AutoTaskEval: Towards Domain-Specific and Fine-Grained Evaluation for LLMs

Qingqing Lyu[1*], Linjuan Wu[1*], Yongliang Shen[1], Hengwei Liu[1], Hao Li[2], Shengpei Jiang[2], Yin Zhang[1†], Weiming Lu [1†]
{lyuqingqing, wulinjuan525, syl, hengweiliu, yinzh, luwm}@zju.edu.cn
{lihao799, shengpeijiang}@sf-express.com

[1]Zhejiang University, [2]SF Technology

Abstract

Despite the rapid progress of LLMs, their evaluation remains hindered by static, manually curated benchmarks with limited task coverage and poor adaptability to emerging domains. Existing automated approaches typically operate within fixed task schemas and often fail to autonomously discover new evaluation dimensions, limiting both scalability and effectiveness. To address these gaps, we propose AUTOTASKEVAL, an automated framework that constructs domain-specific benchmarks directly from unstructured corpora. Using a refined Bloom's Taxonomy, the framework systematically discovers tasks, enriches contextual grounding via iterative Socratic prompting, and generates diverse, progressively challenging evaluation instances. Applied to the complex and knowledge-intensive legal domain, AUTOTASKEVAL uncovers a broader and more fine-grained task space than expert-curated benchmarks while producing high-quality instances that preserve established model-level evaluation trends. We further validate its robustness in a low-structure e-commerce review domain. Together, these results show that AUTOTASKEVAL enables scalable, adaptive, and high-fidelity LLM assessment across domains and model families, advancing autonomous and capability-sensitive evaluation.

1 Introduction

Large language models (LLMs) are advancing rapidly (Comanici et al., 2025; Liu et al., 2024; Meta, 2025; OpenAI et al., 2024), yet their evaluation lags behind, relying on static, human-curated benchmarks that are hard to scale and quick to saturate. As model performance approaches or surpasses human levels, these benchmarks lose discriminative power, hindering timely and domain-adaptive assessment of model capabilities (Zhao et al., 2025a; Owen, 2024).

Figure 1: A refined meta-capability taxonomy grounded in Bloom's Taxonomy, comprising 27 Fine-Grained Cognitive Skills. It is designed to guide autonomous task discovery from domain texts. Detailed descriptions of all sub-capabilities are provided in Appendix A.

Manual benchmark construction is costly and yields limited coverage and granularity. Thus, even well-designed datasets quickly become outdated as domains evolve, failing to capture emerging tasks or finer-grained capabilities. Sustaining meaningful evaluation amid rapid model progress requires automated systems that can autonomously discover tasks, generate high-quality instances, and adapt to domain-specific distributions, enabling continuous and fine-grained assessment of LLMs.

Some efforts attempt to automate LLM evaluation through interactive platforms or prompt-based generation. Interactive systems (Chiang et al., 2024; Zhao et al., 2025b) support large-scale model comparisons but primarily target dialogue preferences and reveal little about fine-grained, task-specific skills. Prompt-based frameworks (Pombal et al., 2025; Butt et al., 2024; Li et al., 2025) automate dataset creation but operate within fixed task schemas and lack domain adaptivity. Critically, these methods generate tests only for predefined tasks and cannot autonomously discover new ones, limiting both task coverage and the exploration of

[*]Equal contribution.
[†]Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6191–6223
July 2-7, 2026

domain-level capabilities.

To bridge this gap, we introduce AUTOTASKEVAL, an automated evaluation framework that dynamically constructs domain-specific benchmarks and supports fine-grained capability assessment. Starting from unstructured domain corpora (e.g., textbooks, technical documentation), AUTOTASKEVAL autonomously discovers salient tasks, generates high-fidelity evaluation instances, and organizes them into a cognitively grounded capability taxonomy. This end-to-end automated pipeline enables systematic, scalable, and domain-adaptive evaluation of LLMs in fast-evolving fields.

Specifically, AUTOTASKEVAL constructs benchmarks through three stages: Task Discovery, context expansion, and instance synthesis. In the first stage, Task Discovery identifies domain-relevant tasks from unstructured texts using a refined Bloom's Taxonomy (Figure 1), ensuring systematic coverage of cognitive capabilities. The context expansion stage then applies iterative Socratic prompting (Qi et al., 2023) to extract salient concepts and retrieve supporting knowledge, forming a compact contextual knowledge base. Finally, instance synthesis combines the same iterative procedure with an Instance Evaluator and Dialog Detector to synthesize high-fidelity and progressively complex evaluation examples. Together, these stages enable AUTOTASKEVAL to autonomously construct a cognitively grounded and challenging benchmark for fine-grained LLM assessment using only open-source models.

We instantiate AUTOTASKEVAL in the legal domain to demonstrate its ability to perform systematic and fine-grained capability assessment. By discovering tasks directly from unstructured corpus rather than predefined categories, the framework uncovers a broader and more nuanced capability space than manual benchmarks. The resulting instances align with expert-curated datasets in evaluation trends while providing more challenging and higher-fidelity test cases. Further applied to noisy e-commerce data, AUTOTASKEVAL proves robust in low-structure settings by extracting coherent tasks and synthesizing benchmarks without relying on explicit ontologies. Our key contributions are summarized as follows:

- We propose AUTOTASKEVAL, an automated framework for fine-grained, domain-adaptive evaluation of LLMs.
- The framework enables autonomous task discovery from unstructured texts, reducing manual effort and supporting scalable capability assessment.
- Experiments show that AUTOTASKEVAL constructs high-quality, domain-specific benchmarks with open-source models, enabling clear and systematic evaluation across model series and scales.

2 Related Work

2.1 Automatic LLM Evaluation

Evaluating LLMs remains challenging. Traditional human-curated benchmarks are costly to build and often lack diversity and coverage for complex, domain-specific tasks. Recent work has explored automated evaluation to reduce manual effort. Interactive platforms like Chatbot Arena (Chiang et al., 2024) and SciArena (Zhao et al., 2025b) rely on human feedback to rank models, mainly reflecting dialogue preferences and providing limited insight into fine-grained, domain-specific abilities. Closer to our approach are automatic benchmark generation frameworks: Zero-shot Benchmarking (Pombal et al., 2025) creates datasets using a meta prompt and judgment prompt; BenchAgents (Butt et al., 2024) decomposes benchmark construction into modular LLM agents with optional human feedback; AutoBencher (Li et al., 2025) formulates dataset generation as an optimization guided by declarative criteria. These methods rely on fixed task schemas and cannot discover new tasks or adapt to domain structures. In contrast, AUTOTASKEVAL performs **autonomous task discovery**, extracting domain-relevant competencies from unstructured text and organizing them into a cognitively grounded hierarchy, enabling scalable, systematic, and domain-adaptive evaluation beyond predefined benchmarks.

2.2 Challenging instance synthesis

Some methods generate challenging evaluation instances but often treat difficulty in isolation. Li and Zhang (2024) plan an answer schema and refine it with difficulty-focused annotations. Patel et al. (2025) decompose generation into verifiable subtasks to ensure correctness and control complexity. Shah et al. (2025) increase difficulty by combining multiple skills per problem, following expert-designed datasets like MATH. Other approaches leverage model failures: Li et al. (2025) formulates benchmark construction as error-guided optimization, while Chen et al. (2024) build SC-G4 by an-

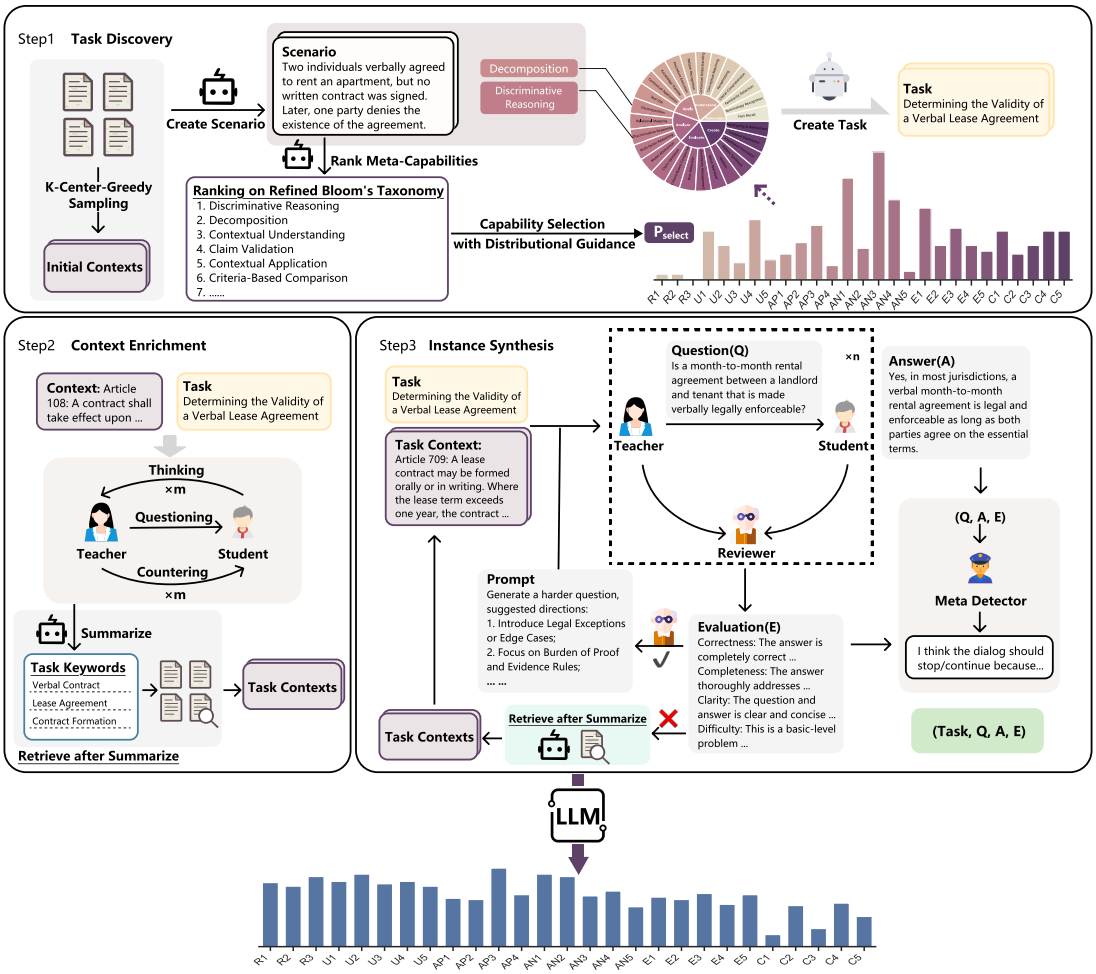

Figure 2: Overview of AUTOTASKEVAL. The automated benchmark construction consists of three steps. Step 1: ***Task Discovery***. Contexts are sampled via K-Center-Greedy from an unstructured corpus to build domain scenarios and select two meta-capabilities, which are used to create tasks. Step 2: ***Context Enrichment***. Iterative Socratic prompting between teacher and student agents extracts key concepts used to retrieve top-k relevant corpus contexts. Step 3: ***Instance Synthesis***. Based on Socratic prompting, the teacher poses questions, the student answers, and the reviewer assesses quality, guiding progressively harder instances generation. The resulting dataset supports comprehensive model evaluation, profiling performance across the refined Bloom's taxonomy.

alyzing GPT-4 mistakes. In contrast, we employ iterative Socratic prompting (Qi et al., 2023) to **progressively refine evaluation instances** from discovered tasks and domain knowledge. Starting from seed concepts, self-questioning elicits deeper reasoning, ensuring semantic accuracy and alignment with Bloom's Taxonomy, producing structured, interpretable instances for systematic capability assessment rather than mere difficulty.

3 Method

Our method introduces an automated framework for constructing domain-specific benchmarks and enabling fine-grained, adaptive evaluation of LLM capabilities. As shown in Figure 2, the pipeline comprises three stages: **Task Discovery** (Sec. 3.1), **Context Enrichment** (Sec. 3.2), and **Instance Synthesis** (Sec. 3.3), operating end-to-end on unstructured domain texts using open-source LLMs. We instantiate AUTOTASKEVAL in the legal domain (Sec. 3.4) and conduct comprehensive evaluation (Sec. 4) to demonstrate its ability to generate high-quality, fine-grained assessments. Formal definitions of key concepts are provided in Appendix A.

3.1 Task Discovery

A central limitation of current LLM evaluation is that benchmarks remain concentrated in well-studied domains, leaving specialized or underrepresented capabilities insufficiently assessed. AUTOTASKEVAL addresses this gap by autonomously discovering tasks from unstructured domain texts using a structured cognitive taxonomy. As shown

in Figure 1, we define a two-level meta-capability hierarchy based on Bloom's Taxonomy, comprising six cognitive dimensions: Remember (R), Understand (U), Apply (AP), Analyze (AN), Evaluate (E), and Create (C), encompassing 27 fine-grained sub-capabilities. This hierarchy provides a systematic scaffold for principled task discovery.

