for all 258 reports that were evaluated by top experts using the Qwen3-Embedding-0.6B model[7]. 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 top

[7]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`.

experts provided quality checks, and then select the most similar reports along with their associated reasoning. For all comparison groups other than the baseline (**no data**) group, we vary the amount of additional data included in the prompt by setting k to 2, 4, 8, 16, 32, or 64. 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 and reasoning data points.
- **Reasoning data:** Add the top k retrieved reasoning data points.

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

After retrieving data and constructing the prompts, we apply the LLM to perform the quality check for each report. We then evaluate the performance of the LLM by measuring the alignment between top experts' quality check results and the LLM's quality check results for each report.

To analyze how different data and model sizes impact LLM's performance, we estimate linear probability models (LPMs) in which the dependent variable is the LLM's alignment with top experts 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 estimate the following linear probability model with report fixed effects for each model family:

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

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

For all non-baseline specifications, $k \in \{2, 4, 8, 16, 32, 64\}$ denotes the total number of retrieved items included in the prompt. When multiple data types are used, the retrieved items are divided evenly among the data types, as described previously. $Alignment_{mrk}$ is a binary indicator equal to one if the LLM's quality check agrees with the top expert'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.

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)	(0.012)	(0.029)
RD	0.015***	0.115***	0.198***
	(0.004)	(0.010)	(0.021)
LargeModel	0.021***	0.038	0.025***
	(0.003)	(0.031)	(0.010)
Report FEs	Yes	Yes	Yes
R-squared (Overall)	0.118	0.254	0.295
Observations	12,900	12,900	12,900

Note: Use NoData as the reference group. The dependent variable is Alignment.
Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$.

Tables 2 report the effects of adding different types of data on LLM's performance across model families. The results show that adding knowledge base, which mainly consists of positive samples, does not significantly improve performance compared to providing no additional data. However, performance 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 performance. When knowledge base, labeled knowledge base, and reasoning data are all provided, performance 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 11 uses the addition of 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 performance 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, in which the gains are much greater. Finally, when reasoning data are added without quality check labels, the effect is smaller and different across different models, indicating that labels are necessary for reasoning data to be effective.

4.2 Impact of Model Size and Data Size

We first examine the impact of model size. Table 2 shows that large models achieve slightly higher performance than small models in deploying the LLM, 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 Family)

Variable	KB	KB_LabeledKB	KB_LabeledKB_RD	RD
log(k + 1)	0.004	0.004	0.030***	0.003***
	(0.004)	(0.003)	(0.004)	(0.001)
LargeModel	0.004	0.021***	0.044***	0.004**
	(0.003)	(0.005)	(0.005)	(0.002)
Report FEs	Yes	Yes	Yes	Yes
R-squared (Overall)	0.020	0.047	0.212	0.018
Observations	4,128	4,128	4,128	4,128

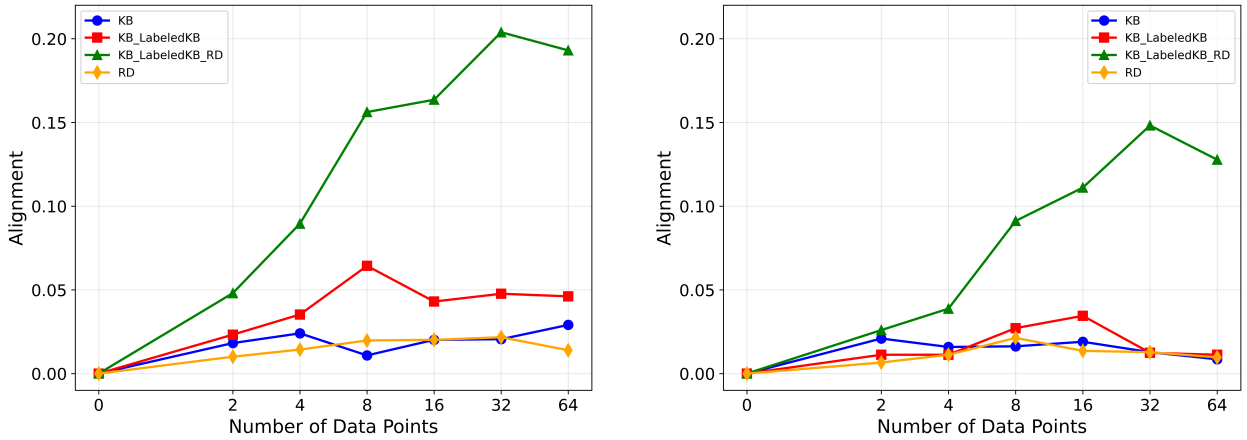
Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$.

