# Fin-R1: A Large Language Model for Financial Reasoning through Reinforcement Learning

**Zhaowei Liu**[1] **Xin Guo**[1] **Fangqi Lou**[1] **Lingfeng Zeng**[1] **Jinyi Niu**[2]
**Zixuan Wang**[1] **Jiajie Xu**[1] **Weige Cai**[1] **Ziwei Yang**[1] **Xueqian Zhao**[1]
**Chao Li**[3] **Sheng Xu**[3] **Dezhi Chen**[3] **Zuo Bai**[3] **Liwen Zhang***[1]
[1]Shanghai University of Finance and Economics [2]Fudan University [3]FinStep

## Abstract

Reasoning large language models (LLMs) are rapidly evolving across various domains. However, their capabilities in handling complex financial problems still require in-depth exploration. In this paper, we introduce Fin-R1, a large language model specifically designed for financial reasoning. With a lightweight parameter scale of 7 billion, this model significantly reduces deployment cost while effectively addresses three major financial pain points: fragmented financial data, uncontrollable reasoning logic, and weak business generalization ability. To boost the model's reasoning capability, we first built Fin-R1-Data, a high-quality dataset with around 60,000 complete chains of thought (CoT) for both reasoning and non-reasoning financial scenarios, through a distillation and screening process from multiple authoritative datasets. Then, we perform Supervised Fine-Tuning (SFT) followed by Reinforcement Learning (RL) based on this dataset. This two-stage training framework significantly enhances the model's ability to perform complex financial reasoning tasks, enabling more accurate and interpretable decision-making in financial AI applications. Despite its compact structure with only 7B parameters, Fin-R1 demonstrates outstanding performance in authoritative benchmarks covering multiple financial business scenarios. It achieves an average score of 75.2, securing second place overall and significantly outperforming other large-scale reasoning LLMs in the evaluation. Notably, Fin-R1 is even better than the 70B-parameter model, DeepSeek-R1-Distill-Llama-70B, demonstrating its efficiency and effectiveness. It achieves the state-of-the-art scores of 85.0 in ConvFinQA and 76.0 in FinQA, which focus on multi-turn and numerical reasoning in financial contexts. In real-world applications, Fin-R1 has demonstrated strong automated reasoning and decision-making abilities in areas like financial compliance and robo-advisory, providing efficient solutions to long-standing financial industry challenges. Our code are available at https://github.com/SUFE-AIFLM-Lab/Fin-R1.

## 1 Introduction

In recent years, the rapid iteration of large language models (LLMs) has significantly propelled the evolution of artificial intelligence towards artificial general intelligence (AGI). OpenAI's o1 [10] series models have enhanced their ability to solve complex reasoning tasks by extending the length of the "chain-of-thought" reasoning process through a mechanism of "exploration-reflection-iteration." Similar o1-like LLMs, such as QwQ [12] and Marco-o1 [33], have achieved notable improvements in various reasoning tasks, including mathematics, programming, and logical reasoning. Financial reproduction versions of o1 models, such as XuanYuan-FinX1-Preview [17] and Fino1 [11], have also demonstrated the immense potential of LLMs in simulating human cognitive processes and

*Correspondence to Liwen Zhang <zhang.liwen@shufe.edu.cn>.

Preprint. Under review.

handling complex tasks. DeepSeek-R1[4] adopts a fundamentally different approach from o1-like models, leveraging pure Reinforcement Learning (RL) to enhance the reasoning capabilities of large language models. Through thousands of steps of unsupervised RL training, combined with a small set of cold-start data and a multi-stage training framework, the model exhibits emergent reasoning abilities in benchmark evaluations. Simultaneously, this training strategy further refines the model's reasoning performance and readability, demonstrating the efficacy of RL-driven methodologies in advancing the inference capabilities of large-scale language models.

However, when general-purpose reasoning models are applied to the financial domain, they still face challenges in adapting to vertical scenarios. Financial reasoning tasks often involve knowledge of legal clauses, economic indicators, and mathematical modeling. These tasks not only require the integration of interdisciplinary knowledge but also demand verifiable, step-by-step decision-making logic. In real-world financial business scenarios, model commonly face the following problems: 1. Fragmentation of financial data makes it difficult to integrate knowledge [23, 5, 24, 6, 1, 27, 26, 23, 30, 31]. The inconsistency of data not only increases the complexity of preprocessing but also may lead to redundant or missing information, further weakening the model's ability to comprehensively understand and reason within the financial domain. 2. Black-box reasoning logic fails to meet regulatory requirements for traceability [22, 32, 19]. the complex structure of existing models makes their reasoning process difficult to interpret intuitively. This creates a contradiction with the regulatory requirements for transparency and traceability in finance, thereby limiting the application of these models in critical financial business areas. 3. Insufficient generalization ability in financial scenarios lead to unreliable outputs in high-risk financial applications[29, 2, 34]. The existing models often perform unstably across different scenarios and struggle to be quickly transferred and generalized to new business contexts. This limitation makes the models prone to instability or inaccuracy in their outputs when facing high-risk financial applications.

To address the challenges faced by general-purpose reasoning models in the financial domain, this paper introduces Fin-R1, the large language model tailored for financial reasoning. By reconstructing a high-quality financial reasoning dataset and employing a two-stage training framework, Fin-R1 effectively tackles the three core issues of fragmented financial data, uncontrollable reasoning logic, and weak business generalization ability. Our main contributions are as follows:

- **High-Quality Financial Knowledge Engine**: We have distilled and filtered the high-quality COT dataset Fin-R1-Data from multiple authoritative financial datasets, specifically designed for professional financial reasoning scenarios. This dataset covers multidimensional professional knowledge in the Chinese and English financial vertical domain and can effectively support multiple core financial business scenarios.
- **Explicit Financial Reasoning Large Language Model**: We develop financial reasoning large language model Fin-R1-7B by integrating multidimensional financial business capability datasets for fine-tuning, which precisely addresses the core demands of the financial industry for decision-making processes, numerical rigor, and strong business generalization capabilities.
- **Two-Stage Hybrid Training Framework**: We propose a two-stage workflow framework that involves constructing a high-quality CoT dataset and training the model through Supervised Fine-Tuning (SFT) and Reinforcement Learning (RL), that can effectively enhance the model's financial reasoning performance.

The remainder of this report is structured as follows. Section 2 provides a detailed description of the methodological framework. Section 3 briefly describes our experiments and results on multiple financial benchmark tests. Section 4 summarizes the technical contributions and outlines future research directions.

## 2 Approach

### 2.1 Overview

We propose a two-stage model construction framework. In the data generation phase, we employ data distillation based on DeepSeek-R1 and a data filtering method using llm-as-judge [25] to create a high-quality financial reasoning dataset, Fin-R1-Data. In the model training phase, we establish

the financial reasoning model Fin-R1 based on Qwen2.5-7B-Instruct, using Supervised Fine-Tuning (SFT) and the Group Relative Policy Optimization algorithm (GRPO) [13] to enhance the model's reasoning capability and standardize its output format.