Scenario Distillation. Unstructured domain corpora are extensive yet uneven in quality. To obtain representative coverage, we apply a *K-center greedy algorithm* to select a diverse subset of text segments, minimizing redundancy while maximizing semantic breadth. For each sampled context, the LLM then constructs a coherent domain scenario, converting raw text into an authentic evaluation setting (see Appendix B for prompt details).

Capability Selection. For each constructed scenario, we select two meta-capabilities to guide task generation. To balance contextual relevance and global coverage, we compute a selection distribution $\mathbf{P}_{\text{select}}$ by integrating three components:

- **Target Distribution** $\mathbf{P}_{\text{target}}$: A preference-weighted distribution specified by human experts. Given non-negative weights $\{w_i\}_{i=1}^{n}$ over $n = 27$ capabilities, we obtain

$$\mathbf{P}_{\text{target}} = \text{Normalize}\left([w_1, w_2, \ldots, w_n]\right).$$

- **Current Distribution** $\mathbf{P}_{\text{current}}$: A normalized frequency vector capturing how often each capability has been selected. Let $\mathbf{f} = [f_1, \ldots, f_n]$ denote the accumulated counts (initialized to zero). After each task, the corresponding entry is incremented and normalized:

$$\mathbf{P}_{\text{current}} = \text{Normalize}\left([f_1, \ldots, f_n]\right).$$

- **Contextual Distribution** $\mathbf{P}_{\text{scenario}}$: Estimated via an LLM by ranking all 27 capabilities by relevance to the scenario. Given ranks $(c_1, \ldots, c_n)$, we convert them into a probability distribution via exponential decay:

$$\mathbf{P}_{\text{scenario}} = \text{Normalize}\left(\left[e^{-\lambda(c_i-1)}\right]_{i=1}^{n}\right),$$

  where $\lambda > 0$ controls the decay rate.

To encourage coverage of underrepresented capabilities, we compute a residual term $\mathbf{P}_{\text{residual}} = \mathbf{P}_{\text{target}} - \mathbf{P}_{\text{current}}$. The final selection distribution interpolates between contextual relevance and this residual signal:

$$\mathbf{P}_{\text{select}} = (1-\alpha)\,\mathbf{P}_{\text{scenario}} + \alpha\,\mathbf{P}_{\text{residual}},$$

where $\alpha \in [0, 1]$ controls the trade-off. We select the top two capabilities with the highest probabilities under $\mathbf{P}_{\text{select}}$.

Task Specification. Given the context and prioritized capabilities, the LLM instantiates a concrete task (e.g., "Verifying Verbal Lease Validity") along with an ideally suited question format. Similarity-based filtering is then applied to remove redundancies, ensuring diversity in the final benchmark.

3.2 Context Enrichment

After task discovery, the initial task corpus often offers limited context, and direct prompting yields shallow or repetitive expansions. To mitigate this, we employ iterative Socratic dialogue between teacher and student LLM agents. Through mutual questioning and exploration of alternatives, the agents surface implicit assumptions, refine conceptual boundaries, and uncover deeper task-relevant insights. A summarization agent then distills the dialogue into concise knowledge points used to perform dense retrieval of the top-k relevant passages, which are appended to the task corpus. The loop terminates once no new passages emerge or after m rounds, balancing depth and cost. By expanding the task corpus via reasoning rather than simple keyword matching, our method ensures a semantically rich and task-aligned corpus, enabling high-quality instance synthesis.

3.3 Instance Synthesis

Given each task's designated capabilities and enriched knowledge context, we iteratively synthesize evaluation instances using a four-agent Socratic framework. Each round proceeds as follows:

1. *Teacher* poses a question grounded in the task and its context.
2. *Student* provides an initial response.
3. *Reviewer* assesses the response. If it meets quality criteria (e.g., coherence, accuracy, and completeness), the *Reviewer* instructs the *Teacher* to escalate difficulty in the next round.
4. If the response is incomplete or incorrect, the system triggers a *context augmentation* step: additional passages are retrieved using the *Reviewer*'s feedback as a query, and the student revises the answer with the expanded context.
5. A *Meta-detector* monitors for task drift, redundancy, or uninformative exchanges and terminates the dialogue when such patterns arise.

Question type	Count
Multiple Choice Question (MCQ)	4014
True/False Question (T/F)	1035
Question Answering (QA)	2625
Classification (C)	2344
Total	10018

Table 1: Instance statistics of **AutoLegalEval**.

Model	Score
claude-sonnet-4	**0.79**
gpt-4.1	0.78
gemini-2.5-pro	0.74
Qwen2.5-72B-Instruct	0.72
Qwen2.5-7B-Instruct	0.6
Qwen2.5-3B-Instruct	0.52
Qwen2.5-1.5B-Instruct	0.42
Gemma-3-27B-it	0.65
Gemma-3-12B-it	0.61
Gemma-3-4B-it	0.55
Meta-Llama-3-70B-Instruct	0.68
Meta-Llama-3-8B-Instruct	0.44

Table 2: Model performance on **AutoLegalEval**.

Pearson	Spearman	Kendall τ	MAE
0.819	0.754	0.694	0.107

Table 3: Agreement between model ratings and human judgments.

The loop continues until halted by the *Meta-detector* or after n rounds. Non-convergent trajectories are discarded to preserve benchmark quality. Remaining instances then undergo a final filtering stage based on predefined criteria (see Appendix I), providing an additional quality screen. Through this structured, feedback-driven pipeline, AUTOTASKEVAL produces diverse, challenging, and cognitively aligned evaluation instances without human involvement.

3.4 Instantiation in Legal Domain

We demonstrate our framework primarily in the legal domain, which demands complex reasoning and domain-specialized knowledge. While the framework is domain-agnostic and later applied to a low-structure e-commerce corpus (Section 4.5), we use the legal domain as the main instantiation to illustrate each stage of the pipeline.

Corpus. We curated legal texts from pile-of-law (Henderson et al., 2022) and extracted 316,495 context paragraphs as seed contexts. These serve as the grounding substrate for **Task Discovery** and **Instance Synthesis**.

Synthesized Data. We generated 200 candidate tasks via **Task Discovery** (examples in Appendix K.1). After filtering for clarity and redundancy, 176 high-quality tasks were retained, spanning diverse legal subdomains and capabilities. For each task, 10 seed contexts were retrieved through **Context Enrichment** and used for **Instance Synthesis** with up to 4 refinement rounds. The final benchmark comprises 10,018 examples (Table 1) across multiple format including open-ended QA, multiple-choice, true/false, and classification formats, enabling structural diversity and fine-grained capability coverage. We refer to the resulting legal-domain benchmark as **AutoLegalEval**.

Model Evaluation. We evaluate multiple LLMs using tailored evaluation metrics: accuracy for multiple-choice and true/false items, and macro-F1 for classification to address label imbalance. QA tasks are evaluated using a combined rubric: binary key-point matching and 5-point Likert ratings on clarity, accuracy, and reasoning depth, which are integrated into a final QA score. Task-level results are aggregated into capability profiles, where each capability score reflects mean model performance across its associated tasks (Figure 6).

4 Experiments

4.1 Setup

We employ Qwen2.5-72B-Instruct as the primary LLM agent and all-mpnet-base-v2 (Reimers and Gurevych, 2019) for *K-center greedy sampling* in Sec. 3.1 and retrieval in Secs. 3.2, 3.3. To enhance diversity during **Task Discovery**, we apply *k-center greedy sampling* with $k = 200$. The maximum iteration counts for **Context Enrichment** and **Instance Synthesis** are set to $m = 2$ and $n = 4$. To model the target capability distribution, we obtain expert ratings that reflect practical priorities in the legal domain (Appendix D). This rating-based formulation offers a more flexible and scalable alternative to directly requesting experts to design tasks. To align the discovered tasks with the desired capability distribution, we set $\lambda = \alpha = 0.2$, as defined in Section 3.1. We assess three closed-source LLMs (GPT-4.1 (OpenAI et al., 2024), Gemini-2.5-Pro (DeepMind, 2025), and Claude-Sonnet-4 (Anthropic, 2025)) and nine open-source instruction-tuned LLMs. The open-source models include

	LawBench	LegalBench	Merged	Ours
diversity	2.81	2.28	2.50	3.90

Table 4: Comparison of task **diversity** across legal benchmarks.

Qwen2.5-Instruct (Team, 2024) (1.5B, 3B, 7B, 72B), Gemma-3-IT (Team, 2025) (4B, 12B, 27B), and LLaMA3-Instruct (Meta, 2025) (8B, 70B).

4.2 Evaluation Results on AutoLegalEval

Table 2 presents model performance on our automatically constructed benchmark **AutoLegalEval**. Within each model family, performance consistently improves with size, confirming that scale is a key factor in handling complex generated tasks. Notably, Gemma-3-12B-it scores 0.61, rising modestly to 0.65 for Gemma-3-27B-it, indicating diminishing returns at larger scales. The highest score of 0.79 shows that our dataset mitigates typical saturation effects and offers a more discriminative benchmark for evaluating future LLM advancements.

Evaluation Reliability. Since automated scoring is required to support end-to-end evaluation, we assess the reliability of the LLM-as-a-judge by measuring its alignment with human judgments. We randomly sampled 100 generated instances and had domain annotators rate them along four dimensions: Correctness, Completeness, Clarity, and Difficulty, following a standardized annotation protocol (see Appendix L.2 for details). Correlation analyses (Table 3) show strong agreement between LLM and human ratings with a low MAE of 0.107. These results confirm that automatic scoring is reliable and that the reported model rankings faithfully reflect underlying performance differences.

4.3 Analysis of Framework Properties

4.3.1 Task Coverage and Diversity

We first evaluate whether the framework can automatically discover a broad and semantically diverse set of legal tasks. To assess coverage, we compare the semantic distribution of discovered tasks with two representative human-curated benchmarks: LegalBench (Guha et al., 2023) and LawBench (Fei et al., 2023). All tasks are embedded using all-mpnet-base-v2 (Reimers and Gurevych, 2019) and visualized via t-SNE (Maaten and Hinton, 2008). As shown in Figure 3, tasks produced by our method occupy a substantially larger portion of the semantic space, including regions sparsely

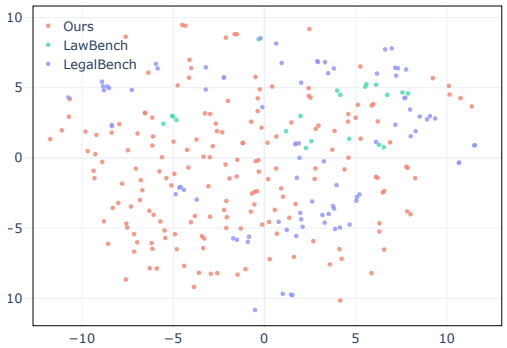

Figure 3: t-SNE visualization comparing task coverage of **LegalBench**, **LawBench**, and **AutoLegalEval**, showing broader semantic and topical diversity in our benchmark.

represented or entirely absent in existing benchmarks. This indicates that automated task discovery uncovers a richer and more fine-grained task landscape than human-engineered collections.

To quantitatively assess diversity, we compute the Shannon entropy of capability distributions by aggregating capability labels across tasks. We refer to this metric as **diversity**. A higher value reflects more uniform and comprehensive coverage. Table 4 shows that our benchmark obtains a diversity of 3.90, markedly exceeding LawBench (2.81), LegalBench (2.28), and their union (2.50). This confirms that the discovered tasks do not collapse around a small set of dominant competencies but instead span a broad range of legal skills.

Finally, using a taxonomy of ten task categories, we manually classified tasks from our benchmark and both baselines. Our dataset covers all categories with substantially higher and more evenly distributed frequency, whereas LawBench and LegalBench exhibit gaps in multiple areas. Detailed category definitions and full comparison results are provided in Appendix C.1. Together, these results demonstrate that **Task Discovery** generates significantly more diverse and balanced legal tasks than human-curated benchmarks, enabling broader and more informative capability evaluation.

