When to Think Deep? Resource-Rational Metacognitive Control for Adaptive Inference in LLM-Based Legal QA

Ao Sun[1] **Shiyuan Liu**[2 *] **Dongliang Zhang**[3] **Xiaoyang Wang**[4] **Ying Zhang**[5]
[1]School of Computer Science and Technology, Zhejiang Gongshang University, Hangzhou, China
[2]Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, Australia
[3]Department of Criminal Investigation, Hunan Police Academy, Changsha, China
[4]School of Computer Science, University of New South Wales, Sydney, Australia
[5]Laboratory for Statistical Monitoring and Intelligent Governance of Common Prosperity, Zhejiang Gongshang University, Hangzhou, China

Abstract

Legal question-answering systems based on large language models (LLMs) encounter reasoning demands that vary significantly across queries. Consequently, static retrieval-and-reasoning pipelines frequently result in inefficient resource allocation. In this paper, we introduce ALEX, a cost-aware adaptive inference architecture grounded in resource-rational accounts of metacognitive control. The proposed framework implements a metacognitive controller that monitors query complexity via multi-dimensional linguistic and confidence cues, and subsequently routes tasks to hierarchical workflows within an expected utility framework. Experimental results on legal benchmarks demonstrate that ALEX achieves superior accuracy while maintaining an efficient trade-off between accuracy and effort, consistent with the principles of bounded rationality. Furthermore, analyses of routing behavior and error correction driven by verification reveal how metacognitive monitoring can mitigate resource misallocation in LLM-based legal QA.

Keywords: Resource-rationality; Metacognitive control; Adaptive inference; Legal reasoning; RAG.

Introduction

Retrieval-augmented generation (RAG) (Lewis et al., 2020) has become central to the development of LLM-based legal question-answering systems (Lee et al., 2024; Gao et al., 2023; Hayashi et al., 2025; Ammann et al., 2025; Zhang et al., 2025). However, most RAG pipelines apply uniform retrieval and reasoning routines across queries, despite the high variability in reasoning demands (Tan et al., 2025; Hu et al., 2025; Liu et al., 2026). This uniformity leads to systematic resource misallocation. Human experts demonstrate an alternative approach, in which they flexibly adjust cognitive effort to match the demands of the task. For instance, a simple statutory lookup may elicit an immediate, intuitive response, whereas a complex multi-jurisdictional scenario requires deliberate analysis involving the retrieval of precedents, statutory reconciliation, and logical verification (Wang and Yuan, 2025; Liu et al., 2025a,b). This establishes a central control problem for legal QA: determining when a system should rely on fast heuristics as opposed to deeper deliberation, and identifying the signals that should govern this transition.

This adaptive behavior aligns with dual-process theories of cognition (Stanovich and West, 2000; Kahneman, 2011), which distinguish between fast, automatic System 1 processing and slow, analytic System 2 reasoning—viewed here as distinct computational modes. Crucially, the transition between these modes is governed by the metacognitive monitoring of task demand and uncertainty, as well as metacognitive control (Nelson, 1990) for strategy selection. When monitoring signals indicate high demand or low confidence, control processes allocate additional computational resources to initiate deeper reasoning or self-verification. From a resource-rational perspective, this reflects a principled trade-off between accuracy and effort. Rather than maximizing effort for every problem—an approach that is computationally prohibitive—rational agents allocate resources to maximize expected utility under specific constraints. Rooted in theories of bounded rationality (Simon, 1955), this principle formalizes intelligence as efficient satisficing (Simon, 1956): the achievement of adequate solutions with minimal expenditure. Recent work demonstrates that such resource-rational principles apply effectively to complex, knowledge-intensive domains (Wu et al., 2024a). To build artificial systems that adaptively allocate effort in legal QA, it is therefore necessary to advance beyond static pipelines and explicitly model this trade-off between resource consumption and accuracy.

Current approaches in legal AI predominantly rely on static processing pipelines that apply a uniform reasoning depth regardless of task complexity (Chan et al., 2024; Zhu et al., 2022; Thanh and Satoh, 2024). This strategy creates a mismatch in which low-demand questions incur unnecessary overhead, while high-demand analyses receive insufficient reasoning depth. Although recent work on RAG has begun to explore adaptive retrieval (Han et al., 2025; Yao et al., 2025), the field lacks a principled computational account of the criteria for deploying different reasoning strategies (Nigam et al., 2025; Wang et al., 2025; Nguyen et al., 2025; Singh et al., 2025; Yang et al., 2025; Wu et al., 2024b; Wang et al., 2024), as well as testable models that formalize the monitoring signals and control policies underlying adaptive reasoning.

To address this gap, we propose ALEX (Adaptive Legal Executor), a resource-rational (Lieder and Griffiths, 2020) computational model of metacognitive control designed for adaptive inference in legal QA. The model implements a loop consisting of monitoring, control, and execution. The process begins with metacognitive monitoring to estimate query de-

*Corresponding author: shiyuan.liu-1@student.uts.edu.au

In G. Boleda, I. Dautriche, M. Maldonado, C. Saldana and T. Torrent (Eds.), *Proceedings of the 48th Annual Conference of the Cognitive Science Society*

mand via multi-dimensional signals, which include linguistic complexity, model uncertainty, and evidence conflict. Subsequently, a resource-rational control policy selects from three hierarchical strategies—Heuristic (S), Hypothesis Refinement (M), and Deliberative Verification (C)—by maximizing a utility function that balances expected accuracy against computational effort (Callaway et al., 2018). Effort encompasses inference costs such as token usage and model calls, while a resource-pressure parameter modulates the sensitivity to these costs. By mapping these graded strategies onto the dual-process continuum, ALEX operationalizes the transition from the fast heuristic processing of System 1 to the slow deliberative reasoning of System 2.