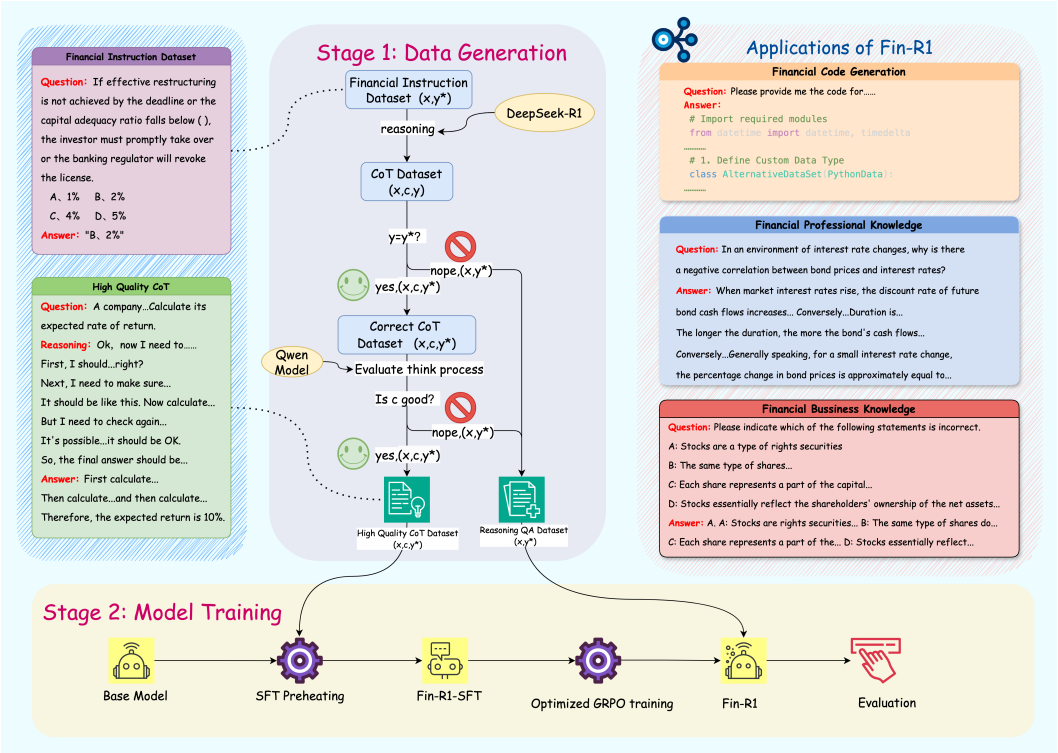



Figure 1: The pipeline for constructing Fin-R1.The diagram depicts the two-stage construction framework of Fin-R1: Data Generation (using DeepSeek-R1 for reasoning to generate CoT data, followed by quality filtering with the Qwen2.5-72B-Instruct) and Model Training (including SFT pretraining and GRPO optimization for Fin-R1). Additionally, the right side highlights the performance of Fin-R1 in financial code generation, professional knowledge, and business knowledge.

## 2.2 Data Construction

Our objective is to develop Fin-R1-Data, a high-quality, supervised fine-tuning (SFT) dataset specifically designed for financial domains. To achieve this goal, we have designed a robust and comprehensive data construction pipeline, including data distillation and data filtering, aimed at ensuring the accuracy and reliability of the dataset. The pipeline of data construction is shown in Figure 3.

### 2.2.1 Data Source

Fin-R1-Data comprises a total of 60,091 distinct entries, encompassing both Chinese and English content in a bilingual format. The dataset is organized into two primary components: open-source datasets and proprietary datasets. The open-source datasets include Ant_Finance [14], FinanceIQ [15], Quant-Trading-Instruct (FinanceQT) [8], ConvFinQA, FinQA, Twitter-Financial-News-Sentiment (TFNS) [20], Finance-Instruct-500K [3], FinCorpus [16], and FinCUGE [7].

The proprietary component of the dataset, the Financial Postgraduate Entrance Exam (FinPEE) dataset, consists of 350 calculation problems derived from financial postgraduate entrance examinations. The construction of FinPEE followed a rigorous, multi-stage process. Initially, the dataset was collected in PDF format and then processed in bulk using Mineru [21] for conversion into markdown format. Following this, structured question-answer (Q-A) pairs were extracted using regularization techniques. To ensure the integrity and accuracy of the data, all extracted Q-A pairs underwent a manual review and validation process, resulting in a high-quality dataset specifically tailored for

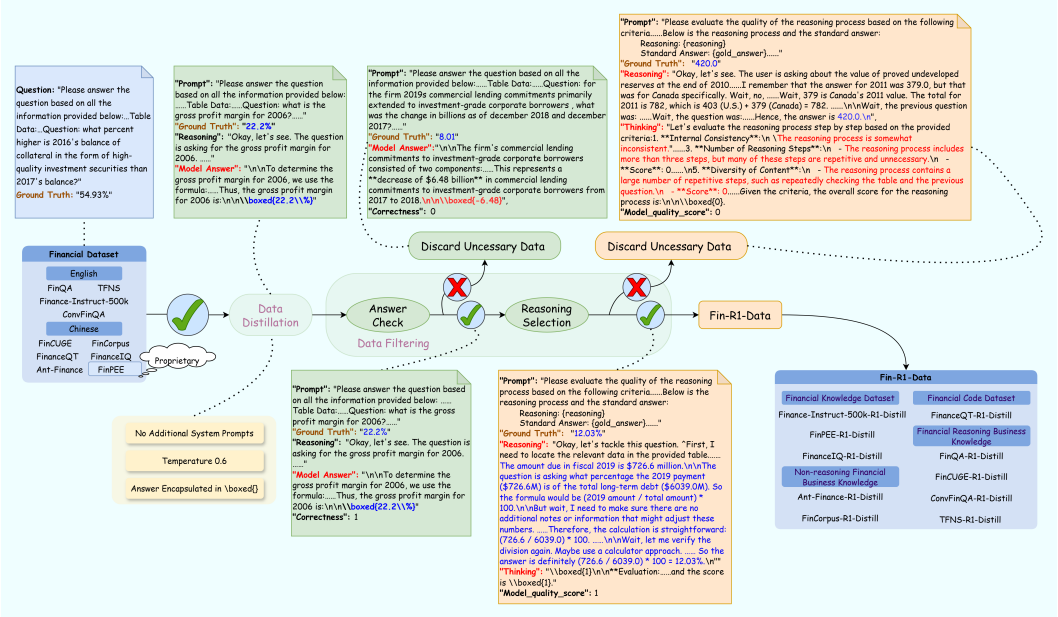



Figure 2: Stage 1 – The Data Construction Pipeline: (1) Data Distillation, (2) Answer Check, where an LLM evaluates the accuracy of responses generated by DeepSeek-R1, and (3) Reasoning Selection, where an LLM assesses and scores reasoning trajectories to ensure logical coherence and quality.

financial postgraduate examination problems. The composition structure of Fin-R1-Data across its various components is illustrated in Figure 3.

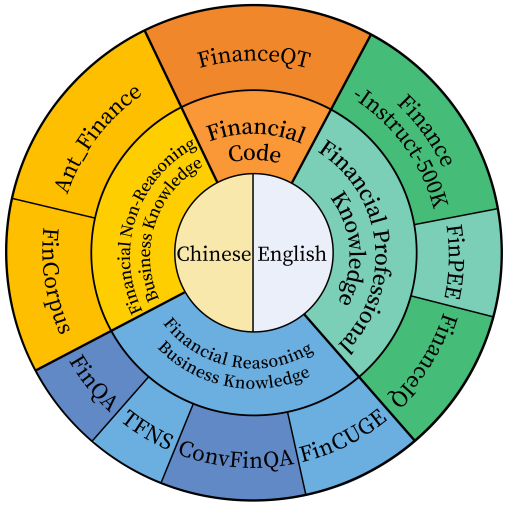



Figure 3: Composition structure of Fin-R1-Data: (1) Financial Code, (2) Financial Professional Knowledge, (3) Financial Reasoning Knowledge, and (4) Financial Non-Reasoning Knowledge.