4.3.2 Iterative Difficulty Progression

Having established that AUTOTASKEVAL discovers a broad and diverse task space, we next examine whether it also produces progressively harder evaluation instances. While diversity determines the breadth of coverage, difficulty progression indicates how effectively a benchmark probes model limits. We use model accuracy as a proxy for instance difficulty, where lower accuracy reflects

Task	Article Prediction (Scenario-Based)					Charge Prediction				
Model	Lawbench	ZSB	BenchAgents	Ours	Ours-hard	Lawbench	ZSB	BenchAgents	Ours	Ours-hard
claude-sonnet-4	38.75	46.12	51.15	40.73	**35.66**	42.67	64.93	49.73	40.19	**37.36**
gpt-4.1	38.67	44.37	56.07	41.22	**36.73**	43.33	62.64	47.99	41.61	**36.92**
gemini-2.5-pro	**37.68**	47.23	53.47	42.64	37.96	44.0	67.08	47.78	41.85	**38.8**
Qwen2.5-72B-Instruct	33.42	44.47	49.43	37.29	**31.08**	39.63	59.97	42.85	35.64	**30.57**
Qwen2.5-7B-Instruct	23.87	27.31	32.05	22.36	**17.67**	35.75	52.43	29.7	25.66	**16.97**
Qwen2.5-3B-Instruct	17.57	19.16	18.75	19.12	**14.59**	29.08	48.05	30.16	24.42	**17.46**
Qwen2.5-1.5B-Instruct	19.99	32.27	33.24	17.56	**13.64**	22.58	36.22	25.16	21.56	**14.35**
Meta-Llama-3-70B-Instruct	32.84	29.66	29.63	27.22	**21.42**	**23.6**	54.79	32.02	30.34	27.1
Meta-Llama-3-8B-Instruct	27.75	23.09	19.9	20.56	**14.8**	7.6	27.59	11.41	7.71	**4.73**
Gemma-3-27B-it	24.4	28.5	31.07	22.93	**20.48**	20.87	53.39	25.7	21.91	**18.53**
Gemma-3-12B-it	23.38	25.01	27.53	21.97	**20.24**	**15.63**	52.67	24.16	22.44	19.34
Gemma-3-4B-it	21.86	21.87	22.64	16.57	**12.18**	7.61	38.42	13.51	9.25	**6.86**

Table 5: Generation Methods Comparison Results. ***LawBench*** denotes the original datasets for each task. ***ZSB*** and ***BenchAgents*** represent datasets derived from the Zero-shot Benchmarking and BenchAgents pipelines. ***Ours*** corresponds to **AutoLegalEval**, while ***Ours-Hard*** represents the most challenging subset selected from the hardest instance refinement iteration. Bolded scores indicate the lowest model performance, reflecting the most challenging data generation method.

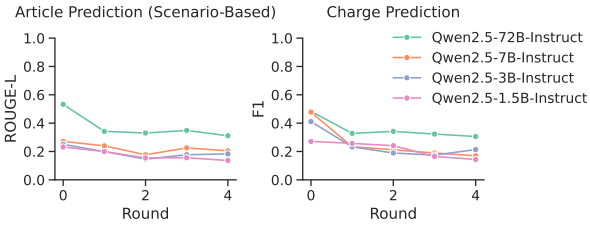

Figure 4: Iterative refinement in AUTOTASKEVAL leads to progressively harder instances, as reflected by declining model performance.

higher complexity.

Analysis of Difficulty Evolution. To assess whether instance difficulty increases across refinement rounds, we track model performance over iterations. As shown in Figure 4 for two representative LawBench tasks, model accuracy exhibits a clear downward trend, indicating that later rounds generally produce more challenging instances. Additional results are provided in Appendix C.3. While minor performance upticks occur in some later rounds (e.g., rounds 3–4), closer inspection reveals that these stem from semantically redundant variants, particularly in narrow tasks, which temporarily weaken adversarial pressure. Despite such non-monotonicity, the overall trend confirms that iterative synthesis effectively yields progressively harder evaluation instances. We further analyze instance quality in Appendix C.4 and diagnostic value of harder instances in Appendix C.6.

4.3.3 Capability-Sensitive Evaluation

We next assess whether the generated instances support capability-sensitive model evaluation.

Alignment with Human-Curated Benchmark. To test whether the generated instances capture skill-specific performance differences in a manner consistent with human-curated datasets. We use ***LawBench*** as a reference and select four representative tasks: Dispute Focus Identification, Article Prediction (Scenario-Based), Charge Prediction, and Legal Case Analysis. Instances are generated using the same procedure as in Step 3 (Sec. 3.3). If model performance on our instances mirrors performance on LawBench, this suggests that our method yields capability-sensitive evaluation data that faithfully preserve task-specific skill hierarchies. Detailed task information is provided in Appendix E.

Comparison with Baselines. We further benchmark AUTOTASKEVAL against two representative automatic generation approaches: Zero-shot Benchmarking (***ZSB***) and ***BenchAgents***, under matched task types and instance counts (Appendix G). Multiple model sizes are evaluated to test consistency and sensitivity to task difficulty across generators.

Results. Table 5 presents performance across nine language models on two representative tasks from each benchmark, with results for the remaining tasks provided in Appendix C.2. Model rankings on our generated instances exhibit strong agreement with those on ***LawBench***, yielding a Spearman correlation of 0.9. This alignment indicates that AUTOTASKEVAL accurately preserves task-specific difficulty structure and supports capability-sensitive, fully automated legal evaluation. In terms of difficulty, instances produced by AUTOTASKEVAL are markedly more

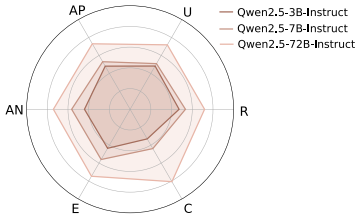

Figure 5: Capability Comparison of Qwen2.5 Models by Bloom's Taxonomy.

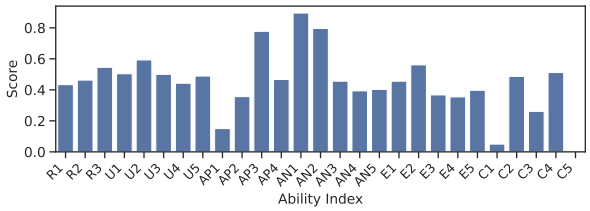

Figure 6: Model capability profile of Qwen2.5-7B-Instruct across bloom-based meta-capabilities.

challenging than those generated by ***ZSB*** and ***BenchAgents***, reaching difficulty levels comparable to the expert-curated ***LawBench***. Moreover, the ***Ours-Hard*** subset, constructed by selecting the most challenging instances through iterative refinement, exceeds the difficulty of LawBench, demonstrating the framework's ability to synthesize genuinely high-complexity cases.

4.4 Hierarchical Capability Analysis

Scaling across Bloom's Levels. Figure 5 shows Qwen2.5 performance across six Bloom cognitive levels. Performance improves with model size, with the largest gains at the Create level, indicating that larger models better handle complex generative tasks, while smaller models remain limited in abstraction, synthesis, and creative reasoning.

Fine-grained Capability Profiling. Furthermore, we evaluate Qwen2.5-7B-Instruct across Bloom's cognitive skills. The model performs well in analytical reasoning (AN) but lags in Create (C) subcategories, particularly C1: Ideation and C5: Prototyping and Refinement, highlighting areas for improvement in tasks requiring original idea generation and expressive outputs.

4.5 Cross-Domain Generalization

To assess whether our framework generalizes beyond domains with well-defined ontologies, we conduct a cross-domain study in the e-commerce product-review domain. Unlike law, this domain is characterized by noisy, subjective, and highly heterogeneous user-generated content, with no standardized taxonomies or stable task structures. These properties make it an ideal stress test for evaluating whether AUTOTASKEVAL can discover coherent tasks and generate meaningful evaluation data in low-structure environments. A detailed description of the setup used to construct **AutoCommerceEval** is provided in Appendix C.7.

Model	CFR	FIA	CIA
Qwen2.5-72B-Instruct	0.85	0.83	0.76
Qwen2.5-32B-Instruct	0.78	0.77	0.72
Qwen2.5-7B-Instruct	0.73	0.75	0.70
Qwen2.5-3B-Instruct	0.67	0.64	0.67
gemma-3-27b-it	0.8	0.84	0.75
gemma-3-12b-it	0.77	0.81	0.70
gemma-3-4b-it	0.71	0.69	0.66
llama-3-70b-instruct	0.82	0.78	0.81
llama-3-8b-instruct	0.77	0.72	0.76

Table 6: Model performance on the three evaluation tasks from **AutoCommerceEval**: CFR = *Customer Feedback Response*, FIA = *Feature Implication Analysis*, CIA = *Concern Impact Analysis*.

Task Diversity. To quantify capability coverage, we compute Shannon entropy over the capability distribution of discovered tasks. The **diversity** for the discovered e-commerce tasks reaches 4.36, exceeding that of structured legal benchmarks and indicating broad, non-collapsed capability coverage despite the domain's unstructured nature.

Ranking Consistency. To verify the discriminative capacity of the generated instances, we evaluate LLM families on three representative tasks, sampling 60 instances per task. Results in Table 6 show that performance scales monotonically with model size across all tested domains. Such consistency underscores the framework's ability to provide a granular assessment of model capabilities, faithfully reflecting scaling effects even in minimally structured domains.

These results demonstrate that AUTOTASKEVAL generalizes well to ontology-light settings, preserving task diversity, difficulty sensitivity, and evaluation reliability beyond the legal domain.

5 Conclusion

In this paper, we propose AUTOTASKEVAL, an automated framework for fine-grained, domain-adaptive evaluation of LLMs. The framework enables taxonomy guided task discovery from unstructured texts, reducing manual effort and sup-

porting scalable capability assessment. Through comprehensive experiments, we show that AUTOTASKEVAL builds high-quality, domain-specific benchmarks with open-source models, supporting systematic evaluation across LLM series and scales, and yielding novel insights into model capabilities and limitations.

Limitations

There are multiple ways for further improvement of this work to alleviate the following limitations:

- While our method generates a diverse and comprehensive task space, the tasks are primarily grounded in the model's own knowledge and experience. Consequently, their alignment with real-world requirements or practical scenarios cannot be guaranteed. This model-centric reliance may introduce biases or omit critical aspects relevant to realistic settings. Future work could integrate real-world user data, domain expert feedback, or human-in-the-loop validation to enhance both the ecological validity and practical relevance of the generated tasks.
- Our approach employs an iterative instance synthesis process, supervised by a Reviewer agent to identify potential errors at each step. Nonetheless, the Reviewer cannot ensure perfect error detection, and undetected mistakes in early iterations may propagate and amplify through subsequent rounds. This limitation may lead to flawed or semantically inconsistent instances, potentially impacting evaluation reliability. Future work could investigate more robust verification mechanisms, ensemble detection strategies, or selective human-in-the-loop oversight to mitigate error accumulation.
- The quality of generated tasks and instances is inherently tied to the underlying corpora. If the source data is noisy, incomplete, or lacks sufficient coverage, the resulting benchmark may inherit these limitations. Future work could explore automated retrieval from open dataset repositories or domain-specific sources to enhance the robustness, representativeness, and overall reliability of the generated dataset.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (No. 62376245), the Key Research and Development Program of Zhejiang Province, China (No. 2024C01034), China Knowledge Centre for Engineering Sciences and Technology (CKCEST-2022-1-7), and MOE Engineering Research Center of Digital Library.

References

Anthropic. 2025. Claude sonnet 4. https://huggingface.co/CometAPI/Claude_Sonnet4. Accessed: 2025-10-06.

Natasha Butt, Varun Chandrasekaran, Neel Joshi, Besmira Nushi, and Vidhisha Balachandran. 2024. Benchagents: Automated benchmark creation with agent interaction. *Preprint*, arXiv:2410.22584.

Yulong Chen, Yang Liu, Jianhao Yan, Xuefeng Bai, Ming Zhong, Yinghao Yang, Ziyi Yang, Chenguang Zhu, and Yue Zhang. 2024. See what llms cannot answer: A self-challenge framework for uncovering llm weaknesses. *Preprint*, arXiv:2408.08978.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. Chatbot arena: An open platform for evaluating llms by human preference. *Preprint*, arXiv:2403.04132.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Google DeepMind. 2025. Gemini 2.5 pro: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. https://arxiv.org/abs/2507.06261. Accessed: 2025-10-06.

Hu Ding, Haikuo Yu, and Zixiu Wang. 2019. Greedy strategy works for k-center clustering with outliers and coreset construction. *arXiv preprint arXiv:1901.08219*.

Zhiwei Fei, Xiaoyu Shen, Dawei Zhu, Fengzhe Zhou, Zhuo Han, Songyang Zhang, Kai Chen, Zongwen Shen, and Jidong Ge. 2023. Lawbench: Benchmarking legal knowledge of large language models. *Preprint*, arXiv:2309.16289.

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, Adam Chilton, Aditya Narayana, Alex Chohlas-Wood, Austin Peters, Brandon Waldon,

Daniel N. Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory M. Dickinson, Haggai Porat, Jason Hegland, and 21 others. 2023. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Preprint*, arXiv:2308.11462.

Peter Henderson, Mark S. Krass, Lucia Zheng, Neel Guha, Christopher D. Manning, Dan Jurafsky, and Daniel E. Ho. 2022. Pile of law: Learning responsible data filtering from the law and a 256gb open-source legal dataset. *Preprint*, arXiv:2207.00220.

Kunze Li and Yu Zhang. 2024. Planning first, question second: An LLM-guided method for controllable question generation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4715–4729, Bangkok, Thailand. Association for Computational Linguistics.