We structure our analysis around four hypotheses. First, the resource rationality hypothesis (H1) predicts that increasing resource pressure will shift strategy selection from deliberative to heuristic processing, tracing a Pareto-optimal frontier in the accuracy-effort space. Second, regarding monitoring-control coupling (H2), we expect that signals indicating higher cognitive demand will predict an increased selection of deliberative strategies, reflecting the monitoring-control loop posited by metacognitive theories. Third, the error monitoring and self-verification hypothesis (H3) posits that the verification mechanism in Strategy C will systematically detect and correct errors made with high confidence. Finally, we propose human-model alignment (H4) as a validation measure, suggesting that the strategy boundaries of ALEX will correlate with human judgments of difficulty for verification.

Our work contributes a computational formalization of adaptive strategy selection bridging dual-process theory, resource rationality, and metacognitive control. We introduce ALEX as a framework that implements these principles for legal QA. Through benchmark evaluations and behavioral analyses, we demonstrate that this adaptive routing improves performance and efficiency, while auxiliary studies confirm the alignment of the model with human judgment. Beyond the legal domain, this framework illustrates how metacognitive monitoring can facilitate the development of resource-efficient LLM systems in knowledge-intensive professional settings.

ALEX for Adaptive Legal QA

This section presents ALEX as a computational model of cost-aware adaptive strategy selection in LLM-based legal QA. It instantiates a resource-rational metacognitive controller that monitors cues of query demand and uncertainty, selects a reasoning strategy under resource constraints, and performs self-verification to detect and correct errors. ALEX is not only evaluated by end-task performance, but also by process-level signatures predicted by theories of metacognition and resource rationality, including systematic strategy shifts under resource pressure and improved accuracy through self-verification.

Task Setting and Strategy Space

We consider a legal question answering setting in which an input query q is answered using an external legal corpus $\mathcal{D}$. A retrieval module returns a set of evidence passages

$$E(q) = \{e_1, e_2, \ldots, e_k\} \subset \mathcal{D}. \tag{1}$$

A reasoning procedure takes $(q, E(q))$ as input and outputs an answer $\hat{y}$. ALEX treats effort allocation as a strategy selection problem. For each query q, ALEX selects a reasoning strategy m from a discrete strategy set

$$\mathcal{M} = \{S, M, C\}, \tag{2}$$

where strategies differ in computational effort and expected accuracy. Strategy S implements a heuristic response using a single-pass retrieve-and-generate procedure. Strategy M implements corrective refinement by generating an initial hypothesis and using it to guide additional evidence retrieval and revision. Strategy C implements deliberation with explicit self-verification and reconstruction, enabling error monitoring and self-correction. We operationalize computational effort using observable proxies such as token usage and the number of model calls. These proxies are measurable and provide a consistent way to compare strategy efficiency across queries.

Monitoring: Query Demand Estimation

Adaptive strategy selection requires estimating when additional computational effort is valuable. ALEX implements metacognitive monitoring by computing monitoring signals that serve as proxies for query demand and uncertainty. These signals are designed to be computable from the query and are motivated by the observation that legal questions vary widely in linguistic complexity and reasoning requirements.

Monitoring Signals ALEX estimates query demand by integrating three signals: a domain-specific linguistic burden measure, a general readability measure, and an intrinsic reasoning-demand score produced by a language model.

First, to capture domain-specific linguistic burden, ALEX uses a modified Gunning Fog Index (GFI-Mod) (Gunning, 1952) that incorporates the density of legal terminology. Let $w_{legal}(q)$ be the number of legal-domain words in the query and $w(q)$ the total number of words. Let $s(q)$ be the number of sentences and $w_{complex}(q)$ the number of complex words. The modified index is defined as

$$S_{\mathrm{GFI}}(q) = 0.4 \left(\frac{w(q)}{s(q)} + 100 \cdot \frac{w_{\mathrm{complex}}(q)}{w(q)} \right) \cdot \left(1 + \alpha \cdot \frac{w_{\mathrm{legal}}(q)}{w(q)} \right). \tag{3}$$

where α controls the contribution of legal terminology to perceived linguistic burden.

Second, ALEX uses the Flesch–Kincaid Grade Level (Kincaid et al., 1975) score to estimate general readability:

$$S_{\mathrm{FKGL}}(q) = 0.39 \cdot \frac{w(q)}{s(q)} + 11.8 \cdot \frac{syll(q)}{w(q)} - 15.59, \tag{4}$$

where $syll(q)$ is the total number of syllables in the query.