As presented in Table 1, the table systematically details the descriptions, data sources, and proportional distribution of various data categories within Fin-R1-Data. The dataset is predominantly composed of financial non-reasoning business knowledge and financial reasoning business knowledge, which collectively constitute 77.9% of the total. These two categories comprehensively capture a wide

range of real-world financial business scenarios, ensuring extensive coverage of operational processes. Additionally, financial professional knowledge represents a significant component of the dataset, encompassing key concepts across multiple financial subfields and accounting for 21.9% of the total data. Furthermore, Fin-R1-Data includes a specialized subset of financial code data, designed for the development of quantitative trading strategies, though this category comprises only 0.2% of the dataset.

Table 1: Financial Data Categories and Sources

| **Data Category** | **Data Category Description** | **Source** | **Proportion** |
|---|---|---|---|
| Financial Code | Financial Quantitative Strategy Code Generation | FinanceQT | 0.2% |
| Financial Expertise | Financial Terminology Explanation, Q&A on Financial Expertise, Financial Calculations | Finance-Instruct-500K | 18.2% |
| | | FinanceIQ | 3.4% |
| | | FinPEE | 0.3% |
| Non-reasoning Financial Business Knowledge | Content Generation in Financial Business, Regulatory Compliance, Financial Knowledge, Financial Cognition, Financial Logic | Ant-Finance | 2.0% |
| | | FinCorpus | 48.4% |
| Financial Reasoning Business Knowledge | Numerical Reasoning on Financial Data, Financial News Sentiment Classification, Financial News Classification, Financial Causal Relationship Extraction | FinQA | 4.8% |
| | | ConvFinQA | 12.3% |
| | | TFNS | 4.0% |
| | | FinCUGE | 6.4% |

#### 2.2.2 Data Processing

Data Processing comprises both data distillation and data filtering. During the distillation phase, parameter configurations were rigorously aligned with the official DeepSeek-R1 specifications. The specific settings applied are outlined as follows:

(1) The temperature was set to 0.6.

(2) For mathematical data, the standardized prompt "Please use `\boxed{}` to wrap the final answer" was employed to ensure consistency in answer formatting.

(3) To maintain alignment with the intended reasoning pattern, a "\n" was forcibly appended at the beginning of each output before initiating data generation.

Data Filtering involves two primary components: answer check (evaluating the accuracy of model-generated responses) and reasoning selection (assessing the quality of reasoning trajectories). In the answer check phase, data filtering is conducted by retaining only responses that precisely align with the reference answers. Specifically, if a response generated by DeepSeek-R1 deviates from the standard answer provided in the dataset, it is immediately discarded. For objective questions, we employ exact matching to ensure correctness, while for subjective questions, we adopt the LLM-as-Judge to evaluate response validity. Regarding model selection and prompt optimization, we conducted a comparative assessment of GPT-4o [9] and Qwen2.5-72B-Instruct [28] across various prompting strategies. Experimental results demonstrate that Qwen2.5-72B-Instruct achieves an

accuracy rate of 99.6% in LLM-as-Judge tasks, surpassing GPT-4o. Consequently, Qwen2.5-72B-Instruct was selected as the judge model, and we determined the optimal prompt for evaluation. Further details on the experimental setup and findings are provided in Appendix A.3.

**Example of low-quality reasoning trajectory**

**"Prompt":** "Please evaluate the quality of the reasoning process based on the following criteria:.....Below is the reasoning process and the standard answer:
Reasoning: {reasoning}
Standard Answer: {gold_answer}..."
**"Reasoning":** "Okay, let's see. The user is asking about the value of proved undeveloped reserves at the end of 2010.....I remember that the answer for 2011 was 379.0, but that was for Canada specifically. Wait, no, .....Wait, 379 is Canada's 2011 value. The total for 2011 is 782, which is 403 (U.S.) + 379 (Canada) = 782. .....\n\nWait, the previous question was: .....Wait, the question was:.....Hence, the answer is 420.0.\n",
**"Thinking":** "Let's evaluate the reasoning process step by step based on the provided criteria:1. **Internal Consistency**:\n \The reasoning process is somewhat inconsistent."......3. **Number of Reasoning Steps**:\n - The reasoning process includes more than three steps, but many of these steps are repetitive and unnecessary.\n - **Score**: 0.....\n5. **Diversity of Content**:\n - The reasoning process contains a large number of repetitive steps, such as repeatedly checking the table and the previous question.\n - **Score**: 0.....Given the criteria, the overall score for the reasoning process is:\n\n\\boxed{0}.
**"Model_quality_score":** 0

**Example of high-quality reasoning trajectory**

**"Prompt":** "Please evaluate the quality of the reasoning process based on the following criteria:.....Below is the reasoning process and the standard answer:
Reasoning: {reasoning}
Standard Answer: {gold_answer}..."
**"Ground Truth":** "12.03%"
**"Reasoning":** "Okay, let's tackle this question. ^First, I need to locate the relevant data in the provided table..... The amount due in fiscal 2019 is $726.6 million.\n\nThe question is asking what percentage the 2019 payment ($726.6M) is of the total long-term debt ($6039.0M). So the formula would be (2019 amount / total amount) * 100.\n\nBut wait, I need to make sure there are no additional notes or information that might adjust these numbers. .....Therefore, the calculation is straightforward: (726.6 / 6039.0) * 100. .....\n\nWait, let me verify the division again. Maybe use a calculator approach. ..... So the answer is definitely (726.6 / 6039.0) * 100 = 12.03%.\n""
**"Thinking":** "\\boxed{1}\n\n**Evaluation:.....and the score is \\boxed{1}."
**"Model_quality_score":** 1

Figure 4: Examples of high-quality and low-quality reasoning selections filtering

In the reasoning selection phase, we drew inspiration from the study by Xie et al. [24] and distilled seven key dimensions from it: internal consistency, term overlap rate, number of reasoning steps, logical coherence, content diversity, task-domain relevance, and alignment with task instructions. These dimensions were employed to comprehensively evaluate the model's reasoning trajectory data. To ensure the robustness of the filtering process, we conducted experiments comparing the correlation scores between human annotators and models. The results, detailed in Appendix A.2, showed that the scores of Qwen2.5-72B-Instruct closely aligned with human judgments, exhibiting only minor discrepancies, while GPT-4o displayed larger deviations. Based on these findings, we selected Qwen2.5-72B-Instruct to assess the quality of the reasoning trajectories. Based on these evaluations, we systematically scored and filtered the reasoning paths, retaining only high-quality trajectories, which were subsequently curated into a refined dataset for supervised fine-tuning (SFT). In Figure 4, we present an example of a high-quality reasoning trajectory alongside a low-quality example, illustrating the distinction between them in the reasoning selection process.

### 2.3 Training method

We present Fin-R1, a large language model designed for financial optimization with enhanced financial reasoning capabilities, trained using financial reasoning data. The model is first trained via Supervised Fine-Tuning (SFT) using a high-quality financial reasoning dataset to enhance its reasoning ability. Building on this, we employ reinforcement learning to implement Group Relative Policy Optimization (GRPO), leveraging financial Q&A data and incorporating a dual reward

mechanism to improve both the accuracy of response formatting and content. Figure 5 intuitively summarizes the comprehensive training framework, illustrating the synergistic integration of the supervised learning and reinforcement learning components.