Xiang Lisa Li, Farzaan Kaiyom, Evan Zheran Liu, Yifan Mai, Percy Liang, and Tatsunori Hashimoto. 2025. Autobencher: Towards declarative benchmark construction. *Preprint*, arXiv:2407.08351.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605.

AI Meta. 2025. The llama 4 herd: The beginning of a new era of natively multimodal ai innovation. *https://ai. meta. com/blog/llama-4-multimodal-intelligence/, checked on*, 4(7):2025.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

David Owen. 2024. How predictable is language model benchmark performance? *Preprint*, arXiv:2401.04757.

Arkil Patel, Siva Reddy, and Dzmitry Bahdanau. 2025. How to get your llm to generate challenging problems for evaluation. *Preprint*, arXiv:2502.14678.

José Pombal, Nuno M. Guerreiro, Ricardo Rei, and André F. T. Martins. 2025. Zero-shot benchmarking: A framework for flexible and scalable automatic evaluation of language models. *Preprint*, arXiv:2504.01001.

Jingyuan Qi, Zhiyang Xu, Ying Shen, Minqian Liu, Di Jin, Qifan Wang, and Lifu Huang. 2023. The art of SOCRATIC QUESTIONING: Recursive thinking with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4177–4199, Singapore. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Aymeric Roucher. 2023. amazon$_{p}roduct_{r}eviews_{d}atafiniti$..

Vedant Shah, Dingli Yu, Kaifeng Lyu, Simon Park, Jiatong Yu, Yinghui He, Nan Rosemary Ke, Michael Mozer, Yoshua Bengio, Sanjeev Arora, and Anirudh Goyal. 2025. Ai-assisted generation of difficult math questions. *Preprint*, arXiv:2407.21009.

Gemma Team. 2025. Gemma 3.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Fang-Fang Zhao, Han-Jie He, Jia-Jian Liang, Jingyun Cen, Yun Wang, Hongjie Lin, Feifei Chen, Tai-Ping Li, Jian-Feng Yang, Lan Chen, and 1 others. 2025a. Benchmarking the performance of large language models in uveitis: a comparative analysis of chatgpt-3.5, chatgpt-4.0, google gemini, and anthropic claude3. *Eye*, 39(6):1132–1137.

Yilun Zhao, Kaiyan Zhang, Tiansheng Hu, Sihong Wu, Ronan Le Bras, Taira Anderson, Jonathan Bragg, Joseph Chee Chang, Jesse Dodge, Matt Latzke, Yixin Liu, Charles McGrady, Xiangru Tang, Zihang Wang, Chen Zhao, Hannaneh Hajishirzi, Doug Downey, and Arman Cohan. 2025b. Sciarena: An open evaluation platform for foundation models in scientific literature tasks. *Preprint*, arXiv:2507.01001.

APPENDIX

A Definition of Key Concepts

For clarity and consistency, we define the core terms used throughout this work:

Sub-capabilities Fine-grained cognitive skills that operationalize Bloom's Taxonomy into 27 distinct, actionable capability dimensions. Each sub-capability specifies a concrete reasoning or generation skill (e.g., Ideation, Comparative Analysis, Error Detection), and serves as the atomic unit for capability-aligned task discovery. Detailed descriptions of all sub-capabilities are provided in Table A.1.

Context A domain-specific textual corpus (e.g., books, papers, reports) that serves as the primary knowledge source for task discovery and scenario construction.

Scenario A semantic setting that captures real-world usage patterns in the target domain. Scenarios are summarized by the LLM from the context and guide capability selection and downstream task generation.

Task A domain-specific evaluation unit identified during the Task Discovery stage. Each task includes a task name, description, question type, and its targeted capability dimensions.

Instance A concrete example under a given task, consisting of a question and its answer. Instances are produced during the instance synthesis stage.

k-center Greedy Algorithm (Ding et al., 2019) A classical greedy coverage algorithm used to select representative samples from the corpus, maximizing diversity in the extracted context.

B Prompts

This section mainly supplements the prompts used in our AUTOTASKEVAL. The parts enclosed in curly brackets indicate the required input. For example, "{context}" represents the input context.

B.1 Prompts for Task Discovery

Create Scenarios

```
System Prompt
You are an expert in designing scenarios in the
domain of domain. Your task is to generate a
concise but realistic legal scenario that could be
the basis for a legal reasoning or comprehension
task. The scenario should involvess a real-world
legal issue, such as a dispute, regulatory
decision, contractual clause, or judicial dilemma.
The scenario implicitly contains:
- The involved parties (e.g., individual, company,
agency);
- The situation or triggering event;
- The key legal issue.
You can draw inspiration from following context:
```

Meta-capability	Description
R1: Fact Recall	Retrieving basic facts or data points
R2: Terminology Recognition	Recognizing terms, labels, and definitions
R3: Familiarity Detection	Recognizing previously encountered configurations or patterns
U1: Literal Comprehension	Understanding meaning at a surface level
U2: Paraphrasing	Rewriting or explaining in different terms
U3: Conceptual Matching	Mapping new input to known conceptual categories
U4: Contextual Understanding	Interpreting meaning based on context
U5: Pattern Recognition	Identifying structures or recurring conceptual patterns
AP1: Procedure Execution	Following known procedures or algorithms
AP2: Knowledge Transfer	Applying known methods to new but similar problems
AP3: Contextual Application	Adjusting known methods to fit situational demands
AP4: Tool Use	Appropriately using conceptual or computational tools
AN1: Decomposition	Breaking down a complex input into components
AN2: Relational Mapping	Understanding how parts relate to each other
AN3: Discriminative Reasoning	Distinguishing between relevant and irrelevant information
AN4: Multi-factor Reasoning	Handling interactions between multiple variables
AN5: Model Abstraction	Formulating abstract representations from complex systems
E1: Claim Validation	Assessing the truth, reliability, or credibility of a statement
E2: Criteria-Based Comparison	Judging options based on explicit standards
E3: Bias Detection	Identifying assumptions or logical fallacies
E4: Uncertainty Management	Making decisions under incomplete information
E5: Value Appraisal	Assessing trade-offs and prioritizing based on goals or values
C1: Ideation	Generating new ideas or alternatives
C2: Pattern Synthesis	Combining concepts to create novel structures
C3: Hypothesis Formation	Proposing explanations or models
C4: Creative Expression	Producing original content beyond recombination
C5: Prototyping & Refinement	Iteratively testing and improving generated outputs

Table A.1: Fine-grained descriptions of the 27 sub-capabilities used in our meta-capability taxonomy.

```
context
Note:
- If the context does not contain enough information to form a proper scenario or is irrelevant to the given domain (domain), simply output "pass" and nothing else.
- The context is for inspiration only. DO NOT reuse any specific names, wording, facts, or legal references from it.
Output requirements: Several sentences in plain text, maximum 200 words.
Here's an example: "Luna signed a 3-year lease for a retail shop, with a clause requiring the landlord to handle major repairs. In the second year, the roof began leaking repeatedly, damaging Luna's merchandise. Despite multiple repair requests, the landlord only made minimal fixes. Facing ongoing losses, Luna decided to terminate the lease early and seek compensation. The landlord argues that Luna had no right to unilaterally terminate the contract and that the losses stemmed from poor business performance."
```

Rank Meta-Capabilities

```
System Prompt
You are an expert in task design. Your goal is to create evaluation tasks in the domain of domain that assess a model's general cognitive abilities.
Given the scenario below, select the top-k most relevant capabilities from the list provided. These capabilities are abstract, general-purpose cognitive abilities.
When selecting, prioritize capabilities that:
- Best reflect the core cognitive demands implied
```

by the scenario;
- Involve deeper processing rather than surface-level recall;
- Are more likely to elicit performance differences across models with varying capabilities.
List of 27 Capabilities (with brief descriptions):
R1: Fact Recall, Retrieving basic facts or data points
R2: Terminology Recognition, Recognizing terms, labels, and definitions
R3: Familiarity Detection, Recognizing previously encountered configurations or patterns
U1: Literal Comprehension, Understanding meaning at a surface level
U2: Paraphrasing, Rewriting or explaining in different terms
U3: Conceptual Matching, Mapping new input to known conceptual categories
U4: Contextual Understanding, Interpreting meaning based on context
U5: Pattern Recognition, Identifying structures or recurring conceptual patterns
AP1: Procedure Execution, Following known procedures or algorithms
AP2: Knowledge Transfer, Applying known methods to new but similar problems
AP3: Contextual Application, Adjusting known methods to fit situational demands
AP4: Tool Use, Appropriately using conceptual or computational tools
AN1: Decomposition, Breaking down a complex input into components
AN2: Relational Mapping, Understanding how parts relate to each other
AN3: Discriminative Reasoning, Distinguishing between relevant and irrelevant information
AN4: Multi-factor Reasoning, Handling interactions between multiple variables
AN5: Model Abstraction, Formulating abstract representations from complex systems
E1: Claim Validation, Assessing the truth, reliability, or credibility of a statement
E2: Criteria-Based Comparison, Judging options based on explicit standards
E3: Bias Detection, Identifying assumptions or logical fallacies
E4: Uncertainty Management, Making decisions under incomplete information
E5: Value Appraisal, Assessing trade-offs and prioritizing based on goals or values
C1: Ideation, Generating new ideas or alternatives
C2: Pattern Synthesis, Combining concepts to create novel structures
C3: Hypothesis Formation, Proposing explanations or models
C4: Creative Expression, Producing original content beyond recombination
C5: Prototyping & Refinement, Iteratively testing and improving generated outputs
Scenario: context
Output the indexes of top-k most relevant capabilities, separated by commas (","). DO NOT make up indexes. Choose the capabilities listed ablove. DO NOT make up indexes.
Try to avoid overused capabilities. Currently underrepresented ones should be prioritized, unless clearly irrelevant to the context.

Create Tasks

System Prompt

You are an expert in task design. Your goal is to create an evaluation task that assesses a model's performance in the domain of domain.
You are given:
- A real-world scenario in the domain of domain.
- Two abstract, general-purpose cognitive capabilities selected for evaluation.
Important notes:
- Your task should be independent, realistic, and not mention models, evaluation purposes or .
- Think of it as designing a task a human professional (e.g., lawyer, analyst, judge) might be expected to complete in practice.
Your goals:
1. From the following {len(candidate_question_types)} question types: {', '.join(candidate_question_types)}, choose only one that best suits the task you create. Do not invent or use any other types.
2. Write a one-sentence task description that clearly defines what the task is about and what should be done. The task should:
- Require the use of both provided cognitive capabilities;
- Reflect a realistic, non-trivial challenge within the scenario;
- Be capable of distinguishing between strong and weak reasoning or decision-making;
- Do not refer to specific names, places, or details from the scenario. The task description should be independent and generalized;
- Avoid using words like "model", "evaluate", "AI", or any mention of benchmarking.
3. Create a concise and specific task name that reflects the real-world nature of the task. The name should distinguish the task from other similar tasks and remain suitable for use in academic or benchmark settings.
4. Clearly define the expected input and output format:
- input_format: What kind of content or information will be given to the model;
- output_format: What kind of structured output is expected in response.
5. Please do not generate tasks related to program execution. Focus on tasks involving text generation and legal reasoning.
Output format:
{{"question_type": <one of: {' / '.join(candidate_question_types)}>, "task_description": "<a one-sentence brief description of the task>", "task_name": "<a concise and meaningful name>", "input_format": "<briefly describe what the model receives as input>", "output_format": "<briefly describe the expected output format>"}}
Here's an example: {{"question_type": "Multi-Label Classification", "task_description": "Predict the likely criminal charges based on the factual description of a legal case", "task_name": "Charge Prediction", "input_format": "A paragraph describing the events and facts related to a criminal case.", "output_format": "A list of one or more predicted charges, e.g., [Ḟraud, Öbstruction of Justice]"}} Scenario: {context}
Target Capabilities:
1. capabilities_dict[picked_capabilities[0]]
2. capabilities_dict[picked_capabilities[1]]

Ensure the output is valid JSON. Escape any quotation marks inside strings using ‘\”‘ if necessary.
Output json:

B.2 Prompts for Context Enrichment

Teacher: Clarify the Statement

System Prompt
You are an expert in domain. Your goal is to initiate an in-depth discussion under the task setting: "{sub_task_name}", based on the following context passage: "{context}".
Follow these steps then provide your answer:
Step 1: Carefully read the context and identify not only key concepts, but also any implicit assumptions, methods, or conclusions related to the task.
Step 2: Propose a clear and concise viewpoint about them, and align with the domain: domain. Your aim is to raise a point that could spark a meaningful response.
Step 3: Justify your viewpoint briefly, using reasoning grounded in the passage. Be precise, avoid vague language, and do not repeat the context.
Be thoughtful, analytical, direct, and avoid speculation. Avoid generic summaries or vague conclusions.
Output format:
Viewpoint: <your claim>
Justification: <your reasoning and evidence>

