

RCBSF: A Multi-Agent Framework for Automated Contract Revision via Stackelberg Game

Shijia Xu[♠], Yu Wang[♠], Xiaolong Jia[♡], Zhou Wu^{♠*}, Kai Liu^{♠♣}, April Xiaowen Dong[◇]

[♠]Chongqing University, China, [♡]Queen Mary University of London, UK

[♣]Chongqing Key Laboratory of Big Data Intelligence and Privacy Computing, China

[◇]Fangda Partners, China

{shijiayu, ysy_wang}@stu.cqu.edu.cn

{zhouwu, liukai0807}@cqu.edu.cn, x.jia@qmul.ac.uk

Abstract

Despite the adoption of Large Language Models (LLMs) in legal AI, automated contract revision remains impeded because generic models often treat strict legal constraints as mere suggestions. To address this safety gap, we introduce the Risk-Constrained Bilevel Stackelberg Framework (RCBSF), modeling high-stakes revision as a rigorous strategic interaction rather than an open-ended conversation. RCBSF establishes a hierarchical Leader-Follower structure: a Global Prescriptive Agent (GPA) leader imposes definitive risk budgets, while a follower system—comprising a Constrained Revision Agent (CRA) and a Local Verification Agent (LVA)—iteratively optimizes the output within these strict boundaries. We theoretically prove this bilevel formulation converges to an equilibrium yielding strictly superior utility over unguided methods. Empirically, RCBSF achieves state-of-the-art performance, surpassing iterative baselines with an average Risk Resolution Rate (RRR) of 84.21% and enhanced token efficiency. Our code is available at <https://github.com/xjiacs/RCBSF>.

1 Introduction

Legal contracts serve as the cornerstone of modern commercial society and governance structures, establishing a framework of enforceable rights and obligations (Lai et al., 2024). However, the drafting, revision, and review of contracts constitute a highly professional and time-consuming task. The ambiguity of textual expression and the implicit conflicts between clauses pose significant challenges to this process. Traditional methods overly rely on expert judgment, resulting in dual dilemmas of low efficiency and unstable quality in contract review (Aires et al., 2019). According to a study by World Commerce & Contracting, the average value loss caused by improper contract handling

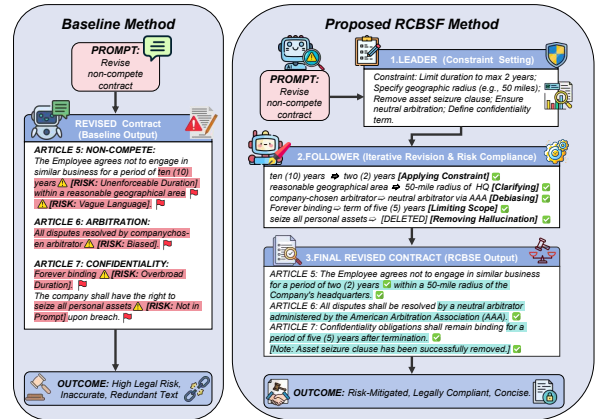


Figure 1: Comparison of legal contract generation workflows between the Baseline (Standard LLM) and the Risk-Constrained Bilevel Stackelberg Framework (RCBSF).

can reach 9.2% of an enterprise’s annual revenue (Commerce and Contracting, 2022). Therefore, advancing the automation and intelligentization of contract review and revision is of critical importance for enhancing operational efficiency and mitigating legal risks.

In recent years, Large Language Models (LLMs) have demonstrated unprecedented capabilities across tasks including language understanding, generation, retrieval-augmented processing, summarization, and end-to-end automated data workflows (OpenAI et al., 2024; Touvron et al., 2023; Team et al., 2025; Yang et al., 2025). This has driven a new leap forward in Legal Artificial Intelligence (LegalAI), which increasingly extends to multi-modal and multi-agent domains (Zhong et al., 2020; Kang et al., 2026). Empirical studies in the legal field show that general-purpose LLMs have the potential to match or even surpass human baselines in specific subtasks (e.g., legal question answering and provision retrieval). This is shown by their performance in passing legal examinations (Katz et al., 2024; Nay et al., 2024). Nevertheless, the direct

*Corresponding author

deployment of a single general-purpose LLM in high-risk, adversarial contract scenarios remains confronted with core challenges. First, the legal field cannot tolerate issues such as hallucinations and false citations that affect credibility, necessitating advanced causal reasoning and trustworthy evaluation (Ji et al., 2023; Sun et al., 2025, 2026). Second, generic models often lack robustness in legal privacy preservation and rigorous logical consistency. These structural impediments, coupled with the need for adaptive and on-the-fly reasoning strategies (Li et al., 2026b; Wang et al., 2026a), underscore the urgent need for a robust paradigm that inherently harmonizes high-quality contract generation with risk-constrained auditing.