### 2.3.1 Training Template

In this section, we explain the training format of the data, and the specific prompt template will be illustrated in Figure 5, which details our training process.

**SFT Training Data** During the Supervised Fine-Tuning (SFT) phase, each sample $v$ in the training dataset $V$ comprises three components, i.e., $v = (x, c, y^*)$, where $x$ denotes the question, $c$ represents the reasoning trace formatted as `<reasoning>...</reasoning>`, and $y^*$ corresponds to the answer, formatted as `<answer>...</answer>`. During the SFT stage, $x$ is used as the input of the training set, and $c$ and $y^*$ are used as the output of the training set. This phase enables the model to learn structured financial reasoning patterns, refining its parameters to generate well-formed reasoning traces and accurate answers.

**RL Training Data** During the reinforcement learning (RL) phase, each sample $v$ in the training dataset $V$ consists of two components, i.e., $v = (x, y^*)$, where $x$ denotes the question and $y^*$ represents the model's output, which includes only the answer without reasoning traces. Reinforcement learning further enhances output quality by improving answer accuracy and ensuring compliance with the expected format.

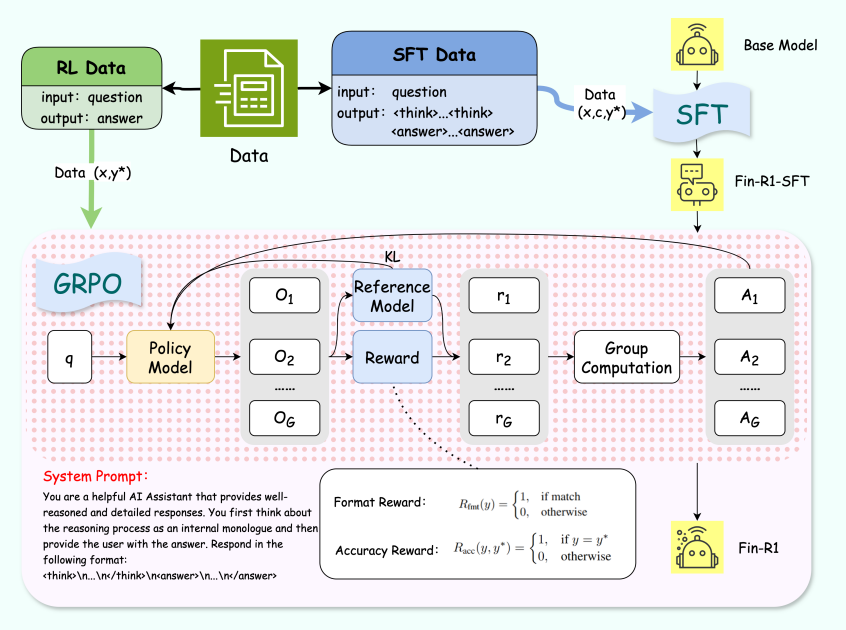



Figure 5: Stage 2-The pipeline of training construction. During the SFT phase, the base model undergoes SFT using a structured reasoning-augmented dataset, focusing on enhancing its ability to perform financial reasoning. During the RL phase, we apply GRPO for RL, which introduces a group computation mechanism to provide two reward signals—one for format correctness and one for content accuracy.

### 2.3.2 Supervised Fine-Tuning

We initially performed Supervised Fine-Tuning (SFT) on Qwen2.5-7B-Instruct, specifically optimizing key aspects of financial reasoning. This fine-tuning process effectively mitigated the reasoning failures observed when applying the general-purpose model to financial reasoning tasks. The training data consisted of the ConvFinQA and FinQA datasets. Following SFT, the model demonstrated enhanced performance in financial reasoning, as detailed in Table 2.

### 2.3.3 Group Relative Policy Optimization

During the reinforcement learning phase, we employ the Group Relative Policy Optimization (GRPO) algorithm, an advanced method that extends Proximal Policy Optimization (PPO).

For each training iteration, we sample $G$ candidate outputs $\{o_i\}_{i=1}^{G}$ from the old policy $\pi_{\text{old}}$. Each output receives a reward $r_i$, from which we compute the group-relative advantage $A_i$:

$$A_i = \frac{r_i - \mu_{\{r\}}}{\sigma_{\{r\}}}$$

where $\mu_{\{r\}}$ and $\sigma_{\{r\}}$ denote the mean and standard deviation of reward values within the group. Outputs exceeding group averages receive higher advantage values for prioritized optimization. The policy update now maximizes the following objective function:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{\mathbf{v}\sim P(\mathbf{V})}\, \mathbb{E}_{\{o_i\}\sim\pi_{\theta_{\text{old}}}}$$
$$\left[\frac{1}{G}\sum_{i=1}^{G} \min\left(r_i^{\text{ratio}} A_i,\ \text{clip}\left(r_i^{\text{ratio}}, 1-\epsilon, 1+\epsilon\right) A_i\right)\ -\beta\, D_{\text{KL}}(\pi_{\theta_{\text{old}}}(\cdot|\mathbf{v}) \,\|\, \pi_\theta(\cdot|\mathbf{v}))\right]$$

where $r_i^{\text{ratio}} = \frac{\pi_\theta(o_i|\mathbf{v})}{\pi_{\theta_{\text{old}}}(o_i|\mathbf{v})}$ represents the importance sampling ratio that quantifies the relative likelihood of generating output $o_i$ under the new policy $\pi_\theta$ compared to the old policy $\pi_{\theta_{\text{old}}}$; $A_i$ denotes the group-relative advantage, calculated by normalizing each reward with respect to the group's mean and standard deviation to emphasize outputs that surpass the group average; the clipping operator $\text{clip}\left(r_i^{\text{ratio}}, 1-\epsilon, 1+\epsilon\right)$ restricts the update magnitude within the trust region $[1-\epsilon, 1+\epsilon]$ to avoid destabilizing large parameter changes; the minimum operation between the unclipped term $r_i^{\text{ratio}} A_i$ and its clipped counterpart ensures a conservative update that balances aggressive improvements with training stability; and finally, the KL divergence term $D_{\text{KL}}(\pi_{\theta_{\text{old}}}(\cdot|\mathbf{v}) \,\|\, \pi_\theta(\cdot|\mathbf{v}))$, scaled by $\beta$, serves as a regularizer that penalizes significant deviations from the old policy, thereby promoting smoother and more controlled policy transitions.

### 2.3.4 Reward Function Design

In the process of training the reward model based on GRPO, we employs two reward mechanisms: format reward and accuracy reward.

**Format Reward** We encourage outputs that include a sequence of reasoning steps enclosed within `<reasoning>...</reasoning>` tags and a concise final answer enclosed within `<answer>...</answer>` tags. A format incentive score of 1 is awarded if all four tags appear exactly once with no extraneous content outside these tags; otherwise, a score of 0 is assigned. The format reward function is defined as follows:

$$R_{\text{fmt}}(y) = \begin{cases} 1, & \text{if the format matches} \\ 0, & \text{otherwise} \end{cases}$$

where $y$ denotes the model's output. Format matching indicates that the output strictly adheres to the specified format by containing exactly one pair of `<reasoning>` tags and one pair of `<answer>` tags, with no additional content outside these tags.