Third, ALEX computes an intrinsic reasoning-demand score $S_{\mathrm{LLM}}(q)$ by prompting a language model to estimate

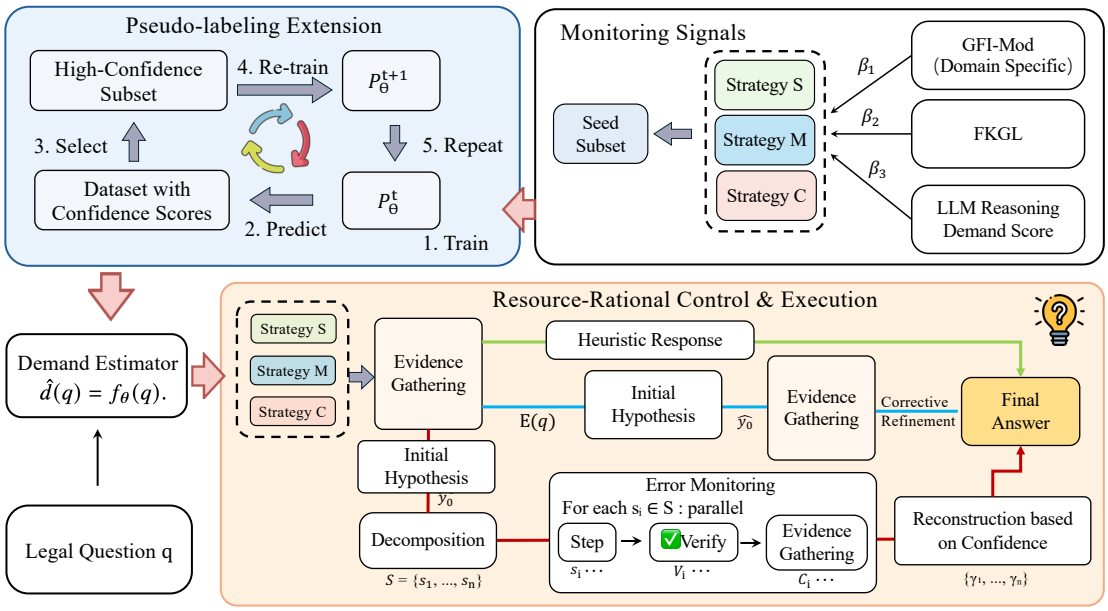

Figure 1: ALEX implemented as a monitoring–control–execution loop.

how much multi-step reasoning is needed to answer the query. This signal is intended to capture query demand not fully explained by surface-level linguistic metrics, such as conditional structure, jurisdictional qualifiers, or the need to reconcile multiple legal sources.

Demand Aggregation ALEX combines the three monitoring signals into a composite demand estimate. We first normalize each component to control scale differences, and then fuse them using a weighted sum:

$$S_{\text{final}}(q) = \beta_1 \hat{S}_{\text{GFI}}(q) + \beta_2 \hat{S}_{\text{FKGL}}(q) + \beta_3 \hat{S}_{\text{LLM}}(q), \quad (5)$$

where $\hat{S}$ denotes normalized scores and $\beta_1, \beta_2, \beta_3$ are aggregation weights. For discrete strategy selection, we discretize $S_{\text{final}}(q)$ into a categorical demand estimate $\tilde{d}(q)$ using thresholds $T_1 < T_2$:

$$\tilde{d}(q) = \begin{cases} \text{low}, & S_{\text{final}}(q) < T_1, \\ \text{medium}, & T_1 \le S_{\text{final}}(q) < T_2, \\ \text{high}, & S_{\text{final}}(q) \ge T_2. \end{cases} \quad (6)$$

The continuous score $S_{\text{final}}(q)$ remains available for analyses of continuous strategy switching boundaries, while $\tilde{d}(q)$ provides a tractable discretization for pseudo-labeling.

Due to the scarcity of expert-annotated demand labels in the legal domain, we bootstrap a robust demand estimator using confidence-driven pseudo-labeling (Arazo et al., 2020; Wang et al., 2016). A seed subset is pseudo-labeled by discretizing the multi-metric demand score into discrete demand categories, yielding $\tilde{d}(q)$. A classifier f_θ is trained to predict these pseudo-labels and iteratively expanded using confidence-based self-training:

$$\hat{d}(q) = f_\theta(q). \quad (7)$$

The goal of this procedure is not demand classification accuracy as an end in itself, but a monitoring signal that is sufficiently stable to support reliable control decisions.

Control: Resource-Rational Routing

Monitoring estimates demand, but it does not directly specify how much computational effort should be invested. ALEX implements metacognitive control as a resource-rational decision policy that trades off expected accuracy against computational effort. For each query q and candidate strategy $m \in \mathcal{M}$, the controller computes a utility:

$$U(m \mid q) = \widehat{\mathbb{E}}[\text{Accuracy} \mid q, m] - \lambda \cdot \text{Effort}(m; q), \quad (8)$$

where $\widehat{\mathbb{E}}[\text{Accuracy} \mid q, m]$ denotes the estimated expected accuracy of strategy m, $\text{Effort}(m; q)$ denotes computational effort, and $\lambda \ge 0$ is a resource pressure parameter controlling sensitivity to effort. The parameter λ provides a theoretically meaningful control knob. When λ is small, the controller is less effort-averse and tends to select more deliberative strategies. As λ increases, the controller becomes increasingly effort-sensitive, shifting toward lower-effort strategies. This formulation yields testable predictions about systematic strategy drift along an accuracy–effort trade-off frontier.