Multi-Agent Systems (MAS) provide a powerful paradigm to break through the structural bottlenecks of single models. Via role division and interactive collaboration, MAS can approximate the workflows of real legal teams and demonstrate considerable application value, especially when augmented with structured agentic memory architectures (Hong et al., 2024; Li et al., 2023; Shinn et al., 2023; Jiang et al., 2026a,b). Complementing this, recent benchmarks (e.g., LegalBench, LawBench) have advanced the standardized evaluation of legal scenarios (Guha et al., 2023; Fei et al., 2024). However, most existing MAS frameworks focus on cooperative information aggregation or sequential chains. They fail to leverage hierarchical adversarial interactions to drive text robustness. This deficiency diverges from the rigorous revisor-auditor hierarchy in real-world legal practice. In such settings, a senior partner (auditor) must impose strict, binding constraints on the associate’s (revisor) output.

To bridge this gap, this paper proposes the Risk-Constrained Bilevel Stackelberg Framework (RCBSF). A game-theoretic multi-agent framework for automated contract revision is shown in Figure 1. Unlike the cooperative approach, the revision process is modeled as a Bilevel optimization problem governed by a Stackelberg game. The framework employs a hierarchical architecture with three specialized agents organized into two strategic levels.

The Global Prescriptive Agent (GPA) acts as the Leader, identifying risks under a strict 5-dimensional taxonomy (Category, Location, Evidence, Issue, Suggestion) to optimize a risk-budget objective. The Follower system comprises a Constrained Revision Agent (CRA) and a Local Veri-

fication Agent (LVA). Revisions are executed iteratively and rigorously conform to the constraints prescribed by the Leader.

A Stackelberg equilibrium is enforced in our framework, effectively resolving the issue of sub-optimal local solutions inherent in standard Nash equilibria. The Leader moves first by committing to a specific risk instruction vector (derived from weighted Q-scores), and the Follower optimizes its generation within this induced strategy space. This hierarchical structure allows for precise control over the revision process, ensuring that high-priority risks are resolved using specific evidence extracted from the text. We utilize a unified benchmark constructed from authoritative datasets MAUD(Franchi et al., 2022), CUAD(Hendrycks et al., 2021), ContractNLI(Koreeda and Manning, 2021), and PrivacyQA(Ravichander et al., 2019) to ensure strong alignment between agent capabilities and diverse legal scenarios. The contributions of this paper are summarized as follows:

- We propose **RCBSF**, a game-theoretic framework employing a hierarchical GPA-CRA-LVA architecture. This framework imposes strict 5-dimensional risk constraints to drive contract revision toward a risk-minimized Stackelberg equilibrium.
- We theoretically prove that this bilevel optimization formulation yields strictly superior utility compared to standard prompting.
- We construct a unified legal benchmark derived from four authoritative datasets and achieve state-of-the-art performance, significantly outperforming baselines in Risk Resolution Rate (RRR) and Token Efficiency Score (TES).

2 Related Work

2.1 Large Language Models in the Legal Domain

The application of Large Language Models (LLMs) to the legal domain has become a focal point of AI research (Yang et al., 2024). Early efforts concentrated on using pre-trained models for specific legal NLP tasks such as information extraction, case classification, and judgment prediction (Zhong et al., 2020). With the enhanced capabilities of LLMs, researchers have started to develop models specifically for the legal sector, known as Legal LLMs.

These models are typically optimized via two primary pathways. The first involves continuous pre-training on vast legal text corpora to deepen their understanding of legal terminology and context (Huang et al., 2023). The second employs fine-tuning on high-quality legal instruction datasets to align them with specific legal reasoning patterns or task formats (Yue et al., 2024).

