

# Evaluating Legal Reasoning Traces with Legal Issue Tree Rubrics

Jinu Lee<sup>1</sup>, Kyoung-Woon On<sup>2</sup>, Simeng Han<sup>3</sup>, Arman Cohan<sup>4</sup>, Julia Hockenmaier<sup>1</sup>

<sup>1</sup>University of Illinois Urbana-Champaign <sup>2</sup>LBOX

<sup>3</sup>Stanford <sup>4</sup>Yale

{jinulee2, juliahmr}@illinois.edu kyoungwoon.on@lbox.kr {shan6}@stanford.edu

## Abstract

Evaluating the quality of LLM-generated reasoning traces in expert domains (*e.g.*, law) is essential for ensuring credibility and explainability, yet remains challenging due to the inherent complexity of such reasoning tasks. We introduce LEGIT (LEGal Issue Trees), a novel large-scale (24K instances) expert-level legal reasoning dataset with an emphasis on reasoning trace evaluation. We convert court judgments into hierarchical trees of opposing parties' arguments and the court's conclusions, which serve as rubrics for evaluating the issue coverage and correctness of the reasoning traces. We verify the reliability of these rubrics via human expert annotations and comparison with coarse, less informative rubrics. Using the LEGIT dataset, we show that (1) LLMs' legal reasoning ability is seriously affected by both legal issue coverage and correctness, and that (2) retrieval-augmented generation (RAG) and RL with rubrics bring complementary benefits for legal reasoning abilities, where RAG improves overall reasoning capability, whereas RL improves correctness albeit with reduced coverage.<sup>1</sup>

## 1 Introduction

Large language models (LLMs) can solve complex reasoning problems by generating an intermediate reasoning trace ("*Chain-of-thoughts*") before outputting the final answer (Wei et al., 2022; Guha et al., 2025). Evaluating the quality of these reasoning traces is crucial for understanding as well as improving the reasoning ability of LLMs, for example, through selecting the best responses or reinforcement learning (Lanham et al., 2023; Yao et al., 2023; Han et al., 2024; Lee and Hockenmaier, 2025). However, evaluating reasoning traces for expert-level tasks, *e.g.*, law and medicine, requires substantial domain expertise that even advanced

LLMs do not yet fully possess (Mishra et al., 2025; Wang et al., 2025). Moreover, automatic evaluators often lack an understanding of domain-specific and nuanced *desirability* shaping human judgments (Kim et al., 2025b; Starace et al., 2025). For instance, when asked to predict the outcome of a court judgment, a good reasoning trace should not only avoid logical and factual errors but also exhaustively identify and analyze legal issues (Yu et al., 2025). This underscores the need for a more sophisticated evaluation strategy for expert-level, domain-specific reasoning traces. We choose law as our testbed, as its textual and logical nature enables objective evaluation of reasoning traces compared to domains that rely on probabilistic and data-driven reasoning, *e.g.*, medicine and finance.

In this paper, we introduce **LEGIT (LEGal Issue Trees, Figure 1)**, a large-scale Korean **legal judgment prediction (LJP)** benchmark with an additional emphasis on *reasoning trace evaluation*. In contrast to other LJP datasets where the task is simply to predict the court's final order given basic facts (Cui et al., 2022), LEGIT includes additional **rubrics for reasoning traces** that are extracted from court judgments. These rubrics can be used to assess whether the reasoning traces fully cover the issues addressed during the trial (**issue coverage**) and if the reasoning about those issues is correct (**issue correctness**). We argue that they enable reliable LLM-as-a-judge evaluations of legal reasoning traces that are comparable to licensed lawyers and that these rubrics facilitate principled and detailed analyses of the legal reasoning ability of LLMs.

Using LEGIT, we evaluate the legal reasoning ability of different LLMs, and discover that even state-of-the-art LLMs are not fully capable of complex legal judgment prediction. We identify and analyze two major error types: either failing to identify relevant issues (*decomposition error*) or failing to reason correctly about facts (*deduction error*).

Finally, we explore how retrieval augmented gen-

<sup>1</sup>We fully release the [code](#) and [data](#).

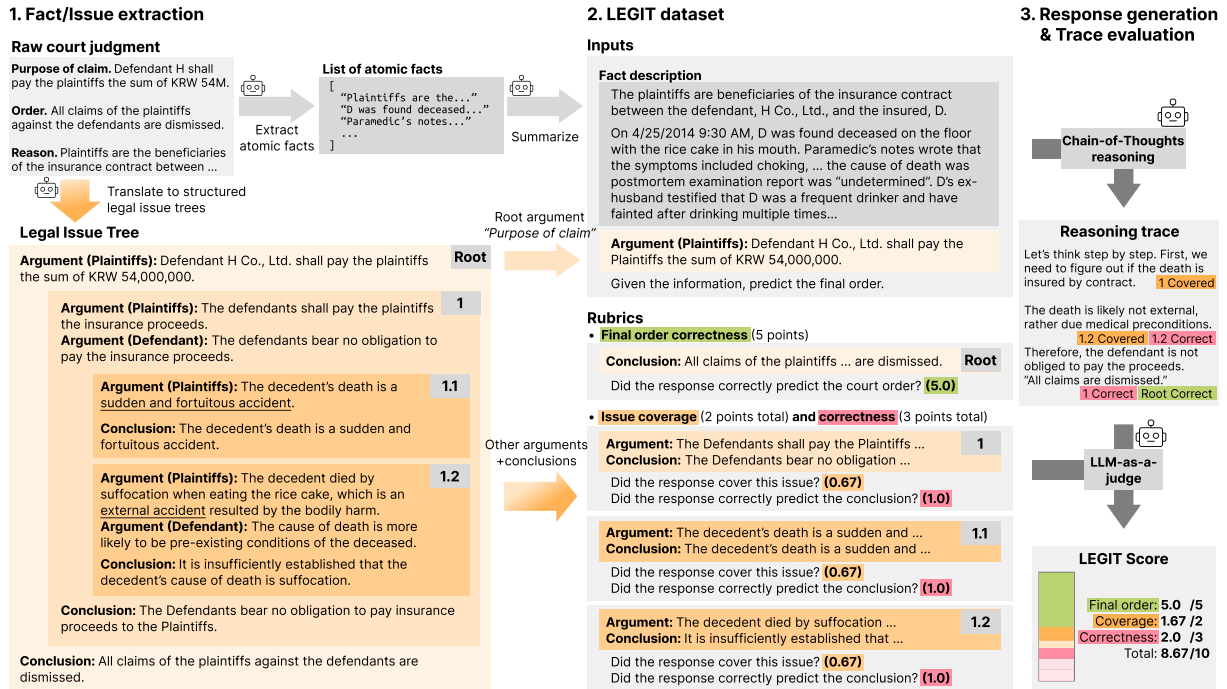


Figure 1: Overview of the LEGIT dataset and task. Facts and issue trees are extracted from real-world court judgments to serve as inputs and rubrics for the LEGIT task. See Appendix A for another example.

eration (RAG) and reinforcement learning (RL) affect legal reasoning performance. These two approaches show complementary benefits: RAG improves both issue coverage and correctness, while RL maximizes correctness at the expense of issue coverage.

## 2 Background

### 2.1 Evaluating reasoning traces

While reasoning traces dramatically increase LLM performance on complex tasks (Wei et al., 2022), they might contain factual or logical errors even if the final answer is correct (Lanham et al., 2023). Such errors can be critical in high-stakes fields such as law and medicine, if end-users rely on LLM responses for important decision-making (Mishra et al., 2025). Evaluating reasoning traces can help reduce this risk by filtering out incorrect responses or training the model via reinforcement learning (Lee and Hockenmaier, 2025).

LLMs can be prompted to evaluate reasoning traces by finding errors or assessing the overall quality ("LLM-as-a-judge") (Yao et al., 2023). Despite its simplicity, LLM-as-a-judge achieves strong performance in error detection and best response selection in math and programming (Huang et al., 2024a; Kim et al., 2025c). However, using generic evaluation criteria might not be sufficient

to capture deeper insights into what constitutes a good reasoning trace. Defining desirability is particularly challenging in expert domains, where one must evaluate both precise factual knowledge and deeper domain-specific insights.

To incorporate diverse, fine-grained forms of desirability in reasoning traces, recent works introduced **instance-specific rubrics** (Gunjal et al., 2025; Pathak et al., 2025). For instance, BigGenBench (Kim et al., 2025b) includes handcrafted rubrics tailored for individual problems that evaluate knowledge and problem-solving strategies. While such rubrics allow LLM judges to assess diverse criteria beyond correctness with high human-LLM agreement, the human effort required to define and validate rubrics hinders scaling this approach beyond evaluation-only benchmarks. By contrast, LEGIT automatically extracts high-quality rubrics from court judgments, resulting in scalable and expert-aligned evaluation of reasoning traces.

### 2.2 Legal Judgment Prediction (LJP)

Legal judgment prediction (LJP) is the task of predicting the court's final order given the facts and claims of a case (Cui et al., 2022). It is a representative legal reasoning task that simultaneously assesses an LLM's legal knowledge (statutes, case laws) and reasoning ability (applying laws to given

facts) (Jiang and Yang, 2023; Huang et al., 2024b).

While traditional reasoning tasks like math often use final answer correctness as the sole criterion (Lightman et al., 2023; Gao et al., 2025), issue coverage and correctness are also critical in LJP (Yu et al., 2025). Imagine a case where the beneficiary of an insurance policy (plaintiff) sues the insurance company (defendant) for payment, and the ground-truth order is to dismiss the case (Figure 1). Even if two reasoning traces both predict that the case should be dismissed, if one thoroughly considers the contractual conditions of an insured accident (e.g., whether the accident was external) when the other does not, legal experts will find the former more useful in practice. Unfortunately, existing benchmarks account only for the final order prediction accuracy, and ignore both issue coverage and correctness (Aletras et al., 2016; Hwang et al., 2022; Huang et al., 2024b; Adarsh et al., 2024).

Current LJP benchmarks also fail to cover the full range of legal cases. As LJP was traditionally considered a classification/regression task, many works have focused on either criminal cases with a continuous spectrum of sentencing labels (Hwang et al., 2022; Huang et al., 2024b), or a minimal set of cases with clear binary labels (Aletras et al., 2016; Adarsh et al., 2024). To the best of our knowledge, no LJP dataset has covered arbitrary case types observed in civil and administrative cases, which account for up to 84.1% of total court cases in the US (United State Court, 2024) and 69.4% in Korea (Court of Korea, 2024).

### 3 The LEGIT Dataset

#### 3.1 Legal issue trees

Legal issues are inherently hierarchical (Jiang and Yang, 2023). Returning to our example of beneficiaries suing an insurance company (Figure 1), the *purpose of claim*<sup>2</sup> is that the company should pay the plaintiffs. The reason is that the event of death is insured by contract. To prove that the event is insured, the plaintiff must show that it is a sudden, fortuitous, and external accident. For instance, an event is external if the death is directly caused by an external event (e.g., choking), and not the consequence of pre-existing medical conditions.

This motivates structuring legal issues as a tree, where each node should include the arguments

---

<sup>2</sup>The topmost claim, which only contains the subject-matter (money, property, disposition) and the action (payment, register, revocation) but not any of the underlying reason.

made by the parties and the judge’s conclusion regarding whether to accept or decline the argument. The tree structure reflects how a legal issue can be decomposed into child issues. While higher-level issues are bound to *laws* like "If the accident is sudden, fortuitous, and external, the insurance company bears obligation to pay the beneficiary", lower issues require common sense inference on *facts*, e.g., "If the deceased fainted after drinking multiple times before, it is likely that such pre-existing condition is the cause of death". The root node’s argument is always the purpose of claim, and the conclusion corresponds to the court’s final order.

We can therefore view the legal judgment prediction task as a *backward chaining* (top-down traversal) of the legal issue tree, iterating between two procedures: identifying the relevant child issues from the given issue (*decomposition*), and reasoning about an issue using the base facts and the conclusions of child issues (*deduction*) (Kazemi et al., 2023; Lee and Hwang, 2025). This allows us to evaluate LLMs not just on the correctness of their final order prediction, but also on whether they can properly decompose issues into their children (*issue coverage*) and can reason about them correctly (*issue correctness*).

#### 3.2 Data source and statistics

To construct LEGIT, we sample judgments issued by Korean District courts from the LBOX database (lbox.kr). We only gather judgments where the final order is *deterministic* by law (Appendix B.1), avoiding the cases where the final decisions are at the discretion of judges. For instance, in the Korean Criminal Act, the judge has the discretion to determine the sentence within a certain range (e.g., between 6 months and 2 years in jail). After filtering judgments that contain keywords related to such non-deterministic orders, the final dataset includes 24,406 judgments from diverse areas of civil and administrative law, which are divided into a training (24,106) and test (300) split. See Appendix B.2 for more statistics.

#### 3.3 Fact extraction

Facts serve as the core inputs for LLMs in the LJP task (Cui et al., 2022). While existing legal datasets commonly use the "Facts" section of the judgment without further modification (Hwang et al., 2022; Huang et al., 2024b), this section often neglects *indirect facts* (which imply important facts by Rule of

Thumb) and *supplementary facts* (which (dis)prove the probative power of an evidence) that are essential for a complete investigation of a case. Therefore, instead of only using the "Facts" section, we use an LLM (Gemini-2.0-Flash) to (1) extract unit factual events in simple sentences from the entire judgment (Min et al., 2023) and (2) generate a coherent case description from the list of unit facts, both with 1-shot examples (Appendix H).

### 3.4 Issue structure extraction

We prompt LLMs to generate rich legal issue trees directly from the court judgment, using Gemini-2.0-Flash and 3-shot examples manually curated by the authors. To improve the quality and minimize errors, we run the LLM twice: first, to create the issue tree structure from raw judgments, and second, to refine the results with another prompt to eliminate errors and mistakes that were often observed in the first round (Appendix H).