We then examine the impact of data size. In the literature on scaling laws, model performance often follows a power-law relationship with increases in data size. Guided by this insight, we first test whether a similar pattern emerges when adding more knowledge base data and reasoning data. To investigate this, we take the logarithm of the number of added data points plus one, log(k + 1), and regress the performance of the LLM on this transformed variable. Below is the regression equation:

$$\text{Alignment}_{mr} = \beta_0 + \beta_1 \log(k_{mr} + 1) + \beta_L \, \text{LargeModel}_{mr} + \alpha_r + \varepsilon_{mr} \tag{2}$$

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

Table 3 summarizes these results. We find that the coefficient for scaling reasoning data is statistically significant, whereas the coefficient for scaling knowledge base data is not. Specifically, increasing the number of data points in the KB_LabeledKB_RD group by 1% increases performance by 0.030 percentage points in the GPT-OSS family, while a 1% increase in the RD group raises performance by 0.003 percentage points.



(a) GPT-OSS 120B

(b) GPT-OSS 20B

Figure 5: The Effect of Adding k Data Points on Alignment of LLMs (GPT-OSS Family)

To investigate whether performance improvements plateau as more data are added, we replaced the continuous measure of $\log(\text{k} + 1)$ 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. As illustrated in Figure 5 and Figure 7, adding reasoning data (KB_LabeledKB_RD group) enables LLMs to achieve substantially higher performance compared to adding only knowledge base and labeled knowledge base data (KB_LabeledKB group). This results in an improvement of approximately 10.8 to 37.2 percentage points at its peak compared to adding only the raw knowledge base across different models. The performance improvements peak between 16 and 32 added data points.

We also find that for all groups, performance drops after reaching its peak. This decline is likely attributable to the long-context limitations of LLMs, as existing literature documents that model performance deteriorates when the prompt becomes excessively long (Liu et al. 2025a). Since the quality check task is inherently complex and the consulting reports are typically lengthy, adding too many data points rapidly pushes the prompt toward the model's context limit, thereby reducing performance.

To jointly illustrate the effects of model size and data size, we plot the average alignment at each value of k for both large and small models in Figure 6 and Figure 8. The results demonstrate that a small model supplied with reasoning data (KB_LabeledKB_RD group) can achieve higher performance than a large model without reasoning data (KB_LabeledKB group). Furthermore, the addition of reasoning data elevates the model's performance to a new height.

In summary, this section demonstrates that the labeled knowledge base can achieve higher performance than the knowledge base. Furthermore, reasoning data significantly improve the

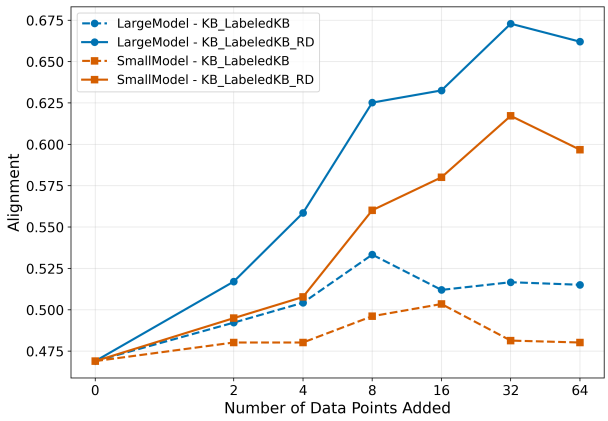


Figure 6: Average Alignment of LLMs when Adding k Data Points (GPT-OSS Family)

performance of LLMs, elevating performance to a new height, with the effect becoming larger as more data points are provided. Notably, when supplied with reasoning data, smaller models are able to achieve higher performance than larger models.

5 Mechanisms

The previous section demonstrated the important role that reasoning data play in deploying LLMs. In this section, we seek to understand the underlying mechanisms through which both the knowledge base and reasoning data contribute to the training of LLMs.

5.1 Why Does the Knowledge Base Not Work?

In Table 1, the human-generated reports used for quality checks are randomly sampled from past delivered reports. Notably, 27.5% of these reports are rejected by experts, and 21.6% receive an indifferent evaluation. This indicates that the knowledge base contains a significant proportion of low-quality reports, introducing considerable noise for the LLM to learn from. This is an important reason why the knowledge base alone does not substantially improve performance.

To obtain more robust results, we investigate whether varying the composition of the knowledge base leads to significant differences in performance. First, we randomly sample half of the reports

in the knowledge base five separate times, using each subset as the knowledge base to train the LLM via RAG as before. We then compare the performance across these different subsets.