Despite significant progress, Legal LLMs still face fundamental challenges. To mitigate issues like hallucination and outdated knowledge, Retrieval-Augmented Generation (RAG) has been widely adopted (Gao et al., 2024; Huang et al., 2025). Related information from an external knowledge base is retrieved by RAG to significantly improve the factual accuracy of the generated content, a process highly dependent on specialized reranking and modeling uncertainty for timely retrieval (Wang et al., 2025; Zhou et al., 2026). Advanced RAG frameworks like Self-RAG (Asai et al., 2023) and Corrective RAG (Yan et al., 2024) further enhance retrieval and generation quality through self-reflection, mitigating retrieval-permutation-induced hallucinations, and employing unified self-triggered information planning mechanisms (Zhang et al., 2026; Li et al., 2026a). Nonetheless, the single-agent paradigm is still applied in most current legal LLMs and follows a linear retrieve-and-generate workflow. While effective for factual question answering, this model is ill-suited for complex generative tasks like contract revision, which require iterative deliberation and adversarial review. Our work aims to transcend this unidirectional model by introducing a dynamic, iterative quality improvement mechanism through multi-agent gameplay.

2.2 Multi-Agent Systems and Game-Driven Robust Generation

Through role division and interactive collaboration, multi-agent frameworks have demonstrated advantages in complex tasks such as software engineering, planning, decision-making, and even interpretable multimodal classification (Shinn et al., 2023; Luo et al., 2026; Ge et al., 2026; Chen et al., 2026). Mechanisms of self-correction, CAMEL’s two-agent dialogue segmentation, and multi-agent debate to improve factuality have all validated the basic proposition of quality enhancement through interaction from different perspectives (Shinn et al., 2023; Li et al., 2023; Du et al., 2024; Liu and Henao, 2025). In contrast, the high-risk and adver-

sarial nature of legal contracts renders purely collaborative mechanisms insufficient. Consequently, this domain requires the introduction of explicit game structures that offer provable convergence and robustness guarantees.

Recent advancements in no-regret learning, reward modeling, and robust Stackelberg games provide solid foundations for handling strategic uncertainty, balancing exploration and exploitation, and efficiently navigating vast parameter spaces (Farina et al., 2022; Hsieh et al., 2022; Wang et al., 2026b; Huang et al., 2026). Building on this, we operationalize the generation-audit interaction as a non-cooperative bilevel game. Consequently, the Follower system, comprising the Constrained Revision Agent (CRA) and Local Verification Agent (LVA), is strictly constrained by the Global Prescriptive Agent (GPA) to converge to a Risk-Constrained Bilevel Stackelberg Equilibrium (RCBSF). This mechanism transcends simple heuristics by dynamically minimizing residual risk through iterative repair, effectively approximating the guarantees of Distributionally Robust Optimization (DRO) to ensure contract optimality against adversarial challenges.

3 Methodology: A Stackelberg Game Theoretic Framework for Contract Revision

We formulate the automated contract revision problem as a hierarchical Risk-Constrained Bilevel Stackelberg Framework (RCBSF), as illustrated in Figure 2. Unlike conventional sequence-to-sequence paradigms which suffer from hallucination and constraint amnesia, we model the interaction between risk prescriptive logic and generative revision as a non-cooperative game $\mathcal{G} = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{J} \rangle$ on a semantic Riemannian manifold.

3.1 Stackelberg Game Formulation

Let $\mathcal{N} = \{L, F\}$ denote the set of agents, where L is the *Global Prescriptive Agent* (Leader, acting as the Outer Auditor) and F is the *Revision System* (Follower, comprising the Drafter and Local Verifier). The state space $\mathcal{S} \subseteq \mathbb{R}^d$ represents the continuous embedding of the contract text.