As cases with more issues tend to involve more facts and relevant statutes, we divide the dataset into three subsets based on the number of issues (easy:  $\leq 25\%$ , medium: 25-75%, hard:  $\geq 75\%$ ). LEGIT’s test split includes 300 examples, with 100 from each difficulty subset, where the paper authors have manually inspected and fixed errors in the dataset.

See Appendix B.1 for manual evaluation results of the automatic data construction pipeline.

### 3.5 Issue-to-rubric conversion

For LLM-as-a-judge evaluation, we convert issues into rubrics. The LLM judge will evaluate each issue individually. Specifically, for each extracted issue, the LLM jointly predicts if that issue was mentioned in the given response (coverage) and the response includes the correct conclusion of the given issue (correctness) with a short Chain-of-thoughts rationale. The scores are calculated by the point scheme described below.

### 3.6 LEGIT score calculation

The LEGIT score has three components: final order correctness, issue coverage, and issue correctness. **Final order correctness** represents whether the LLM-predicted final order is correct, which is the goal of the LJP task. Since LEGIT intentionally uses only deterministic judgments, the score is binary: 5 points if the final order exactly matches the ground truth, and 0 otherwise. **Issue coverage** (max: 2 points) measures whether the response has

identified the legal issues mentioned in the court judgment. If the issue tree has  $N \geq 1$  non-root nodes, we assign  $2/N$  points for each issue that is covered. Finally, for **issue correctness** (max: 3 points), we assign  $3/N$  points for each issue that is covered *and* the predicted conclusion is correct. The maximum total LEGIT score is 10 points.

The score distribution between the three components is based on two core principles. First, final order correctness (5 points) is the most important criterion, as LJP’s fundamental goal is to accurately predict the final order. Second, issue correctness (3/10) is more rewarded than issue coverage (2/10) because predicting the conclusion of the issue requires finding the exact relevant rationales and reasoning about them, but predicting the existence of an issue does not need to be as precise.

### 3.7 Pointers to relevant appendices

We present LEGIT data statistics and manual quality inspection results in Appendix B. Prompts used for data construction are included in Appendix H.

## 4 Reliability of LLM-as-a-judge

In this section, we show that LEGIT rubrics allow reliable evaluation of reasoning traces by measuring inter-rater agreement of human experts and LLMs, and comparing with coarser rubrics.

### 4.1 Inter-rater agreement

#### 4.1.1 Experimental settings

**Reasoning trace generation** To begin with, we sample solutions from 12 frontier LLMs, including OpenAI {o3, GPT-4.1, GPT-4.1-mini} (OpenAI, 2024), Google Gemini {2.5-Pro, 2.5-Flash, 2.0-Flash} (Gemini Team, 2025), Gemma 3 {27B, 12B, 4B} (Gemma Team, 2025), EXAONE {3.5-32B, 3.5-7.8B, 3.0-7.8B} (LG AI Research, 2024)<sup>3</sup>. A few widely used open-weight models, including Qwen, LLaMA, and DeepSeek, were discarded due to their inability to generate fluent Korean responses for LEGIT inputs. Each model was given the facts and the purpose of claim, and instructed to predict the final court order with emphasis on identifying possible issues in its reasoning trace. The temperature was set to 0 for reproducibility.

**LLM-as-a-judge evaluation** We employ 10 different LLMs to evaluate reasoning traces with LEGIT rubrics. Specifically, we use OpenAI

<sup>3</sup>All models are instruction-tuned versions unless noted otherwise.

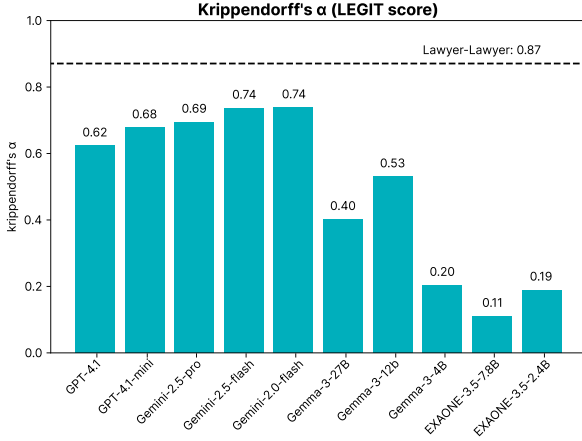


Figure 2: Lawyer-LLM inter-rater agreement in LEGIT score evaluation. Lawyers achieve strong agreement, ensuring that the generated rubrics are sound and effective. While strong LLMs (Gemini, GPT) achieve significant agreement with human experts, weaker open-sourced LLMs exhibit limited agreement.

{GPT-4.1, GPT-4.1-mini}, Gemini {2.5-Pro, 2.5-Flash, 2.0-Flash}, Gemma 3 {27B, 12B, 4B}, and EXAONE {3.5-7.8B, 3.5-2.4B}. For each issue, the LLM evaluators are prompted to assess if the given solution mentions the issue (issue coverage) and if the issue’s conclusion is accurately predicted (issue correctness). We also set the temperature to 0 for evaluators.

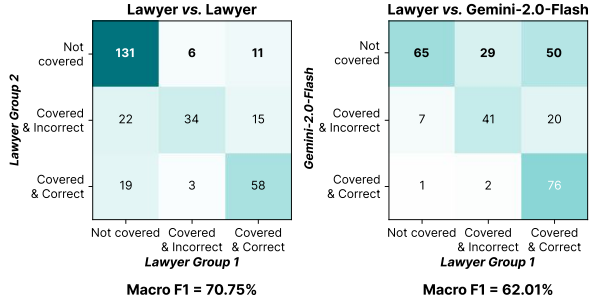
**Human expert evaluation** Two independent groups of licensed Korean lawyers were provided with 44 problems randomly sampled from the test split (/300), with responses obtained from a random LLM. The lawyers were instructed to annotate whether the LLM responses cover the given issue and reach the correct conclusion for the issue, mirroring the LLM-as-a-judge setting.

We use Krippendorff’s  $\alpha$  to compute the inter-rater agreement between lawyers and LLM evaluators, with the error function  $\delta(s_1, s_2) = (s_1 - s_2)^2$ .

#### 4.1.2 Results

**Human experts achieve excellent inter-rater agreement with LEGIT rubrics.** First, we analyze how two lawyers agree on evaluating the LEGIT dataset. Lawyers achieve Krippendorff’s  $\alpha$  score of 0.87, where the typically recommended threshold is  $\alpha > 0.67$  (Krippendorff, 2006; Stefanovitch and Piskorski, 2023). Such a strong agreement indicates that the rubrics are objective and unambiguous to human experts, proving the clarity and reliability of LEGIT rubrics.

(a) Lawyers are more stricter than LLM evaluators



(b) Smaller models are more stricter than larger counterparts

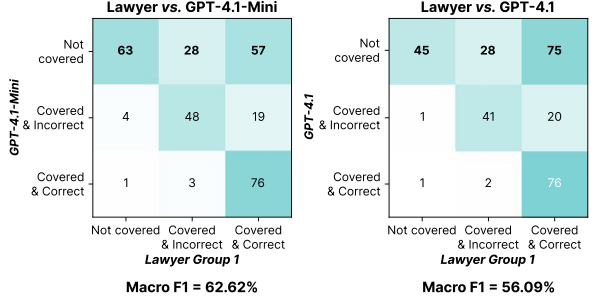


Figure 3: Confusion matrices of individual issue labels between rater groups. (a) Lawyers are the strictest evaluators in judging issue coverage and correctness, and (b) smaller models tend to be stricter than larger counterparts (e.g., GPT-4.1 vs. GPT-4.1-mini, Gemini-2.5-Flash vs. Gemini-2.5-Pro).

**Strong LLMs can reliably evaluate reasoning traces with LEGIT rubrics.** The LLM-Lawyer agreement on LEGIT scores shows the reliability of LLM-as-a-judge evaluations. Closed-source models (GPT and Gemini) achieve Krippendorff’s  $\alpha=0.62-0.74$  with human experts. While larger open-weight models show reasonable agreement with lawyers, smaller open-weight models tend to demonstrate lower performance. For instance, Gemma-3-12B achieves Krippendorff’s  $\alpha=0.53$  while the 4B version only reaches  $\alpha=0.20$ , indicating that evaluating legal reasoning traces requires substantial domain knowledge and reasoning ability even with highly informative LEGIT rubrics.

**Error analysis.** We further analyze the discrepancy between lawyers and LLM evaluators. Figure 3 presents two confusion matrices (lawyers vs. lawyers, and lawyers vs. Gemini-2.0-Flash). Comparison of these matrices shows that the LLM tends to overestimate both issue coverage and correctness, while rarely misclassifying covered or correct issues as uncovered or incorrect. In other words, human evaluators apply stricter standards when assessing the equivalence between legal concepts in the issue description and the reasoning trace,

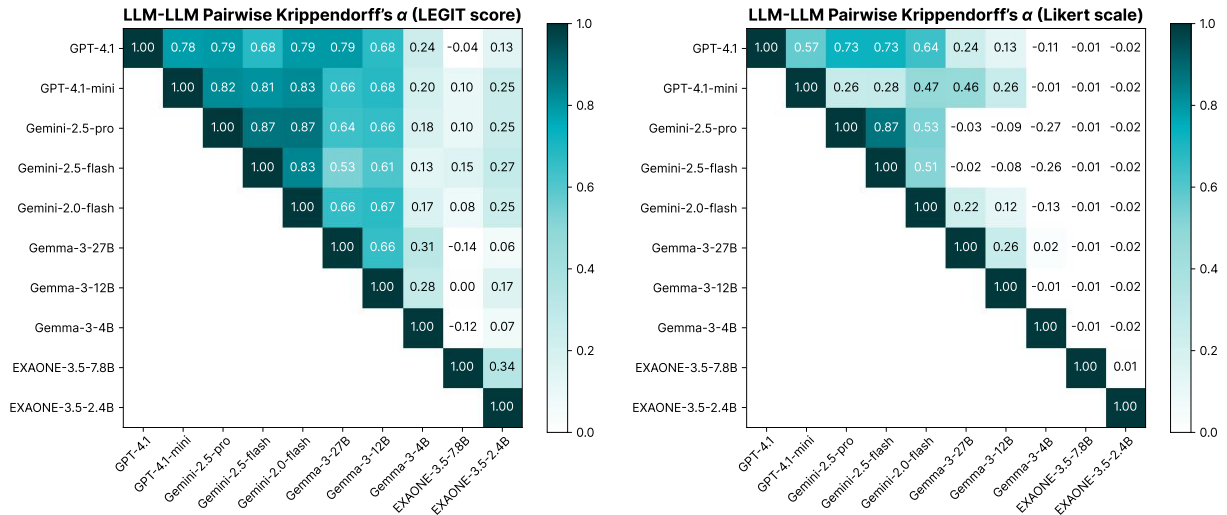


Figure 4: Comparison between LLM-evaluated scores between LEGIT score and Likert scale. Even though the Likert scale prompt includes the ground truth court judgments and rubrics, the coarse granularity limits the inter-rater agreement of LLM-as-a-judge compared to modular LEGIT rubrics.

whereas the LLM-as-a-judge adopts looser criteria by mistaking similar but actually different legal concepts. This observation also explains the better lawyer-LLM agreement of smaller models, as they evaluate issues with stricter criteria than their larger counterparts.

Further details regarding human expert evaluations and LLM prompts are included in Appendix C and H, respectively.

## 4.2 Comparing LEGIT with coarser rubrics

### 4.2.1 Experimental settings

LLM-as-a-judge is typically applied with the Likert scale (Li et al., 2024b), where the evaluator assigns a fixed-range score to the entire response. Multiple studies have proved that these evaluation results are inconsistent across different evaluator models (Li et al., 2024a; Han et al., 2025). We propose that LEGIT-style issue-based rubrics are more consistent compared to coarse Likert scale rubrics that attempt to evaluate the reasoning trace at once.

As a baseline, we use Likert scale rubrics with an integer score between 0 and 10. To ensure fair comparison with LEGIT, we provide descriptions for intermediate scores (0, 3, 7, 10) regarding issue coverage and correctness, and give the entire judgment text for reference. While it has access to all arguments and the court’s conclusions, this rubric is *coarse* in that it evaluates the entire trace within a single inference, and *underspecified* because there is no explicit guidance on what should be included in the trace. We use the same set of generator and

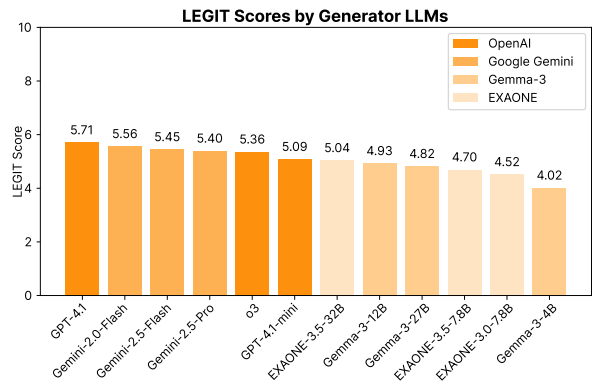


Figure 5: LEGIT score of 12 generator LLMs, evaluated with Gemini-2.0-Flash.

evaluator models as in the previous section.

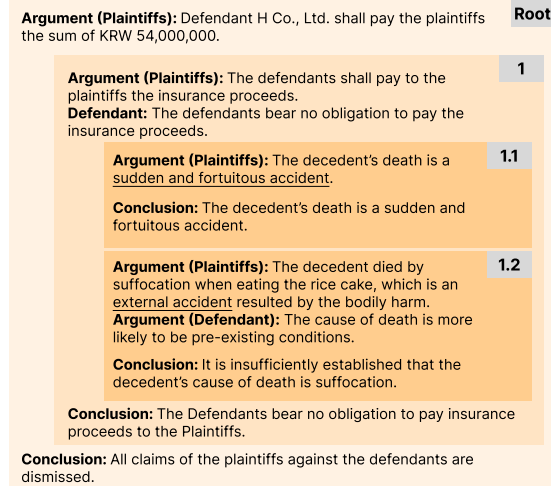
### 4.2.2 Results

**Modular LEGIT rubrics show higher evaluation consistency than the coarse Likert scale.** Figure 4 shows the pairwise LLM-LLM agreement (Krippendorff’s  $\alpha$ ) of LEGIT scores and Likert scale evaluations. Even though LLM-as-a-judge for the Likert scale has full information about the case, the modular nature of LEGIT rubrics allows a significant boost in pairwise inter-rater agreement for all LLMs. This shows that LEGIT rubrics are more robust to the selection of the evaluator LLM, likely bringing consistent evaluation results in practice.