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

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

Variable	Sample_1	Sample_2	Sample_3	Sample_4	Sample_5
AddingData	0.009	0.017*	0.011	0.014	0.009
	(0.010)	(0.009)	(0.010)	(0.009)	(0.010)
LargeModel	0.000	0.002	-0.000	0.013*	0.003
	(0.006)	(0.007)	(0.007)	(0.007)	(0.007)
Report FEs	Yes	Yes	Yes	Yes	Yes
R-squared (Overall)	0.016	0.032	0.018	0.036	0.019
Observations	4,128	4,128	4,128	4,128	4,128

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. 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 5: The Effect of Adding Knowledge Base Samples from Different Time Periods (GPT-OSS Family)

Variable	Older Knowledge Base	Newer Knowledge Base
AddingData	0.010	0.012
	(0.008)	(0.011)
LargeModel	-0.001	0.009
	(0.006)	(0.006)
Report FEs	Yes	Yes
R-squared (Overall)	0.016	0.030
Observations	4,128	4,128

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

performance 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 performance. Taken together, these findings suggest that the knowledge base may include a substantial number of low-quality reports across both time periods, which introduces noise and makes it challenging for LLMs to effectively learn from the data.

5.2 Why Do Reasoning Data Work?

5.2.1 Rating vs. Comment

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	0.029*	0.044***	0.080***	0.114***
	(0.015)	(0.017)	(0.017)	(0.020)
LargeModel	0.021***	0.047***	0.048***	0.044***
	(0.005)	(0.009)	(0.008)	(0.005)
Report FEs	Yes	Yes	Yes	Yes
R-squared (Overall)	0.081	0.132	0.193	0.257
Observations	4,128	4,128	4,128	4,128

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

We next aim to understand why reasoning data work. 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 vary the conditions and provide the LLM with only ratings or only comments, and then compare their performance. 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 uses the first column, which shows the effect of adding only the knowledge base and labeled knowledge base compared to no additional data, as the baseline. Building on this, the second column incorporates ratings in addition to the baseline, resulting in only a small improvement in alignment (from 2.9 percentage points to 4.4 percentage points). In contrast, the third column adds comments to the baseline, leading to a much larger improvement (from 2.9 percentage points to 8.0 percentage points), which is close to the effect of adding both ratings and comments together (11.4 percentage points). These results suggest that the detailed and specific comments are the primary factor driving the effect of adding reasoning data.

5.2.2 Similar Patterns

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	KB_LabeledKB_RD
AvgSimilarity	0.737***
	(0.140)
Large Model	0.050***
	(0.006)
Report FEs	Yes
R-squared (Overall)	0.629
Observations	3,612

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

Next, we investigate why these comments improve the LLM's performance. 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. Therefore, the reasoning data will be more effective if the patterns described in the comment 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 will mention which patterns make this a good or bad report). For each testing report, we calculate the average similarity (ranging from

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	0.029*	0.040**	0.058***	0.080***
	(0.015)	(0.017)	(0.014)	(0.017)
LargeModel	0.021***	0.039***	0.052***	0.048***
	(0.005)	(0.009)	(0.008)	(0.008)
Report FEs	Yes	Yes	Yes	Yes
R-squared (Overall)	0.081	0.112	0.157	0.193
Observations	4,128	4,128	4,128	4,128

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

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 performance. Table 7 shows that the average similarity strongly predicts the LLM's performance: when the average similarity increases from 0 to 1, alignment increases by 73.7 percentage points on average. This indicates that when the comments from the added reasoning data provide information about patterns found in the testing report, the LLM's performance will improve.

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 top 50% most similar comments to the testing report's comment, and "LessSimilarComments," which includes the bottom 50% most 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 E, 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 comment contribute to the performance improvement, we examine whether comments about the style of the report or the content of the report are more effective. For each comment, we split it 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 leads to a larger improvement in alignment than adding comments about the style of the report. This suggests that the content of the comments is the primary factor driving the effect of adding comments.

Table 9: Comments about style of the report vs. comments about content of the report (GPT-OSS family)

Variable	KB_LabeledKB	KB_LabeledKB _CommentsAbout Style	KB_LabeledKB _CommentsAbout Content	KB_LabeledKB _Comments
AddingData	0.029*	0.044**	0.072***	0.080***
	(0.015)	(0.017)	(0.018)	(0.017)
LargeModel	0.021***	0.039***	0.039***	0.048***
	(0.005)	(0.010)	(0.010)	(0.008)
Report FEs	Yes	Yes	Yes	Yes
R-squared (Overall)	0.081	0.111	0.154	0.193
Observations	4,128	4,128	4,128	4,128

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

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, we provide the report, quality check result, and rating to the LLM and prompt it to generate reasoning data for each report. We then use these LLM-generated reasoning data to deploy the LLM and compare its performance with that of the LLM deployed using expert-generated reasoning data.

Because the expert-generated reasoning data are mostly one paragraph in length, we consider two conditions when generating LLM-based reasoning data: 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 top expert, 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.

From Table 10, we can see that the effect of adding LLM-generated reasoning data is much smaller than the effect of adding expert-generated reasoning data, and only slightly larger than the effect of not adding reasoning data (KB_LabeledKB group). This suggests that LLM-generated reasoning data is not as effective as expert-generated reasoning data. For robustness, in Appendix G, we use a top commercial model (Gemini 3 Pro) to generate LLM-generated reasoning data and find consistent results.

