TaxReasoning: Benchmarking Knowledge-Intensive Mathematical Reasoning with Evolving Tax Laws

Nan Hu[1,2*], Yike Wu[1,2*], Jiaye Li[1,2], Huikang Hu[1,2], Guilin Qi[1,2†], Songlin Zhai[1,2], Yongrui Chen[1,2], Tianxing Wu[1,2], Tongtong Wu[3], Jiaoyan Chen[4], Jeff Z. Pan[5]

[1]School of Computer Science and Engineering, Southeast University, China
[2]Key Laboratory of New Generation Artificial Intelligence Technology and its Interdisciplinary Applications (Southeast University), Ministry of Education, China
[3] Monash University, Australia
[4]The University of Manchester, United Kingdom
[5]The University of Edinburgh, United Kingdom
{nanhu, yike.wu, gqi}@seu.edu.cn

Abstract

Recent studies have explored the capabilities of large language models (LLMs) in solving knowledge-intensive mathematical reasoning problems. However, existing benchmarks predominantly involve static theorems that LLMs have encountered during pretraining, failing to assess dynamic knowledge integration. In this work, we introduce TAXREASONING, a novel benchmark designed to evaluate LLMs' abilities in real-world tax calculation scenarios. These tasks require not only mathematical reasoning and numerical computation, but also the extraction and application of complex, frequently updated tax regulations. Through extensive experiments with state-of-the-art LLMs using diverse prompting strategies and knowledge augmentation techniques, we uncover substantial limitations in their ability to handle dynamic, knowledge-intensive questions—primarily due to missing domain-specific knowledge and ineffective retrieval. Even the best-performing models fall significantly short of human-level performance. Our analysis points to key avenues for improvement, including enhancing LLMs' reasoning capabilities, developing more effective knowledge summarization techniques, and improving retrieval strategies. TaxReasoning offers a critical testbed for advancing LLMs in dynamic knowledge-intensive domains.

Introduction

Large language models (LLMs) demonstrate remarkable capabilities in tackling complex real-world reasoning tasks (Jaech et al. 2024b; DeepSeek-AI et al. 2025; Yang et al. 2025). Among these, mathematical reasoning has emerged as a key lens for evaluating LLMs' reasoning abilities (Lu et al. 2023b; Chen et al. 2023), as it requires not only contextual understanding but also precise logical inference and numerical computation (Zheng, Lapata, and Pan 2024).

A wide range of benchmarks have been developed to assess mathematical reasoning, spanning educational levels from elementary to university mathematics (Koncel-Kedziorski et al. 2016; Wang, Liu, and Shi 2017; Amini et al. 2019; Miao, Liang, and Su 2020; Patel, Bhattamishra, and Goyal 2021; Hendrycks et al. 2021; Lu et al. 2023a). However, most of these benchmarks are limited to general mathematical problems and do not incorporate domain-specific knowledge. In practice, LLMs are often expected to solve highly specialized and knowledge-intensive tasks that go beyond general-purpose reasoning.

Recent research has started to explore LLMs' performance in knowledge-intensive mathematical reasoning, such as in scientific reasoning benchmarks (Chen et al. 2023; Wang et al. 2024; Sun et al. 2024) and financial reasoning datasets (Zhao et al. 2024; Tang et al. 2025), where domain knowledge plays a central role. Yet, these datasets primarily focus on the application of static or well-known domain knowledge, which is already memorized during pretraining. For example, Wang et al. report that LLMs make fewer than only 10% of errors in their benchmark stem from missing external knowledgee. Similarly, Tang et al. show that DeepSeek-R1 achieves over 90% accuracy on most financial reasoning benchmarks. These results suggest that existing benchmarks are insufficient for evaluating LLMs' ability to reason over new or frequently updated domain knowledge.

To address this gap, we present TaxReasoning, a benchmark for evaluating LLMs' ability to reason over complex and evolving tax law knowledge. It contains 940 expert-annotated questions grounded in real-world scenarios, each requiring interpretation and integration of multiple clauses from recent regulatory documents. As shown in Figure 1, solving a single question often involves synthesizing piecewise formulas, tabular data, conditional logic, and textual rules from diverse tax sources.

We conduct a comprehensive evaluation of 13 LLMs, including both reasoning-augmented and general-purpose models, under various prompting and knowledge augmentation strategies. These include Chain-of-Thought (CoT) (Wei et al. 2022) and Program-of-Thought (PoT) (Chen et al. 2022) to assess internal reasoning capabilities, as well as

*These authors contributed equally.
†Corresponding author.