ALEX uses monitoring outputs to approximate the marginal value of additional computational effort. Intuitively, when estimated demand is low, refinement and verification are less likely to improve accuracy and may be unnecessary. When estimated demand is high, deliberative strategies can provide larger benefits by retrieving additional evidence, resolving conflict, and correcting plausible but unsupported inferences. In practice, the controller can approximate $\widehat{\mathbb{E}}[\text{Accuracy} \mid q, m]$ using demand-conditioned empirical estimates or monotonic mappings from $S_{\text{final}}(q)$, reflecting the assumption that higher-demand queries benefit more from deeper strategies.

Effort Definition ALEX operationalizes computational effort using directly observable resource proxies such as token usage and the number of model calls. These proxies support the intended ordering

$$\text{Effort}(S; q) < \text{Effort}(M; q) < \text{Effort}(C; q), \tag{9}$$

reflecting that refinement and verification require additional retrieval, generation, and evaluation steps.

Strategy Selection Rule ALEX selects a strategy by maximizing the resource-rational utility in Equation 8. In our implementation, we use the monitored demand estimate to obtain a tractable threshold controller:

$$m^*(q) = \arg\max_{m \in \mathcal{M}} U(m \mid q) \quad \approx \quad \begin{cases} S, & \hat{d}(q) = \text{low}, \\ M, & \hat{d}(q) = \text{medium}, \\ C, & \hat{d}(q) = \text{high}. \end{cases} \tag{10}$$

The left-hand side expresses the normative utility-maximizing policy, while the right-hand side gives a tractable threshold operationalization using the monitoring output.

Execution: Strategy Workflows

Once selected, ALEX executes the strategy. All strategies share a common retrieval substrate but differ in iterative evidence gathering and verification depth.

Strategy S: heuristic response Strategy S implements a single-pass retrieve-and-generate procedure. ALEX retrieves evidence $E(q)$ and generates an answer:

$$\hat{y}_S = \text{LLM}(q, E(q)). \tag{11}$$

This strategy corresponds to a low-effort strategy that relies on direct pattern-based responding supported by retrieved passages, without explicit revision or self-verification.

Strategy M: corrective refinement Strategy M implements limited reconsideration and targeted evidence seeking. ALEX first generates an initial draft answer (a hypothesis) and then performs a second retrieval conditioned on that draft:

$$\hat{y}_0 = \text{LLM}(q, E(q)), \tag{12}$$

$$E'(q) = \text{Retrieve}(q, \hat{y}_0), \tag{13}$$

$$\hat{y}_M = \text{LLM}(q, E(q) \cup E'(q)). \tag{14}$$

This strategy captures an intermediate-effort strategy in which an initial draft answer (a hypothesis) guides additional evidence gathering and correction, without full decomposition and step-wise self-verification.

Strategy C: deliberation with self-verification Strategy C implements the highest-effort strategy, featuring explicit decomposition, step-wise self-verification, and reconstruction. ALEX first generates an initial response, then decomposes it into reasoning steps:

$$\{s_1, s_2, \ldots, s_n\} = \text{StepExtract}(\hat{y}_0). \tag{15}$$

For each reasoning step s_i, ALEX retrieves supporting evidence and evaluates whether the step is supported. Let C_i denote evidence retrieved for step s_i and let γ_i denote a verification outcome such as a confidence score:

$$(C_i, \gamma_i) = \text{Verify}(q, s_i). \tag{16}$$

Finally, ALEX reconstructs an improved answer conditioned on verified steps and their evidence:

$$\hat{y}_C = \text{Reconstruct}(q, \{s_i\}_{i=1}^n, \{C_i, \gamma_i\}_{i=1}^n). \tag{17}$$

This self-verification-driven deliberation loop supports explicit error checking, conflict resolution, and revision.

Self-Verification and Error Monitoring

Self-verification in Strategy C is intended to implement error monitoring and self-correction rather than serving as a purely performance-driven engineering add-on. In legal reasoning, failures are often not random: a model may produce an answer that is fluent and internally coherent yet unsupported or contradicted by authoritative evidence. Such failures can produce high-confidence errors, where an initial response appears reliable but is incorrect.

ALEX addresses this failure mode by decomposing reasoning into testable sub-claims and validating each claim against retrieved evidence. Step-wise self-verification yields intermediate outcomes γ_i that identify unsupported or conflicting steps. Reconstruction then revises the final answer by relying more heavily on verified reasoning paths and correcting low-confidence steps. In addition to improving correctness, self-verification provides process-level traces for downstream analysis. For each query, ALEX records the selected strategy, computational effort, and self-verification outcomes when applicable. These process variables enable tests of metacognitive predictions in subsequent experiments, including calibration analyses and measurements of whether self-verification preferentially corrects high-confidence errors.