We continue to investigate why LLM-generated reasoning data is not as effective as expert-generated reasoning data. We use an LLM (GPT-OSS 120B) to compare the differences between expert-generated reasoning and LLM-generated reasoning. Specifically, we use the LLM to decompose both expert-generated and LLM-generated reasoning into individual points, and then count the number of points that appear in expert-generated reasoning but not in LLM-generated reasoning, as

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	0.029*	0.050***	0.043**	0.080***
	(0.015)	(0.018)	(0.017)	(0.017)
LargeModel	0.021***	0.027**	0.018	0.048***
	(0.005)	(0.011)	(0.011)	(0.008)
Report FEs	Yes	Yes	Yes	Yes
R-squared (Overall)	0.081	0.110	0.090	0.193
Observations	4,128	4,128	4,128	4,128

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

well as the number of points that appear in LLM-generated reasoning but not in expert-generated reasoning. We show the distribution of the number of such points in Figure 9 and Figure 10. The results show that many points in expert-generated reasoning are not covered by LLM-generated reasoning, suggesting that LLMs are not able to capture those important patterns mentioned by experts. Furthermore, LLM-generated reasoning is likely to contain a large number of points that are not covered in expert-generated reasoning. These points are likely to create noise and may make it difficult for the LLM to learn from them.

6 Discussions and Conclusion

In summary, our findings show that when deploying LLMs, scaling the labeled knowledge base can achieve higher performance than scaling the raw knowledge base. Scaling reasoning data can significantly improve the performance of LLMs, elevating performance to a new height, and their effect becomes greater as more data points are added. With reasoning data, smaller models are able to achieve the performance of larger models. We also show that reasoning data are effective because they include information about which patterns constitute a good or bad report. When conducting a

quality check, if the report contains similar patterns to those described in the reasoning data, the LLM's performance improves.

These findings have important implications. First, companies are investing heavily in LLMs but are failing to generate value from these investments. Our findings suggest that there is a bottleneck in LLM deployment, which cannot be solved by only utilizing the knowledge base, scaling the model size, or scaling the data size. However, scaling experts' reasoning data can push performance to a new height. Second, while reasoning data significantly enhance the performance of LLMs, organizations must develop processes to systematically collect and integrate such data into their workflows. This may involve training employees to document and formalize their reasoning during decision-making, or designing tools that can capture expert reasoning during routine tasks. Second, our results show that increasing the amount of reasoning data allows smaller models to perform better than much larger models. This presents two types of costs: the upfront cost associated with collecting and documenting reasoning data, and the continuous marginal cost of using larger models, which require ongoing computational resources. Managers need to consider this trade-off when making decisions about deploying LLMs.

References

Alur R, Raghavan M, Shah D (2025) 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.

Arora N, Chakraborty I, Nishimura Y (2025) AI–Human Hybrids for Marketing Research: Leveraging Large Language Models (LLMs) as Collaborators. *Journal of Marketing* 89(2):43–70, ISSN 0022-2429, URL http://dx.doi.org/10.1177/00222429241276529.

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.

Berglund L, Tong M, Kaufmann M, Balesni M, Stickland AC, Korbak T, Evans O (2024) The Reversal Curse: LLMs trained on "A is B" fail to learn "B is A". URL http://dx.doi.org/10.48550/arXiv.2309.12288.

Brand J, Israeli A, Ngwe D (2023) Using LLMs for Market Research. URL http://dx.doi.org/10.2139/ssrn.4395751.

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) STATE OF AI IN BUSINESS 2025 .

Chi MT, Bassok M, Lewis MW, Reimann P, Glaser R (1989) Self-Explanations: How Students Study and

Use Examples in Learning to Solve Problems. *Cognitive Science* 13(2):145–182, ISSN 1551-6709, URL http://dx.doi.org/10.1207/s15516709cog1302_1.

Chi MTH, De Leeuw N, Chiu MH, Lavancher C (1994) Eliciting self-explanations improves understanding. *Cognitive Science* 18(3):439–477, ISSN 0364-0213, URL http://dx.doi.org/10.1016/0364-0213(94)90016-7.

databricks (2024) State of Data + AI. Technical report.

Dell'Acqua F, McFowland E, Mollick ER, Lifshitz-Assaf H, Kellogg K, Rajendran S, Krayer L, Candelon F, Lakhani KR (2023) Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of AI on Knowledge Worker Productivity and Quality. URL http://dx.doi.org/10.2139/ssrn.4573321.

Doshi AR, Bell JJ, Mirzayev E, Vanneste BS (2024) Generative artificial intelligence and evaluating strategic decisions. *Strategic Management Journal* n/a(n/a), ISSN 1097-0266, URL http://dx.doi.org/10.1002/smj.3677.