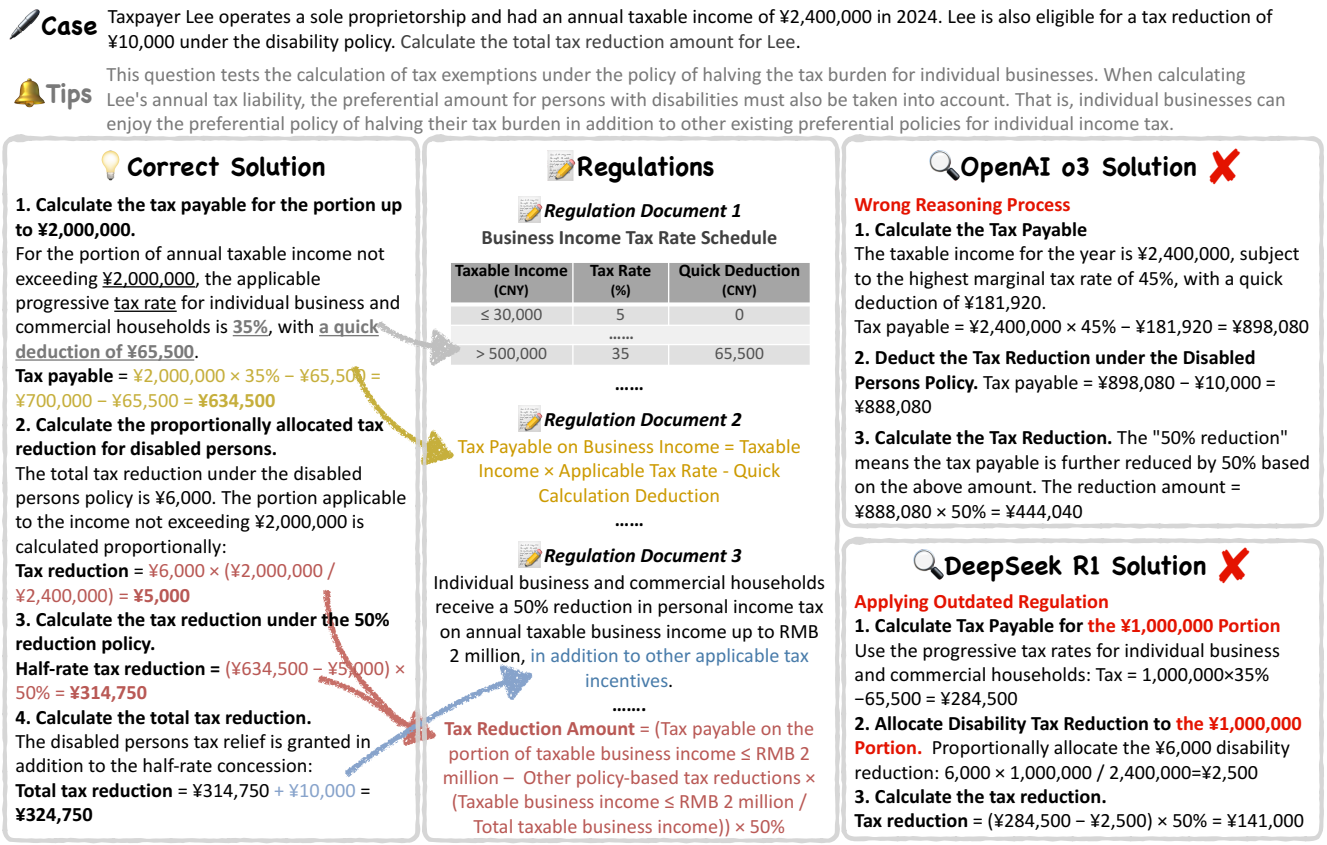

Figure 1: Example from TaxReasoning benchmark: question case, gold-standard solution, and tax regulations. Arrows denote regulatory knowledge used in the calculation. OpenAI o3 lacks relevant tax knowledge, while DeepSeek-R1 applies outdated regulatory information.

several retrieval-augmented approaches such as Standard retrieval (Brown et al. 2020), Summary-based retrieval (Gao et al. 2023; Xu, Shi, and Choi 2024), and multi-turn strategies like Interact and InlineSearch (Gao et al. 2023).

Our findings show that most leading models, including OpenAI o3, Gemini 2.5 Pro, and Claude Opus 4, achieve less than 30% accuracy using parametric knowledge alone (Pan et al. 2023). Even when equipped with knowledge augmentation techniques, their performance remains well below that of human experts. These results underscore the difficulty of TaxReasoning and highlight the limitations of LLMs in reasoning over evolving, domain-specific knowledge.

Further analysis reveals two critical bottlenecks: (1) the reasoning capacity of LLMs is vital, as stronger reasoning models benefit more from retrieval; and (2) even state-of-the-art models are frequently hindered by inadequate summarization and inefficient retrieval strategies, limiting their ability to locate and apply relevant tax information. Motivated by these insights, we propose to restructure tax law documents into a structured knowledge graph (KG) (Pan et al. 2017), enabling more accurate summarization and fine-grained retrieval of tax regulations during the reasoning process.

Our key contributions are as follows:

1. We introduce TaxReasoning, the first benchmark targeting evolving, knowledge-intensive mathematical reasoning in the domain of tax, explicitly designed to test LLMs' ability to incorporate and reason over newly introduced or updated knowledge.
2. We conduct a thorough evaluation of 13 LLMs under diverse prompting and knowledge augmentation strategies. The results expose significant limitations in current models and point toward essential avenues for improvement.
3. We propose to restructure tax documents into a structured KG, enabling both more accurate summarization and efficient retrieval, offering practical improvements and insights for enhancing LLM performance on dynamic, knowledge-intensive mathematical reasoning.

Related Work

Mathematical reasoning is a cornerstone for the development of general-purpose intelligent systems and has attracted sustained interest from the research community. A broad range of mathematical reasoning benchmarks has been proposed to assess LLMs across different educational

stages, from elementary school to college-level curricula (Koncel-Kedziorski et al. 2016; Wang, Liu, and Shi 2017; Amini et al. 2019; Miao, Liang, and Su 2020; Patel, Bhattamishra, and Goyal 2021; Hendrycks et al. 2021; Lu et al. 2023a). However, these benchmarks primarily focus on general math problems and do not typically involve specialized domain knowledge, limiting their applicability to real-world, domain-specific reasoning scenarios.