Student: Think on the Statement

System Prompt
You are an expert in domain, engaging in a constructive debate with another expert.
You are given:
- Domain: "domain"
- The task setting: "sub_task_name"
- A context passage related to the task: "context"
- An opinion from the opponent: "old_statement"
Your goal is to critically examine the opponentś viewpoint, and present a distinct perspective which aligns with the domain: domain.
You may challenge their assumptions, offer an alternative interpretation, or highlight overlooked aspects — but only if justified by the passage or domain knowledge. Avoid contradicting facts without evidence. Your disagreement should be reasonable, meaningful, and grounded. Do not oppose for the sake of opposition — if the opponentś point is valid, you can instead answer from another perspective. Be concise, analytical, and accurate.
Output format:
Counterpoint: <your core claim or alternate view>
Justification: <briefly explaining your reasoning and support>

Teacher: Reflect

System Prompt
You are an expert in domain, participating in a critical debate with another expert.
You are given:
- Domain: "domain"
- The task setting: "sub_task_name"
- A context passage related to the task: "context"
- Your original opinion: "old_statement"
- The opponentś opinion: "new_statement"
Your goal is to revise your original opinion in light of the opponentś view. Clearly indicate which parts of their reasoning you accept and integrate, and which parts you respectfully maintain disagreement with. Do not change your position unless proved wrong based on the context or domain knowledge.
Be analytical, concise, and avoid vague statements or repetition. Base your reasoning on evidence from the context or widely accepted knowledge in the field. Do not speculate without support. Do not repeat the full context or both opinions verbatim.
Output format:
Updated View: <your refined or reinforced opinion>
Justification: <briefly explaining your agreement, disagreement, and reasoning>

Student: Reflect

System Prompt
You are an expert in domain, participating in a critical debate with another expert.
You are given:
- Domain: "domain"
- The task setting: "sub_task_name"
- A context passage related to the task: "context"
- Your opponentś view in the first round: "old_statement"
- Your view in the first round: "new_statement"
- Your opponentś view in the second round: "reflect_statement"
Your goal is to revise your original opinion in light of the opponentś view. Clearly indicate which parts of their reasoning you accept and integrate, and which parts you respectfully maintain disagreement with. Do not change your position unless proved wrong based on the context or domain knowledge.
Be analytical, concise, and avoid vague statements or repetition. Base your reasoning on evidence from the context or widely accepted knowledge in the field. Do not speculate without support. Do not repeat the full context or both opinions verbatim.
Output format:
Updated View: <your refined or reinforced opinion>
Justification: <briefly explaining your agreement, disagreement, and reasoning>

Summarize the Statements as Keywords

System Prompt
You are an expert in domain. You are given: - Domain: "domain"
- The task setting: "sub_task_name"
- A context passage related to the task: "context"

- An opinion from one expert: "pro_statement"
- An opinion from another expert: "con_statement"
Your task is to extract up to 8 highly relevant keywords or key phrases that represent the core concepts, methods, or points of disagreement involved in the passage and the two opinions. These keywords should:
- Reflect important domain-specific ideas discussed or implied;
- Cover both the original passage and differing perspectives;
- Be useful for retrieval or organizing related knowledge;
Separate the keywords or phrases with commas. Do not include explanations or full sentences.

B.3 Prompts for Instance Synthesis

Teacher: Raise the Initial Question

System Prompt
You are an expert in domain_adj education and exam design, skilled in crafting high-quality questions based on domain_adj contexts. Given the following input:
- Context: "contexts-plain"
- Task Name: "task_name"
- Task Description: "task_description"
Your job is to generate one question according to the following guidelines: question_prompt
Notes:
1. Do not directly quote anything from the context. If you need to refer to information from the context, paraphrase or naturally integrate it into the scenario.
2. Only generate the question. Do not include the correct answer or any explanations.
3. Output should be in plain text format only. No more than 500 words.

Teacher: Raise the Following Question

System Prompt
You are an expert in domain_adj education and exam design, proficient in crafting high-quality questions based on domain_adj contexts. You are currently participating in an iterative question generation process aimed at producing progressively more challenging domain_adj questions. Your task is to generate a harder question based on a prior round of question.
Given:
- Context: "contexts-plain"
- Task Name: "task_name"
- Task Description: "task_description"
- Original instance generated from the context: "old_question"
- Suggestions for deepening the question: "deepen_question_prompt"
Your job is to generate one question according to the following guidelines: question_prompt
Notes:
1. Do not directly quote anything from the context. Instead, paraphrase or integrate relevant content into the question naturally if needed.
2. The new question should be a deeper extension of the original, increasing cognitive complexity and the effort required to answer.
3. You may use the original answer as a given premise in the new question, but do not refer to it as "the above case". Ensure the new question is standalone and self-contained.
4. Only generate the question. Do not include the correct answer or any explanations.
5. Output should be in plain text format only. No more than 500 words.

Student: Answer the Initial Question

System Prompt
You are an expert in the domain of domain. Given a domain_adj context you may refer to: "contexts-plain"
You need to answer the question under the task: "task_name: task_description"
Note: Output in plain text only. No more than 500 words. Please list only the key points. Respond concisely like a legal advisor.
Question: question
Answer:

Student: Answer the Following Question

System Prompt
You are an expert in the domain of domain. You are given:
- A domain_adj context you may refer to: "contexts-plain"
- An example instance: "Question": old_question, "Answer": old_answer
You need to answer the question under the task: "task_name: task_description"
Note: Output in plain text only. No more than 500 words. Please list only the key points. Respond concisely like a legal advisor.
Question: question
Answer:

Meta-Detector: Check if The Dialog Should Stop

System Prompt
You are an expert in the domain of domain. You are participating in an iterative question-answering process aimed at generating increasingly challenging domain_adj instances. In each round, a new question builds upon the previous one, with the goal of deepening domain_adj reasoning and analytical complexity. Given:
- domain: "domain"
- Task Name: "task_name"
- Task Description: "task_description"
- Context: "contexts-plain"
- Previous instance generated based on the context: "Question": old_question; "Answer": old_answer
- Current instance generated based on the context: "Question": question; "Answer": answer
Your goal:
Decide whether the iterative process should

be terminated based on the following criteria. Termination is warranted if any of the following conditions are met:
1. The instance is irrelevant to the domain or the assigned task;
2. The answer is logically invalid, lacks justification, or deviates from the topic;
3. There are signs of hallucination, contradiction, confusion, or repetition in the instance;
4. The new question does not show any increase in difficulty or depth compared to the previous question.
If you decide that the iterative process be terminated, output "###STOP###" followed by brief reason.
If the iterative process is still valid and can continue, only output "###CONTINUE###".
Do not include any additional explanation or reasoning. Only output one of the two options above.

Reviewer: Score the QA pair

System Prompt
You are an expert evaluator in the domain of domain. Your task is to assess the quality of a question-answer instance based on five defined criteria. Each score should be an integer from 1 (very poor) to 5 (excellent). Given:
- Task Name: "task_name"
- Task Description: "task_description"
- Context: "contexts-plain"
- An instance generated based on the context: "Question": question; "Answer": answer
Your evaluation should score the QA instance on the following five dimensions:
1. Correctness: Does the answer correctly address the domain_adj question based on the context and applicable rules?
2. Completeness: Does the answer fully address all key aspects or sub-issues implied in the question?
3. Clarity: Is the language of both the question and the answer clear, concise, and free of ambiguity?
4. Difficulty: How much reasoning, multi-step analysis, or domain-specific knowledge is required to correctly answer the question?
Assign each dimension a score between 1 (very poor) and 5 (excellent). Output your scores using the exact format below. Do not provide explanations, just the JSON-formatted score.

Output format:
```
{{
  "Correctness": x,
  "Completeness": x,
  "Clarity": x,
  "Difficulty": x
}}
```

Ensure the output is valid JSON.

Reviewer: Check if More Contexts Is Needed

System Prompt
You are an expert in the domain of domain. Given:
- Task Name: "task_name"
- Task Description: "task_description"
- A context relevant to the task: "contexts-plain"
- An instance generated based on the context: "Question": question; "Answer": answer
- Evaluation of the instance: "evaluation"
Your goal is to assess whether the current context provides sufficient information to support a correct and complete answer to the question.
If the existing context is sufficient, output ###YES### only. If the context lacks key information, output ###NO###, followed by up to 8 domain-specific knowledge elements (keywords or phrases) that are missing but necessary for accurately answering the question. Separate them with commas.
Be precise. Focus on concrete domain_adj or domain-relevant knowledge rather than vague or overly general terms. Avoid hallucinated concepts not inferable from the question and task.

Reviewer: Prompt the Teacher to Ask Harder Question

System Prompt
You are a prompt engineer and an expert in the domain of domain. Your task is to design an instruction for another language model, which will be used to generate complex legal instances. You must not generate any questions or answers yourself. Instead, you are to craft a prompt that instructs another model on how to generate a new domain_adj instance that is more complex, more challenging, and more in-depth than the original. Given:
- Domain: "domain"
- Task Name: "task_name"
- Task Description: "task_description"
- A context relevant to the task: "contexts-plain"
- An instance generated based on the context: {{"Question": {question}; "Answer": {answer}}}
- Evaluation of the instance: "evaluation"
Your prompt must take into account the following principles:
1. Provide a clear, actionable, and specific prompt that guides the model to generate a higher-level, domain-relevant instance.
2. Strictly adhere to the task definition "task_name: task_description". If the original instance deviated from the task format, guide the model to refocus accordingly.
3. Address weaknesses identified in the evaluation.
4. Use precise, concise, and execution-ready language, no more than 500 words.
5. Output only the designed prompt. Do not include any sample question, answer, or explanation.
6. Your prompt must focus solely on how to create a more challenging legal question. Do not add unrelated instructions.
7. Format your prompt as a numbered list of clear, specific suggestions (e.g., "1.", "2.", "3.") written in plain text, three suggestions at most.
Here's an example output under the task of "crime charge prediction":
1. The conduct itself has a certain degree

```
of reasonableness or blurred boundaries (e.g.,
overlap between civil and criminal law, confusion
between contract fraud and ordinary breach of
contract);
2. Multiple charges may apply, increasing the
difficulty of legal qualification (e.g., the
boundary between fraud and contract fraud);
3. Adding complex circumstances, such as crimes
committed through third parties, presence of
accomplices with unclear principal-subordinate
relationships, or attempted crimes;
Your Output:
```

C Additional Results

C.1 Manually Categorized Tasks Comparison

To support a controlled comparison across benchmarks, we construct a ten-category taxonomy that captures the major competencies required in legal reasoning and legal NLP tasks. Each benchmark (LawBench, LegalBench, and Ours) was manually annotated by two trained annotators following standardized category definitions. Disagreements were resolved through discussion, and category frequencies were computed at the task level, ensuring cross-benchmark comparability.

Table C.2 reports the distribution of tasks across the ten categories for all three benchmarks. Our automatically discovered benchmark exhibits complete category coverage with substantially higher frequencies in most categories. In contrast, LawBench and LegalBench show sparse or absent representation in several areas, particularly in **Evidence Evaluation and Admissibility**, **Legal Outcome Prediction**, **Liability and Risk Assessment** and **Legal Decision and Action Recommendation**. This broader task landscape supports a more comprehensive and discriminative evaluation of legal reasoning capabilities across models.

C.2 Additional Results for Comparison with Existing Automatic Methods

As shown in Appendix E, the selected tasks from LawBench fall into four categories. Results for *Charge Prediction* and *Scenario-Based Article Prediction* are reported in the main text, while the remaining two tasks are presented in Table C.3.

C.3 Additional Results for Iterative Generation of Difficult Instances

As noted in Section 4.3.2, the trends are particularly pronounced on task *Dispute Focus Identification*. Most models achieve their lowest performance on the second iteration, reflecting the increased difficulty introduced by early-stage iterative refinement. Notably, performance recovers in the third and fourth iterations, which can be attributed to the introduction of redundant information in later iterations that reduces the effective difficulty of some instances.

C.4 Instance Quality Analysis

To evaluate the quality of synthesized instances, we sample 50 instances from each task type and obtain expert annotations using a 5-point Likert scale across four dimensions: Correctness, Completeness, Clarity, and Difficulty. Table C.4 reports the averaged scores. The results indicate that the generated instances maintain high correctness and clarity, with moderate difficulty, an appropriate balance for evaluating fine-grained legal reasoning capabilities. This confirms that our iterative generation process preserves instance quality while producing progressively more challenging evaluation data. To further understand how quality is enforced within this process, we also examine the error types flagged by the Reviewer during instance refinement. A detailed breakdown is provided in Appendix C.5.

C.5 Error Analysis of the Reviewer Agent

The Reviewer agent addresses a key bottleneck in automated data construction: detecting errors in synthesized answers. While not eliminating all failure cases, its utility is supported by the strong agreement between LLM-based scoring and expert ratings (Section 4.2), indicating that the Reviewer provides a practical and sufficiently reliable mechanism for filtering low-quality outputs.