Dziri N, Lu X, Sclar M, Li XL, Jiang L, Lin BY, West P, Bhagavatula C, Bras RL, Hwang JD, Sanyal S, Welleck S, Ren X, Ettinger A, Harchaoui Z, Choi Y (2023) Faith and Fate: Limits of Transformers on Compositionality. URL http://dx.doi.org/10.48550/arXiv.2305.18654.

Feigenbaum EA, et al. (1977) The art of artificial intelligence: Themes and case studies of knowledge engineering .

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.

Gao Y, Lee D, Burtch G, Fazelpour S (2025) Take Caution in Using LLMs as Human Surrogates: Scylla Ex Machina. URL http://dx.doi.org/10.48550/arXiv.2410.19599.

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.

Große CS, Renkl A (2007) Finding and fixing errors in worked examples: Can this foster learning outcomes? *Learning and Instruction* 17(6):612–634, ISSN 0959-4752, URL http://dx.doi.org/10.1016/j.learninstruc.2007.09.008.

Hattie J, Timperley H (2007) The Power of Feedback. *Review of Educational Research* 77(1):81–112, ISSN 0034-6543, URL http://dx.doi.org/10.3102/003465430298487.

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, Casas DdL, 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, Rae JW, Vinyals O, Sifre L (2022) Training Compute-Optimal Large Language Models URL http://dx.doi.org/10.48550/arXiv.2203.15556.

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.

Hovland CI, Weiss W (1953) Transmission of information concerning concepts through positive and negative instances. *Journal of Experimental Psychology* 45(3):175–182, ISSN 0022-1015, URL http://dx.doi.org/10.1037/h0062351.

Ji J, Qiu T, Chen B, Zhang B, Lou H, Wang K, Duan Y, He Z, Vierling L, Hong D, Zhou J, Zhang Z, Zeng F, Dai J, Pan X, Ng KY, O'Gara A, Xu H, Tse B, Fu J, McAleer S, Yang Y, Wang Y, Zhu SC, Guo Y, Gao W (2025) AI Alignment: A Comprehensive Survey. URL http://dx.doi.org/10.48550/arXiv.2310.19852.

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.

Kim J, Kovach M, Lee KM, Shin E, Tzavellas H (2024) Learning to be Homo Economicus: Can an LLM Learn Preferences from Choice. URL http://dx.doi.org/10.48550/arXiv.2401.07345.

Köpf A, Kilcher Y, von Rütte D, Anagnostidis S, Tam ZR, Stevens K, Barhoum A, Nguyen D, Stanley O, Nagyfi R, Es S, Suri S, Glushkov D, Dantuluri A, Maguire A, Schuhmann C, Nguyen H, Mattick A (2023) OpenAssistant Conversations - Democratizing Large Language Model Alignment. *Advances in Neural Information Processing Systems* 36:47669–47681.

Law KKF, Shen M (2024) How Does Artificial Intelligence Shape Audit Firms? *Management Science* ISSN

0025-1909, URL http://dx.doi.org/10.1287/mnsc.2022.04040.

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521(7553):436–444, ISSN 1476-4687, URL http://dx.doi.org/10.1038/nature14539.

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. Larochelle H, Ranzato M, Hadsell R, Balcan MF, Lin HT, eds., *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, Virtual* (arXiv).

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 (2025a) 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.

Liu Z, Wang P, Xu R, Ma S, Ruan C, Li P, Liu Y, Wu Y (2025b) Inference-Time Scaling for Generalist Reward Modeling. URL http://dx.doi.org/10.48550/arXiv.2504.02495.

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.

Mitchell TM (1997) *Machine Learning* (New York: McGraw-Hill Education), 1st edition edition, ISBN 978-0-07-042807-2.

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.

Newell A, Simon H (1956) The logic theory machine–A complex information processing system. *IRE Transactions on Information Theory* 2(3):61–79, ISSN 2168-2712, URL http://dx.doi.org/10.1109/TIT.1956.1056797.

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.

Russell S, Norvig P (2020) *Artificial Intelligence: A Modern Approach* (Hoboken: Pearson), 4th edition edition, ISBN 978-0-13-461099-3.

Schwartz DL, Bransford JD (1998) A Time For Telling. *Cognition and Instruction* 16(4):475–5223, ISSN 0737-0008, 1532-690X, URL http://dx.doi.org/10.1207/s1532690xci1604_4.

Shen T, Jin R, Huang Y, Liu C, Dong W, Guo Z, Wu X, Liu Y, Xiong D (2023) Large Language Model Alignment: A Survey. URL http://dx.doi.org/10.48550/arXiv.2309.15025.

Smoke KL (1933) Negative instances in concept learning. *Journal of Experimental Psychology* 16(4):583–588, ISSN 0022-1015, URL http://dx.doi.org/10.1037/h0073724.