To address this limitation, recent work has begun to explore knowledge-intensive reasoning benchmarks. For example, Chen et al. (2023c) propose a theorem-driven QA task targeting scientific reasoning, while SciBench (Wang et al. 2024) and SciEval (Sun et al. 2024) evaluate LLMs' ability to apply domain-specific concepts in science and engineering. In parallel, benchmarks such as FinanceMath (Zhao et al. 2024) and FinanceReasoning (Tang et al. 2025) assess models' performance in financial tasks. These datasets emphasize the importance of domain knowledge in reasoning, but often rely on static or widely known knowledge and formulas that may have been encountered during model pretraining.

In contrast to these existing efforts, our work introduces TaxReasoning, a benchmark specifically designed to test LLMs' reasoning abilities in evolving, regulation-driven domains, with a focus on tax law. Unlike prior benchmarks, TaxReasoning emphasizes the need to apply newly introduced or frequently updated legal knowledge, often in combination with multi-step numerical reasoning and interpretation of conditional rules. This dynamic and legally grounded setting introduces unique challenges, including regulation tracking, clause integration, and symbolic-expressive reasoning, that are not captured in prior datasets.

TaxReasoning Benchmark

In this section, we describe the construction process of the TaxReasoning dataset. We begin by collecting a series of long tax regulation documents, covering a wide range of tax-related rules and policies. Next, we gather a set of tax calculation cases. Based on these cases, expert annotators are instructed to formulate knowledge-intensive questions by referencing the relevant tax regulation documents, and to provide corresponding step-by-step solutions grounded in the applicable legal provisions.

Tax Regulations Collection

To establish a robust knowledge foundation for TaxReasoning, we curate a comprehensive collection of 90 latest official tax regulation documents from authoritative Chinese government sources. These documents span 12 major tax categories within China's legal system. Given the heterogeneous formats and inconsistent structures of these documents, we conducted extensive preprocessing and normalization to ensure consistency and usability. Many regulations contain critical information in tabular form, such as progressive tax brackets, quick deduction formulas, and exemption conditions. However, these tables are often presented with irregular formatting or embedded as images, posing challenges for machine-readable extraction. To address this, we employed ChatGPT to convert non-standard or image-based tables into structured Markdown format. All converted outputs were manually reviewed and validated by expert annotators to ensure fidelity to the original legal content and regulatory semantics. The resulting curated and structured corpus of tax regulations forms the authoritative knowledge base for generating realistic, knowledge-intensive questions in the TaxReasoning benchmark.

TaxReasoning Question Annotation

For each tax clause, we retrieve relevant calculation scenarios from university exams and instruct annotators to create corresponding math reasoning questions. When possible, questions are designed to yield numerical answers, aligning with the benchmark's focus on quantitative reasoning grounded in regulation.

Annotators follow a strict set of guidelines to ensure both quality and originality of the questions: (1) Surface-level paraphrasing is not sufficient. Annotators must also modify all associated numerical values to produce a genuinely novel variant of the question. (2) To mitigate risks of data contamination, annotators are instructed to perform an Internet search for each composed question. If a similar problem appears on the first page of search results, the question must be revised or discarded. (3) Recognizing that many tax and financial reasoning problems involve structured tabular data, which poses additional comprehension challenges for LLMs, we encourage and reward the inclusion of relevant, realistic tables that represent rule conditions, deduction rates, or progressive tax brackets. Following this rigorous process, we curated a total of 1000 tax reasoning questions.

Identifying Question-Relevant Regulations After drafting each question, annotators are required to identify the specific regulatory clauses that are essential for solving it. They search for the specific terms, formulas, or rules within our curated regulatory corpus and verify both their presence and contextual relevance. If a referenced term is not found or is ambiguously defined in the original corpus, the question is discarded to maintain benchmark integrity.

To encourage corpus refinement and high-quality annotations, annotators receive additional incentives for contributing clarifications or highlighting inconsistencies in the underlying regulations. After quality control and filtering, we retain a final set of 940 validated questions, each aligned with concrete regulatory sources.

Solution Annotation Each question is paired with a detailed expert-annotated solution. These solutions consist of natural language explanations that outline the step-by-step reasoning process, accompanied by mathematical equations that formalize the underlying computational logic.

Data Quality Validation

To ensure the accuracy and reliability of the TaxReasoning benchmark, we implement a rigorous multi-stage validation protocol. Each annotated example undergoes secondary review by an independent expert, who verifies the following three dimensions: (1) Question Quality: The question must

Knowledge Corpus	
Documents	90
Avg. document length	3,126
With formula/rule	91.1%

TaxReasoning Dataset	
Avg. question length	263.3
Avg. solution length	232.1
Clauses per example	2.8
Dev. set size	187
Test set size	753

Table 1: Key statistics of the tax regulation corpus (left) and the TAXREASONING benchmark (right), highlighting the size and structure of both components.

be clearly formulated, grammatically correct, and meaningful in the context of real-world tax reasoning. (2) Knowledge Accuracy: All referenced tax regulations must be correctly identified, contextually relevant, and sufficiently complete to enable problem-solving. (3) Solution Validity: The annotated solution must be logically consistent, mathematically correct, and presented in a clear and interpretable manner.

If any of these criteria are not met, validators are required to either revise the example or flag it for re-annotation. This double-blind validation process ensures that all benchmark examples are coherent, legally grounded, and solvable by both humans and machines. To further assess annotation quality, we conduct a human evaluation over 200 randomly sampled examples. As shown in Table 3 in the Appendix, TaxReasoning has a high annotation quality.

Data Statistics

Table 1 summarizes key statistics of the TaxReasoning dataset. Following previous work (Zhao et al. 2024), we randomly partition the dataset into two subsets: development and test. The development set contains 187 examples for model development and validation. The test set contains the remaining 753 examples for standard evaluation. To mitigate risks of data contamination, gold-standard answers for the test set are withheld from public release. Instead, we provide an online evaluation platform featuring a submission interface and public leaderboard. This ensures standardized evaluation while supporting broad community participation.