## 5 Evaluating LLMs in legal reasoning

We now analyze the performance of different LLMs on the LEGIT benchmark. In this section, we use

### Legal Issue Tree



### OpenAI GPT-4.1

The core issue of this case is whether D's death is a sudden, fortuitous, and external accident.

...  
 Given the circumstances (pre-existing conditions, undetermined death of cause ...), existence of a food inside the mouth does not sufficiently prove the death is an external accident.

...  
 Therefore, the defendant is not obliged to pay the insurance proceeds.

**1 Covered**  
**1.1 Covered**  
**1.2 Covered**  
**1.2 Correct**  
**1 Correct**  
**Root Correct**

### OpenAI o3 - Deduction error

The plaintiffs will claim that the cause of death is the choking, which is a sudden, fortuitous, and external accident insured by the contract.

...  
 Given the paramedic's notes, it is natural to assume that the cause of death is foreign body airway obstruction by the rice cake.

...  
 Therefore, the defendant must pay the plaintiffs the sum of KRW 54M.

**1 Covered**  
**1.1 Covered**  
**1.2 Covered**  
**1.2 Incorrect**  
**1 Incorrect**  
**Root Incorrect**

### EXAONE-3.5-7.8B - Decomposition error

We need to determine if the death is an insured accident\*. Generally, if the cause of death is undetermined, it can be considered as accident.

...  
 Therefore, the final order can be predicted as:

- The defendant H should pay the plaintiffs KRW 54M.

\* Insured accident should be fortuitous, along other conditions depending on the insurance type (e.g., personal injury insurance: sudden and external)

**1 Covered**  
**1.1 Not covered**  
**1.2 Not covered**  
**1 Incorrect**  
**Root Incorrect**

Figure 6: Three LLM responses obtained from the example LEGIT problem in Figure 1. o3 fails to reason correctly about Issue 1.2, while EXAONE-3.5 does not further decompose Issue 1 into subissues.

Gemini-2.0-Flash as the evaluator based on its high agreement with human experts (Figure 2).

Figure 5 displays the performance of different LLMs. Even the LLMs with strong reasoning capabilities do not saturate the LEGIT task, where the highest LEGIT score is 5.71/10 achieved by GPT-4.1. Since all instances are generated from real court cases and guaranteed to be answerable, this score indicates that even the most powerful LLMs achieve suboptimal performance in complex legal judgment prediction scenarios.

Qualitatively, we identify two types of error, **deduction error** and **decomposition error**. During the top-down reasoning process, deduction errors occur when the model fails to reason correctly about complex and conflicting facts, while decomposition errors happen when the LLM fails to identify subissues due to a lack of legal knowledge. Figure 6 shows examples of these errors obtained from the example in Figure 1. o3 makes a deduction error by incorrectly claiming that the cause of death is choking due to distracting evidence (paramedic report). On the other hand, EXAONE-3.5-7.8B fails to list three conditions of an insured accident defined by law, which led to an irrelevant case analysis and incorrect reasoning.

We perform a quantitative analysis of how deduction and decomposition errors affect the reasoning performance. Specifically, for each depth-1 subtree of LEGIT, we measure the parent issue's correctness when (1) no child issues are covered (decomposition error), (2) at least one child issue is covered but incorrect (deduction error), and (3) all

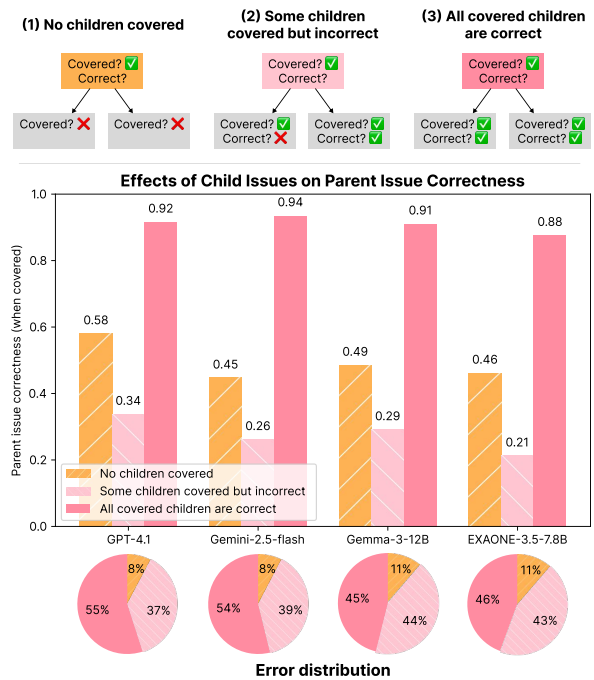


Figure 7: Correctness rate of covered parent issue, depending on child issue results. Failing to identify child issues or to reason about them correctly seriously degrades the correctness of parent issues.

covered child issues are correct. Compared to the ideal case (3), both errors cause serious degradation in the parent issue's correctness (Figure 7). In other words, when LLMs fail to identify the subissues or their reasoning on the child issues is incorrect, these errors are likely to propagate to the higher issues, leading to incorrect reasoning traces and final order prediction. Hence, it is crucial to reduce deduction and decomposition errors by improving

issue coverage and correctness in legal reasoning.

Appendix D includes further quantitative results about how problem difficulty, reasoning trace quality, and final order correctness relate.

## 6 Improving legal reasoning capabilities

As shown in the preceding analysis, low issue coverage and correctness can negatively affect the overall reasoning performance. In this section, we explore two of the most commonly used approaches for augmenting the legal reasoning capability of LLMs, retrieval-augmented generation (RAG) and reinforcement learning (RL). We test a minimal variant of each method to isolate and understand its effects on the LEGIT score.

### 6.1 Experimental settings

**Retrieval-augmented generation (RAG)** RAG can increase the factuality of LLMs by retrieving and prepending relevant documents to the input (Lewis et al., 2020; Gao et al., 2024). In the context of legal RAG, we explore *citation retrieval* (Zhang et al., 2025), a challenging form of *retrieval for reasoning* (Su et al., 2025; Shao et al., 2025) where the retriever should search for relevant statutes and leading cases to support the reasoning process. We extract all citations from the LEGIT dataset (both train and test splits) as the retrieval base, and use the fact and purpose of claim as the query.

We employ lexical matching-based BM25 (Robertson and Walker, 1994) and dense vector-based mContriever (Izacard et al., 2022), two commonly used retrievers for legal RAG (Rosa et al., 2021; Kim et al., 2025a). We also test RAG with ground truth (GT) citations as an ideal case. Ten retrieved citations obtained from each retriever (all citations for GT) are prepended to the original LEGIT problem and passed to Gemma-3-4B. We again use the same evaluator model (Gemini-2.0-Flash) for evaluation.

**Reinforcement learning (RL) with LEGIT rubrics** RL provides a way to directly optimize LLMs for any specified metric (Shen et al., 2016; Ouyang et al., 2022), including rubric-based scores (Viswanathan et al., 2025; Gunjal et al., 2025). To improve its legal reasoning ability, we train a Gemma-3-4B checkpoint using the GRPO objective (Shao et al., 2024) using LEGIT scores as rewards. We use Gemma-3-27B as the evaluator during the training phase, and Gemini-2.0-flash for

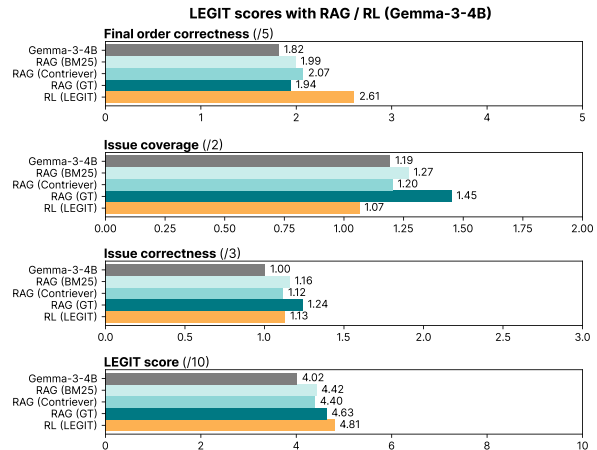


Figure 8: Comparison of LEGIT scores of Gemma-3-4B with RAG and RL. While RAG improves all components of LEGIT scores, RL significantly improves the final order/issue correctness while *reducing* issue coverage.

evaluation at test time. Using separate LLM-as-a-judge models for training and testing improves evaluation robustness by preventing the policy from overfitting to the training-time evaluator.

Refer to Appendix E-F for preprocessing, hyperparameters, computation budgets, and additional experiments (fine-tuned Contriever and more generators for RAG, final answer-only rewards for RL).

### 6.2 Results

We analyze the effects of RAG and RL by comparing their impact on the three components of the LEGIT score (Figure 8).

**RAG improves all reasoning abilities.** Prepending citations retrieved by BM25 or Contriever improves LEGIT scores by around +0.4 points. As shown in Figure 8, these improvements are distributed across all three components, as retrieved laws directly indicate possible arguments and serve as an inference rule during reasoning. Closed-sourced LLMs show consistent results; see Appendix E.

**RL prioritizes correctness at the cost of coverage.** In contrast to RAG, reinforcement learning with the LEGIT reward significantly increases final order and issue correctness, but *reduces* issue coverage. This trade-off is consistent with the analysis in Figure 7, which shows that the penalty for incorrect reasoning is more severe than omitting the issue altogether. As a result, the policy trained with RL will likely favor covering only those issues that are

straightforward, while avoiding fuzzier or subtler issues that might negatively affect the parent nodes.

Overall, these results highlight the complementary effects of RAG and RL. RAG allows broader exploration and accurate reasoning by providing relevant law, while RL sharpens the model’s reasoning correctness by pruning uncertain issues. Therefore, combining the two approaches holds promise for improving the legal reasoning ability of LLMs.

## 7 Conclusion

This work proposes LEGIT, a high-quality dataset of LJP problems and legal issue tree-based rubrics for reasoning traces. LEGIT’s rubrics allow reliable and consistent LLM-as-a-judge evaluation, achieving strong agreement with human experts. Furthermore, we show that LLMs often fail to identify relevant issues or reason about them correctly, which affects the correctness of higher-level issues and harms the response quality. Finally, we show that RAG and RL with LEGIT rubrics have complementary benefits, where RAG benefits general reasoning ability while LEGIT reward improves correctness by reducing issue coverage.

Evaluating reasoning traces is crucial for developing AI for high-stakes domains like law. Structured rubrics extracted from court judgments allow reliable evaluation of issue correctness and coverage, provide insights about how different failures propagate through the issue hierarchy, and can directly improve LLMs via RL. We believe our work represents an important step toward developing expert-level reasoning LLMs.

## 8 Limitations

First, the proposed LEGIT dataset only addresses the Korean legal system and is limited to the Korean language. However, we believe that this work has the potential to generalize beyond a single legal system and language. A proposal for evaluating and improving issue coverage of LLM legal reasoning can be found in multiple previous works across different jurisdictions (Izzidien et al., 2024; Yu et al., 2025). Furthermore, we believe that our findings on LLMs’ performance in legal reasoning (Section 5) and how RAG and RL improve it (Section 6) can be generalized beyond the data scope of this work. We believe extending LEGIT to diverse jurisdictions and languages is a promising direction, and we leave it as future work.

Furthermore, LEGIT’s rubric-based evaluation requires more compute than other LLM-as-a-judge methods, *i.e.*, Likert scale or evaluating only final order accuracy. We view this as a tradeoff between computation and evaluation reliability, echoing works from other fields that show scaling evaluation compute leads to better evaluation performance (Hashemi et al., 2024; Lee et al., 2025; Kim et al., 2025c).

Finally, LEGIT does not directly address *citation accuracy*, evaluating whether the LLM cites the correct source document for the quoted legal knowledge. Such a decision was intentional, as we frequently observe false negatives when evaluating citation accuracy. For instance, we identify at least 25 different Supreme Court case identifiers that are associated with the same case law (about claims that can be protected by the rights to revoke fraudulent conveyances) in the LEGIT dataset, while each case judgments cite only one or two of them. To properly handle such cases, we need to verify whether the cited document actually exists and is relevant given the context (Byun et al., 2024; Wu et al., 2025). However, as Korean court judgments are not freely disclosed to the public, we find this approach infeasible at the time of writing.