We further analyzed 100 sampled test cases flagged by the Reviewer agent for low Correctness, with the error types summarized as follows:

1. Incorrect statutory references (13 cases): The model cites non-existent or contextually irrelevant statutes, resulting in answers that are inconsistent with the provided legal context.
2. Procedural errors (23 cases): Mistakes in legal procedures, such as confusing the authorities of enforcement versus judicial bodies, or inaccurately describing litigation processes.
3. Reasoning errors (45 cases): Logical flaws in case analysis or interpretation of legal requirements, including superficial analogies or misapplication of legal elements.
4. Conceptual misunderstandings (11 cases): Confusions between distinct legal concepts or prin-

Task Type	Task Description	LawBench	LegalBench	Ours
Legal Reasoning	Analyze case facts, statutes, and precedents	5	6	29
Evidence Evaluation and Admissibility	Assess whether evidence meets legal standards	0	2	12
Compliance and Procedural Assessment	Analyze compliance with laws, policies, or procedures	0	7	21
Contractual and Agreement Reasoning	Interpret contract clauses and assess breaches	1	31	35
Legal Outcome Prediction	Predict legal dispute outcomes based on rules and facts	2	0	30
Liability and Risk Assessment	Analyze potential legal liabilities and risks	1	0	14
Legal Decision and Action Recommendation	Recommend optimal legal actions or strategies	2	0	28
Issue/Fact Identification	Identify legal issues and disputes from facts	4	20	3
Legal Text Generation	Generate, summarize, or edit legal texts	4	7	3
Specialized Legal Knowledge Application	Apply specialized legal knowledge	1	15	1
Total		20	88	176

Table C.2: Coverage of task categories in our dataset versus LawBench and LegalBench.

ciples, leading to inaccurate or misleading answers.

5. Ambiguous or Non-committal Answers (8 cases): Vague or overly qualified responses that avoid making a clear legal judgment. Such answers may appear correct but fail to meet the required decisiveness and clarity of legal tasks.

These findings demonstrate that the Reviewer agent effectively identifies diverse and consequential error types, thereby playing a critical role in maintaining the quality of the constructed evaluation data.

C.6 Error Patterns Revealed by High-Difficulty Instances

To substantiate the evaluation value of iteratively synthesized high-difficulty instances, we further analyze the novel error patterns they uncover. While lower-difficulty instances predominantly expose surface-level issues, such as factual inaccuracies or misapplied procedures, harder instances elicit qualitatively distinct and more diagnostic failure modes that reflect deeper limitations in legal reasoning.

Specifically, higher-difficulty instances reveal the following error categories with markedly higher frequency:

- **Long-context consistency failures**: breakdowns in maintaining coherent tracking of facts, parties, or temporal relations across extended and interleaved case narratives.
- **Multi-statute interaction errors**: failures to jointly reason over multiple interdependent legal provisions, including incorrect resolution of conflicts between general and special rules.
- **Multi-party attribution errors**: misallocation of rights, obligations, or liabilities in scenarios involving multiple actors, hierarchical responsibilities, or indirect agency.

Task	Dispute Focus Identification					Legal Case Analysis				
Model	Lawbench	ZSB	BenchAgents	Ours	Ours-hard	Lawbench	ZSB	BenchAgents	Ours	Ours-hard
Qwen2.5-1.5B-Instruct	27.88	50.67	53.69	45.33	40.0	47.0	75.33	74.67	76.67	60.0
Qwen2.5-3B-Instruct	27.6	55.33	53.02	35.33	26.67	55.6	68.0	74.0	74.67	56.67
Qwen2.5-7B-Instruct	38.0	62.67	73.83	48.0	40.0	66.6	79.33	84.67	74.67	60.0
Qwen2.5-72B-Instruct	47.6	62.42	73.86	71.33	60.0	67.2	92.0	89.61	85.33	80.0
Gemma-3-4B-it	20.8	51.33	51.68	22.67	10.0	14.4	25.33	24.0	12.0	6.67
Gemma-3-12B-it	43.8	64.0	65.77	62.67	56.67	6.8	9.33	17.33	10.04	3.0
Gemma-3-27B-it	41.6	60.0	59.73	57.33	50.0	19.0	46.0	28.67	24.67	13.33
Meta-Llama-3-8B-Instruct	28.0	44.67	36.91	43.33	30.0	30.0	75.33	64.67	62.0	46.67
Meta-Llama-3-70B-Instruct	45.4	59.33	65.1	58.0	50.0	42.6	84.0	82.67	74.0	56.67

Table C.3: Generation Methods Comparison Results. ***LawBench*** denotes the original datasets for each task. ***ZSB*** and ***BenchAgents*** represent datasets derived from the Zero-shot Benchmarking and BenchAgents pipelines. ***Ours*** corresponds to datasets generated via our proposed AUTOTASKEVAL method, and ***Ours-Hard*** indicates the subset composed of the most challenging examples, selected from the hardest instance refinement iteration.

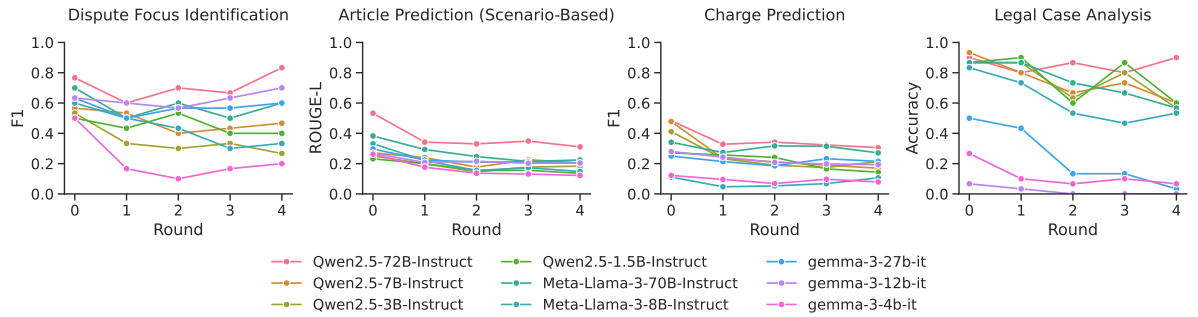

Figure C.1: Additional results for iterative generation of difficult instances. Iterative refinement in AUTOTASKEVAL leads to progressively harder instances, as reflected by declining model performance.

Question Type	Correctness	Completeness	Clarity	Difficulty
MCQ	4.27	-	4.93	3.88
T/F	4.16	-	4.89	3.65
QA	4.67	4.76	4.91	4.04
C	4.13	-	4.92	3.93

Table C.4: Average instance quality scores (1–5) across four task types, rated by human experts on four dimensions.

- **Analogy boundary errors**: over-extension or misapplication of analogical reasoning beyond legally permissible or precedentially supported boundaries.
- **Principle-based reasoning errors**: incomplete or incorrect application of abstract legal principles, such as proportionality, duty of care, balancing tests, or reasonableness standards.

These failure modes rarely surface in low-complexity tasks but become prominent in instances that require multi-step reasoning, integration of multiple norms, or abstraction over legal principles. Their emergence demonstrates that progressively harder instances do not merely reduce model accuracy, but instead probe fundamentally different reasoning weaknesses. This supports the claim that iterative difficulty escalation yields evaluation data with higher diagnostic value, improving both the reliability and coverage of model assessment.

C.7 Experimental Setup for AutoCommerceEval

Using a corpus of approximately 8K product-review contexts from the Amazon Product Reviews (Roucher, 2023) dataset as seed materials, we apply the same pipeline used in the legal domain. The system automatically produced 20 task candidates, which were reduced to 17 unique tasks after deduplication (Table K.2). The resulting task set spans sentiment interpretation, aspect-based reasoning, quality assessment, and user-intent inference, demonstrating that the framework can infer diverse task structures even without explicit domain ontologies. Evaluation instances are then generated following the same refinement and filtering procedures. We refer to the resulting e-commerce benchmark as **AutoCommerceEval**.

D Expert Ratings for Task Discovery

We engaged a legal expert to score the 27 refined meta-capabilities based on Bloom's taxonomy according to their relevance and importance in the legal domain. The scoring principle was straightforward: the more critical a capability is for legal tasks, the higher its score.To guide the scoring process, we established the following criteria:

- High importance (score 8-10): Capabilities essential for accurate legal reasoning, interpretation, or decision-making, frequently required across diverse legal scenarios.
- Moderate importance (score 4-7): Capabilities useful in many legal contexts but not universally critical.
- Low importance (score 1-3): Capabilities occasionally relevant or supportive but not fundamental to legal expertise.

The results of the expert scoring are listed in Table D.1.

However, After evaluating the expert's scores against experimental task distributions and real-world legal practice requirements, we observed that tasks related to the "Remember" category were overrepresented relative to their actual importance. Consequently, we adjusted the weighting by moderately reducing the emphasis on "Remember" meta-capabilities to better align with practical needs.

The final calibrated scores are presented as follows: [3, 3, 2, 7, 5, 6, 10, 6, 5, 6, 8, 3, 8, 9, 10, 9, 5, 10, 8, 6, 7, 5, 3, 5, 6, 2, 4]

E Overview of the Selected LawBench Tasks

The details of four selected tasks from LawBench are presented below.

- Dispute Focus Identification: A single-label classification task evaluated by F1 score. Given a sentence from a legal case, identify the specific type of dispute focus it expresses.
- Article Prediction (Scenario-Based): A text generation task evaluated by ROUGE-L score. Given a specific legal scenario and question, generate the exact legal provision applicable by providing the relevant article text.
- Charge Prediction: A multi-label classification task evaluated by F1 score. Given the facts of a case, simulate a judge by identifying the applicable crime name(s).
- Legal Case Analysis: A multiple-choice task evaluated by Accuracy. Given a legal case and a corresponding question, select the correct answer from 4 candidates.

F Comparison with Related Work

While prior methods have advanced the automation of LLM evaluation, important limitations remain. Interactive platforms such as Chatbot Arena (Chiang et al., 2024) and SciArena (Zhao et al., 2025b) primarily capture human preference rankings in open-ended dialogue, lacking systematic coverage and fine-grained capability assessment. More recent automatic benchmark generation frameworks including Zero-shot Benchmarking (Pombal et al., 2025), BenchAgents (Butt et al., 2024), and AutoBencher (Li et al., 2025) improve task diversity and reduce human effort through prompt-based synthesis or modular agent pipelines. However, as summarized in F.5, these approaches still rely on fixed or declaratively specified task schemas, limiting their adaptability to emerging domains and novel task formulations. Notably, they do not perform autonomous task discovery or organize competencies into structured taxonomies. In contrast, AUTOTASKEVAL provides fully automated, domain-adaptive model evaluation by identifying latent evaluation dimensions in unstructured corpora and generating task-capability pairs that support systematic, fine-grained assessment. This enables controllable difficulty calibration, fine-grained evaluation, and scalable coverage of both known and previously unarticulated model abilities.

G Two Generation Methods Used in Baseline Comparison

We compare our method with the following two automatic generation approaches:

- Zero-Shot Benchmarking ((Pombal et al., 2025)): The Zero-Shot Benchmarking (ZSB) framework offers an innovative solution to the challenges of automatic language model evaluation by leveraging the models themselves to generate diverse test data and perform assessments without relying on human annotations. By using carefully designed meta-prompts for data generation and judgment prompts for evaluation, ZSB can flexibly create large-scale, varied benchmarks across multiple tasks and languages. Its model-agnostic and scalable design allows it to adapt seamlessly to evolving model capabilities and emerging tasks, making it a powerful and efficient ap-

Method	Difficulty Controllable	Automated Benchmark Design	Domain Adaptability	Task Discovery	Fine-grained Evaluation
(Chiang et al., 2024)	✗	✗	✗	✗	✗
(Zhao et al., 2025b)	✗	✗	✓	✗	✗
(Pombal et al., 2025)	✓	✓	✓	✗	✗
(Butt et al., 2024)	✓	✓	✓	✗	✓
(Li et al., 2025)	✓	✓	✓	✗	✓
Ours	✓	✓	✓	✓	✓

Table F.5: Comparison of automatic evaluation frameworks for LLMs.

proach for robust and comprehensive model assessment.

- BenchAgents ((Butt et al., 2024)): BenchAgents is a multi-agent framework that decomposes benchmark creation into four specialized roles: planning, data generation, verification, and evaluation, each handled by dedicated large language model agents. It integrates developer-in-the-loop feedback and combines LLM-driven automation with executable code to efficiently produce high-quality, diverse benchmarks. This hybrid, modular approach enables scalable and controllable generation of complex evaluation datasets with human oversight.

H Pseudocode of Multi-Agent Interaction in Sections 3.2 and 3.3

H.1 Pseudocode of Multi-Agent Interaction in Sections 3.2

The pseudocode for Multi-Agent Interaction in Sections 3.2 is presented in Algorithm 1.

H.2 Pseudocode of Multi-Agent Interaction in Sections 3.3

The pseudocode for Multi-Agent Interaction in Sections 3.3 is presented in Algorithm 2.