Shared Retrieval and Outputs

All strategies rely on a shared retrieval substrate that combines sparse and dense signals. Given a query q and a document $d \in \mathcal{D}$, ALEX computes a hybrid relevance score $s(q, d) = w_1 \cdot s_{\text{BM25}}(q, d) + w_2 \cdot s_{\text{dense}}(q, d)$ to provide a consistent evidence interface across workflows. This shared foundation helps isolate the effects of metacognitive control and verification depth rather than changes in the retrieval mechanism itself. For each query, ALEX generates the selected strategy $m^*(q)$, a final answer $\hat{y}$, and an effort estimate based on resource proxies along with process-level signals such as self-verification outcomes under Strategy C. These outputs enable downstream analyses of accuracy–effort trade-offs and strategy-switching boundaries as a function of estimated demand and resource pressure while characterizing the error-correction effects of deliberative reasoning.

Table 1: Comparison of inference performance and computational effort against baseline models.

Method	Bar Exam QA					Housing Statute QA				
	Acc	F1-M	F1-W	Tokens	Calls	Acc	F1-M	F1-W	Tokens	Calls
Deepseek-V3 (Baseline)	62.18	62.04	62.11	6.03×10^5	1196	52.23	51.56	51.93	4.61×10^6	9297
Selfrag (llama2-13B)	45.06	44.57	45.37	1.25×10^6	2857	41.34	40.22	40.86	9.89×10^6	24038
Activerag (deepseek-V3)	65.10	65.23	65.24	1.31×10^6	2413	60.23	26.60	58.73	9.78×10^6	20726
RAG+ (deepseek-V3)	66.78	53.39	66.76	9.83×10^5	1196	57.09	**55.71**	58.07	6.85×10^6	9297
ALEX (Ours)	**71.21**	**71.13**	**71.17**	1.15×10^6	2246	**64.03**	40.68	**64.27**	9.21×10^6	19089
Δ *vs. Best Baseline*	*+4.43*	*+5.90*	*+4.41*	—	—	*+3.80*	—	*+5.54*	—	—

Experimental evaluation

Experimental Settings

Datasets The efficacy of ALEX is evaluated on two challenging legal question-answering benchmarks: Bar Exam QA and Housing Statute QA (Zheng et al., 2025). Bar Exam QA comprises authentic Multistate Bar Examination (MBE) questions testing comprehensive legal reasoning. Housing Statute QA contains questions derived from statutory law, requiring precise interpretation of regulatory text. The processed Bar Exam QA dataset includes 1195 questions, categorized as 37% low-demand, 38% medium-demand, and 25% high-demand. The Housing Statute QA dataset consists of 9297 questions, distributed as 32% low-demand, 31% medium-demand, and 37% high-demand. To enrich coverage, we augment the native corpora with 381 additional U.S. case PDFs from CourtListener, applying OCR to image-based files to ensure searchable text. We report Accuracy and F1 (Macro/Weighted) for end-to-end QA performance, and Demand Classification Accuracy to evaluate the quality of our demand estimator (monitoring module).

Baselines To validate its performance, ALEX is benchmarked against a standard RAG baseline (Deepseek-V3) and three state-of-the-art adaptive RAG systems. SELF-RAG (Asai et al., 2024) uses self-reflection to determine retrieval timing and evaluate outputs. ActiveRAG (Jiang et al., 2023) dynamically retrieves context based on model uncertainty during generation. RAG+ (Wang et al., 2025) is a strong baseline specialized for the legal domain that enhances reasoning with application-specific examples.

Implementation Details We employ the Deepseek-V3 for legal term identification and reasoning score evaluation. For multi-metric demand estimation, aggregation weights are $\beta_1 = 0.4$, $\beta_2 = 0.2$, and $\beta_3 = 0.4$. The seed dataset is 20% of the data, with 50% reserved for pseudo-labeling ($\tau = 0.8$). The self-training process runs for ten iterations. For retrieval, BGE-M3 (Chen et al., 2024) embeddings are indexed in Chroma with HNSW (Malkov and Yashunin, 2018). Hybrid retrieval uses weights $w_1 = 0.3$ (BM25) and $w_2 = 0.7$ (dense) to reflect a 7:3 semantic-to-BM25 ratio. The corpus is chunked into 1000-token segments. All experiments are conducted on an NVIDIA A800 GPU.

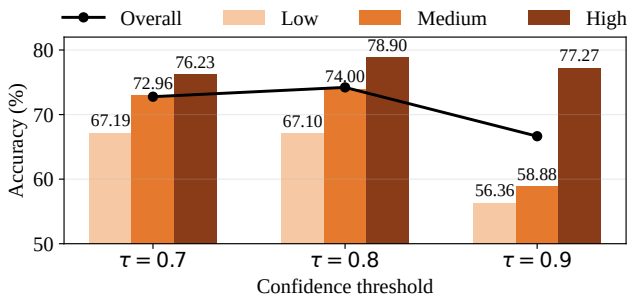

Figure 2: Monitoring accuracy across confidence threshold τ.