Evaluated Systems

This section presents a detailed overview of the evaluated human and large language model (LLM) performance on the TaxReasoning benchmark, including the experimental settings, model selection, and evaluation methodology.

Human-Level Performance Evaluation

To establish a meaningful performance reference for LLMs, we conducted a human evaluation with certified domain experts. A sample of 50 questions was randomly selected from our benchmark and independently completed by two Certified Public Accountants (CPAs) under two distinct settings: (1) **Closed-book setting.** Participants were prohibited from accessing external resources (e.g., internet, textbooks) and were required to complete the task within three hours. Under these conditions, the two CPAs achieved accuracies of 58% and 64%, respectively (average: 61%). (2) **Open-book setting.** Two additional experts were allowed to consult any external materials while solving the same set of questions. This setting simulates how access to authoritative tax documents can support real-world decision-making. Performance improved substantially, with scores of 78% and 86% (average: 82%).

These results underscore both the inherent difficulty of the benchmark and the significant role of external knowledge in improving performance. Full human evaluation instructions and protocols are detailed in the Appendix.

Large Language Models

We evaluate 13 LLMs spanning both reasoning-specialized and general-purpose families, enabling a comprehensive analysis across diverse architectural capabilities.

Reasoning-Augmented Models. The following models are optimized specifically to enhance reasoning, complex problem solving, or agentic behavior: `OpenAI o1` (Jaech et al. 2024a), `OpenAI o3` (OpenAI 2024), `OpenAI o4-mini` (OpenAI 2024), `DeepSeek-R1` (Guo et al. 2025), `Gemini 2.5 Pro` (Comanici et al. 2025), `Claude Opus 4` (anthropic 2024), `QwQ-32B` (Team 2024d) and `Qwen-3-32B` (Yang et al. 2025).

General-Purpose Models. We include several widely used general-purpose LLMs as baselines: `Gemini 2.0 Flash` (Team 2024a), `GPT-4o` (Hurst et al. 2024), `Claude 3.5 Sonnet` (Team 2024b), `DeepSeek-V3` (Liu et al. 2024a), and `Qwen2.5-72B` (Team 2024c).

Evaluation Methods

Vanilla Method. Following (Chen et al. 2023; Zhao et al. 2024), we apply two standard prompting strategies that rely solely on the LLMs' internal knowledge: (1) **Chain-of-Thought (CoT)** (Wei et al. 2022): Prompts the model to generate intermediate reasoning steps before producing a final answer. (2) **Program-of-Thought (PoT)** (Chen et al. 2022): Encourages the model to write executable code (e.g., Python) to perform structured computations in reasoning.

Knowledge-Augmented Methods We also evaluate several retrieval-based augmentation strategies (Brown et al. 2020; Gao et al. 2023) that enable LLMs to incorporate external knowledge. **Single-turn Retrieval:** (1) **Standard**: The model receives the top-5 most relevant tax documents retrieved via a dense retriever and is instructed to use them for solving the task. (2) **Summary**: To mitigate long-context forgetting (Liu et al. 2024b), we summarize the retrieved documents using `GPT-4o` and filter them for relevance using a classifier. Irrelevant passages excluded; concise summaries retained only. **Multi-turn Retrieval:** To overcome limitations of one-time retrieval, i.e., the difficulty of obtaining all the necessary information for complex reasoning tasks through a single query, we adopt two iterative retrieval strategies based on (Gao et al. 2023): (1) **Interact**: Enhances the Summary method by enabling interactive document inspection. During generation, the model starts with summaries of the top-10 documents and then iteratively execute one of the following three actions: (a) `Check: Document [i]` to reveal a full content based on the reasoning; (b) `Output:` to generate reasoning based on the

documents; (c) `End.` to terminate the generation. (2) **InlineSearch**: Allows LLMs to dynamically trigger "Search: query" actions during generation to retrieve knowledge on demand, interleaving retrieval with reasoning. The model can execute one of three actions: (1) `Search:{query}` to retrieve the currently needed information through generated query; (2) `Output:` to generate the reasoning steps and the query for retrieving; and (3) `End.` to terminate the generation. This mimics agent-style interactions with external tools, akin to LLM agents (Qu et al. 2025).

Oracle Setting: We further include an **Oracle** setting, where the model is provided with human-annotated gold documents relevant to the question. This simulates the upper-bound performance of standard retrieval-augmented methods. Implementation details are provided in the Appendix.

Answer Extraction and Accuracy Evaluation

Following Zhao et al., we apply distinct mechanisms for extracting final answers: For PoT, we directly execute the model-generated Python code to obtain the output. For other methods, we use GPT-4o-mini to extract the final numerical answer from the generated reasoning trace. Evaluation is based on exact numerical matching, with a tolerance of ±0.2% to account for minor rounding discrepancies. This ensures both mathematical precision and practical leniency in real-world financial contexts.

Experiments and Analysis

Results under the Vanilla Method

We first examine the performance of large language models under the vanilla setting, using only internal knowledge without any retrieval augmentation. Results under COT and POT prompting are reported in the leftmost columns of Table 2. Several key findings emerge:

TaxReasoning presents a substantial challenge for current LLMs and serves as an effective benchmark for evaluating models' ability to incorporate external knowledge in real-world mathematical reasoning tasks. Most models fail to perform well using internal knowledge alone. Even the best-performing model, DeepSeek-R1, achieves only 42.16% accuracy—substantially below the 61% achieved by human experts under a closed-book setting. Notably, advanced models such as OpenAI o1 and o3 perform below 20%, indicating that much of the specialized tax-related knowledge required by this benchmark is missing from their pretraining corpora.