Snell C, Lee J, Xu K, Kumar A (2024) Scaling LLM Test-Time Compute Optimally can be More Effective than Scaling Model Parameters. URL http://dx.doi.org/10.48550/arXiv.2408.03314.

Tully T, Joff Redfern, Deedy Das, Derek Xiao (2025) 2025: The State of Generative AI in the Enterprise.

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.

Yue Z, Zhuang H, Bai A, Hui K, Jagerman R, Zeng H, Qin Z, Wang D, Wang X, Bendersky M (2024) Inference Scaling for Long-Context Retrieval Augmented Generation. *The Thirteenth International Conference on Learning Representations*.

Zečević M, Willig M, Dhami DS, Kersting K (2023) Causal Parrots: Large Language Models May Talk Causality But Are Not Causal. URL http://dx.doi.org/10.48550/arXiv.2308.13067.

Zhao WX, Zhou K, Li J, Tang T, Wang X, Hou Y, Min Y, Zhang B, Zhang J, Dong Z, Du Y, Yang C, Chen Y, Chen Z, Jiang J, Ren R, Li Y, Tang X, Liu Z, Liu P, Nie JY, Wen JR (2024) A Survey of Large Language Models. URL http://dx.doi.org/10.48550/arXiv.2303.18223.

Zhou ZH (2021) *Machine Learning* (Singapore: Springer), ISBN 978-981-15-1966-6 978-981-15-1967-3, URL http://dx.doi.org/10.1007/978-981-15-1967-3.

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 AI knowledge worker.

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

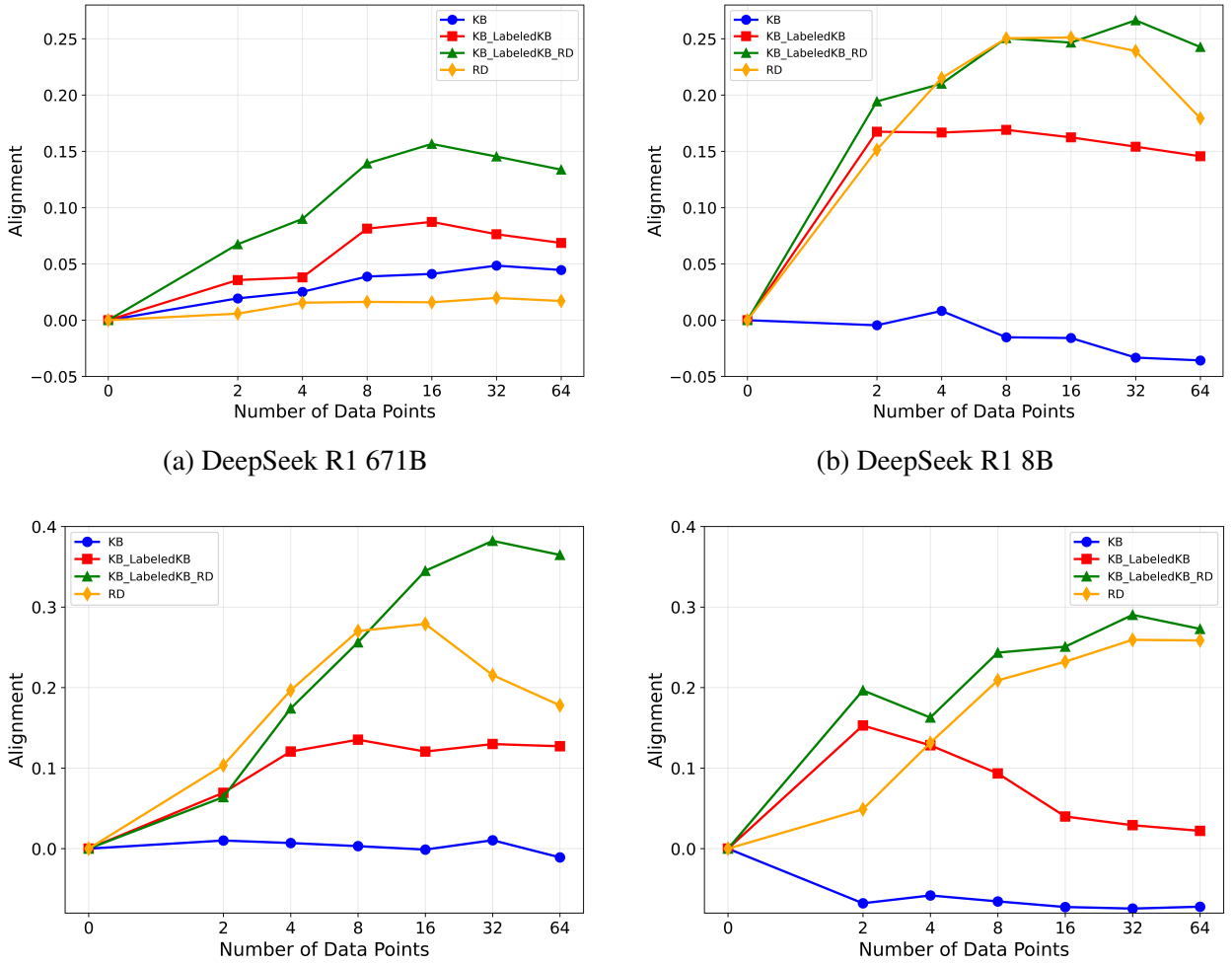
C The Effects of Adding Knowledge Base and Reasoning Data (KB_LabeledKB as the Reference Group)