Main Results

1) Overall Performance. Table 1 summarizes the overall performance. The results demonstrate that dynamically matching reasoning depth to cognitive demand yields substantial performance gains across both benchmarks. On the Bar Exam QA dataset, ALEX achieves 71.21% accuracy, outperforming the strongest baseline (RAG+) by over 4.4%, underscoring its superior capacity to handle the nuanced, multi-step reasoning. On the Housing Statute dataset, ALEX attains the highest accuracy (64.03%) and Weighted F1-score (64.27%). These results suggest that while existing adaptive baselines incorporate dynamic elements, our explicit, demand-driven routing more effectively resolves the demand-processing mismatch in legal QA. Despite its multi-stage reasoning architecture, ALEX demonstrates superior resource efficiency compared to other adaptive retrieval methods. On the Bar Exam dataset, ALEX reduces token usage by 12.2% and the number of model calls by 7.0% relative to ActiveRAG. While single-pass baselines such as RAG+ naturally consume fewer tokens, the substantial accuracy gains provided by ALEX justify the modest increase in effort. This confirms that the resource-rational controller effectively balances the trade-off between accuracy and effort by avoiding over-computation on low-demand queries.

2) Classifier Performance. The effectiveness of our adaptive framework hinges on accurate demand classification. Figure 2 shows the classifier performs optimally with a 0.8 confidence threshold, achieving 74.21% overall accuracy. It exhibits particularly strong performance on high-demand questions, reaching a high of 78.90%, which is vital for ensuring that high-demand queries are correctly routed to the highest-effort strategy when warranted.

Table 2: Component ablation results on legal QA benchmarks. "w/o" denotes removing the specified component.

Model Configuration	Strategy M	Strategy C	Hybrid Retrieval	Bar Acc. (%)	Housing Acc. (%)
Full ALEX (Ours)	✓	✓	✓	**71.21**	**64.03**
w/o Complex Workflow	✓	✗	✓	67.45	60.14
w/o Strategy Selection	✗	✓	✓	73.65	65.58
w/o Hybrid Retrieval	✓	✓	✗	70.54	63.76
w/o Strategies M & C	✗	✗	✓	62.18	52.23

Table 3: Self-verification effects in Strategy *C* (by EM).

Datasets	w→r	r→w	CorrRate	DegrRate	ΔAcc
Bar Exam	81	6	60.45%	3.64%	+25.09
Housing	831	27	33.05%	2.92%	+23.37

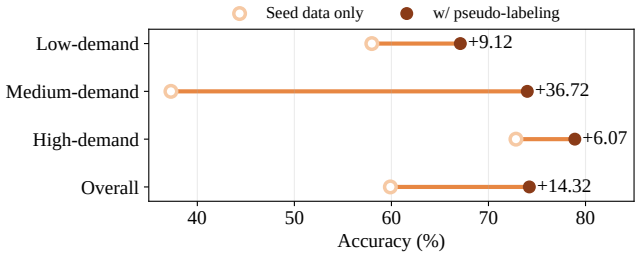

Figure 3: Effect of pseudo-labeling on the monitoring signal.

3) Verification as Error Monitoring and Self-Correction. We examine whether Strategy C exhibits process signatures that are consistent with metacognitive error monitoring. By logging the initial (pre-answer) and final (post-answer) responses, we isolate the contribution of self-verification from other end-to-end effects and directly attribute changes to the verification step. Table 3 reports exact-match (EM) transition counts, where the wrong→right cell quantifies induced corrections and the right→wrong cell captures potential degradation. Across benchmarks, self-verification yields substantial corrections with minimal degradation, indicating that Strategy C serves as a targeted corrective mechanism rather than random perturbation. Pre/post accuracy gains quantify the net payoff of this correction–degradation trade-off.

Ablation Study

1) Impact of Pseudo-Labeling. As shown in Figure 3, the confidence-driven pseudo-labeling algorithm is crucial for overcoming data scarcity in demand classification. Without this procedure, the overall accuracy of the classifier drops from 74.21% to 59.89%, and performance on medium-demand questions falls from 74% to 37.28%.

2) Impact of Core Components. Table 2 reveal that the hierarchical strategy space is the primary driver of performance. Removing the complex workflow (Strategy C) results in an accuracy drop from 71.21% to 67.45% on Bar Exam and from 64.03% to 60.14% on Housing Statute. When both adaptive strategies M and C are removed, performance collapses to the baseline levels of 62.18% and 52.23%, indicating that nearly

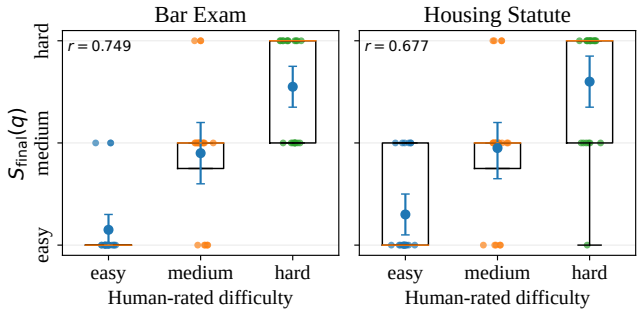

Figure 4: Human–model alignment on difficulty.

10% to 12% of the absolute performance gain is attributable to adaptive reasoning depthpresents a systematic evaluation of our specialized strategies and hybrid retrieval mechanism. The w/o Strategy Selection (Always C) setting yields 73.65% accuracy on Bar Exam and 65.58% on Housing Statute, while consuming approximately 1.7× more tokens and 1.6× more model calls than ALEX.