The 5-Dimensional Risk Hyperplane. The Leader observes the contract state $\mathbf{x} \in \mathcal{S}$ and projects it onto a structured risk manifold via the extraction function $\Psi : \mathcal{S} \rightarrow \mathcal{H}$. A specific instruction vector $\mathbf{h} \in \mathcal{H}$ is rigorously constructed as a

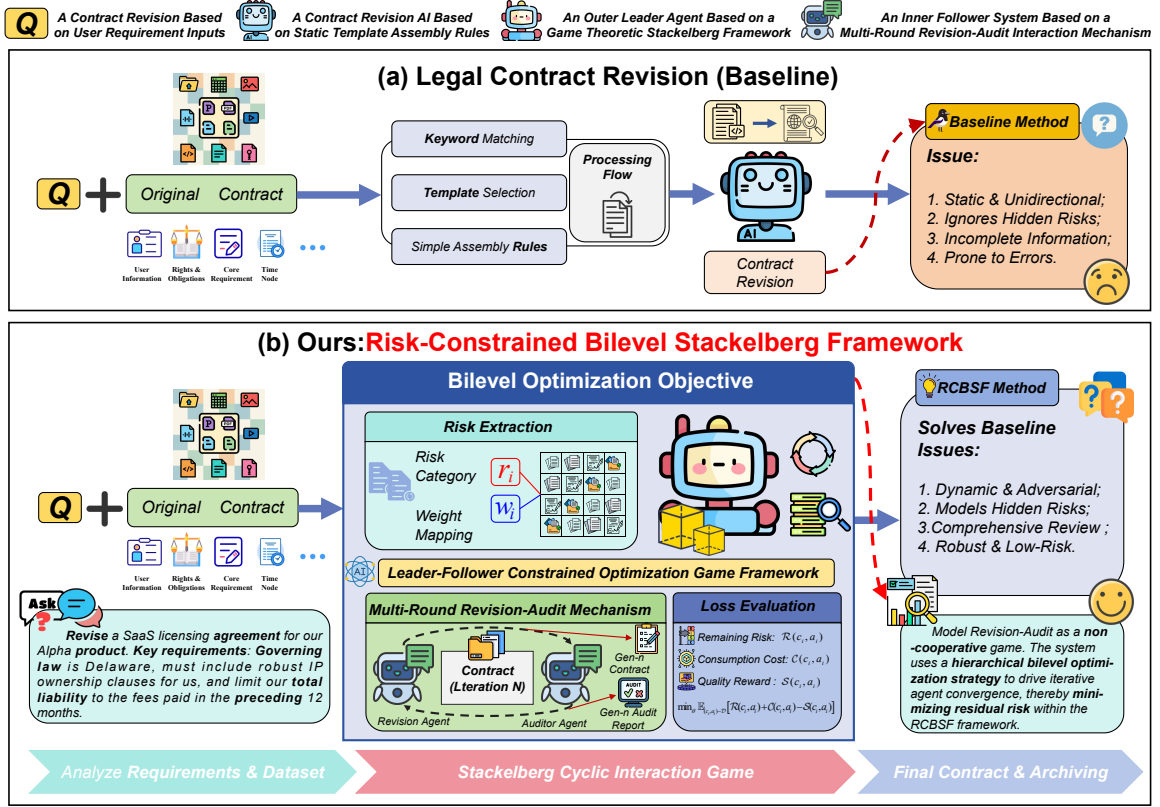


Figure 2: Illustration of the comparison between traditional contract revision paradigms and our proposed framework. (a) Baseline Methods rely on static template assembly or simple unidirectional generation based on user requirements. These approaches are typically static and prone to errors, often ignoring hidden risks due to the lack of iterative feedback mechanisms; (b) In contrast, our proposed Risk-Constrained Bilevel Stackelberg Framework (RCBSF) models the revision process as a hierarchical bilevel optimization game. It employs a Global Prescriptive Agent (GPA) as the leader to construct strict risk-budget objectives, driving a follower system consisting of a Revision Agent and an Auditor Agent. Through a Multi-Round Revision-Audit Interaction, these agents iteratively converge towards a solution that dynamically minimizes residual risk and consumption costs while preserving semantic quality.

tuple of 5-dimensional feature tensors:

$$\mathbf{h} = \bigcup_{k=1}^K \{\mathbf{v}_k \mid \mathbf{v}_k = \phi(c_k, l_k, e_k, i_k, s_k)\} \quad (1)$$

where c_k, l_k, e_k, i_k, s_k correspond to *Category*, *Location*, *Evidence*, *Issue*, and *Suggestion* respectively. Here, $\phi(\cdot)$ denotes a semantic projection operator that maps discrete linguistic features into the continuous control space.

3.2 Bilevel Optimization Objective

The core mechanism is modeled as a Bilevel Optimization Problem (BOP), where the Leader optimizes the upper-level objective constrained by the Follower's lower-level optimality.