**Accuracy Reward** In the financial scenario, we observed that it is challenging to exhaustively enumerate answer regular expressions using rule-based methods. The examples that are difficult to identify are presented in Figure 6. Consequently, we adopt Qwen2.5-Max [18]as the judge for answer evaluation. The content enclosed within the `<answer>` ... `</answer>` tags is extracted from the completions model output using regular expressions, with the resulting solution serving as the standard answer. If the output within the `<answer>` ... `</answer>` tags is semantically consistent with the standard answer, a reward of 1 is assigned; otherwise, the reward is 0. The specific prompts for the LLM as judge are provided in Appendix A.3. The accuracy reward function is defined as follows:

$$R_{\text{acc}}(y, y^*) = \begin{cases} 1, & \text{if } y = y^* \\ 0, & \text{otherwise} \end{cases}$$

where $y$ is model's output (from <answer> ... </answer> tags). $y^*$ is the standard answer.

### 2.4 Evaluation

#### 2.4.1 Evaluation Datasets

We establishe a financial domain multi-task benchmarking framework by systematically validating five representative open-source heterogeneous datasets: FinQA, ConvFinQA, Ant-Finance, TFNS, and Finance-Instruct-500k. Notably, except for Finance-Instruct-500k where a custom 10% test subset was extracted through stratified sampling from the complete data preprocessing pipeline, all other datasets strictly adhere to their original publicly available official evaluation splits.To control costs and maintain relatively uniform data distribution, for each evaluation set, we randomly sample 1,000 data entries for evaluation. If an evaluation set has fewer than 1,000 entries, we evaluate all of them.


(a) Mismatched reserved decimal places.

(b) Different format.

Figure 6: The format difference between the model output and the ground truth is shown. Figure 6a illustrates the difference in decimal placement, while Figure 6b shows the difference in expression.

#### 2.4.2 Evaluation Method

The financial evaluation datasets employed in this study, except Finance-Instruct-500k, feature objective question formats with definitive and unique reference answers. Given that numerical calculation problems may induce discrepancies between model outputs and reference answers in representational formats, as shown in Figure 6 (manifested as equivalent conversion issues between percentage and decimal representations or differences in significant digit retention), we implements a large language model as an automated evaluation judge for answer check, adopting the prompt design and evaluation methodology proposed by Zhu et al. [35]. Notably, although this evaluation paradigm operates at a low level of surface complexity, systematic prompt engineering optimization strategies were implemented to ensure assessment reliability. Multi-dimensional tuning experiments were conducted on critical parameters of the prompt template, including but not limited to format specification directives, numerical precision constraints, and fault tolerance rule configurations. Detailed experimental designs and results are analyzed in the Appendix A.3.

## 3 Experiment

### 3.1 Baselines

To comprehensively evaluate the reasoning capabilities of Fin-R1 in financial scenarios, we conducted a thorough comparative assessment against multiple state-of-the-art models. These models include DeepSeek-R1, DeepSeek-R1-Distill-Qwen-7B, DeepSeek-R1-Distill-Qwen-14B, DeepSeek-R1-Distill-Qwen-32B, DeepSeek-R1-Distill-Llama-70B, Fin-R1-SFT, Qwen-2.5-7B-Instruct, Qwen-2.5-14B-Instruct, and Qwen-2.5-32B-Instruct. The selection of these models encompasses a spectrum ranging from lightweight to high-performance architectures, taking into account factors such as reasoning capability and computational resource consumption. This comprehensive comparison aims to provide a holistic evaluation of Fin-R1's performance within financial applications.

### 3.2 Results

In our comprehensive benchmarking evaluation covering multiple financial business scenarios, Fin-R1 demonstrated remarkable performance advantages despite its lightweight 7B parameter scale. Ultimately, it achieved an average score of 75.2, securing second place overall. Notably, Fin-R1 outperformed all participating models of similar scale, with only a 3.8-point performance gap compared to the industry benchmark DeepSeek-R1 (78.2). Furthermore, it surpassed DeepSeek-R1-Distill-Llama-70B (69.2) by 8.7 points. Furthermore, Fin-R1 achieved top rankings in two critical tasks:

FinQA, which focuses on real-world financial table-based numerical reasoning, and ConvFinQA, which evaluates multi-turn interactive financial reasoning. It obtained scores of 76.0 and 85.0, respectively, surpassing all competing models. These results highlight Fin-R1's strong capabilities in both financial reasoning and non-reasoning scenarios. Although Fin-R1 underwent specialized training primarily for FinQA and ConvFinQA, it exhibited significant performance improvements in other financial benchmarks, including Ant_Finance, TFNS, and Finance-Instruct-500K. This suggests the model possesses robust cross-task generalization capabilities, further underscoring its effectiveness in diverse financial applications.

Table 2: Financial Scenario Evaluation Results.

| Model | Parameters | FinQA | ConvFinga | Ant_Finance | TFNS | Finance-Instruct-500K | Average |
|---|---|---|---|---|---|---|---|
| DeepSeek-R1 | 671B | 71.0 | 82.0 | **90.0** | 78.0 | **70.0** | **78.2** |
| Qwen-2.5-32B-Instruct | 32B | 72.0 | 78.0 | 84.0 | 77.0 | 58.0 | 73.8 |
| DeepSeek-R1-Distill-Qwen-32B | 32B | 70.0 | 72.0 | 87.0 | **79.0** | 54.0 | 72.4 |
| Fin-R1-SFT | 7B | 73.0 | 81.0 | 76.0 | 68.0 | 61.4 | 71.9 |
| Qwen-2.5-14B-Instruct | 14B | 68.0 | 77.0 | 84.0 | 72.0 | 56.0 | 71.4 |
| DeepSeek-R1-Distill-Llama-70B | 70B | 68.0 | 74.0 | 84.0 | 62.0 | 56.0 | 69.2 |
| DeepSeek-R1-Distill-Qwen-14B | 14B | 62.0 | 73.0 | 82.0 | 65.0 | 49.0 | 66.2 |
| Qwen-2.5-7B-Instruct | 7B | 60.0 | 66.0 | 85.0 | 68.0 | 49.0 | 65.6 |
| DeepSeek-R1-Distill-Qwen-7B | 7B | 55.0 | 62.0 | 71.0 | 60.0 | 42.0 | 58.0 |
| **Fin-R1** | 7B | **76.0** | **85.0** | 81.0 | 71.0 | 62.9 | 75.2 |

## 4 Conclusion and Future work

We introduces the financial reasoning large language model Fin-R1, which effectively addresses three core challenges in financial AI applications: fragmented financial data, uncontrollable reasoning logic, and weak business generalization ability. By constructing the high-quality financial reasoning CoT dataset Fin-R1-Data followed by model training through SFT (Supervised Fine-Tuning) and RL (Reinforcement Learning), forms a two-stage workflow framework within the financial domain, Fin-R1 achieves the state-of-the-art performance among evaluated models, scoring 85.2 and 76.1 in the ConvFinQA and FinQA , respectively. Our approach has significantly advanced the application of large language models in the financial domain. In the future, we will focus on advancing the integration and innovation of fintech field. On one hand, we will refine our architecture for financial multimodal scenarios and deepen its application exploration in cutting - edge areas, promoting the financial industry's intelligent and compliant development. On the other hand, we will drive the widespread adoption of large language models in finance, fostering deeper integration with financial applications to enhance risk management and regulatory compliance, ultimately expanding the practical utility of the model.

## Limitations

Although the model has achieved significant improvements in the financial domain, our study still has three main limitations:

– **Limited training dataset coverage**: The current training data of the model is confined to ConvFinQA and FinQA only, and it has not yet reached the satisfactory target. Future training will be expanded to more diverse financial datasets.

– **Single-modality architecture limitation**: The current model, based on a pure text architecture, struggles to handle financial reports containing visual elements. We will consider multimodal extension solutions for financial chart understanding and reasoning in the future.