Procedural reasoning (PoT) is generally weaker than textual reasoning (CoT) across most LLMs in complex, knowledge-intensive mathematical tasks, with the exception of OpenAI's o-series models. For the majority of models, PoT yields lower accuracy than CoT, revealing significant limitations in procedural reasoning capabilities. This observation diverges from prior work (Zhao et al. 2024; Tang et al. 2025), which found PoT to be superior in domain-specific reasoning. We attribute this discrepancy to the greater complexity of TaxReasoning tasks: unlike previous benchmarks that typically required short and self-contained calculations, our tasks often involve generating and executing tens of lines of code encompassing multiple variables, logical conditions, and cross-dependent constraints. Under such complexity, most models struggle to maintain consistent and correct reasoning across execution steps. An exception is observed with OpenAI's o-series models (especially o3 and o4-mini), which perform substantially better under PoT than CoT. This suggests stronger procedural reasoning abilities, likely due to architectural optimizations and extensive exposure to structured programming tasks during their training.

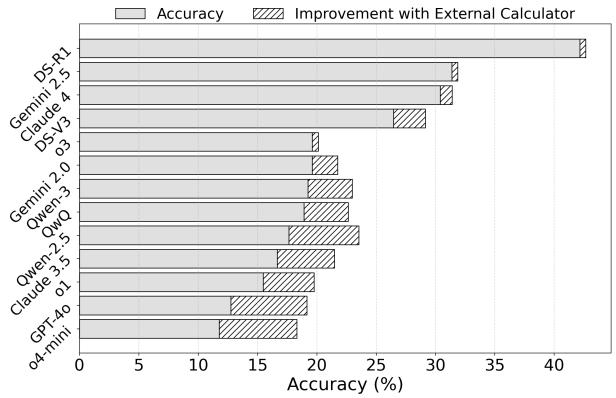

Figure 2: Calibrated performance of CoT with external calculator support for mathematical reasoning, showing LLMs' challenges in complex numerical problem-solving.

Error Analysis To better understand the limitations of current LLMs on our benchmark, we conducted an in-depth error analysis and case study review. Specifically, we examined 50 failed examples produced by the best-performing model, DeepSeek-R1, under the CoT prompting strategy. Our goal was to categorize the types of errors that persist even in top-tier models. Note that some instances exhibit multiple error types. A detailed breakdown is provided in the Appendix. We identified four primary categories of failure: (1) **Misidentification or Misapplication of Required Knowledge** (29/50): In over half the cases, the model failed to correctly identify or apply the relevant tax law concepts. This often led to flawed reasoning paths, incomplete deduction chains, or the use of incorrect assumptions. These errors reveal a gap in the model's ability to align legal knowledge with problem context. (2) **Hallucinated or Outdated Tax Law Knowledge** (14/50): In these cases, the model confidently produced structured solutions but relied on fabricated or obsolete legal knowledge. This highlights issues with factual grounding, especially in dynamic domains like tax law, where regulations are frequently updated. (3) **Incomplete Understanding of the Question** (5/50): Some failures were due to shallow comprehension of the problem statement. In these instances, the model missed implicit cues or overlooked necessary constraints, resulting in incomplete or misaligned calculation procedures. (4) **Mathematical Computation Errors** (2/50): Although rare in DeepSeek-R1, a few cases showed numerical inaccuracies despite correct reason-

Model \ Method	Vanilla		Single-Turn RAG		Multi-Turn RAG		Oracle
	CoT	PoT	Standard	Summary	Interact	InlineSearch	
Reasoning-Augmented Models							
OpenAI o1	15.48	16.67	35.24	29.41	40.16	44.56	48.14
OpenAI o3	19.61	21.57	38.24	36.27	41.97	45.26	51.26
OpenAI o4-mini	11.76	16.00	27.45	21.57	24.17	22.51	31.37
DeepSeek-R1	**42.16**	**38.00**	46.63	43.39	53.75	59.48	62.50
Gemini 2.5 Pro	31.37	29.41	**57.22**	**56.33**	**58.26**	**60.25**	**64.12**
Claude Opus 4	30.39	25.49	44.92	40.64	52.29	55.96	61.16
QwQ-32B	18.91	16.97	26.61	25.14	27.05	28.51	32.16
Qwen-3-32B	19.24	19.25	27.61	26.16	27.16	28.91	32.51
General-Purpose Models							
GPT-4o	12.75	7.84	11.67	13.82	13.01	12.21	14.76
DeepSeek-V3	26.47	19.80	34.31	35.29	36.16	37.03	38.35
Gemini 2.0 Flash	19.61	10.78	18.63	18.63	19.51	19.04	24.51
Claude 3.5 Sonnet	16.67	12.75	20.59	19.61	19.58	19.02	21.57
Qwen 2.5-72B	17.65	10.11	21.51	20.14	17.14	17.56	20.67
Human (Closed-book)	61.00		–	–	–	–	–
Human (Open-book)	–		–	–	–	–	82.00

Table 2: Overall performance of 13 LLMs on the TaxReasoning benchmark. Bold numbers indicate the best performance in each column, while underlined values denote the second-best. Shading highlights the top-2 results in each row, with darker shades indicating higher scores.

ing structures. These errors included arithmetic mistakes, rounding issues, or incorrect handling of expressions.

We observe that several other models still suffer from frequent calculation errors. To distinguish between reasoning failures and pure computation errors, we follow Zhao et al. and implement an external calculator pipeline. Specifically, we use GPT-4o to extract the final mathematical expression from each model's CoT output, and then evaluate the expression using an external symbolic execution engine. Figure 2 compares the original model accuracy with post-correction performance using this calculator pipeline. The results reveal that many open-source LLMs produce valid reasoning traces but fail at precise computation—underscoring a key limitation in symbolic math capabilities.