Table 11: 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)	0.054	0.024	0.112
Observations	12,900	12,900	12,900

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

D Impact of Data Size and Model Size (DeepSeek-R1 Family and Llama-3.1 Family)



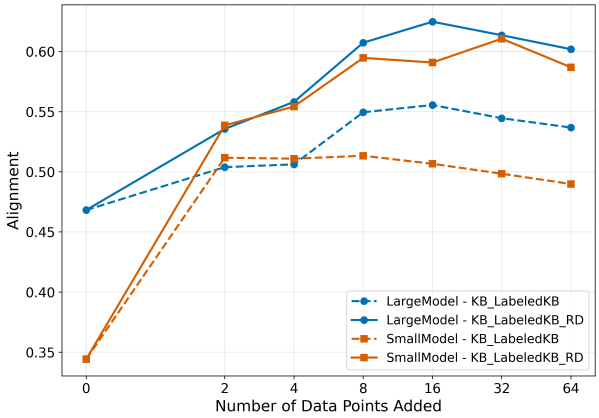
(a) DeepSeek R1 671B

(b) DeepSeek R1 8B

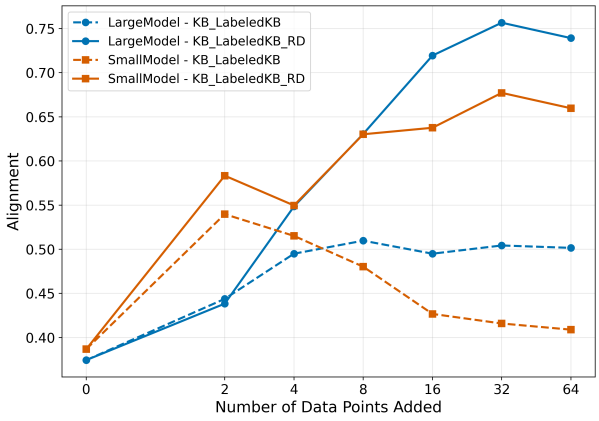
(c) Llama-3.1 70B

(d) Llama-3.1 8B

Figure 7: The Effect of Adding k Data Points on Alignment of LLMs (DeepSeek-R1 Family and Llama-3.1 Family)



(a) DeepSeek R1 Family



(b) Llama 3 Family

Figure 8: Average Alignment of LLMs when Adding k Data Points (DeepSeek-R1 Family and Llama-3.1 Family)

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

Table 12: The Effects of Adding Knowledge Base and Reasoning Data (with LargeModel Interaction) (GPT-OSS Family)

Variable	KB	KB_LabeledKB	KB_LabeledKB_RD
AddingData	-0.020	0.077***	0.183***
	(0.016)	(0.014)	(0.015)
LargeModel	0.037**	0.037**	0.037**
	(0.017)	(0.017)	(0.017)
AddingData × LargeModel	0.037***	-0.005	-0.010
	(0.010)	(0.011)	(0.013)
Report FEs	Yes	Yes	Yes
R-squared (Overall)	0.041	0.184	0.348
Observations	12,384	12,384	12,384

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. 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 is 1 if any type of data is being added. Each column in Table 12 represents the addition of a different type of data.

Table 12 demonstrates that large models are better able to utilize the knowledge base, as indicated by the significant effect sizes of adding the knowledge base: 3.7 percentage points for KB compared with small models. 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 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.

F Does the Style of the Comments Matter?

Next, we examine whether the improvement in performance comes solely from the content of the comments, or if 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 company's experts could help the AI knowledge worker learn more effectively. To investigate this, we use an LLM to remove the style from the comments, retaining only their keywords, and then compare performance when only content (not style) is present.

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 13: The Effect of Removing the Style of the Comments (GPT-OSS Family)

Variable	KB_LabeledKB	KB_LabeledKB _CommentsWithoutStyle	KB_LabeledKB _Comments
AddingData	0.029*	0.076***	0.080***
	(0.015)	(0.017)	(0.017)
LargeModel	0.021***	0.038***	0.048***
	(0.005)	(0.011)	(0.008)
Report FEs	Yes	Yes	Yes
R-squared (Overall)	0.081	0.160	0.193
Observations	4,128	4,128	4,128

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

Results in Table 13 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.