– **Closed-scenario focus bias**: The current evaluation mainly targets reasoning questions with clear standard answers, and open-ended financial text question answering has not been designed.

Although we currently have the above limitations, in the future, we will redouble our efforts to address these potential shortcomings. We believe that these improvements will significantly enhance the model's applicability and effectiveness in real-world financial scenarios.

## References


[1] Zihan Dong, Xinyu Fan, and Zhiyuan Peng. "Fnspid: A comprehensive financial news dataset in time series". In: *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2024, pp. 4918–4927.

[2] Georgios Fatouros et al. "Can large language models beat wall street? unveiling the potential of ai in stock selection". In: *arXiv preprint arXiv:2401.03737* (2024).

[3] Joseph G. Flowers. *Finance Instruct 500K*. Dataset. Accessed: 2025-03-18. 2025. URL: https://huggingface.co/datasets/Josephgflowers/Finance-Instruct-500k.

[4] Daya Guo et al. "Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning". In: *arXiv preprint arXiv:2501.12948* (2025).

[5] Xin Guo et al. "FinEval: A Chinese Financial Domain Knowledge Evaluation Benchmark for Large Language Models". In: *arXiv preprint arXiv:2308.09975* (2024).

[6] Xiang Li et al. "AlphaFin: Benchmarking Financial Analysis with Retrieval-Augmented Stock-Chain Framework". In: *arXiv preprint arXiv:2403.12582* (2024).

[7] Dakuan Lu et al. "BBT-Fin: Comprehensive Construction of Chinese Financial Domain Pre-trained Language Model, Corpus and Benchmark". In: *arXiv preprint arXiv:2302.09432* (2023). URL: https://arxiv.org/abs/2302.09432.

[8] Lukas Malik. *Quant-Trading-Instruct*. Dataset. Accessed: 2024-03-18. 2024.

[9] OpenAI. *GPT-4o Model Documentation: Parameter Configuration and Data Formats*. https://platform.openai.com/docs/models/gpt-4o. Accessed: 2025-03-18. 2024.

[10] OpenAI. *Learning to reason with llms*. 2024. URL: https://openai.com/index/learnin%20g-to-reason-with-llms/..

[11] Lingfei Qian et al. "Fino1: On the Transferability of Reasoning Enhanced LLMs to Finance". In: *arXiv preprint arXiv:2502.08127* (2025).

[12] Qwen. *Qwq: Reflect deeply on the boundaries of the unknown*. 2024. URL: https://github.com/QwenLM/QwQ.

[13] Zhihong Shao et al. "DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models". In: *arXiv preprint* (2024). Accessed: 2024-03-18. arXiv: 2402.03300 [cs.CL]. URL: https://arxiv.org/abs/2402.03300.

[14] Alipay Team. *Financial Evaluation Dataset*. Software. Accessed: 2024-03-18. 2023. URL: https://github.com/alipay/financial_evaluation_dataset.

[15] Duxiaoman-DI Team. *FinanceIQ*. Software. Accessed: 2024-03-18. 2023. URL: https://github.com/Duxiaoman-DI/XuanYuan/tree/main/FinanceIQ.

[16] Duxiaoman-DI Team. *FinCorpus*. Dataset. Accessed: 2024-03-18. 2023. URL: https://huggingface.co/datasets/Duxiaoman-DI/FinCorpus.

[17] Duxiaoman-DI Team. *XuanYuan-FinX1-Preview*. Software. Accessed: 2024-03-18. 2024. URL: https://github.com/Duxiaoman-DI/XuanYuan.

[18] Qwen Team. "Qwen 2.5 Technical Report". In: *arXiv preprint* (2024). DOI: 10.48550/arXiv.2412.15115. arXiv: 2412.15115. URL: https://arxiv.org/abs/2412.15115.

[19] Hanshuang Tong et al. "Ploutos: Towards interpretable stock movement prediction with financial large language model". In: *arXiv preprint arXiv:2403.00782* (2024).

[20] *Twitter Financial News Sentiment*. Dataset. Accessed: 2024-03-18. 2024. URL: https://huggingface.co/datasets/zeroshot/twitter-financial-news-sentiment.

[21] Bin Wang et al. *MinerU: An Open-Source Solution for Precise Document Content Extraction*. 2024. arXiv: 2409.18839 [cs.CV]. URL: https://arxiv.org/abs/2409.18839.

[22] Saizhuo Wang et al. "Alpha-gpt: Human-ai interactive alpha mining for quantitative investment". In: *arXiv preprint arXiv:2308.00016* (2023).

[23] Saizhuo Wang et al. "Quantagent: Seeking holy grail in trading by self-improving large language model". In: *arXiv preprint arXiv:2402.03755* (2024).

[24] Qianqian Xie et al. "Finnlp-agentscen-2024 shared task: Financial challenges in large language models-finllms". In: *Proceedings of the Eighth Financial Technology and Natural Language Processing and the 1st Agent AI for Scenario Planning*. 2024, pp. 119–126.

[25] Cheng Xu et al. "MMBench: Benchmarking End-to-End Multimodal DNNs and Understanding Their Hardware-Software Implications". In: *2023 IEEE International Symposium on Workload Characterization (IISWC)*. IEEE. Reno, NV, USA, Oct. 2023, pp. 154–166.

[26] Congluo Xu, Zhaobin Liu, and Ziyang Li. "FinArena: A Human-Agent Collaboration Framework for Financial Market Analysis and Forecasting". In: *arXiv preprint arXiv:2503.02692* (2025).

[27] Siqiao Xue et al. "FAMMA: A Benchmark for Financial Domain Multilingual Multimodal Question Answering". In: *arXiv preprint arXiv:2410.04526* (2024).

[28] An Yang et al. "Qwen2. 5 technical report". In: *arXiv preprint arXiv:2412.15115* (2024).

[29] Yangyang Yu et al. "Fincon: A synthesized llm multi-agent system with conceptual verbal reinforcement for enhanced financial decision making". In: *arXiv preprint arXiv:2407.06567* (2024).

[30] Yangyang Yu et al. "FinMem: A Performance-Enhanced LLM Trading Agent with Layered Memory and Character Design". In: *Proceedings of the AAAI Symposium Series 2024*. Austin, TX, USA: AAAI Press, Feb. 2024, pp. 595–597. URL: https://www.aaai.org/ocs/index.php/SSS/SSS24/paper/view/29999.

[31] Wentao Zhang et al. "Finagent: A multimodal foundation agent for financial trading: Tool-augmented, diversified, and generalist". In: *arXiv e-prints* (2024), arXiv–2402.

[32] Haiyan Zhao et al. "Explainability for large language models: A survey". In: *ACM Transactions on Intelligent Systems and Technology* 15.2 (2024), pp. 1–38.

[33] Yu Zhao et al. "Marco-o1: Towards open reasoning models for open-ended solutions". In: *arXiv preprint arXiv:2411.14405* (2024).

[34] Wenxuan Zhou et al. "Universalner: Targeted distillation from large language models for open named entity recognition". In: *arXiv preprint arXiv:2308.03279* (2023).

[35] Lianghui Zhu, Xinggang Wang, and Xinlong Wang. *JudgeLM : Fine-tuned Large Language Models are Scalable Judges*. 2024. URL: https://openreview.net/forum?id=87YOFayjcG.