## References

- Shivam Adarsh, Elliott Ash, and Stefan Bechtold. 2024. [Automating Abercrombie: Machine-learning trademark distinctiveness](#). *Journal of Empirical Legal Studies*, 21(4):826–880.
- Nikolaos Aletras, Dimitrios Tsarapatsanis, Daniel Preoŕtiuc-Pietro, and Vasileios Lampos. 2016. [Predicting judicial decisions of the European Court of Human Rights: a Natural Language Processing perspective](#). *PeerJ Computer Science*, 2:e93. Publisher: PeerJ Inc.
- Chen Amiraz, Florin Cuconasu, Simone Filice, and Zohar Karnin. 2025. [The Distracting Effect: Understanding Irrelevant Passages in RAG](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18228–18258, Vienna, Austria. Association for Computational Linguistics.
- Dorian Brown. 2025. [dorianbrown/rank\\_bm25](#). Original-date: 2019-01-20T23:26:01Z.
- Courtnei Byun, Piper Vasicek, and Kevin Seppi. 2024. [This Reference Does Not Exist: An Exploration of LLM Citation Accuracy and Relevance](#). In *Proceedings of the Third Workshop on Bridging Human-Computer Interaction and Natural Language Processing*, pages 28–39, Mexico City, Mexico. Association for Computational Linguistics.
- Court of Korea. 2024. [2024 Annual Judicial Statistics](#).
- Junyun Cui, Xiaoyu Shen, Feiping Nie, Zheng Wang, Jinglong Wang, and Yulong Chen. 2022. [A Survey on Legal Judgment Prediction: Datasets, Metrics, Models and Challenges](#). *arXiv preprint*. ArXiv:2204.04859 [cs].
- Bofei Gao, Zefan Cai, Runxin Xu, Peiyi Wang, Ce Zheng, Runji Lin, Keming Lu, Dayiheng Liu, Chang Zhou, Wen Xiao, Tianyu Liu, and Baobao Chang. 2025. [LLM Critics Help Catch Bugs in Mathematics: Towards a Better Mathematical Verifier with Natural Language Feedback](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 14588–14604, Vienna, Austria. Association for Computational Linguistics.
- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. [Retrieval-Augmented Generation for Large Language Models: A Survey](#). *arXiv preprint*. ArXiv:2312.10997 [cs].
- Gemini Team. 2025. [Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities](#). *arXiv preprint*: 2507.06261.
- Gemma Team. 2025. [Gemma 3 Technical Report](#). *arXiv preprint*. ArXiv:2503.19786 [cs].
- Etash Guha, Ryan Marten, Sedrick Keh, Negin Raouf, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, Ashima Suvarna, Benjamin Feuer, Liangyu Chen, Zaid Khan, Eric Frankel, Sachin Grover, Caroline Choi, Niklas Muennighoff, Shiye Su, and 31 others. 2025. [Openthoughts: Data recipes for reasoning models](#). *Preprint*, arXiv:2506.04178.
- Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Bing Liu, and Sean Hendryx. 2025. [Rubrics as Rewards: Reinforcement Learning Beyond Verifiable Domains](#). *arXiv preprint*. ArXiv:2507.17746 [cs].
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, and 175 others. 2025. [DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning](#). *Nature*, 645(8081):633–638.
- Simeng Han, Aaron Yu, Rui Shen, Zhenting Qi, Martin Riddell, Wenfei Zhou, Yujie Qiao, Yilun Zhao, Semih Yavuz, Ye Liu, Shafiq Joty, Yingbo Zhou, Caiming Xiong, Dragomir Radev, Rex Ying, and Arman Cohan. 2024. [P-FOLIO: Evaluating and improving logical reasoning with abundant human-written reasoning chains](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16553–16565, Miami, Florida, USA. Association for Computational Linguistics.
- Sophia Simeng Han, Yoshiki Takashima, Shannon Zeng, Chen Liu, Yixin Liu, Roque K. Thuo, Sonia Knowlton, Ruzica Piskac, Scott J Shapiro, and Arman Cohan. 2025. [CourtReasoner: Can LLM agents reason like judges?](#) In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 35279–35294, Suzhou, China. Association for Computational Linguistics.
- Helia Hashemi, Jason Eisner, Corby Rosset, Benjamin Van Durme, and Chris Kedzie. 2024. [LLM-Rubric: A Multidimensional, Calibrated Approach to Automated Evaluation of Natural Language Texts](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13806–13834. ArXiv:2501.00274 [cs].
- Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024a. [Large Language Models Cannot Self-Correct Reasoning Yet](#). In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.
- Wanhong Huang, Yi Feng, Chuanyi Li, Honghan Wu, Jidong Ge, and Vincent Ng. 2024b. [CMDL: A Large-Scale Chinese Multi-Defendant Legal Judgment Prediction Dataset](#). In *Findings of the Association for*

- Computational Linguistics: ACL 2024*, pages 5895–5906, Bangkok, Thailand. Association for Computational Linguistics.
- Wonseok Hwang, Dongjun Lee, Kyoungyeon Cho, Hanuhl Lee, and Minjoon Seo. 2022. [A Multi-Task Benchmark for Korean Legal Language Understanding and Judgement Prediction](#). *Advances in Neural Information Processing Systems*, 35:32537–32551.
- Piotr Indyk and Rajeev Motwani. 1998. [Approximate nearest neighbors](#). *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, pages 604–613.
- Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. [Unsupervised Dense Information Retrieval with Contrastive Learning](#). *Transactions on Machine Learning Research*.
- Ahmed Izzidien, Holli Sargeant, and Felix Steffek. 2024. [LLM vs. Lawyers: Identifying a Subset of Summary Judgments in a Large UK Case Law Dataset](#). *arXiv preprint*. ArXiv:2403.04791 [cs].
- Cong Jiang and Xiaolei Yang. 2023. [Legal Syllogism Prompting: Teaching Large Language Models for Legal Judgment Prediction](#). *arXiv preprint*. ArXiv:2307.08321 [cs].
- Mehran Kazemi, Najoung Kim, Deepti Bhatia, Xin Xu, and Deepak Ramachandran. 2023. [LAMBADA: Backward Chaining for Automated Reasoning in Natural Language](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6547–6568, Toronto, Canada. Association for Computational Linguistics.
- Chaeun Kim, Jinu Lee, and Wonseok Hwang. 2025a. [LegalSearchLM: Rethinking Legal Case Retrieval as Legal Elements Generation](#). *arXiv preprint*. ArXiv:2505.23832 [cs].
- Seungone Kim, Juyoung Suk, Ji Yong Cho, Shayne Longpre, Chaeun Kim, Dongkeun Yoon, Guijin Son, Yejin Cho, Sheikh Shafayat, Jinheon Baek, Sue Hyun Park, Hyeonbin Hwang, Jinkyung Jo, Hyowon Cho, Haebin Shin, Seongyun Lee, Hanseok Oh, Noah Lee, Namgyu Ho, and 13 others. 2025b. [The BiGGen Bench: A Principled Benchmark for Fine-grained Evaluation of Language Models with Language Models](#). *arXiv preprint*. ArXiv:2406.05761 [cs].
- Seungone Kim, Ian Wu, Jinu Lee, Xiang Yue, Seongyun Lee, Mingyeong Moon, Kiril Gashteovski, Carolin Lawrence, Julia Hockenmaier, Graham Neubig, and Sean Welleck. 2025c. [Scaling Evaluation-time Compute with Reasoning Models as Process Evaluators](#). *arXiv preprint*. ArXiv:2503.19877 [cs].
- Klaus Krippendorff. 2006. [Reliability in Content Analysis](#). *Human Communication Research*, 30(3):411–433.
- Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilë Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, and 11 others. 2023. [Measuring Faithfulness in Chain-of-Thought Reasoning](#). *arXiv preprint*. ArXiv:2307.13702 [cs].
- Jinu Lee and Julia Hockenmaier. 2025. [Evaluating Step-by-step Reasoning Traces: A Survey](#). *arXiv preprint*. ArXiv:2502.12289 [cs].
- Jinu Lee and Wonseok Hwang. 2025. [Symba: Symbolic Backward Chaining for Structured Natural Language Reasoning](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2468–2484, Albuquerque, New Mexico. Association for Computational Linguistics.
- Mincheol Lee. 2024. [Kiwi: Developing a Korean Morphological Analyzer Based on Statistical Language Models and Skip-Bigram](#). *Korean Journal of Digital Humanities*, 1(1):109–136.
- Yukyung Lee, Joonghoon Kim, Jaehee Kim, Hyowon Cho, Jaewook Kang, Pilsung Kang, and Najoung Kim. 2025. [CheckEval: A reliable LLM-as-a-Judge framework for evaluating text generation using checklists](#). *arXiv preprint*. ArXiv:2403.18771 [cs].
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich K uttler, Mike Lewis, Wen-tau Yih, Tim Rock-t aschel, Sebastian Riedel, and Douwe Kiela. 2020. [Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.
- LG AI Research. 2024. [EXAONE 3.5: Series of Large Language Models for Real-world Use Cases](#). *arXiv preprint*. ArXiv:2412.04862 [cs].
- Haitao Li, Qian Dong, Junjie Chen, Huixue Su, Yujia Zhou, Qingyao Ai, Ziyi Ye, and Yiqun Liu. 2024a. [LLMs-as-Judges: A Comprehensive Survey on LLM-based Evaluation Methods](#). *arXiv preprint*. ArXiv:2412.05579 [cs].
- Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, Hai Zhao, and Pengfei Liu. 2024b. [Generative Judge for Evaluating Alignment](#). In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.
- Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. [Let’s Verify Step by Step](#). In *The Twelfth International Conference on Learning Representations*.

- Sewon Min, Kalpesh Krishna, Xinxu Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. [FActScore: Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.
- Venkatesh Mishra, Bimsara Pathiraja, Mihir Parmar, Sat Chidananda, Jayanth Srinivasa, Gaowen Liu, Ali Payani, and Chitta Baral. 2025. [Investigating the Shortcomings of LLMs in Step-by-Step Legal Reasoning](#). In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 7795–7826, Albuquerque, New Mexico. Association for Computational Linguistics.
- Carlos Olea, Holly Tucker, Jessica Phelan, Cameron Pattison, Shen Zhang, Maxwell Lieb, Doug Schmidt, and Jules White. 2024. [Evaluating Persona Prompting for Question Answering Tasks](#). *Security, Privacy and Trust Management*, pages 63–81. Conference Name: 12th International Conference of Security, Privacy and Trust Management ISBN: 9781923107281 Publisher: Academy & Industry Research Collaboration Center.
- OpenAI. 2024. [GPT-4 Technical Report](#). *arXiv preprint*. ArXiv:2303.08774 [cs].
- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. [Training language models to follow instructions with human feedback](#). *arXiv preprint*. ArXiv:2203.02155 [cs].
- Aditya Pathak, Rachit Gandhi, Vaibhav Uttam, Arnav Ramamoorthy, Pratyush Ghosh, Aaryan Raj Jindal, Shreyash Verma, Aditya Mittal, Aashna Ased, Chirag Khatri, Yashwanth Nakka, Devansh, Jagat Sesh Challa, and Dhruv Kumar. 2025. [Rubric Is All You Need: Enhancing LLM-based Code Evaluation With Question-Specific Rubrics](#). In *Proceedings of the 2025 ACM Conference on International Computing Education Research V.1*, pages 181–195. ArXiv:2503.23989 [cs].
- S. E. Robertson and S. Walker. 1994. [Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval](#). In *SIGIR '94*, pages 232–241, London. Springer.
- Guilherme Moraes Rosa, Ruan Chaves Rodrigues, Roberto Lotufo, and Rodrigo Nogueira. 2021. [Yes, BM25 is a Strong Baseline for Legal Case Retrieval](#). *arXiv preprint*. ArXiv:2105.05686 [cs].
- Rulin Shao, Rui Qiao, Varsha Kishore, Niklas Muennighoff, Xi Victoria Lin, Daniela Rus, Bryan Kian Hsiang Low, Sewon Min, Wen-tau Yih, Pang Wei Koh, and Luke Zettlemoyer. 2025. [ReasonIR: Training Retrievers for Reasoning Tasks](#). *arXiv preprint*. ArXiv:2504.20595 [cs].
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. [DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models](#). *arXiv preprint*. ArXiv:2402.03300 [cs].
- Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. [Minimum Risk Training for Neural Machine Translation](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1683–1692, Berlin, Germany. Association for Computational Linguistics.
- Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2025. [HybridFlow: A Flexible and Efficient RLHF Framework](#). In *Proceedings of the Twentieth European Conference on Computer Systems*, pages 1279–1297. ArXiv:2409.19256 [cs].
- Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, Johannes Heidecke, Amelia Glaese, and Tejal Parniani. 2025. [Paperbench: Evaluating ai’s ability to replicate ai research](#). *Preprint*, arXiv:2504.01848.
- Nicolas Stefanovitch and Jakub Piskorski. 2023. [Holistic Inter-Annotator Agreement and Corpus Coherence Estimation in a Large-scale Multilingual Annotation Campaign](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 71–86, Singapore. Association for Computational Linguistics.
- Hongjin Su, Howard Yen, Mengzhou Xia, Weijia Shi, Niklas Muennighoff, Han-yu Wang, Haisu Liu, Quan Shi, Zachary S. Siegel, Michael Tang, Ruoxi Sun, Jinsung Yoon, Serkan O. Arik, Danqi Chen, and Tao Yu. 2025. [BRIGHT: A Realistic and Challenging Benchmark for Reasoning-Intensive Retrieval](#). *arXiv preprint*. ArXiv:2407.12883 [cs].
- United State Court. 2024. [Federal Judicial Caseload Statistics 2024](#).
- Vijay Viswanathan, Yanchao Sun, Shuang Ma, Xiang Kong, Meng Cao, Graham Neubig, and Tongshuang Wu. 2025. [Checklists Are Better Than Reward Models For Aligning Language Models](#). *arXiv preprint*. ArXiv:2507.18624 [cs].
- Hanyin Wang, Chufan Gao, Qiping Xu, Bolun Liu, Guleid Hussein, Hari Prasad Korsapati, Mohamad El Labban, Kingsley Iheasirim, Mohamed Hassan, Gokhan Anil, Brian Bartlett, and Jimeng Sun. 2025. [Process-Supervised Reward Models for Verifying Clinical Note Generation: A Scalable Approach Guided by Domain Expertise](#). *arXiv preprint*. ArXiv:2412.12583 [cs].

- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. [Chain-of-Thought Prompting Elicits Reasoning in Large Language Models](#). *Advances in Neural Information Processing Systems*, 35:24824–24837.
- Kevin Wu, Eric Wu, Kevin Wei, Angela Zhang, Allison Casasola, Teresa Nguyen, Sith Riantawan, Patricia Shi, Daniel Ho, and James Zou. 2025. [An automated framework for assessing how well LLMs cite relevant medical references](#). *Nature Communications*, 16(1):3615. Publisher: Nature Publishing Group.
- Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. [Tree of Thoughts: Deliberate Problem Solving with Large Language Models](#). *Advances in Neural Information Processing Systems*, 36:11809–11822.
- Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. [Making Retrieval-Augmented Language Models Robust to Irrelevant Context](#). *arXiv preprint*. ArXiv:2310.01558 [cs].
- Ying-Chu Yu, Sieh-Chuen Huang, and Hsuan-Lei Shao. 2025. [Structured Evaluation of Legal Reasoning in LLMs: Chain-of-Thought Prompting and Human Scoring for Retrieval Robustness](#).
- Kepu Zhang, Weijie Yu, Sunhao Dai, and Jun Xu. 2025. [CitaLaw: Enhancing LLM with Citations in Legal Domain](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 11183–11196, Vienna, Austria. Association for Computational Linguistics.
- Eric Zhu. 2025. [ekzhu/datasketch](#). Original-date: 2015-03-20T01:21:46Z.

## A Additional LEGIT example

As another representative (and more complex) example of LEGIT, we show a *fraudulent conveyance* case and responses obtained from two LLMs, Gemini-2.5-Flash and EXAONE-3.5-32B (see Figure 9 for the English version, and Figure 10 for Korean).

In the Korean Civil Act, a fraudulent conveyance is when the obligor (someone in debt) makes a contract with a third party (e.g., selling/giving their property, setting off debt, etc.), knowing that it will harm the obligee’s rights to collect the debt back. Within a year of when the obligee acknowledged that a fraudulent conveyance had happened, the obligee can sue the third party to cancel the fraudulent contract. In this particular scenario, the obligor (company C) pays off the debt of only one obligee (defendant), which reduces the joint security for other obligees and harms their right to reclaim debt.

### Article 406 (Obligee’s Right of Revocation)

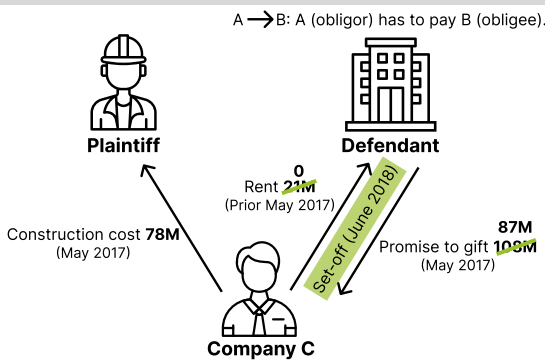
(1) If the obligor has performed any juristic act which has a property right for its subject, with the knowledge that it would prejudice the obligee, the obligee may apply to the court for its revocation and restitution of its original status: Provided, That this shall not apply where a person who has derived a benefit from such act or a subsequent purchaser was, at the time of the act or of the purchase, unaware of the fact that it would prejudice the obligee.

Although both LLMs make an incorrect final order prediction, the LEGIT rubrics clearly distinguish between the two models, since the LEGIT issue coverage and correctness scores provide richer signals than the final order-only evaluation of traditional LJP benchmarks. First, Gemini is fully aware of the three conditions to declare fraudulent conveyance (identifying issues 2.1.1-2.1.3), correctly reasoning that the set-off contract of this case is a fraudulent conveyance. EXAONE also reaches the conclusion that the set-off is fraudulent, but only points out one of the three conditions (issue 2.2.1). Here, the LEGIT rubrics clearly indicate that Gemini’s response is more accurate and informative. Second, both models incorrectly conclude that the defendant shall directly pay the plaintiff (issue 2.2),

**Example case - Translated (English)** (Judgment ID: 서울남부지방법원-2020가단273896)

**Facts**

The defendant, who operated the "E Building" under commission from the district office, leased part of the center's second floor to a company C.  
 In May 2017, C contracted with the plaintiff for structural reinforcement and interior work at the hall, agreeing to pay 108 million KRW for construction completed in June 2017. However, C paid only 30 million KRW, leaving 78 million KRW unpaid. The defendant promised C to pay for the construction cost with the district budget.  
 In June 2018, the defendant and C entered into a new agreement extending the lease and allowing mutual set-off between rent claims and agreed payments. At that time, the company was already insolvent, with liabilities of about **400 million KRW**, where the only active assets were the **108 million KRW** promised by the defendant.



**Set-off contract** here is a fraudulent act that harms other obligees. Assume C goes bankrupt, and obligees are paid proportional to debt:

- Before set-off: Plaintiff gets  $108M \times 78M / 400M \approx 21M$
- After set-off: Plaintiff gets  $87M \times 78M / 379M \approx 18M$

**Legal Issue Tree**

**Argument (Plaintiff):**  
 1. Cancel the set-off contract between the defendant and C.  
 2. Defendant shall pay the plaintiff the sum of KRW 78,000,000. Root

**Argument (Defendant):** Plaintiff's period of filing lawsuit has expired. 1  
**Conclusion:** Not sufficiently proved, pleading dismissed.

**Argument (Plaintiff):** The set-off contract is a fraudulent conveyance. It shall be canceled and the defendant shall pay 78M. 2

**Argument (Plaintiff):** The set-off contract is fraudulent. 2.1  
**Argument (Plaintiff):** The construction debt was established before the set-off contract. 2.1.1  
**Conclusion:** Claim accepted.

**Argument (Plaintiff):** The set-off contract harms the plaintiff and other obligees' right to collect the debt. 2.1.2  
**Conclusion:** The contract is a fraudulent act, unless the defendant did not know it is a fraudulent act (*good faith*).

**Argument (Defendant):** Defendant had good faith. 2.1.3  
**Conclusion:** Not sufficiently proved, pleading dismissed.

**Conclusion:** The set-off contract is a fraudulent act.

**Argument (Plaintiff):** Defendant shall pay the plaintiff 78M. 2.2  
**Conclusion:** Claim dismissed. (*It is C, not the defendant, who shall pay the Plaintiff.*)

**Conclusion:** The set-off contract between the defendant and C shall be canceled, but the defendant is not obliged to pay the debt in C's behalf.

**Conclusion:**  
 1. The set-off contract between the defendant and C shall be canceled.  
 2. The payment claim is dismissed.

**Example LLM outputs**

**Gemini-2.5-Flash**

...  
 The key issues of this case are as follows:  
 1. Is the set-off contract a fraudulent conveyance?  
 • Is the plaintiff's claim established before the set-off?  
 • Did the set-off worsen the insolvent status of C?  
 • Were C and the defendant with good faith?  
 2. If fraudulent conveyance, should defendant pay plaintiff?  
 ...  
 • It is trivial that the plaintiff's construction payment was established before the set-off contract.  
 • ... reducing the joint security, hence constitutes a fraudulent conveyance.  
 • .. The defendant would likely have known C's insolvency ...  
 ...  
 The plaintiff can claim that the defendant shall pay plaintiff.  
 ...  
**Final order prediction:**  
 • The set-off contract of issue shall be canceled.  
 • The defendant shall pay the plaintiff the sum of KRW 78M. Root Incorrect

$$\begin{aligned} \text{LEGIT} &= 0 \times 5 \text{ (Final order)} \\ &+ 6/7 \times 2 \text{ (Issue coverage)} \\ &+ 4/7 \times 3 \text{ (Issue correctness)} \\ &= 3.43 \end{aligned}$$

**EXAONE-3.5-32B**

...  
**### Key considerations**  
 1. Fraudulent conveyance (Civil Act, Article 406) 2 Covered  
 • ... 2.1 Covered  
 2. Plaintiff's right to claim payment 2.1.1 Covered  
 • ... Provided that construction is complete, the existence of such right [to collect payment from C] is trivial. 2.1.1 Correct  
 ...  
**### Possible judgment outcomes**  
 1. Provided that the fraudulent conveyance is acknowledged,  
 • The set-off contract is nullified. 2.1 Correct  
 • Plaintiff can indirectly claim payment to defendant. 2.2 Covered  
 • Specifically, the court will order that the defendant shall pay 78M to the plaintiff. 2.2 Incorrect  
 ...  
**### Conclusion**  
 ... it is likely that the set-off contract is a fraudulent conveyance, and the defendant shall pay the plaintiff. 2 Incorrect  
 ... Root Incorrect

$$\begin{aligned} \text{LEGIT} &= 0 \times 5 \text{ (Final order)} \\ &+ 4/7 \times 2 \text{ (Issue coverage)} \\ &+ 1/7 \times 3 \text{ (Issue correctness)} \\ &= 1.29 \end{aligned}$$

Figure 9: A detailed example of a LEGIT case (fraudulent conveyance, top), including facts and the legal issue tree, as well as two LLM outputs and their LEGIT scores (bottom), translated into English. Refer to Figure 10 for the original version of the data and LLM responses.

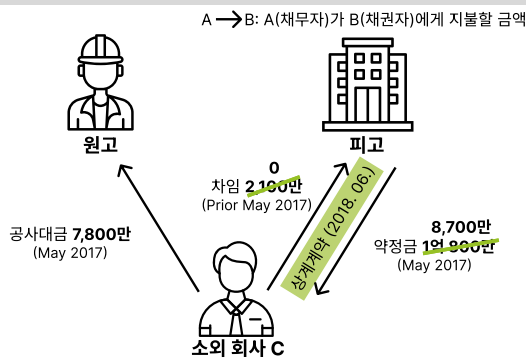
**Example case - Original (Korean)** (Judgment ID: 서울남부지방법원-2020가단273896)

**Facts**

피고는 구청으로부터 E센터 건물을 위탁받아 운영하였다. 소외 회사 C는 피고로부터 이 사건 2층 부분(F 체험관)을 3년간 연 차임 7,000만원으로 정하여 임차하였다.

2017년 5월 경, 원고는 C와 이 사건 체험관의 구조보강 및 인테리어 공사에 관하여 공사대금 1억 800만원, 공사기간은 2017년 6월까지로 정하여 공사계약을 체결하였다. 그러나, C는 원고에게 이 사건 공사대금 중 3,000만 원을 지급하였고 나머지 7,800만원은 지급하지 않았다. 피고는 C에게 이 사건 공사대금을 지원해주겠다고 약정하였다.

피고와 C는 2018년 6월, 이 사건체험관 운영기간 연장 및 장소사용료 등에 대한 약정을 체결하였다. 위 약정에서 피고와 C는 이 사건 약정금 채권과 이 사건 차임 채권을 대등액의 범위 내에서 상계하기로 합의하였다. 상계계약 당시 소외 회사의 적극재산은 사실상 이 사건의 약정금 채권 (1억 800만원)에 해당하였다. 소외 회사의 소극재산은 이 사건 공사대금 채무 및 차임 채무 등 합계 4억 원 상당의 채무가 있어 채무초과 상태였다.



**이 사건 상계계약은 다른 일반채권자들을 해하는 사해행위에 해당한다.** 예컨대, C가 파산할 경우 피고가 배당받을 수 있는 금액은 아래와 같다.

- 상계계약 이전:  $1.08억 \times 0.78억 / 4억 \approx 2,100만 원$
- 상계계약 이후:  $0.87억 \times 0.78억 / 3.79억 \approx 1,800만 원$

**Legal Issue Tree**

**Argument (Plaintiff):** 1. 피고와 C 사이에 체결된 상계계약을 7,800만 원의 한도 내에서 취소한다. 2. 피고는 원고에게 7,800만원[및 지연이자]을 지급하라. **Root**

**Argument (Defendant):** 원고의 제척기간이 도과하였다. **1**  
**Conclusion:** 피고의 분안전 항변은 받아들여지지 않는다.

**Argument (Plaintiff):** 이 사건 상계계약은 다른 일반채권자의 이익을 행하는 사해행위로, 상계계약의 취소와 그 원상회복으로 그 상당액의 반환을 구한다. **2**

**Argument (Plaintiff):** 이 사건 상계계약은 사해행위이다. **2.1**

**Argument (Plaintiff):** 원고의 이 사건 공사대금 채권이 상계 계약 이전에 발생하였다. **2.1.1**  
**Conclusion:** 원고의 피보전채권을 인정할 수 있다.

**Argument (Plaintiff):** 이 사건 상계계약은 원고 등 다른 일반채권자의 이익을 해하는 사해행위에 해당한다. **2.1.2**  
**Conclusion:** 이 사건 상계계약은 원고 등 다른 일반채권자의 이익을 해하는 사해행위에 해당하고, 채무자의 사해의사는 추정된다.

**Argument (Defendant):** 피고는 사건 당시 선의였다. **2.1.3**  
**Conclusion:** 피고가 선의였음을 인정할 증거가 없다.

**Conclusion:** 이 사건 상계계약은 사해행위이다.

**Argument (Plaintiff):** 원상회복으로 그 상당액의 반환을 구한다. **2.2**  
**Conclusion:** 원상회복을 청구할 여지는 없다. (피고가 아닌 C에게 공사대금 반환의 소를 제기해야 함)

**Conclusion** 원고의 사해행위 취소 청구는 이유 있어 인용하고, 나머지 청구는 이유 없어 기각한다.

**Conclusion:** 1. 피고와 C 사이에 체결된 상계계약을 7,800만 원의 한도 내에서 취소한다. 2. 원고의 나머지 청구를 기각한다.

**Example LLM outputs**

**Gemini-2.5-Flash**

본 사안의 핵심 쟁점은 다음과 같습니다.

1. 피고와 C 간의 상계 계약이 사해행위에 해당하는지 여부
  - 원고의 채권이 상계 계약 이전에 성립하였는지
  - 상계 계약으로 인해 C의 채무초과 상태가 심화되었는지
  - C(채무자)와 피고(수익자)에게 사해의사가 있었는지
2. 원상회복의 방법으로 피고가 원고에게 금전을 지급해야 하는지 여부
  - 원고의 피보전채권 존재: 원고의 공사대금 채권(7,800만 원)은 상계 계약 이전에 발생하였음이 명백합니다.
  - ... 공동담보를 현저히 부족하게 만들고 채무초과 상태를 더욱 심화시키는 행위이므로, \*\*객관적으로 사해행위에 해당합니다.
  - ... 피고는 C의 재정 상황을 충분히 알 수 있었다 ...