Human–Model Alignment

We test whether ALEX's demand estimates track human judgments of question difficulty. Five legally trained participants rated 120 items sampled across demand tiers, and we use the majority vote as the human difficulty label. Figure 4 shows that ALEX's demand score $S_{\text{final}}(q)$ increases monotonically across the human difficulty categories, suggesting that the monitoring signal reflects dimensions of difficulty that matter to people in practice. The Pearson correlation between $S_{\text{final}}(q)$ and the ordinal human labels (1–3) is strong for both Bar Exam QA ($r = 0.7490$) and Housing Statute QA ($r = 0.6772$), supporting the monotonic trend in Figure 4. Overall, these results indicate that ALEX's internal monitoring signals are aligned with human difficulty judgments and can usefully guide the allocation of computational effort under resource constraints.

Conclusion

This paper introduces ALEX, a resource-rational model that operationalizes metacognitive control to bridge the gap between static workflows and human-like cognitive flexibility. Our results validate demand-driven strategy allocation as a computational account of metacognitive control, establishing a principled path for accurate and resource-efficient legal AI.

Acknowledgments

AI Use Disclosure. The authors used an AI-based writing assistant for limited language polishing of selected sentences. All content was reviewed by the authors. This work was supported by the China Scholarship Council Program (202408330124), and the Zhejiang Gongshang University Graduate Student Research and Innovation Fund Project.

References

Ammann, L., Ott, S., Landolt, C. R., and Lehmann, M. P. (2025). Securing rag: A risk assessment and mitigation framework. *arXiv preprint arXiv:2505.08728*. https://doi.org/10.48550/arXiv.2505.08728.

Arazo, E., Ortego, D., Albert, P., O'Connor, N. E., and McGuinness, K. (2020). Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *2020 International joint conference on neural networks (IJCNN)*, pages 1–8. IEEE.

Asai, A., Wu, Z., Wang, Y., Sil, A., and Hajishirzi, H. (2024). Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=hSyW5go0v8.

Callaway, F., Lieder, F., Das, P., Gul, S., and Krueger, P. M. (2018). A resource-rational analysis of human planning. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 40. https://escholarship.org/uc/item/11j0r4gj.

Chan, C.-M., Xu, C., Yuan, R., Luo, H., Xue, W., Guo, Y., and Fu, J. (2024). Rq-rag: Learning to refine queries for retrieval augmented generation. *arXiv preprint arXiv:2404.00610*. https://doi.org/10.48550/arXiv.2404.00610.

Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., and Liu, Z. (2024). Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*. https://doi.org/10.48550/arXiv.2402.03216.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, H., and Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2(1). https://doi.org/10.48550/arXiv.2312.10997.

Gunning, R. (1952). The technique of clear writing. *(No Title)*.

Han, W., Xiao, X., Li, Y., Wang, J., Pechenizkiy, M., and Fang, M. (2025). Adaptive iterative retrieval for enhanced retrieval-augmented generation. *Neurocomputing*, page 132272. https://doi.org/10.1016/j.neucom.2025.132272.

Hayashi, K., Kamigaito, H., Kouda, S., and Watanabe, T. (2025). Iterkey: Iterative keyword generation with llms for enhanced retrieval augmented generation. *arXiv preprint arXiv:2505.08450*. https://doi.org/10.48550/arXiv.2505.08450.

Hu, Y., Wang, X., Liu, Q., Xu, X., Fu, Q., Zhang, W., and Zhu, L. (2025). Mmapg: A training-free framework for multimodal multi-hop question answering via adaptive planning graphs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 14195–14211.

Jiang, Z., Xu, F. F., Gao, L., Sun, Z., Liu, Q., Dwivedi-Yu, J., Yang, Y., Callan, J., and Neubig, G. (2023). Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992. https://doi.org/10.18653/v1/2023.emnlp-main.495.

Kahneman, D. (2011). *Thinking, fast and slow*. macmillan.

Kincaid, J. P., Fishburn, R. P., and Chissom, B. S. (1975). Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. *Adult Basic Education*, page 49. https://doi.org/10.21236/ADA006655.

Lee, M.-C., Zhu, Q., Mavromatis, C., Han, Z., Adeshina, S., Ioannidis, V. N., Rangwala, H., and Faloutsos, C. (2024). Agent-g: An agentic framework for graph retrieval augmented generation. https://openreview.net/forum?id=g2C947jjjQ.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474. https://doi.org/10.48550/arXiv.2005.11401.

Lieder, F. and Griffiths, T. L. (2020). Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources. *Behavioral and brain sciences*, 43:e1. https://doi.org/10.1017/S0140525X1900061X.

Liu, S., Feng, Y., Yang, S., Ma, B., and Li, C. (2025a). An efficient ray tracing framework for urban scenarios powered by heterogeneous graph neural networks. *IEEE Antennas and Wireless Propagation Letters*.

Liu, S., Li, C., Chen, C., Ma, B., Chen, Z., Wang, X., and Zhang, Y. (2025b). Geek-explainer: an efficient interpretation method for graph neural networks in sdn. *IEEE Transactions on Cognitive Communications and Networking*.

Liu, S., Wang, J., Lin, X., Qin, L., Zhang, W., and Zhang, Y. (2026). Hyperjoin: Llm-augmented hypergraph link prediction for joinable table discovery. *arXiv preprint arXiv:2601.01015*.