# A Appendix

## A.1 The Prompt of data construct

During the entire data construction process, we constructed prompts in three key processes respectively. Firstly, in the data distillation stage, we referred to the official prompt setting of DeepSeek - R1 and constructed the prompt shown in Figure 7. Secondly, in the first stage of data screening, for the task of "LLM-as-Judge", we compared various prompts and finally determined the optimal prompt. The details are shown in Figure 10. Finally, in order to obtain high-quality inference trajectories, in the second stage of data screening, we proposed seven indicators for evaluating the inference trajectories of the model and carefully constructed the prompt in Figure 8.

**The Prompt for data distillation**

```
"""
Task description: {task_desc}
Task input: {task_input}
Task instruction: {task_instruction}
Please analyze the given input based on the above task description and instruction, and
generate the output that meets the requirements.

Notes:
1. Reason according to the task instruction and provide the final answer.
2. Please enclose the final answer in the format \\boxed{{}}, for regular extraction. For
example: \\boxed{{Final answer}}
3. Add "\\n" at the beginning of each output before generating the data.
"""
```

Figure 7: The prompt of data distillation that we used

### A.2 The Prompt of reasoning selection

To compare the scoring outcomes between human annotators and language models, we conducted supplementary experiments. Specifically, we randomly selected 20 data points from the dataset filtered in the initial preprocessing step and evaluated their reasoning performance using Qwen2.5-72B-Instruct and GPT-4o. The evaluation followed seven predefined judgment criteria. Each data point received a score of 1 if its reasoning satisfied a given criterion and 0 otherwise. The total score for each data point was obtained by summing across all criteria, resulting in a range from 0 (minimum) to 7 (maximum). Given the scoring framework, we effectively employed a binary scoring approach (0/1) at the criterion level.

**The Prompt for reasoning selection**

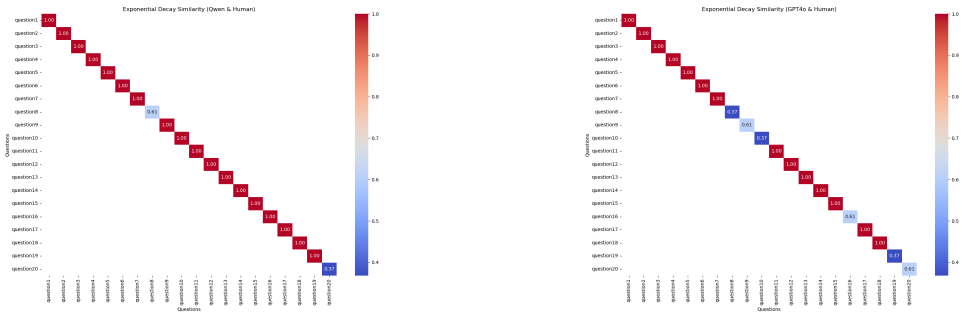
```
Please evaluate the quality of the reasoning process based on the following criteria:
        1. **Internal Consistency**: Check if the steps in the reasoning process are
consistent and logically lead to the correct answer.
        2. **Term Overlap**: Check the overlap between terms used in the reasoning
process and the standard answer. The higher the overlap, the better.
        3. **Number of Reasoning Steps**: Ensure the reasoning process includes enough
steps (at least 3 steps).
        4. **Logical Consistency**: Ensure that the steps in the reasoning process align
with the standard answer in terms of logic, with no major errors or omissions.
        5. **Content Diversity**: Check if the reasoning process contains repetitive
steps.
        6. **Task Domain Relevance**: Ensure the reasoning is relevant to the task domain
(task domain: {task_domain}). Higher relevance to the task domain should result in a
higher score.
        7. **Task Instruction Relevance**: Check if the reasoning closely follows the
task instructions. The more closely the reasoning matches the task instructions, the
higher the score. Task instructions: {task_instruction}

        Below is the reasoning process and the standard answer:
        Reasoning: {reasoning}
        Standard Answer: {gold_answer}

        Please provide a score based on the following criteria:
        - The seven indicators mentioned above are scored 1 point each. If the total
score is 7, it is considered a high-quality reasoning trajectory, and the output is 1.
Otherwise, the output is 0.
        - Please wrap the score in \\boxed{{}} format for easy extraction using regex.
For example: \\boxed{{final_score}}
```

Figure 8: The Prompt for reasoning selection

To establish a reference baseline, human annotators independently scored the reasoning for the same data points. We then visualized the correlation between the scoring distributions of Qwen2.5-72B-Instruct, GPT-4o, and human annotations using heatmaps (see Figure 9) to assess their alignment and discrepancies. The results show that Qwen2.5-72B-Instruct exhibits high concordance with human annotations, with most questions having a correlation score of 1, and only minor deviations in a few cases. In contrast, GPT-4o shows larger discrepancies, indicating lower alignment with human judgments. Based on these findings, we ultimately selected Qwen2.5-72B-Instruct as the scoring model for reasoning selection.




(a) Qwen2.5-72B-Instruct vs. Human

(b) GPT-4o vs. Human

Figure 9: Heatmap comparison of reasoning scores between LLMs and human annotators. Figure 9a and 9b represent the correlation between the scores of Qwen2.5-72B-Instruct and GPT-4o with human scores.

### A.3 The Prompt of Judging

In the research on answer verification tasks based on LLM-as-Judge, we reveals that although the surface task format appears relatively simple (i.e. determining binary output 1 or 0 based on consistency between model-generated answers and reference answers), different prompt wording strategies significantly influence the performance of evaluation models. To quantitatively analyze this phenomenon, we randomly selected 100 sample instances from the FinQA dataset and conducted five repeated experiments for each prompt strategy to assess result stability. This process yielded 500 comparative results per prompt group, with model performance evaluated through consistency analysis against human-annotated results. Experiments were conducted using both GPT-4o and Qwen2.5-72B-Instruct.

To systematically evaluate the impact of different prompting strategies on evaluation model performance, this study established a quantitative evaluation metrics system comprising two core indicators:

- **Classification Inaccuracy:** defined as the proportion of samples where model judgments disagree with human annotations.
- **Format Irregularity:** reflecting the degree to which model outputs fail to strictly adhere to binary constraints (0/1).

Through statistical analysis of 500 comparative results under each prompting strategy, the performance comparison data are shown in Table 3. The specific prompt formats include our used format as OF, the format where the content to be judged is at the end as CIE, the format with the original question passed in as WQ, the format with the original question passed in and the question-and-answer content placed at the end as CIE-WQ and the Chinese format as ZH. The prompts are shown in Figure 10, 11, 12, 13, 14.

| | **Inaccuracy** | | **Irregularity** | |
|---|---|---|---|---|
| **Format** | GPT-4o | Qwen2.5-72B-Instruct | GPT-4o | Qwen2.5-72B-Instruct |
| OF | 2.8% | 0.4% | 0.8% | 0.0% |
| CIE | 2.0% | 2.0% | 0.0% | 0.0% |
| WQ | 6.0% | 8.0% | 3.6% | 3.2% |
| CIE-WQ | 4.8% | 9.6% | 1.6% | 3.2% |
| ZH | 5.2% | 1.6% | 0.0% | 0.0% |

Table 3: Comparison of GPT-4o and Qwen2.5-72B-Instruct on answer judgment inaccuracy and irregularity across different prompt formats.The specific prompt formats include our used format as OF, the format where the content to be judged is at the end as CIE, the format with the original question passed in as WQ, the format with the original question passed in and the question-and-answer content placed at the end as CIE-WQ and the Chinese format as ZH.