원고는 ... 피고에게 ... 자신에게 금전을 지급할 것을 청구할 수 있습니다. **2.2 Incorrect**  
**2 Incorrect**

**예상 판결:**

- 피고와 소외 회사 사이에 체결된 상계계약을 취소한다.
- 피고는 원고에게 78,000,000원[및 지연이자]을 지급하라.

**Root Incorrect**

$$\text{LEGIT} = 0 \times 5 \text{ (Final order)} + 6/7 \times 2 \text{ (Issue coverage)} + 4/7 \times 3 \text{ (Issue correctness)} = 3.43$$

**EXAONE-3.5-32B**

### 주요 법적 고려사항

1. 사해행위 (민법 406조)
  - ...
2. 원고의 채권 성립 및 이행
  - 공사 완료, 일부지급을 고려하면, 원고의 채권 성립은 명확하다.

### 예측 가능한 판결 결과

1. 사해행위 인정 시
  - 상계계약은 사해행위로 판단되어 취소된다.
  - 원고는, ... 피고에게도 간접적으로 채무 이행을 청구할 수 있다.
  - 구체적으로, 피고는 원고에게 7,800만원과 이에 대한 지연손해금을 지급하도록 명령할 가능성이 높다.

### 결론

... 사해행위의 인정 가능성이 높은 상황에서, 원고의 청구액 중 상당 부분인 7,800만 원과 지연손해금 지급 명령이 될 것으로 보입니다.

**2.1 Covered**  
**2.1 Covered**  
**2.1.1 Covered**  
**2.1.2 Covered**  
**2.1.3 Covered**  
**2.1.1 Correct**  
**2.2 Covered**  
**2.1 Correct**  
**2.2 Correct**  
**2.1.3 Correct**  
**2.1 Correct**  
**2.2 Incorrect**  
**2 Incorrect**

**Root Incorrect**

$$\text{LEGIT} = 0 \times 5 \text{ (Final order)} + 4/7 \times 2 \text{ (Issue coverage)} + 1/7 \times 3 \text{ (Issue correctness)} = 1.29$$

Figure 10: A detailed example of a LEGIT case (fraudulent conveyance, top), including facts and the legal issue tree, as well as two LLM outputs and their LEGIT scores (bottom), in Korean. Refer to Figure 9 for the English-translated version of the data and LLM responses.

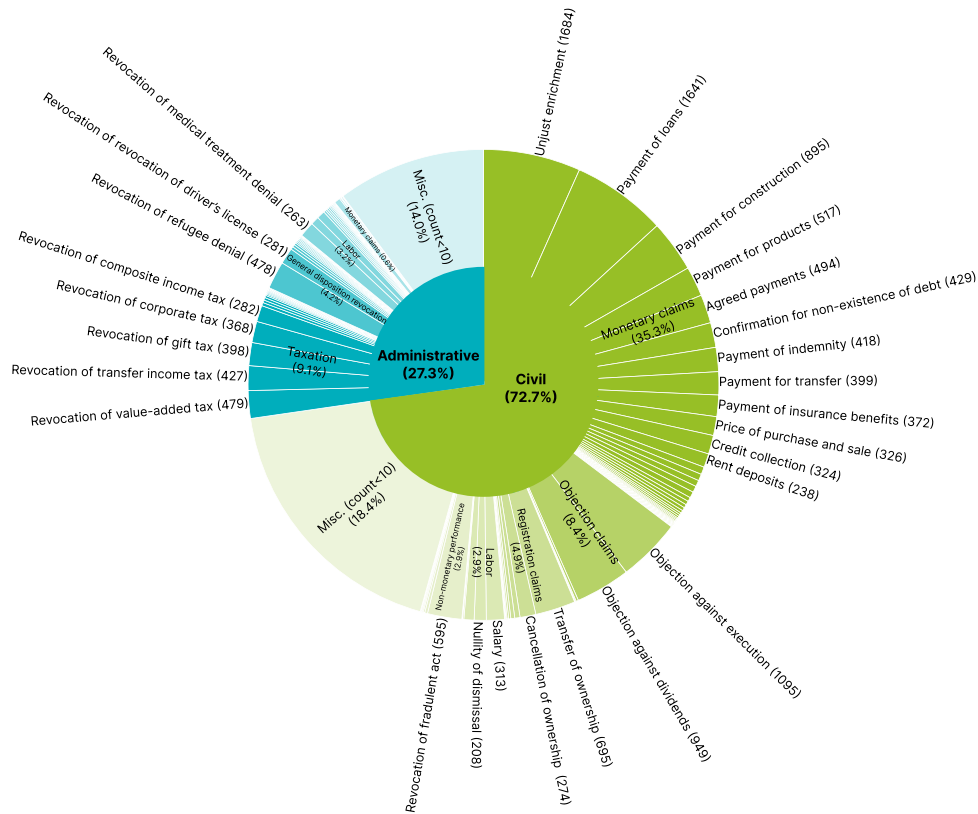


Figure 11: Distribution of case types in LEGIT. Case types that have more than 200 instances are shown with their instance count. *Misc.* subcategory includes all case types with under 10 instances. Compared to other LJP benchmarks, LEGIT includes an unprecedented variety of civil and administrative cases.

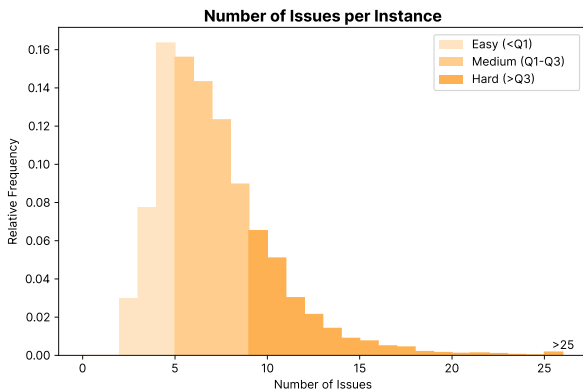


Figure 12: Histogram showing the number of issues for each LEGIT instance. The dataset is divided into easy/medium/hard difficulty subsets based on the number of issues.

which pinpoints the deduction error that led to the wrong final order prediction.

## B LEGIT dataset details

### B.1 Filtering out non-deterministic cases

We use rules to filter out non-deterministic final orders. First, we only maintain civil and administra-

tive cases, which can be identified by the document ID. Then, we filter out any judgments that include the following keywords curated by a lawyer: "compensation for damages (손해배상)", "negligence rate (과실비율)", "liability rate (책임비율)", and "compensation for pain and suffering (위자료)". Cases that include these terminologies are likely to involve the judge's discretion, which we do not allow in the LEGIT dataset.

### B.2 Dataset statistics

**Issue count** The number of issues indicates the logical complexity of a case. The issue count distribution of the entire LEGIT dataset is shown in Figure 12. The median issue count is 7, indicating that most cases in the LEGIT dataset carry a complex, nontrivial set of legal issues.

**Case types** In Korea, plaintiffs can assign *case types* when filing the lawsuit to indicate the nature of the case, *e.g.*, claim, payment of loans, revocation of corporate tax, etc. While there are no predefined lists of case type identifiers, there are many common case types that are shared by the court and legal practitioners. Note that one judgment can

Error types	Description	Occurrence
<b>Fact extraction</b> (/50 facts)		
Missing antecedents	The antecedents of some pronouns and redacted symbols are not directly mentioned (can be inferred by context).	14.0%
Overspecification	Fact paragraph includes information that is not a neutral statement.	10.0%
Missing numbers <sup>†</sup>	Important numbers (e.g., money) are redacted in the raw data, affecting multiple issues.	6.0%
Facts in attachments*	A major portion of the facts are included in the attachments, not in the original judgment.	2.0%
<b>Issue structure extraction</b> (/306 issues)		
Ungrounded claims	Parties' claims are not grounded in the facts, thus rejected by judges due to a complete lack of evidence.	5.2%
Vacuous issues	No meaningful claim and conclusion are presented.	0.7%
Duplicate issues	There are two issues with identical claim and conclusion.	0.3%
<b>Overall answerability</b> (/50)		
Answerable		92%
Partially answerable( <sup>†</sup> )		6%
Unanswerable(*)		2%

Table 1: Dataset error types and rates were analyzed from a randomly sampled subset of LEGIT’s training split (50 examples, 306 issues). Dagger (<sup>†</sup>) represents the case where it is impossible to deduce a significant amount of the issues (>25%). Asterisks (\*) denote critical errors that render a question unanswerable. All such cases stem from defects in the raw data. Other errors are minor and do not prevent experts from correctly predicting the final order or identifying the relevant issues.

have multiple case types, which occurs when the defendant files a counterclaim or two cases are merged during the trial. Figure 11 shows the distribution of case types in LEGIT, after applying string regularization. Among 3,697 distinct case types in LEGIT, there are 27 types with more than 200 instances, and 111 types with more than 10 instances. This diversity in civil/administrative cases is unrivaled by existing legal judgment prediction datasets, which often focus on either criminal cases or very narrow subsets of civil cases.

### B.3 Training set quality inspection

The analysis in Section 4 shows that lawyers achieve significant agreement with the extracted rubrics, proving the quality of the test split of the LEGIT dataset. However, unlike the test split, where the authors manually corrected errors in the inputs and rubrics, the training set remains unmodified from the automatically labeled version. Here, we report manual inspection results on the training set.

To assess the quality of extracted facts and legal issue trees, we manually inspect a small subset of the training split. The error types and respective error rates observed from 50 randomly sampled training problems are shown in Table 1. Overall, only one problem was not answerable due to missing information in the raw data, and all errors made in the LLM-based annotation stage were minor

since the experts could still deduce the correct final order and identify relevant legal issues. This shows that the LEGIT dataset’s automatic annotation is of reasonable quality, which justifies the rubric-based RL.

## C Expert annotation details

In Section 4, we collect human expert annotations to assess the quality of LEGIT rubrics and the reliability of LLM-as-a-judge methods. For this process, we hire 7 licensed Korean lawyers with sufficient knowledge in civil and administrative law. To measure inter-human agreement, we divide the lawyers into two groups and instruct them to annotate the same samples, while disallowing communication between groups during the annotation process.

The lawyers are instructed to evaluate LLM-generated reasoning traces obtained from 44 different problems using LEGIT rubrics. The sample set contains 15 easy/15 medium/14 hard problems, so that the total number of issues sums up to 300. For each response, annotators are provided (problem, LLM response, issue) tuples as input, and were asked if the LLM response covered the issue and predicted the conclusion correctly. Note that the instruction did not ask to evaluate the overall quality of the reasoning trace, nor were the lawyers knowledgeable about how the final LEGIT score was computed from their annotation. The full prompt can be seen in Figure 13.



## Annotation guide for lawyers

### Guidelines - translated (English)

The evaluation consists of two parts: whether the LLM's answer addresses the given issue (yellow column), and if it does, whether it correctly predicts the conclusion of that issue (pink column). For each issue (row), write 1 if the LLM's answer satisfies the condition, and 0 if it does not. Specifically:

#### Whether the issue is addressed (yellow):

Issues can range from those encompassing the entire judgment (e.g., purpose of claim/final order) to very specific factual questions (e.g., whether the expenditure made by Company OO on Date XX is included in business expenses). If the LLM's answer explicitly mentions the issue or refers to an argument that a party could make based on the relevant facts, mark 1; otherwise, mark 0.

- An issue is covered when explicitly mentioning the issue's title or referring to a potential argument.
- When determining whether the issue was mentioned, apply a loose standard. If the LLM's answer presents an argument that can be legally regarded as equivalent, mark 1. For example, for an issue involving the plaintiff's argument that "the administrative disposition is excessive, considering that there was no intent or negligence in the violation, that the number and degree of violations are minor, and that the resulting disadvantage is too severe," if the LLM only discusses whether the disposition was excessively harsh or an abuse of discretion, you may still mark 1 even if it does not reproduce the exact grounds (e.g., no criminal punishment, number of violations, etc.).
- Each row includes the party's argument as part of the "issue." You should determine whether the LLM correctly identified and examined that argument, or presented a legally equivalent one as described above.

#### Whether the conclusion is correct (pink):

Evaluate whether the LLM accurately predicts the judge's decision on the specific issue.

- If there is a clear legal difference between the judgment and the LLM's answer, such as dismissal vs. rejection, joint obligation vs. quasi-joint obligation, amount of payment, or the starting date for statutory interest, it should be marked as incorrect.
- If a claim is partially upheld, you must check not just whether the LLM identified it as "partially upheld," but also whether it reached the same specific conclusion (e.g., for monetary claims partially accepted, the exact amount, interest start date, and payment method [cash vs. in-kind] must all match).
- For the "root" issue in each judgment (purpose of claim/final order), only the main claim needs to match when assessing whether the order was correctly predicted; do not consider litigation costs or provisional dispositions.

Issues that appear once per row may either involve calculating specific values such as amount or interest start date (like in the final order), or simply determining the validity of a claim. For the former, the exact figures and dates must match according to the guidelines; for the latter, you only need to check whether the LLM's conclusion (e.g., "claim upheld" / "claim dismissed") agrees with the judge's decision.

### Guidelines - original (Korean)

평가하는 LLM의 답변에 해당 쟁점이 다루어졌는지 (황색 열), 그리고 다루어졌다면 해당 쟁점의 결론도 정확히 예측했는지 (분홍색 열)로 이루어져 있습니다. 해당 행의 쟁점에 대해 LLM의 답변이 각 조건을 만족하면 숫자 1, 그렇지 않다면 숫자 0을 적어주시면 됩니다. 구체적으로,

#### 해당 쟁점이 다루어졌는지 (황색):

쟁점은 판결문 전체를 아우르는 것 (청구취지/주문)부터 굉장히 세부적인 사실관계에 대한 것 (ex. OO기업이 XX일 지출한 내역이 영업비용에 포함되는지 여부)까지 다양할 수 있습니다. 답변이 해당 쟁점을 정확히 명시하거나, 사실관계를 바탕으로 당사자가 할 수 있는 주장을 언급했다면 1을, 그렇지 않다면 0을 표기합니다.