G 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 14 shows that the effect of adding LLM-generated

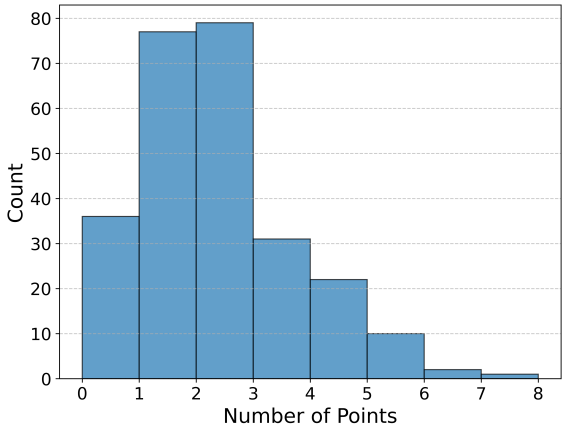
reasoning data with Gemini 3 Pro is similar to that of adding LLM-generated reasoning data with GPT-OSS 120B, both of which are less effective than expert-generated reasoning data.

Table 14: The Effect of LLM-generated Reasoning using Gemini 3 Pro (GPT-OSS family)

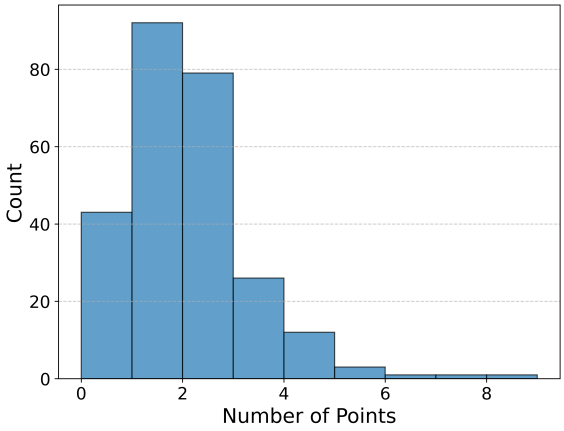
Variable	KB_LabeledKB	KB_LabeledKB _LLMReasoningGemini	KB_LabeledKB _Comments
AddingData	0.029*	0.056***	0.080***
	(0.015)	(0.019)	(0.017)
LargeModel	0.021***	0.035***	0.048***
	(0.005)	(0.010)	(0.008)
Report FEs	Yes	Yes	Yes
R-squared (Overall)	0.081	0.125	0.193
Observations	4,128	4,128	4,128

Note: The dependent variable is Alignment. Significance stars: $^{*}p < 0.1$, $^{**}p < 0.05$, $^{***}p < 0.01$. 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_LLMReasoningGemini as the reference group, which shows that KB_LabeledKB_Comments can significantly improve alignment compared to KB_LabeledKB_LLMReasoningGemini.

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

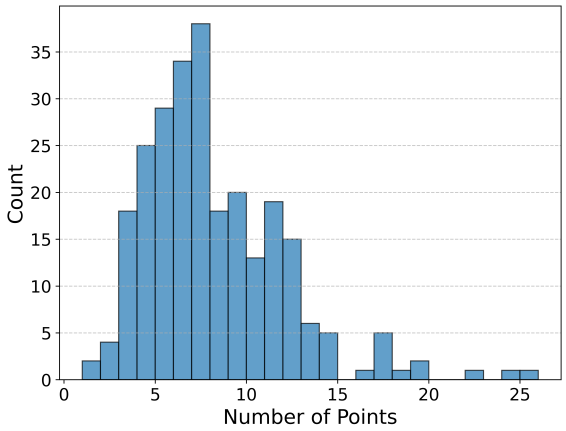


(a) One-paragraph restriction

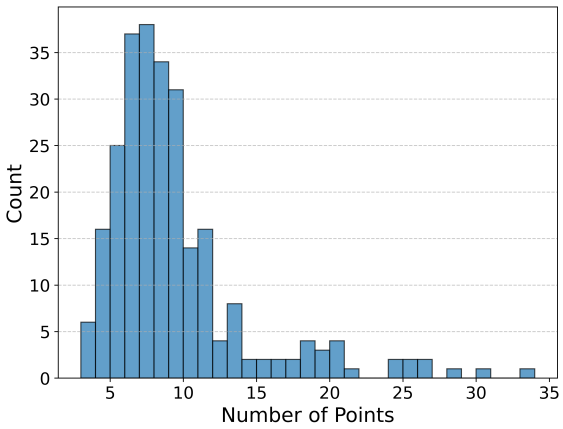


(b) No length restriction

Figure 9: Distribution of the number of points appearing in expert-generated reasoning but not in LLM-generated reasoning. 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.



(a) One-paragraph restriction



(b) No length restriction

Figure 10: Distribution of the number of points appearing in LLM-generated reasoning but not in expert-generated reasoning. 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.