The systematic analysis based on experimental data reveals that different prompt strategies significantly influence the performance of evaluation models. We analyze the results as follows:

- Text positioning strategies demonstrate model-specific differences. GPT-4o shows stable performance under the CIE strategy when reference answers are post-positioned, with an inaccuracy rate of 2.0%, while Qwen2.5-72B-Instruct exhibits superior adaptation to the rule-preceding OF strategy, achieving an extremely high accuracy of 99.6%.
- Although incorporating original questions as contextual information theoretically enhances semantic comprehension, it substantially increases the format deviation rates (Irregularity). Under the WQ strategy, GPT-4o and Qwen2.5-72B-Instruct exhibit 3.6% and 3.2% Irregularity respectively. Manual verification identifies that format deviations predominantly occur in long-text samples, potentially due to input sequence elongation inducing model hallucinations (e.g., Qwen2.5-72B-Instruct's classification error rate under WQ strategy surges from baseline 0.4% to 8.0%).
- Cross-lingual testing indicates that Chinese prompts (ZH), while partially ensuring format compliance, yield significantly higher classification errors than optimal English strategies due to the English evaluation context. Compared with GPT-4o, Qwen2.5-72B-Instruct demonstrates better Chinese prompt adaptability.

Base on the above analyses, we ultimately select Qwen2.5-72B-Instruct as the judge model. Moreover, we proposes the following optimizations: (1) Prompt engineering should account for model architecture characteristics, as employing model-specific prompt structures may enhance evaluation accuracy. (2) In answer verification tasks, unnecessary long contextual inputs should be minimized to effectively reduce format deviations.

neurips$_2$024

**The Prompt for judging(Our used Format:OF)**

```
You are a scoring assistant for financial questions. I will provide you with a
<ground truth> and a <model answer>. Please determine whether the <model answer> has the
same meaning as the <ground truth> according to the following rules. If they are
consistent, output 1, otherwise output 0.

<ground truth>
{Truth}
<ground truth>

<model answer>
{Answer}
<model answer>

### Rules:
1.If the <ground truth> is a numerical value, and the format of the <model answer> is
different from that of the <ground truth>, but the numerical values are the same, then
it is considered that the meanings are consistent. For example, if the
<ground truth> is 0.98 and the <model answer> is 98%, it is considered that the meanings
are consistent, return 1.
2.If the <ground truth> is a numerical value, and the finalresult of the <model answer> is
consistent with the <ground truth> after rounding, then it is considered that the meanings
are consistent. For example, if the <ground truth> is 2 and the <model answer> is 1.98, it
is considered that the meanings are consistent, return 1.
### Output Format:
Make the judgment according to the above rules, and finally put the judgment result 1 or 0
 in boxed{{}}, for example, boxed{{1}} or boxed{{0}}
```

Figure 10: The prompt for judging the model answer that we used.

**The Prompt for judging(Content-In-End Format:CIE)**

```
You are a scoring assistant for financial questions. I will provide you with a
<ground truth> and a <model answer>. Please determine whether the <model answer> has the
same meaning as the <ground truth> according to the following rules. If they are
consistent, output 1, otherwise output 0.
### Rules:
1.If the <ground truth> is a numerical value, and the format of the <model answer> is
different from that of the <ground truth>, but the numerical values are the same, then
it is considered that the meanings are consistent. For example, if the
<ground truth> is 0.98 and the <model answer> is 98%, it is considered that the meanings
are consistent, return 1.
2.If the <ground truth> is a numerical value, and the final result of the
<model answer> is consistent with the <ground truth> after rounding, then it is considered
that the meanings are consistent. For example, if the <ground truth> is 2 and the
<model answer> is 1.98, it is considered that the meanings are consistent, return 1.
### Output Format:
Make the judgment according to the above rules, and finally put the judgment result 1 or 0
 in boxed{{}}, for example, boxed{{1}} or boxed{{0}}

<ground truth>
{Truth}
<ground truth>

<model answer>
{Answer}
<model answer>
```

Figure 11: The prompt for judging the model answer that the content comes at the end.

**The Prompt for judging the answer(With-Question Format:WQ)**

```
You are a scoring assistant for financial questions. I will provide you with a <question>
 and its <ground truth> and  <model answer>. Please determine whether the
<model answer> has the same meaning as the <ground truth> according to the following
rules. If they are consistent, output 1, otherwise output 0.

<question>
{Question}
<question>

<ground truth>
{Truth}
<ground truth>

<model answer>
{Answer}
<model answer>

### Rules:
1.If the <ground truth> is a numerical value, and the format of the <model answer> is
different from that of the <ground truth>, but the numerical values are the same, then
it is considered that the meanings are consistent. For example, if the
<ground truth> is 0.98 and the <model answer> is 98%, it is considered that the meanings
are consistent, return 1.
2.If the <ground truth> is a numerical value, and the final result of the
<model answer> is consistent with the <ground truth> after rounding, then it is considered
that the meanings are consistent. For example, if the <ground truth> is 2 and the
<model answer> is 1.98, it is considered that the meanings are consistent, return 1.
### Output Format:
Make the judgment according to the above rules, and finally put the judgment result 1 or 0
 in boxed{{}}, for example, boxed{{1}} or boxed{{0}}
```

Figure 12: The prompt for judging the model answer which is combined with the question.

**The Prompt for judging(Content-In-End-With-Question Format:CIE-WQ)**

```
You are a scoring assistant for financial questions. I will provide you with  a <question>
 and its <ground truth> and  <model answer>. Please determine whether the <model answer>
has the same meaning as the <ground truth> according to the following
rules. If they are consistent, output 1, otherwise output 0.
### Rules:
1.If the <ground truth> is a numerical value, and the format of the <model answer> is
different from that of the <ground truth>, but the numerical values are the same, then
it is considered that the meanings are consistent. For example, if the <ground truth> is
 0.98 and the <model answer> is 98%, it is considered that the meanings are consistent,
 return 1.
2.If the <ground truth> is a numerical value, and the final result of the <model answer>
  is consistent with the <ground truth> after rounding, then it is considered that
the meanings are consistent. For example, if the <ground truth> is 2 and the
<model answer> is 1.98, it is considered that the meanings are consistent, return 1.
### Output Format:
Make the judgment according to the above rules, and finally put the judgment result 1 or 0
 in boxed{{}}, for example, boxed{{1}} or boxed{{0}}

<question>
{Question}
<question>

<ground truth>
{Truth}
<ground truth>

<model answer>
{Answer}
<model answer>
```

Figure 13: The prompt for judging the model answer, which is combined with the question, comes at the end.

**The Prompt for judging the answer(Chinese Language Format:ZH)**

你是一个金融题目结果评分助手，我会给你一个<标准答案>与一个 <模型回答>，请根据以下规则判断<模型回答>是否与<标准答案>的含义一致。如果一致，输出1，否则输出0。

<标准答案>
{Truth}
<标准答案>

<模型回答>
{Answer}
<模型回答>

### 规则:
1.如果<标准答案>是一个数值，<模型回答>与<标准答案>的格式不一样，但是数值一致，则认为含义一致。例如:
<标准答案>是0.98，<模型回答>是98%，认为含义一致，返回1
2.如果<标准答案>是一个数值，<模型回答>的最终结果经四舍五入后与<标准答案>一致，则认为含义一致。例如:
<标准答案>是2，<模型回答>是1.98，认为含义一致，返回1
### 回复格式:
按照以上规则给出判断，并在最后将判断结果1 or 0放在boxed{{}}中，例如boxed{{1}}或boxed{{0}}

Figure 14: The Chinese prompt for judging the model answer.