- 쟁점을 정확히 명시하는 것과 당사자가 할 수 있는 주장을 언급하는 것은 "or"의 관계에 있습니다.
- 쟁점을 언급했는지와 관련하여서는 어떠한 노력도 없는 엄격한 기준을 적용하는 대신, 법적으로 동일시될 수 있는 주장을 언급하기만 하면 1이라고 표기해주시기 바랍니다. 일례로 원고의 "이 사건 위반행위에 관한 고의나 과실이 없음이 인정되어 처벌받지 않은 점, 위반행위의 횟수나 내용 및 정도에 비하여 이 사건 처분으로 인한 불이익이 큰 점 등에 비추어 볼 때, 이 사건 처분은 가혹하다." 주장이 들어간 쟁점의 경우, 처분이 가혹하여 재량권이 일탈/남용되었는지만 검토했다면 앞의 근거(형사처벌 없는 점, 횟수나 내용 등)가 정확히 일치하지 않아도 포괄적으로 1의 점수를 부여할 수 있습니다.
- 각 행에는 "쟁점"의 일부로서 당사자의 주장이 포함되어 있습니다. 해당 주장을 LLM이 정확히 찾아내어 검토하였거나, 위의 예시와 같이 폭넓게 법적으로 동일시될 수 있는 주장을 하였는지를 판단해주시면 될 것 같습니다.

#### 결론이 정확한지 (분홍색):

특정 쟁점에 대한 판사의 판단을 LLM이 정확하게 예측했는지 여부입니다.

- 각항/각각, 연대채무/부진정연대채무, 지급 금액의 차이, 법정이자 기산일 등 판결 내용과 LLM의 답변에 명확한 법률상 차이가 있는 경우는 오답으로 판정해야 합니다.
- 각 쟁점별 주장 (혹은 청구취지)의 일부가 인용된 경우, 일부가 인용됨을 맞춘데 그치지 않고 정확히 같은 결론에 도달하였는지 판단하여야 합니다 (ex. 금전 지급 청구가 부분적으로 인정되었을 시 정확한 금액, 지연이자 기산일, 지급 방법 (가액 배상/현물배상) 등이 모두 일치해야 합니다).
- 각 판결문 별 첫 번째 쟁점 (청구취지/주문)에서 결론인 주문을 정확하게 예측하였는지 판단할 때, 본 청구에 관한 부분만 일치하면 됩니다. 즉, 소송비용 및 가처분 등의 부수적인 내용은 고려하지 않습니다.

각 행에 1개씩 존재하는 쟁점은 그 성격상 주문과 같이 액수, 기산일, ... 등 구체적인 수치를 산출하는 것이 있고, 주장의 당부를 가리기만 하는 것이 있습니다. 쟁점이 전자에 해당하는 경우는 가이드라인에 적혀있는 대로 쟁점에서 다루는 정확한 금액/기산일 등이 일치해야 하고, 후자에 해당하는 경우에는 판사와 모델의 결론(해당 주장이 이유 있음/없음)이 일치하는지만 판단해주시면 됩니다.

Figure 13: Annotation guide presented to the lawyers during expert annotation process in Section 4.

The hourly compensation was set to KRW 264,000 (approx. USD 187), and the total work time of seven lawyers is 12.78 hours.

## D Additional quantitative analyses

**Score per difficulty** LEGIT test data consists of three subsets, *easy*, *medium*, and *hard*, divided based on the number of issues. Figure 14 shows how different LLMs perform on each difficulty bin. All models show a consistent drop in LEGIT score, in all three subsections (final order correctness, issue coverage, issue correctness). This indicates that approximating the difficulty of the given case by the number of issues is a plausible strategy.

**Issue depth and coverage/correctness** We plot the relation between issue depth and their coverage/correctness in Figure 15. Both coverage and correctness exhibit a clear pattern, where issue coverage decreases rapidly as the depth increases, while issue correctness of covered issues ( $\frac{\text{correct}}{\text{covered}}$ )

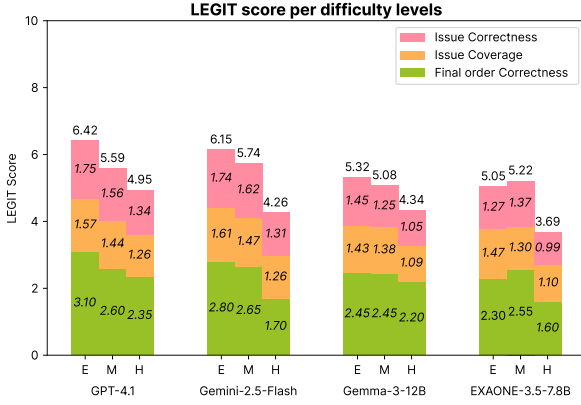


Figure 14: Component-wise LEGIT score of four LLMs, divided by difficulty subsets (E: Easy, M: Medium, H: Hard). Individual score components (final order accuracy, issue coverage, issue correctness) generally drop as the case becomes more complex.

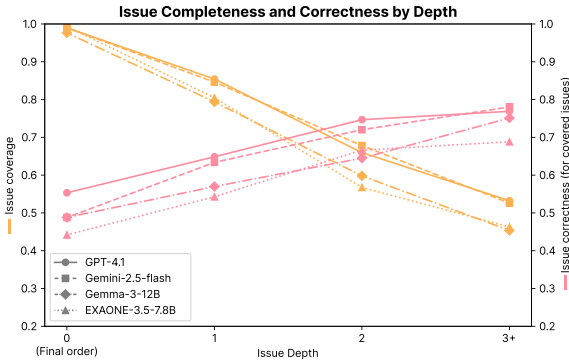


Figure 15: (1) Issue coverage and (2) issue correctness of covered issues ( $\frac{\text{correct}}{\text{covered}}$ ) per issue depth.

decreases as the depth decreases. These observations suggest that these rubrics are inherently structured as a tree, supporting the backward chaining intuition presented in Section 3 and motivating analyses in Section 5. Once the LLM fails to identify a parent node, it will likely fail to cover its children that must be obtained by decomposing the parent. Similarly, once the child issue is incorrect, the parent will also likely be incorrect, as shown in Figure 7.

**Issue coverage/correctness and final order correctness** Figure 16 shows the relationship between the reasoning trace’s quality (issue coverage and correctness) and the final order correctness. The strong correlation suggests that a high-quality reasoning trace is essential for deriving the correct final order, demonstrating the structural nature of legal reasoning.

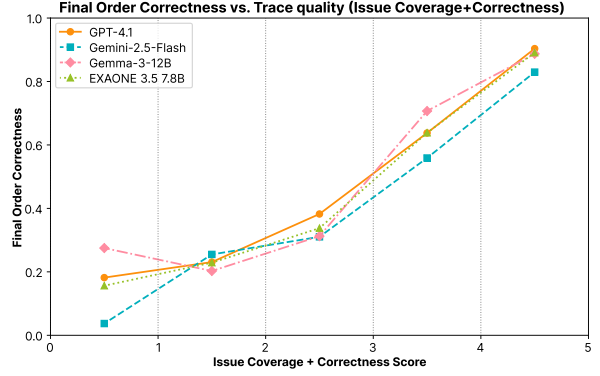


Figure 16: Issue coverage + correctness scores strongly correlate with final order correctness, suggesting that reasoning trace quality has a *causal* effect on the final answer accuracy.

**Parent issue correctness and child coverage/correctness** We include detailed visualizations about the relationships between parent correctness and children coverage/correctness in Figure 17, extending Section 5 and Figure 7. Specifically, we bin all covered parent issues by child coverage and correctness, both ranging from 0 to 1, and evaluate parent issue correctness for each bin. Note that the correctness value is always smaller than coverage, as coverage counts all *covered* issues but correctness counts all *covered and correct* issues. In this figure, the main diagonal represents the contour line of issue correctness for the covered issues ( $\frac{\text{correct child}}{\text{covered child}}$ ).

The results show that the parent accuracy decreases when (1) child issue coverage decreases ( $\checkmark$ ) and (2) correctness of covered child issues decreases ( $\downarrow$ ). However, the latter exhibits more severe degradation than the former, consistent with the intuition that low correctness is more detrimental than low coverage (Figure 7). It is noticeable that the gap between identifying 0-20% of child issues and 20%-40% of child issues is similar to or larger than the gap between 20%-40% and 80%-100%, motivating the separate definition of decomposition error (identifying no child issues).

## E Retrieval-augmented generation details

### E.1 Experimental settings

**Retrieval base preprocessing** We use LLMs to extract any citation to statutes and Supreme Court cases, and filter out any malformed strings (the quote being shorter than 20 characters). Then, we apply the MinHash and LSH algorithm (Indyk and Motwani, 1998) implemented by Zhu (2025) for

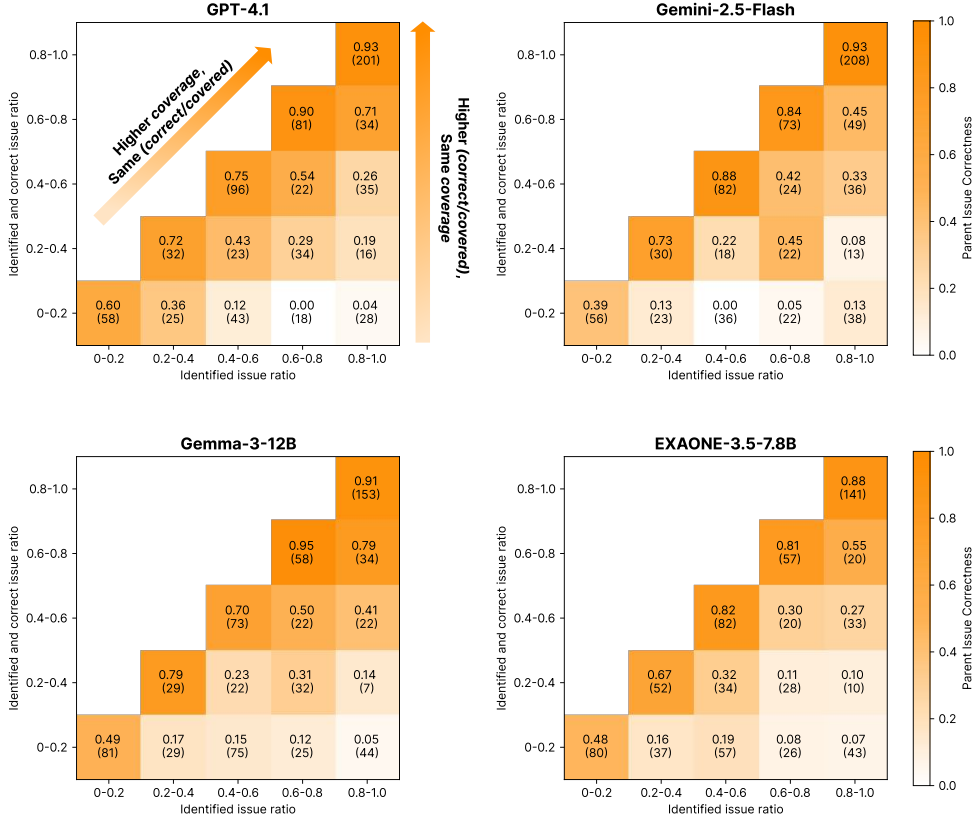


Figure 17: Parent issue correctness binned by child issue coverage and correctness, extending Figure 7. In each cell, the numbers without parentheses (above) are the parent issue correctness values, and the numbers within parentheses (below) are the number of elements in the bin.

deduplication. We set the number of permutations in MinHash to 64, and the LSH threshold to 0.65.

**BM25 details** The text is first segmented into a list of words, where only content words (nouns, verbs, adjectives, adverbs, numbers, foreign characters) are maintained using the Korean part-of-speech tagger Kiwi (Lee, 2024). Then, we use Okapi BM25 (Robertson and Walker, 1994) implemented by Brown (2025) to retrieve the top 10 candidates using the facts as the query. We use  $k_1 = 1.5$  and  $b = 0.75$  for hyperparameters.

**mContriever details** We use mContriever checkpoint fine-tuned on the multilingual MS-MARCO dataset (facebook/mcontriever-msmarco). We truncate both query and target documents to 512 tokens, which is the maximum length supported by the model.

**mContriever fine-tuning details** Extending the results in Section 6, we further fine-tune the mContriever checkpoint above using the LEGIT train split. To apply contrastive loss, we use ground-truth citations as positive documents and BM25 retrieval results that are not positive as negative

documents. We train the model for 2,000 steps (approx. 3 epochs), with a batch size of 64 and a learning rate of  $1e-4$ .

## E.2 Additional results

**RAG improves all three components of LEGIT score.** Figure 18 shows the LEGIT score for all three generators (GPT-4.1, Gemini-2.5-Flash, and Gemma-3-4B) and five retrieval settings (No RAG, BM25, Contriever, Fine-tuned contriever, and Ground-truth citations). The LEGIT score increases from 0.1-0.4 for all combinations for three retrievers, and 0.6-1.3 for ground-truth citations. The improvement happens within all three components of the LEGIT score, consistent with the results from Section 6.