Knowledge Augmentation Experiments

We further investigate whether LLMs can improve their performance on complex, domain-specific mathematical reasoning tasks by incorporating evolving knowledge. Table 2 summarizes the results under various retrieval-augmented methods. Several key observations emerge:

The models with strong reasoning capabilities benefit substantially from external knowledge. Reasoning-enhanced models exhibit significant performance gains when external information is introduced. For instance, Gemini 2.5 Pro achieves a 25.85% improvement under the Standard retrieval setting compared to the CoT baseline (57.22% vs. 31.37%). In contrast, general-purpose models show limited gains. GPT-4o, for example, achieves only a 2.01% improvement under the Oracle setting (14.76%) compared to its CoT baseline (12.75%). These findings suggest that TaxReasoning tasks often require multi-step application of tax knowledge, and only models with robust reasoning abilities can identify when and how to apply external knowledge effectively.

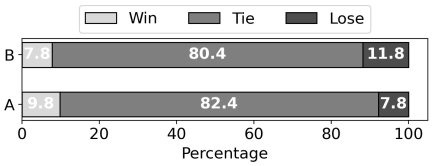

Figure 3: Win–Tie–Lose diagrams comparing (A) Standard vs. Summary and (B) Interact vs. InlineSearch. "Win" means the first method is correct and the second is not; "Lose" means the reverse; "Tie" means both are correct or both incorrect.

Multi-turn retrieval methods yield notable improvements, but only for reasoning-capable models. Advanced multi-round retrieval strategies (e.g., Interact, InlineSearch) are particularly effective for models with strong reasoning skills, such as DeepSeek-R1, Gemini 2.5 Pro, Claude Opus 4, and OpenAI's o-series models (o1 and o3). These models are capable of decomposing complex problems into sub-steps and issuing focused queries to retrieve supporting knowledge. Conversely, models with weaker reasoning ability perform worse under multi-round retrieval than single-turn retrieval. These models struggle to decompose questions accurately and often fail to issue relevant queries, resulting in misleading retrieved content.

Standard v.s. Summary and Interact v.s. InlineSearch To compare the effectiveness of different retrieval strategies, we analyze performance deltas using DeepSeek-R1 as a case study. Figure 3 presents win–tie–lose comparisons for Standard vs. Summary (A in the Figure) and Interact vs. InlineSearch (B in the Figure).

Standard achieves a higher win ratio than Summary.

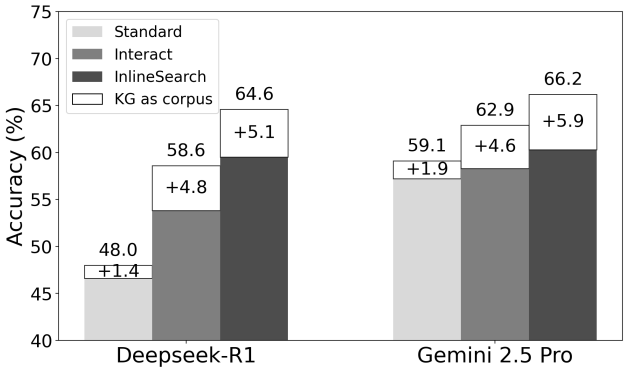

Figure 4: Performance comparison of DeepSeek-R1 and Gemini 2.5 Pro under three retrieval-augmented methods, using either raw tax documents or a structured tax knowledge graph as the retrieval corpus.

This is largely because summaries of tax law documents tend to omit critical details such as formulas, exception clauses, and legal conditions. However, STANDARD also loses in some cases due to noise introduced by lengthy or overly complex documents containing multiple legal topics. **These observations point to the importance of developing effective summarization techniques that preserve salient information while reducing contextual clutter.**

INLINESEARCH achieves a higher win ratio than INTERACT. This is because INLINESEARCH allows the model to generate context-sensitive queries on-the-fly, leading to the retrieval of more targeted and relevant legal content. In contrast, while INTERACT enables interactive document inspection, it relies on fixed input documents and is limited by the quality and coverage of initial retrievals. **These findings underscore the critical role of retrieval strategy in enhancing legal reasoning performance.**

Structural Knowledge Improves Reasoning

Building on the above findings, we observe that the reasoning ability of LLMs is crucial for solving tasks in TaxReasoning, and state-of-the-art reasoning models often underperform due to limitations beyond reasoning itself. In particular, their performance is frequently constrained by inadequate knowledge summarization and suboptimal retrieval strategies, which hinder the effective use of evolving tax knowledge during problem solving. Inspired by GraphRAG (Han et al. 2025), we propose to restructure tax documents into a structured knowledge graph (KG), enabling both more accurate summarization and efficient retrieval.

Tax regulations are inherently hierarchical and modular, typically organized into titles, clauses, subclauses, and condition-based rules. This structure naturally lends itself to a graph representation, where each node corresponds to a defined tax concept or regulation fragment, and edges capture logical dependencies, hierarchical relationships, and citation links. By leveraging this structure, we can more effectively retrieve relevant and granular tax knowledge, avoiding the noise and redundancy often found in full-document retrieval.

Graph Construction and Integration. We follow the methodology proposed by GraphRAG (Edge et al. 2024) to construct a domain-specific knowledge graph from tax law documents. Nodes are extracted from headings, paragraphs, formulas, and legal conditions, and linked via intra- and inter-documental dependencies (e.g., "refers to," "extends," "applies when"). The resulting KG serves as an indexable, structured corpus for downstream retrieval.