Malkov, Y. A. and Yashunin, D. A. (2018). Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE transactions on pattern analysis and machine intelligence*, 42(4):824–836. https://doi.org/10.1109/TPAMI.2018.2889473.

Nelson, T. O. (1990). Metamemory: A theoretical framework and new findings. In *Psychology of learning and motivation*, volume 26, pages 125–173. Elsevier. https://doi.org/10.1016/S0079-7421(08)60053-5.

Nguyen, T., Chin, P., and Tai, Y.-W. (2025). Ma-rag: Multi-agent retrieval-augmented generation via collaborative chain-of-thought reasoning. *arXiv preprint arXiv:2505.20096*. https://doi.org/10.48550/arXiv.2505.20096.

Nigam, S. K., Patnaik, B. D., Mishra, S., Thomas, A. V., Shallum, N., Ghosh, K., and Bhattacharya, A. (2025). Nyayarag: Realistic legal judgment prediction with rag under the indian common law system. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the ACL (Volume 1: Long Papers)*, pages 1709–1726. Association for Computational Linguistics. https://aclanthology.org/2025.ijcnlp-long.92/.

Simon, H. A. (1955). A behavioral model of rational choice. *The quarterly journal of economics*, pages 99–118. https://doi.org/10.2307/1884852.

Simon, H. A. (1956). Rational choice and the structure of the environment. *Psychological review*, 63(2):129. https://doi.org/10.1037/h0042769.

Singh, A., Ehtesham, A., Kumar, S., and Khoei, T. T. (2025). Agentic retrieval-augmented generation: A survey on agentic rag. *arXiv preprint arXiv:2501.09136*. https://doi.org/10.48550/arXiv.2501.09136.

Stanovich, K. E. and West, R. F. (2000). Individual differences in reasoning: Implications for the rationality debate? *Behavioral and Brain Sciences*, 23(5):645–665. https://doi.org/10.1017/S0140525X00003435.

Tan, X., Wang, X., Liu, Q., Xu, X., Yuan, X., and Zhang, W. (2025). Paths-over-graph: Knowledge graph empowered large language model reasoning. In *Proceedings of the ACM on Web Conference 2025*, pages 3505–3522.

Thanh, N. H. and Satoh, K. (2024). Krag framework for enhancing llms in the legal domain. *arXiv preprint arXiv:2410.07551*. https://doi.org/10.48550/arXiv.2410.07551.

Wang, J., Wu, Y., Wang, X., Zhang, Y., Qin, L., Zhang, W., and Lin, X. (2024). Efficient influence minimization via node blocking. *Proceedings of the VLDB Endowment*, 17(10):2501–2513.

Wang, X., Zhang, Y., Zhang, W., Lin, X., and Chen, C. (2016). Bring order into the samples: A novel scalable method for influence maximization. *IEEE Transactions on Knowledge and Data Engineering*, 29(2):243–256.

Wang, Y., Zhao, S., Wang, Z., Fan, M., Zhang, Y., Zhang, X., Wang, Z., Huang, H., and Liu, T. (2025). Rag+: Enhancing retrieval-augmented generation with application-aware reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2025.emnlp-main.1630.

Wang, Z. and Yuan, B. (2025). L-mars–legal multi-agent workflow with orchestrated reasoning and agentic search. *arXiv preprint arXiv:2509.00761*. https://doi.org/10.48550/arXiv.2509.00761.

Wu, S. A., Ren, X., Gerstenberg, T., Choi, Y., and Levine, S. (2024a). Resource-rational moral judgment. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 46. https://escholarship.org/uc/item/4jg1t212.

Wu, Y., Sun, R., Wang, X., Wen, D., Zhang, Y., Qin, L., and Lin, X. (2024b). Efficient maximal frequent group enumeration in temporal bipartite graphs. *Proceedings of the VLDB Endowment*, 17(11):3243–3255.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. (2025). Qwen3 technical report. *arXiv preprint arXiv:2505.09388*. https://doi.org/10.48550/arXiv.2505.09388.

Yao, Z., Qi, W., Pan, L., Cao, S., Hu, L., Weichuan, L., Hou, L., and Li, J. (2025). Seakr: Self-aware knowledge retrieval for adaptive retrieval augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 27022–27043. https://doi.org/10.18653/v1/2025.acl-long.1312.

Zhang, W., Zhou, H., Zhou, Q., Li, Y., Liu, Y., Lou, J., Wu, C., and Li, J. (2025). Towards comprehensive legal document analysis: A multi-round rag approach. In *Proceedings of the 2025 International Conference on Multimedia Retrieval*, pages 1840–1848.

Zheng, L., Guha, N., Arifov, J., Zhang, S., Skreta, M., Manning, C. D., Henderson, P., and Ho, D. E. (2025). A reasoning-focused legal retrieval benchmark. In *Proceedings of the 2025 Symposium on Computer Science and Law*, pages 169–193. https://doi.org/10.1145/3709025.3712219.

Zhu, G., Hao, M., Zheng, C., and Wang, L. (2022). Design of knowledge graph retrieval system for legal and regulatory framework of multilevel latent semantic indexing. *Computational Intelligence and Neuroscience*, 2022(1):6781043. https://doi.org/10.1155/2022/6781043.