**RAG remains helpful even when the retriever’s performance is limited.** Table 1 shows the retrieval performance (Recall@10, NDCG@10) of different retrievers alongside their RAG performance on three generators. Unfortunately, the performance of retrievers is generally low, as directly performing citation retrieval only using the facts is extremely challenging. However, contrary to com-

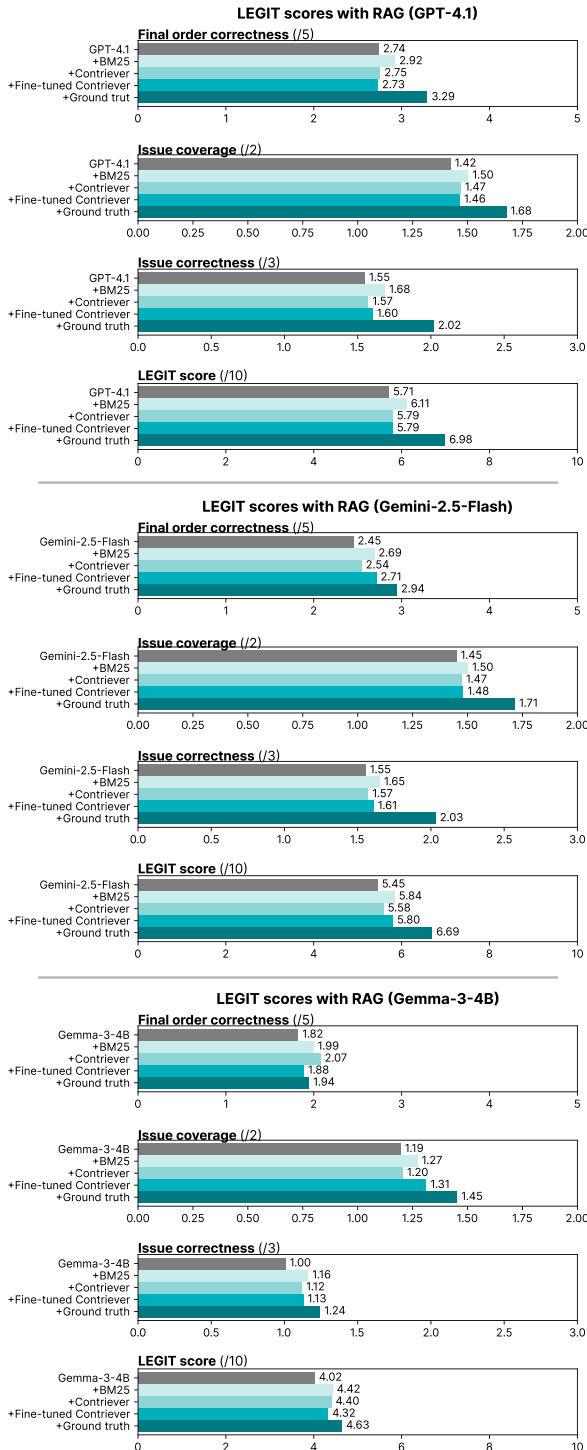


Figure 18: Results of RAG for LEGIT dataset, with three different generators and five retrieval settings (No RAG, BM25, two Contrievers, and Ground-truth citations). RAG improves LEGIT score for around 0.1-0.4 for all (generator, retriever) pairs, with a gain in all three components.

Hyperparam.	Value
Objective	GRPO
KL Div. Coef.	1e-3
Max prompt len.	2048
Max output len.	4096
Batch size	32
Rollouts	8
Optimizer	AdamW
Learning rate	1e-6
Evaluation	every 20 steps
Early stop	60 steps

Table 2: Hyperparameters used for RL training, both LEGIT rewards and final order correctness rewards.

mon belief (Yoran et al., 2024; Amiraz et al., 2025), we observe a significant performance gain with RAG in all three generator LLMs despite the low retriever performance. For instance, BM25 achieves a Recall@10 of around 4%, but the LEGIT scores increase by 0.38-0.42 for all generators, which is between 32.1-64.4% of the performance gain observed with ground truth citations compared to the base setting. While these retrieved laws might not directly relate to the given situation, we conjecture that prepending relevant laws triggers the *persona effect* (Olea et al., 2024) that elicits the legal reasoning ability of the LLMs.

## F Reinforcement learning details

### F.1 Experimental settings

**RL settings** For online RL, we use ver1 (Sheng et al., 2025), an open-source library for training LLMs with RL. We use FSDP as the model training backend, and vLLM for generating rollouts. For reproducibility, we list core training hyperparameters in Table 2. For computing training/validation reward with Gemma-3-27B, we use vLLM as the inference engine.

**Computation** For training, we use 4 NVIDIA A100 GPUs for model training (rollout generation, backpropagation, etc.) and another 4 NVIDIA A100 GPUs for running the online LLM-as-a-judge evaluation. Training with LEGIT reward took approximately 41.6 hours of wall clock time in total, while training with the final order correctness reward took 18.6 hours.

### F.2 LEGIT rewards and final answer rewards

We compare the behavior of models trained with (1) final order correctness rewards, where the model receives a reward of +10 only when the final order prediction is correct and 0 otherwise, and (2)

LEGIT score rewards. Note that both rewards are computed using LLM-as-a-judge. Furthermore, we impose a degeneration penalty of -5 for both settings, where the LLM detects and penalizes undesirable behaviors like code switching and n-gram repetition.

**LEGIT score reward achieves better final order accuracy and trace quality than final answer reward.** When training Gemma-3-4B with LEGIT rewards, we observe a significant performance gain in the test set (4.02→4.77), outperforming RAG with ground truth citations and roughly matching the performance of Gemma-3-27B (4.82). However, training with final answer-only rewards achieves 4.31. Interestingly, the final order correctness of the model is lower than one trained with LEGIT rewards despite being directly trained on that metric. The results show that optimizing the quality of reasoning traces via RL leads to better outcomes in legal reasoning, contrary to the final answer-only RL paradigm in math and programming popularized by DeepSeek-R1 (Guo et al., 2025). Furthermore, we show that high inter-LLM consistency (Section 4) transfers to robust RL performance against using different evaluators in train/test time.

## G Miscellaneous (author checklists)

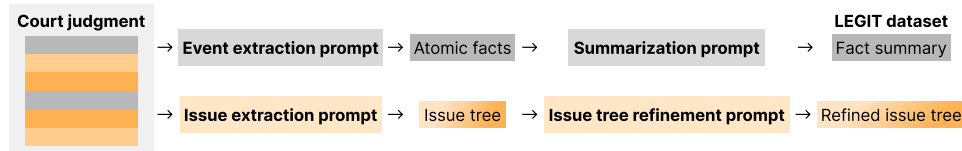
**Copyright and personal information** Korean court judgments are not bound by any copyrights under law (Copyright Act, Article 7). However, most court judgments are not freely distributed but sold by the Court to cover administrative fees. All judgments used in LEGIT are completely anonymized either by the court or by LBOX. All models and software used in this paper permit academic use.

**Randomness** To minimize the randomness, the temperature of the generator and evaluator LLMs was set to 0 in all evaluation experiments. All training experiments (fine-tuning mContriever and RL with rubrics) have been performed once due to high experiment cost and limited compute.

## H Prompts

Figures 19 and 20 present the prompts used for creating the LEGIT dataset and evaluating with rubrics. The original Korean prompts are translated into English for readability, and few-shot examples are omitted due to space. The full code and prompts can be found in our official [GitHub repository](#).

## 1. Extraction/Structuring with LLMs



### Event extraction prompt (3-shot)

Extract all "events" dealt with in this judgment.

- "Event" is an occurrence that actually happened and can have concrete date/time, place, acts, etc.
- Even if specific date/time/place are not identified for various reasons, if it is an apparent/clearly established fact that the occurrence took place (i.e., it is unspecified only because of lack of evidence), it is acceptable to include it.
- Even if the wording appears in the judgment's reasoning sections (e.g., "According to ..., it is recognized that ...", "The following circumstances can be recognized"), include it so long as it concerns a concrete event or circumstance. If an event is briefly described earlier and appears later in more detail, it's fine to include it multiple times.
- Important: Never include claims, suppositions, or speculation (especially phrases like "there is no evidence that \_\_\_"). Include only adjudicated facts (findings of fact / established facts) or content that the judge has officially recognized as events that actually occurred. This is a very important condition—carefully check to avoid mistakes.

Judgment: {Judgment}

### Summarization prompt (0-shot)

Combine the information below into a coherent text. It is acceptable to change the order or refine the wording to make the content easier to understand, but do not replace legal terminology with other expressions. Subjective words should be avoided as much as possible (e.g., ~only, ~merely, ~abundantly acknowledged, ~no reason, etc.) and should be rewritten with neutral terms. Write only the summary, without adding any introductory or concluding remarks.

Case information: {Atomic facts}

### Issue extraction prompt (3-shot)

Analyze the issue structure of the given judgment and represent it in the following JSON format.

- "id" must show the hierarchical relationship of each issue, such as 1, 1.1, 1.2, etc. Note that this is clearly different from the judgment's table of contents. For example, if multiple parties' arguments on the same issue are presented across different paragraphs, you must merge them and represent them as a single issue.
- "summary" should contain the key keywords of the issue.
- "claim" should be written as a Python dict, containing only the parts of the arguments (by plaintiff/defendant/accused/prosecutor, etc.) that are relevant to the issue (summary). The "claimer" must be written exactly as expressed in the judgment text.
- "conclusion" should contain the court's conclusion regarding each argument.

The very first issue must always have an "id" of an empty string "". For this first issue:

- "summary" must be an empty string,
- "claim" must include the entire purpose of claim separated by party,
- "conclusion" must include the entire final order.

Judgment: {Judgment}

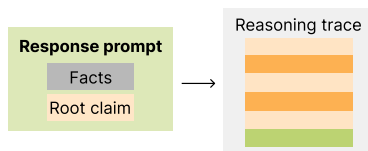
### Refinement prompt (0-shot)

Find and correct the errors in the issue structure just printed. Focus especially on the following points:

- Important: If there are references such as "Same as the final order," "Same as the purpose of claim," "As previously examined," "As seen earlier," etc., replace the quotation with the original text it refers to.
- Important: If opposing arguments on the same issue appear much later and were not included in the tree (e.g., earlier only the plaintiff's argument was included, but later the defendant's counter-argument appears), make sure to add those parts.
- Important: If "relevant\_law" is written as a separate subsection with no content in both "claim" and "conclusion," connect those "relevant\_law" sections appropriately to the right node and delete the original empty node.
- Important: Every node's "conclusion" must include the result concerning the issue mentioned in its summary/claim.
- "conclusion" should only contain the decision (approval/rejection) regarding each argument, not the rationales.

Issue tree: {Issue tree}

## 2. Response generation



### Response prompt (0-shot)

You are a legal expert in Korean law. Carefully read the given facts and the purpose of claim, and then predict the outcome of the judgment. Your prediction should clearly reflect the relevant statutes and Supreme Court precedents. In doing so, thoroughly examine possible arguments, defenses, and counterarguments.

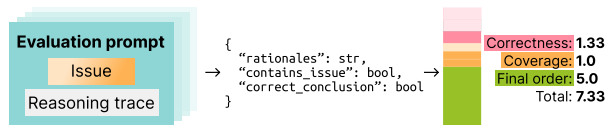
{Facts} {Facts}

{Root claim} {Root claim}

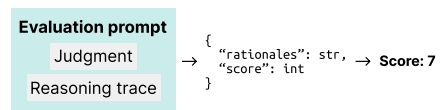
Predict the final order of this case.

Figure 19: Prompts for dataset construction and generating LLM responses.

### 3. Reasoning trace evaluation (LEGIT)



### 3-1. Reasoning trace evaluation (Likert scale)



#### LEGIT evaluation prompt (1-shot)

Read the text about a legal issue and determine: (1) Whether the content of the "summary/claim" is addressed in the TEXT, and (2) Whether the "conclusion" matches the conclusion of the TEXT.

Evaluation method:

- Judge whether the direction of the conclusion is consistent with the TEXT.
  - If the conclusion is in the format of "the claim is correct/incorrect," first determine whether the claim stated in "claim" is discussed in the text, and then determine whether the conclusion (correct/incorrect) is accurate.
  - If the conclusion is in the format of a final order, evaluate whether the predicted amount of payment, performance details, reckoning date, dismissal/rejection, etc., are all correct. However, exclude litigation cost allocation and provisional execution.
  - For any other type of conclusion, consider the overall context of the text and determine whether the same issue was addressed and whether the same conclusion was reached.
- Apply very strict standards for the consistency of legal terms (e.g., dismissal/rejection, ownership/possession, expectation interest/reliance interest).

Format:

- The answer must strictly follow the format: `<OUTPUT>{"rationales": str, "contains_issue": bool, "correct_conclusion": bool}</OUTPUT>`
- In "rationales," provide a brief one-paragraph explanation of the evaluation.
- The final JSON format must be output as a Dict, exactly as in the example.

```
TEXT: {Reasoning trace}
summary: {Issue[summary']}
claim: {Issue['claim']}
conclusion: {Issue[conclusion']}
```

#### Likert scale evaluation prompt (1-shot)

Read the text about a legal issue and determine: (1) whether the content of the "summary/claim" is addressed in the TEXT, and (2) whether the conclusion in "conclusion" matches the corresponding part of the TEXT.

Evaluation Method:

- The following TEXT is a review that predicts the expected final order of the judgment (JUDGMENT) based on the facts and the purpose of claim of the case.
- After reading the judgment, evaluate whether the given TEXT sufficiently addresses the issues and arrives at the correct prediction of the final order. Assign an integer score between 0 and 10.
- The approximate meaning of the scores is as follows:
  - 0 points: The TEXT is completely unrelated to the content of the judgment.
  - 3 points: The TEXT addresses about half of the issues raised in the judgment and accurately predicts the conclusions on those issues, but the prediction of the final order is incorrect.
  - 7 points: The TEXT addresses about half of the issues raised in the judgment and accurately predicts the conclusions on those issues, and also correctly predicts the final order.
  - 10 points: The TEXT identifies and addresses all of the issues raised in the judgment, accurately predicts the conclusions on each issue, and correctly predicts the final order.

Format:

- The answer must strictly follow the format: `<OUTPUT>{"rationales": str, "score": int}</OUTPUT>`
- In "rationales," briefly explain the reason for the evaluation in about one paragraph.
- The final JSON tag format should be output as a dictionary, as shown in the example.

```
JUDGMENT: {Judgment}
TEXT: {Reasoning trace}
```

Figure 20: Prompts for evaluating the reasoning traces, either with LEGIT rubrics (left) or Likert scale (right).