We use the constructed KG as the retrieval corpus in conjunction with three retrieval-augmented reasoning strategies: STANDARD, INTERACT, and INLINESEARCH. The SUMMARY method is excluded in this setting, as the KG inherently contains concise, human-interpretable summaries embedded at the node level.

Experimental Results. Figure 4 presents the performance of two representative reasoning-strong models, DeepSeek-R1 and Gemini 2.5 Pro, nder three retrieval-augmented methods. Results indicate that KG-based retrieval consistently improves performance across all strategies. However, the degree of improvement varies: (1) **STANDARD** retrieval sees only modest gains. Since retrieval is based solely on the question embedding, it often misses concise but crucial knowledge nodes in the KG that lack rich contextual information. To partially address this, we increase the top-k value to improve recall, but challenges remain. (2) **INTERACT** benefits more substantially from KG integration. With an expanded retrieval set and iterative access to full node details, the model can selectively explore relevant tax rules and avoid legal noise. Raising the retrieval k enhances coverage without overwhelming the context window. (3) **INLINESEARCH** achieves the most notable improvement. This strategy allows the model to dynamically query specific parts of the KG during its reasoning process, generating context-aware queries that retrieve focused legal content. This synergy between procedural reasoning and structured knowledge leads to both improved accuracy and efficiency. Overall, structured knowledge like KGs proves most effective when combined with dynamic, interactive retrieval, reducing token overhead and enhancing reasoning precision. More details are provided in the Appendix.

Conclusion

We introduce TaxReasoning, the first benchmark for evaluating LLMs' evolving knowledge-intensive mathematical reasoning in tax regulations. Experiments with 13 LLMs under diverse prompting and retrieval strategies confirm their underperformance compared to humans, attributed to weak reasoning capacity and ineffective knowledge retrieval. Restructuring tax regulations into a structured KG effectively improves performance. TaxReasoning uncovers LLMs' limitations in real-world, domain-specific reasoning and paves the way for future advances. In future work, we plan to extend it to multilingual and cross-jurisdiction evaluations.

On applications, fintech companies can leverage TaxReasoning to enhance the accuracy of tax-related features in their products, such as intelligent tax declaration software and financial management platforms.

Acknowledgments

This work is partially supported by National Nature Science Foundation of China under No. U21A20488. We thank the Big Data Computing Center of Southeast University for providing the facility support on the numerical calculations in this paper.

References

Amini, A.; Gabriel, S.; Lin, S.; Koncel-Kedziorski, R.; Choi, Y.; and Hajishirzi, H. 2019. MathQA: Towards Interpretable Math Word Problem Solving with Operation-Based Formalisms. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2357–2367. Minneapolis, Minnesota: Association for Computational Linguistics.

anthropic. 2024. Introducing Claude 4. https://www.anthropic.com/news/claude-4. Accessed: 2025-07-31.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Chen, W.; Ma, X.; Wang, X.; and Cohen, W. W. 2022. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *arXiv preprint arXiv:2211.12588*.

Chen, W.; Yin, M.; Ku, M.; Lu, P.; Wan, Y.; Ma, X.; Xu, J.; Wang, X.; and Xia, T. 2023. TheoremQA: A Theorem-driven Question Answering Dataset. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 7889–7901. Singapore: Association for Computational Linguistics.

Comanici, G.; Bieber, E.; Schaekermann, M.; Pasupat, I.; Sachdeva, N.; Dhillon, I.; Blistein, M.; Ram, O.; Zhang, D.; Rosen, E.; et al. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

DeepSeek-AI; Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; et al. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *CoRR*, abs/2501.12948.

Edge, D.; Trinh, H.; Cheng, N.; Bradley, J.; Chao, A.; Mody, A.; Truitt, S.; Metropolitansky, D.; Ness, R. O.; and Larson, J. 2024. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*.

Gao, T.; Yen, H.; Yu, J.; and Chen, D. 2023. Enabling Large Language Models to Generate Text with Citations. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 6465–6488. Singapore: Association for Computational Linguistics.

Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; Wang, P.; Bi, X.; et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Han, H.; Shomer, H.; Wang, Y.; Lei, Y.; Guo, K.; Hua, Z.; Long, B.; Liu, H.; and Tang, J. 2025. Rag vs. graphrag: A systematic evaluation and key insights. *arXiv preprint arXiv:2502.11371*.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring Mathematical Problem Solving With the MATH Dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Hurst, A.; Lerer, A.; Goucher, A. P.; Perelman, A.; Ramesh, A.; Clark, A.; Ostrow, A.; Welihinda, A.; Hayes, A.; Radford, A.; et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Jaech, A.; Kalai, A.; Lerer, A.; Richardson, A.; El-Kishky, A.; Low, A.; Helyar, A.; Madry, A.; Beutel, A.; Carney, A.; et al. 2024a. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Jaech, A.; Kalai, A.; Lerer, A.; Richardson, A.; El-Kishky, A.; Low, A.; Helyar, A.; Madry, A.; Beutel, A.; et al. 2024b. OpenAI o1 System Card. *CoRR*, abs/2412.16720.

Koncel-Kedziorski, R.; Roy, S.; Amini, A.; Kushman, N.; and Hajishirzi, H. 2016. MAWPS: A Math Word Problem Repository. In Knight, K.; Nenkova, A.; and Rambow, O., eds., *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1152–1157. San Diego, California: Association for Computational Linguistics.