I Post-Generation Instance Filtering Rules

As depicted in Section 3.3, each instance is evaluated by models across four dimensions using a Likert scale. Instances receiving a score below 4 on any of Correctness, Completeness, or Clarity are subsequently excluded from the dataset.

J Cost Analysis of Benchmark Construction

In our legal-domain instantiation, the overhead of constructing a benchmark to support automated evaluation is primarily divided into three stages: Task Discovery, context enrichment, and instance synthesis. All generation steps across these stages were performed using the Qwen2.5-72B-Instruct model.

- Task Discovery: This stage begins with sampling 200 contexts to construct domain-specific scenarios. Each scenario is then used to select two target meta-capabilities, followed by task formulation. Except for the capability selection step, which must be performed sequentially, all other operations are parallelized with a concurrency of 8. This stage takes approximately 25 minutes in total.
- context enrichment: For the 176 filtered tasks, additional context samples are retrieved to enrich the task corpus. Each task is augmented with 9 extra context instances, requiring 5 API calls per task. With a concurrency level of 8, this stage completes in around 30 minutes.
- instance synthesis: In total, 1760 context are used to generate test cases. For each instance, a set of 6 progressively difficult examples is generated. Producing one such set requires 36 API calls. The process is parallelized across 4 question types, each with a concurrency of 8. This stage is the most time-consuming, taking approximately 13 hours.

In summary, under our framework, constructing a task-rich benchmark containing 10,018 high-quality evaluation examples takes approximately 14 hours.

K Examples from the Instantiation of AutoTaskEval in the Legal Domain

K.1 Task Examples

We list some of the generated tasks from **AutoLegalEval** along with their details in Table K.1. The statistical distribution of task capabilities was computed and is presented in Figure K.2. Tasks from **AutoCommerceEval** are listed in Table K.2

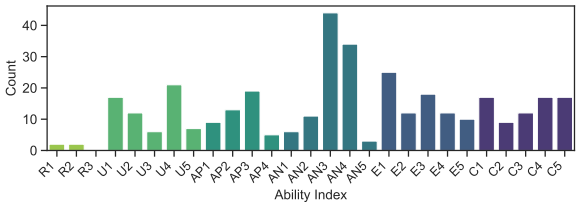

Figure K.2: Capability distribution of **AutoLegalEval** tasks.

K.2 Instance Examples

Task Type: Zoning Ordinance Validity Assessment
Task Description: Determine the most likely outcome of a legal dispute over the validity of a local zoning ordinance in light of a state's comprehensive land use plan.
Question Type: Multiple Choice Question
Assessed Meta-Capabilities: AN5(Model Abstraction), E1(Claim Validation)

Round 0

Question: A city has enacted a new zoning ordinance that restricts the construction of multifamily housing units in certain areas to promote single-family homes. The state's comprehensive land use plan emphasizes the need for more affordable housing and encourages the development of multifamily units to address public housing needs. A local advocacy group challenges the ordinance, arguing that it conflicts with the state's comprehensive land use plan. What is the most likely outcome of this legal dispute? A. The court will uphold the ordinance because local zoning decisions are generally given deference, regardless of state plans. B. The court will strike down the ordinance because it fails to comply with the state's comprehensive land use plan, which prioritizes affordable housing. C. The court will uphold the ordinance if the city can demonstrate that it has a residential antidisplacement and relocation assistance plan in place. D. The court will strike down the ordinance only if the state has explicitly mandated the construction of multifamily units in the comprehensive land use plan.

Answer: B

Deepen Question Prompt: 1. Incorporate multiple layers of legal reasoning by requiring the model to consider both state and federal laws, including the potential for federal preemption and the implications of fair housing certification. 2. Introduce additional stakeholders and competing interests, such as environmental groups, developers, and residents, to create a more nuanced and complex legal scenario. 3. Require the model to evaluate the city's compliance with the state's comprehensive land use plan, including the city's efforts to affirmatively further fair housing and its antidisplacement and relocation assistance plan.

Round 1

Question: A city has enacted a new zoning ordinance that restricts the construction of multifamily housing units in certain areas to promote single-family homes. The state's comprehensive land use plan emphasizes the need for more affordable housing and encourages the development of multifamily units to address public housing needs. Additionally, the city has certified that it will affirmatively further fair housing and has implemented a residential antidisplacement and relocation assistance plan. An environmental group, a local developer, and a residents' association have all filed lawsuits challenging the ordinance. The environmental group argues that the ordinance will exacerbate urban sprawl and increase carbon emissions. The developer claims that the ordinance violates the state's comprehensive land use plan. The residents' association supports the ordinance, citing concerns about increased traffic and strain on local resources. What is the most likely outcome of these legal challenges? A. The court will uphold the ordinance because the city has a residential antidisplacement and relocation assistance plan, which satisfies the state's requirements. B. The court will strike down the ordinance due to its conflict with the state's comprehensive land use plan and the city's certification to affirmatively further fair housing. C. The court will uphold the ordinance if the city can demonstrate that the restrictions are necessary to protect environmental interests and local resources. D. The court will strike down the ordinance only if there is clear evidence of federal preemption or if the state has explicitly mandated the construction of multifamily units in the comprehensive land use plan.

Answer: B

Deepen Question Prompt: 1. Introduce multiple layers of legal reasoning by incorporating additional stakeholders with conflicting interests, such as a state agency, a federal housing organization, and a community advocacy group, each presenting distinct legal arguments.2. Enhance the complexity by including a detailed analysis of the city's compliance with federalism considerations, specifically addressing whether the city's actions are preempted by federal law or if there is a conflict with state mandates.3. Incorporate nuanced scenarios where the city's residential antidisplacement and relocation assistance plan is challenged on grounds of insufficient implementation or effectiveness, requiring the court to assess both procedural and substantive aspects of the plan.

Round 2

Question: A city has enacted a new zoning ordinance that restricts the construction of multifamily housing units in certain areas to promote single-family homes. The state's comprehensive land use plan emphasizes the need for more affordable housing and encourages the development of multifamily units to address public housing needs. Additionally, the city has certified that it will affirmatively further fair housing and has implemented a residential antidisplacement and relocation assistance plan. An environmental group, a local developer, a residents' association, a state housing agency, and a federal housing organization have all filed lawsuits challenging the ordinance. The environmental group argues that the ordinance will exacerbate urban sprawl and increase carbon

emissions. The developer claims that the ordinance violates the state's comprehensive land use plan. The residents' association supports the ordinance, citing concerns about increased traffic and strain on local resources. The state housing agency argues that the ordinance conflicts with the state's mandate to increase affordable housing. The federal housing organization contends that the city's antidisplacement and relocation assistance plan is insufficient and ineffective, failing to meet federal standards for fair housing. What is the most likely outcome of these legal challenges? A. The court will uphold the ordinance because the city has a residential antidisplacement and relocation assistance plan, which satisfies the state's requirements. B. The court will strike down the ordinance due to its conflict with the state's comprehensive land use plan and the city's certification to affirmatively further fair housing. C. The court will uphold the ordinance if the city can demonstrate that the restrictions are necessary to protect environmental interests and local resources, despite the state's comprehensive land use plan. D. The court will strike down the ordinance only if there is clear evidence of federal preemption or if the state has explicitly mandated the construction of multifamily units in the comprehensive land use plan.

Answer: B

Deepen Question Prompt: 1. Introduce additional layers of complexity by incorporating multiple legal doctrines, such as the doctrine of severability, the principle of least restrictive means, and the doctrine of unconstitutional conditions. 2. Include detailed arguments from each party that highlight specific statutory provisions, case law precedents, and regulatory frameworks, emphasizing the interplay between state and federal laws. 3. Present a scenario where the city's zoning ordinance includes conditional provisions that could be interpreted differently, requiring the court to consider the ordinance's intent, the impact on various stakeholders, and the broader policy goals of the state's comprehensive land use plan.

Round 3

Question: A city has enacted a new zoning ordinance that restricts the construction of multifamily housing units in certain areas to promote single-family homes. The state's comprehensive land use plan emphasizes the need for more affordable housing and encourages the development of multifamily units to address public housing needs. The city has certified that it will affirmatively further fair housing and has implemented a residential antidisplacement and relocation assistance plan. However, the ordinance includes a conditional provision that allows multifamily units if they meet specific environmental and safety standards set by the city. An environmental group, a local developer, a residents' association, a state housing agency, and a federal housing organization have all filed lawsuits challenging the ordinance. The environmental group argues that the conditional provision is too vague and unenforceable, the developer claims that the ordinance violates the state's comprehensive land use plan, the residents' association supports the ordinance, the state housing agency argues that the ordinance conflicts with the state's mandate to increase affordable housing, and the federal housing organization contends that the city's antidisplacement and relocation assistance plan is insufficient and ineffective. If the court applies the doctrines of severability, the principle of least restrictive means, and the doctrine of unconstitutional conditions, what is the most likely outcome of these legal challenges? A. The court will uphold the entire ordinance because the conditional provision provides a mechanism for multifamily units, which aligns with the state's comprehensive land use plan. B. The court will strike down the entire ordinance due to the conflict with the state's comprehensive land use plan and the city's certification to affirmatively further fair housing, regardless of the conditional provision. C. The court will uphold the ordinance but invalidate the conditional provision, finding it too vague and unenforceable, thus leaving the restriction on multifamily units intact. D. The court will strike down the restriction on multifamily units but uphold the conditional provision, finding it a less restrictive means to achieve the city's environmental and safety goals while aligning with the state's comprehensive land use plan.

Answer: D

Deepen Question Prompt: 1. Introduce additional layers of complexity by incorporating multiple legal doctrines and principles, such as the doctrine of preemption, the principle of proportionality, and the doctrine of substantive due process, to challenge the validity of the zoning ordinance. 2. Include a broader range of stakeholders with conflicting interests, such as a state environmental agency, a national housing advocacy group, and a local business coalition, each presenting distinct legal arguments that intersect with the state's comprehensive land use plan. 3. Design the scenario to involve a multifaceted legal challenge that requires the court to balance the city's environmental and safety concerns, the state's mandate for affordable housing, and federal requirements for fair housing certification, thereby testing the limits of local autonomy and state oversight.

Round 4

Question: A city has enacted a new zoning ordinance that restricts the construction of multifamily housing units in certain areas to promote single-family homes. The state's comprehensive land use plan emphasizes the need for more affordable housing and encourages the development of multifamily units to address public housing needs. The city has certified that it will affirmatively further fair housing and has implemented a residential antidisplacement and relocation assistance plan. However, the ordinance includes a conditional provision that allows multifamily units if they meet specific environmental and safety standards set by the city. An environmental group, a local developer, a residents' association, a state housing agency, a federal housing organization, a state environmental agency, a national housing advocacy group, and a local business coalition have all filed lawsuits challenging the ordinance. The

environmental group argues that the conditional provision is too vague and unenforceable, the developer claims that the ordinance violates the state's comprehensive land use plan, the residents' association supports the ordinance, the state housing agency argues that the ordinance conflicts with the state's mandate to increase affordable housing, the federal housing organization contends that the city's antidisplacement and relocation assistance plan is insufficient and ineffective, the state environmental agency asserts that the environmental standards are inadequate, the national housing advocacy group claims that the ordinance disproportionately affects low-income families, and the local business coalition argues that the ordinance stifles economic growth. If the court applies the doctrines of severability, the principle of least restrictive means, the doctrine of unconstitutional conditions, the doctrine of preemption, the principle of proportionality, and the doctrine of substantive due process, what is the most likely outcome of these legal challenges? A. The court will uphold the entire ordinance because the conditional provision provides a mechanism for multifamily units, which aligns with the state's comprehensive land use plan, and the environmental standards are deemed adequate. B. The court will strike down the entire ordinance due to the conflict with the state's comprehensive land use plan, the city's certification to affirmatively further fair housing, and the federal requirements for antidisplacement and relocation assistance, regardless of the conditional provision. C. The court will uphold the ordinance but invalidate the conditional provision, finding it too vague and unenforceable, and the environmental standards inadequate, thus leaving the restriction on multifamily units intact. D. The court will strike down the restriction on multifamily units but uphold the conditional provision, finding it a less restrictive means to achieve the city's environmental and safety goals while aligning with the state's comprehensive land use plan, and order the city to revise its antidisplacement and relocation assistance plan to meet federal standards.

Answer: D

Deepen Question Prompt: 1. Introduce a scenario where the zoning ordinance involves multiple layers of regulatory requirements, including environmental, safety, and affordability standards, and where the city has taken steps to comply with some but not all of these requirements, creating a complex web of compliance issues. 2. Include a detailed analysis of the interplay between state and federal laws, particularly focusing on the doctrines of preemption, severability, and substantive due process, and how these doctrines might affect the validity of different provisions within the ordinance. 3. Incorporate a broader range of stakeholders, each with distinct and potentially conflicting interests, such as additional environmental groups, housing advocates, local businesses, and state agencies, to create a more intricate and multifaceted legal challenge.

Algorithm 1 Task Contexts Expansion via Multi-Agent Interaction

Require: Domain context C, task description T
Ensure: Task-relevant keywords K

Initialize dialog history $H \leftarrow \emptyset$
Teacher generates initial task-related statement $S_0 \leftarrow \text{TEACHER.STATEMENT}(C, T)$
Append S_0 to H
Student generates thoughts $R_0 \leftarrow \text{STUDENT.THINK}(S_0)$
Append R_0 to H
Teacher reflects and possibly refutes: $S_1 \leftarrow \text{TEACHER.REFLECT}(R_0)$
Append S_1 to H
Student responds with further reflection/refutation: $R_1 \leftarrow \text{STUDENT.REFLECT}(S_1)$
Append R_1 to H
$K \leftarrow \text{SUMMARIZEKEYWORDS}(H)$
return K

Algorithm 2 instance synthesis via Multi-Agent Interaction

Require: Task T, context C
Ensure: Final instance-answer pair (Q, A) or Termination

Initialize dialog history $H \leftarrow \emptyset$
while not $\text{METADETECTOR.SHOULDSTOP}(H)$ **do**
    $Q \leftarrow \text{TEACHER.ASK}(T, C)$
    Append Q to H
    $A \leftarrow \text{STUDENT.ANSWER}(Q, C)$
    Append A to H
    $R \leftarrow \text{REVIEWER.ASSESS}(A)$
    if $\text{REVIEWER.ISVALID}(R)$ **then**
        $Q \leftarrow \text{REVIEWER.PROMPTHARDERQUESTION}(H)$
        continue
    else
        $C' \leftarrow \text{CONTEXTAUGMENT}(C, R)$
        $A \leftarrow \text{STUDENT.ANSWER}(Q, C')$ ▷ Retry with augmented context
        Append A to H
    end if
end while
return Final (Q, A) or Terminate

Bloom's Taxonomy Level	Meta-capability	Score	Reason
Remember	R1: Fact Recall	7	Frequent recall of precedents and statutory provisions is essential, especially in exams and case reviews.
	R2: Terminology Recognition	9	Accurate understanding and use of legal terminology is a fundamental skill for legal professionals.
	R3: Familiarity Detection	4	Recognizing similar cases or statutes has some value but is less critical overall.
Understand	U1: Literal Comprehension	7	Literal understanding is the foundation given the precision of legal language.
	U2: Paraphrasing	5	Useful in interpreting judgments or explaining statutes to non-lawyers.
	U3: Conceptual Matching	6	Analogical reasoning like classifying "illegal possession" under "civil torts" is important.
	U4: Contextual Understanding	10	Interpretation of contracts and statutes heavily relies on context.
	U5: Pattern Recognition	6	Recognizing patterns in templates or precedent structures aids legal automation.
Apply	AP1: Procedure Execution	5	Used in procedural correctness checks for filing, appeals, etc.
	AP2: Knowledge Transfer	6	Applying known principles to novel legal cases.
	AP3: Contextual Application	8	Requires flexible application of laws based on case facts.
	AP4: Tool Use	3	Tools like LexisNexis or citation generators are supportive but not central.
Analyze	AN1: Decomposition	8	Decomposing complex disputes into legal relations, actors, rights, and duties.
	AN2: Relational Mapping	9	Mapping contractual/legal relationships between parties.
	AN3: Discriminative Reasoning	10	Crucial to differentiate relevant from irrelevant evidence and legal/illegal factors.
	AN4: Multi-factor Reasoning	9	Important for multi-clause or multi-factor judgment (e.g., sentencing).
	AN5: Model Abstraction	5	Abstracting cases into legal frameworks is useful but not required in all tasks.
Evaluate	E1: Claim Validation	10	Assessing legality and validity of legal claims is a core task.
	E2: Criteria-Based Comparison	8	Used in sentencing comparisons or contract compliance evaluation.

Continued on next page

Table D.1 – continued from previous page

Bloom's Taxonomy Level	Meta-capability	Score	Reason
	E3: Bias Detection	6	Awareness of bias is necessary, though many tasks require objective reasoning.
	E4: Uncertainty Management	7	Legal grey areas (e.g., judicial discretion) require uncertainty handling.
	E5: Value Appraisal	5	Evaluation of legal policy or ethical conflicts has auxiliary significance.
Create	C1: Ideation	3	Helps with legal reforms, drafting new laws or contract templates.
	C2: Pattern Synthesis	5	Relevant in generating standardized contracts or summarizing precedent types.
	C3: Hypothesis Formation	6	Used to infer legality of an act hypothetically.
	C4: Creative Expression	2	Rarely used, mainly for legal promotion or education.
	C5: Prototyping & Refinement	4	Applicable in revising contract drafts; practical but narrow.

Table D.1: Capability scores and reasons in the legal domain.

Task Type	Task Description	Question Type	Target Capability
Support Order Modification	Determine the most appropriate legal action for modifying a spousal and child support order based on the provided financial and familial circumstances.	Multiple Choice	AP1, AN3
Diversity Compliance Analysis	Determine the most plausible legal argument that supports or refutes the claim that the selection process complied with the legal requirement to consider diversity.	Multiple Choice	U1, AN3
Legal Outcome Prediction	Determine the most likely legal outcome based on the analysis of a scenario involving incidental data collection and dissemination by law enforcement.	Multiple Choice	AN3, E4
Privacy Compliance Decision	Determine the most appropriate legal action for a government agency to take when faced with a request to share UAS-collected images containing PII, considering privacy laws and data dissemination regulations.	Multiple Choice	AN3, C3
Sentencing Guideline Evaluation	Determine the most compelling argument for or against modifying sentencing guidelines to address the use of encryption in criminal activities.	Multiple Choice	R3, E1
Evidence Admissibility Assessment	Determine the admissibility of a piece of evidence in a defamation case based on its relevance and compliance with hearsay rules.	Multiple Choice	AN3, AN4
Hearsay Admissibility Assessment	Determine the admissibility of a former employee's statements in a legal case based on the hearsay rule and the defendant's Sixth Amendment rights.	Multiple Choice	U4, U5
Expert Testimony Admissibility	Determine the admissibility of an expert's testimony in an arbitration hearing based on the provided legal context and rules of evidence.	Multiple Choice	U4, E4
Mediation Evidence Admissibility	Determine whether settlement offers and technical documents discussed during a mediation can be admitted as evidence in a subsequent lawsuit, considering the exceptions to confidentiality.	Multiple Choice	U4, C3
Legal Exception Justification	Determine the most relevant legal exception that justifies the defendant's actions based on the provided case facts and applicable laws.	Multiple Choice	R2, AN4
Legal Justification for Privacy Restrictions	Determine the potential legal justifications for restricting public access to a privacy impact assessment based on the described scenario.	Classification	AP4, AN3

Continued on next page

Task Type	Task Description	Question Type	Target Capability
Legal Argument Assessment	Determine the potential legal arguments and their validity based on the provided legal context and facts.	Classification	AN5, E1
Violation Identification	Identify potential violations of consumer protection laws based on a description of a debt collection interaction.	Classification	R2, E4
Conflict of Interest Analysis	Determine the relevant legal and ethical issues based on a detailed description of a potential conflict of interest scenario.	Classification	R3, E1
Legal Issue Identification	Identify the key legal issues and potential outcomes based on the factual and procedural context of a court review involving a government directive and a tech company's challenge.	Classification	U1, U5
Violation Identification	Identify potential legal violations based on the factual description of a financial disclosure report and its inaccuracies.	Classification	U1, AN3
Evidence Admissibility Assessment	Determine whether an out-of-court identification should be admitted as evidence, considering the presence of counsel and the reliability of the identification.	Question Answering	AN3, AN5
Expert Witness Qualification	Determine whether an expert witness meets the necessary criteria to provide reliable testimony based on their specialized knowledge and its relevance to the case.	Question Answering	AN3, E2
Lobbying Violation Assessment	Determine whether the described actions constitute a violation of the Lobbying Disclosure Act based on the provided facts and legal context.	Question Answering	R1, AN3
Judicial Recusal Evaluation	Determine whether a judge should recuse themselves based on a given set of facts and ethical guidelines.	Question Answering	R2, U4
Compensation Validity Assessment	Determine whether a party's claim for compensation as an administrative expense is valid based on the provided legal context and contractual terms.	Question Answering	R3, E1
Contract Definitization Validity	Determine the validity of the contracting officer's claim to unilaterally definitize the contract and provide a reasoned argument based on relevant legal principles.	Question Answering	AN3, E1

Continued on next page

Task Type	Task Description	Question Type	Target Capability
Guideline Authority Assessment	Determine whether proposed guidelines for preventing bias in refugee status determinations should have binding authority based on the provided legal arguments and evidence.	True/False question	AN3, AN4
Evidence Admissibility Assessment	Determine whether the evidence obtained by the FBI can be legally used in a criminal prosecution based on the given legal principles and facts.	True/False question	AN4, E1
Contract Clause Justification	Determine whether the 'act of God' clause in a contract justifies a mid-term price increase, considering the provided legal context and precedents.	True/False question	E1, C2

Table K.1: Task examples from AutoLegalEval.

Task Type	Task Description	Question Type	Target Capability
Satisfaction Analysis and Uncertainty Identification	Determine the most likely reasons for a customer's high satisfaction with a product and identify any potential uncertainties in their review.	Question Answering	U5, E4
Issue Identification and Solution Suggestion	Identify the specific issue mentioned in the product review and suggest a potential solution.	Question Answering	U1, U4
Review Claim Analysis	Identify the key claims made in a product review and determine their validity based on contextual information.	Question Answering	U4, E1
User Experience Enhancement	Identify potential improvements to the user experience based on a product review and suggest actionable steps to address the issues.	Question Answering	E2, C1
Bias and Aspect Analysis	Identify potential biases and relate them to specific aspects of the product mentioned in the review.	Question Answering	AN2, E3
Customer Feedback Response	Generate a detailed customer service response addressing the customer's feedback and suggesting potential improvements or uses for the product.	Question Answering	AP3, C4
Feature Extraction and Application	Identify and explain the key features and benefits mentioned in a product review, and suggest how these could be applied to similar products.	Question Answering	U3, AP2
Concern Impact Analysis	Identify the primary concern and its impact on user satisfaction from a product review.	Question Answering	AN3, E5
Malfunction Diagnosis and Action Plan	Identify potential causes for a product's malfunction based on a customer's negative review and propose a course of action.	Question Answering	AN1, C3
Issue Likelihood Analysis	Identify potential issues and their likelihood based on a customer's product review.	Question Answering	AN4, E4
Feature and Limitation Extraction	Identify the key features and limitations of a product based on a customer review.	Question Answering	U1, U5
Usage and Claim Analysis	Identify the primary and secondary uses of the product mentioned in the review and determine if the claims about its performance are supported by the review text.	Question Answering	AN4, E1
Feature Implication Analysis	Identify the key features and their implications based on a customer review of a product.	Question Answering	U5, AN5

Continued on next page

Task Type	Task Description	Question Type	Target Capability
Customer Service Ticket Generation	Generate a formal customer service ticket from an informal product review, including a summary of the issue and the steps taken to address it.	Question Answering	U2, AP1
Component and Sentiment Analysis	Identify the key components and their functionality in a product review, and determine if the review is positive or negative.	Question Answering	AP3, AN3
Review Insight Extraction	Identify and articulate the potential underlying issues or improvements suggested by a positive product review.	Question Answering	AN1, C4
Review Synthesis and Analysis	Identify the key strengths and weaknesses of a product based on a customer review and synthesize a summary.	Question Answering	E2, C2

Table K.2: Task examples from AutoCommerceEval.

L Ethics

L.1 Licenses and Terms of Use

We discuss the licensing and terms of use for all artifacts involved in this work. All existing artifacts are used consistently with their intended research purposes and in compliance with their original licenses and access conditions. The generated artifacts, including automatically constructed tasks, prompts, and evaluation instances, are explicitly restricted to research and evaluation use and are compatible with the access conditions of the source data.

We do not redistribute any raw data from third-party sources. Only derived artifacts that do not expose original data content may be released, and their intended use and distribution scope are clearly specified to ensure legal and ethical compliance.

L.2 Annotation Details

Manual annotations were conducted by two trained annotators with prior experience in legal NLP research. Annotators followed written guidelines specifying task objectives, category definitions, and labeling criteria, accompanied by illustrative examples to ensure consistency. The annotation task involved categorizing benchmark task descriptions only and did not include sensitive, offensive, or personal data. Annotators were informed that the annotations would be used exclusively for research analysis and publication in anonymized form, and informed consent was obtained prior to participation.

Annotators were recruited through academic collaboration rather than crowdsourcing platforms and were compensated according to local research assistant standards, which we consider appropriate given their expertise and workload. Since the study involves minimal-risk annotation of synthetic or publicly available benchmark data and does not collect personal information beyond the annotators themselves, the annotation process falls under standard exemptions for non-invasive research and did not require formal ethics board review.