Liu, A.; Feng, B.; Xue, B.; Wang, B.; Wu, B.; Lu, C.; Zhao, C.; Deng, C.; Zhang, C.; Ruan, C.; et al. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Liu, N. F.; Lin, K.; Hewitt, J.; Paranjape, A.; Bevilacqua, M.; Petroni, F.; and Liang, P. 2024b. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 11: 157–173.

Lu, P.; Qiu, L.; Chang, K.-W.; Wu, Y. N.; Zhu, S.-C.; Rajpurohit, T.; Clark, P.; and Kalyan, A. 2023a. Dynamic Prompt Learning via Policy Gradient for Semi-structured Mathematical Reasoning. In *The Eleventh International Conference on Learning Representations*.

Lu, P.; Qiu, L.; Yu, W.; Welleck, S.; and Chang, K.-W. 2023b. A Survey of Deep Learning for Mathematical Reasoning. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 14605–14631. Toronto, Canada: Association for Computational Linguistics.

Miao, S.-y.; Liang, C.-C.; and Su, K.-Y. 2020. A Diverse Corpus for Evaluating and Developing English Math Word Problem Solvers. In Jurafsky, D.; Chai, J.; Schluter, N.; and Tetreault, J., eds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 975–984. Online: Association for Computational Linguistics.

OpenAI. 2024. Introducing OpenAI o3 and o4-mini. https://openai.com/index/introducing-o3-and-o4-mini/. Accessed: 2025-07-31.

Pan, J.; Vetere, G.; Gomez-Perez, J.; and Wu, H., eds. 2017. *Exploiting Linked Data and Knowledge Graphs for Large Organisations*. Springer. ISBN 978-3-319-45652-2.

Pan, J. Z.; Razniewski, S.; Kalo, J.-C.; Singhania, S.; Chen, J.; Dietze, S.; Jabeen, H.; Omeliyanenko, J.; Zhang, W.; Lissandrini, M.; Biswas, R.; de Melo, G.; Bonifati, A.; Vakaj, E.; Dragoni, M.; and Graux, D. 2023. Large Language Models and Knowledge Graphs: Opportunities and Challenges. *Transactions on Graph Data and Knowledge*, 1–38.

Patel, A.; Bhattamishra, S.; and Goyal, N. 2021. Are NLP Models really able to Solve Simple Math Word Problems? In Toutanova, K.; Rumshisky, A.; Zettlemoyer, L.; Hakkani-Tur, D.; Beltagy, I.; Bethard, S.; Cotterell, R.; Chakraborty, T.; and Zhou, Y., eds., *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2080–2094. Online: Association for Computational Linguistics.

Qu, C.; Dai, S.; Wei, X.; Cai, H.; Wang, S.; Yin, D.; Xu, J.; and Wen, J.-R. 2025. Tool learning with large language models: A survey. *Frontiers of Computer Science*, 19(8): 198343.

Sun, L.; Han, Y.; Zhao, Z.; Ma, D.; Shen, Z.; Chen, B.; Chen, L.; and Yu, K. 2024. Scieval: A multi-level large language model evaluation benchmark for scientific research. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 19053–19061.

Tang, Z.; E, H.; Ma, Z.; He, H.; Liu, J.; Yang, Z.; Rong, Z.; Li, R.; Ji, K.; Huang, Q.; Hu, X.; Liu, Y.; and Zheng, Q. 2025. FinanceReasoning: Benchmarking Financial Numerical Reasoning More Credible, Comprehensive and Challenging. In Che, W.; Nabende, J.; Shutova, E.; and Pilehvar, M. T., eds., *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 15721–15749. Vienna, Austria: Association for Computational Linguistics. ISBN 979-8-89176-251-0.

Team, G. 2024a. Gemini 2.0 Flash. https://deepmind.google/models/gemini/flash/. Accessed: 2025-07-31.

Team, Q. 2024b. Claude 3.5 Sonnet. https://www.anthropic.com/news/claude-3-5-sonnet. Accessed: 2025-07-31.

Team, Q. 2024c. Qwen2.5-Max: Exploring the Intelligence of Large-scale MoE Model. https://qwenlm.github.io/blog/qwen2.5-max/. Accessed: 2025-07-31.

Team, Q. 2024d. QwQ-32B: Embracing the Power of Reinforcement Learning. https://qwenlm.github.io/blog/qwq-32b/. Accessed: 2025-07-31.

Wang, X.; Hu, Z.; Lu, P.; Zhu, Y.; Zhang, J.; Subramaniam, S.; Loomba, A. R.; Zhang, S.; Sun, Y.; and Wang, W. 2024. SciBench: Evaluating College-Level Scientific Problem-Solving Abilities of Large Language Models. In *International Conference on Machine Learning*, 50622–50649. PMLR.

Wang, Y.; Liu, X.; and Shi, S. 2017. Deep Neural Solver for Math Word Problems. In Palmer, M.; Hwa, R.; and Riedel, S., eds., *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 845–854. Copenhagen, Denmark: Association for Computational Linguistics.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Xu, F.; Shi, W.; and Choi, E. 2024. RECOMP: Improving Retrieval-Augmented LMs with Context Compression and Selective Augmentation. In *The Twelfth International Conference on Learning Representations*.

Yang, A.; Li, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; et al. 2025. Qwen3 Technical Report. *CoRR*, abs/2505.09388.

Zhao, Y.; Liu, H.; Long, Y.; Zhang, R.; Zhao, C.; and Cohan, A. 2024. FinanceMATH: Knowledge-Intensive Math Reasoning in Finance Domains. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 12841–12858.

Zheng, D.; Lapata, M.; and Pan, J. Z. 2024. Archer: A Human-Labeled Text-to-SQL Dataset with Arithmetic, Commonsense and Hypothetical Reasoning. In *In Proc. of the 18th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2024)*.