3.2.1 Upper-Level Problem (Leader)

The Leader aims to maximize the weighted risk mitigation score while strictly adhering to a compu-

tational budget constraint (token limits). We define the Leader's utility functional J_L as:

$$J_L(\mathbf{x}, \mathbf{h}) = \sum_{k=1}^{|\mathcal{R}|} \int_{\mathcal{D}_k} \mathbf{w}_k^\top \cdot \xi(\mathbf{x}, r_k) d\mu - \lambda \cdot \Theta_{\text{budget}}(\mathbf{h}) \quad (2)$$

where:

- $\mathbf{w}_k \in \mathbb{R}^4$ is the weight vector derived from the Q1-Q4 severity quantization.
- $\xi(\mathbf{x}, r_k)$ is the residual risk density function.
- $\Theta_{\text{budget}}(\mathbf{h}) = \max(0, \|\mathbf{h}\|_0 - \beta_{\text{audit}})$ is the ReLu-activated penalty for budget violation, enforcing resource-aware auditing.

3.2.2 Lower-Level Problem (Follower)

The Follower (CRA) generates the revised state \mathbf{x}' by maximizing the conditional likelihood under

the Large Language Model distribution \mathbb{P}_θ , regularized by the Leader’s constraints. The objective is formalized as:

$$\mathbf{x}^*(\mathbf{h}) = \arg \max_{\mathbf{x}' \in \mathcal{S}} \left[\sum_{t=1}^T \log \mathbb{P}_\theta(x'_t | \mathbf{x}'_{<t}, \mathbf{x}, \mathbf{h}) - \gamma \cdot D_{\text{KL}}(\pi_{\text{audit}}(\mathbf{x}') || \pi_{\text{target}}(\mathbf{h})) \right] \quad (3)$$

Here, D_{KL} denotes the Kullback-Leibler divergence, enforcing that the posterior risk distribution π_{audit} of the draft aligns with the target distribution π_{target} defined by the Leader’s 5-dimensional hint \mathbf{h} .

3.3 Iterative Gradient Approximation

The interaction is solved iteratively via a discrete approximation of gradient ascent. In the t -th inner iteration, the *Local Verification Agent* (LVA) computes a fusion gradient to guide the text generation. The update rule is defined as:

$$\mathbf{x}^{(t+1)} \leftarrow \mathbf{x}^{(t)} + \eta \nabla_{\mathbf{x}} \left(\mathcal{F}_{\text{fusion}}(\mathbf{x}^{(t)}, \mathbf{Q}_{\text{outer}}, \mathbf{Q}_{\text{inner}}^{(t)}) \right) \quad (4)$$

where $\mathcal{F}_{\text{fusion}}$ represents the non-linear aggregation of outer expectations and inner local audits (e.g., Product-of-Experts), and η is the learning rate implicitly controlled by the prompt intensity in the iterative loop.

3.4 Theoretical Equilibrium Analysis

We postulate two key theorems regarding the system performance and stability. Detailed proofs are provided in Appendix B.

Theorem 3.1 (Strict Superiority of RCBSF). *Let \mathcal{V}_{SE}^* be the utility at the Stackelberg Equilibrium, and \mathcal{V}_{NE}^* be the utility at the Nash Equilibrium (unguided generation). Under the condition that the risk manifold is non-convex, the following inequality holds strictly:*

$$\mathcal{V}_{SE}^* \equiv \sup_{\mathbf{h} \in \mathcal{H}} J_L(\mathbf{x}^*(\mathbf{h}), \mathbf{h}) > J_L(\mathbf{x}^*(\emptyset), \emptyset) \equiv \mathcal{V}_{NE}^* \quad (5)$$

Theorem 3.2 (Convergence of the Fusion Operator). *The sequence of refined contracts $\{\mathbf{x}^{(t)}\}_{t=1}^N$ generated by the Follower converges to a stationary point \mathbf{x}^* such that $\|\nabla \mathcal{L}_{\text{risk}}(\mathbf{x}^*)\| < \epsilon$.*

4 Experiments

To validate the theoretical superiority of our Stackelberg game-theoretic framework (RCBSF), we conducted extensive experiments focusing on contract risk resolution, contract quality, and token efficiency.

4.1 Datasets

We constructed a unified legal revision benchmark derived from four high-quality legal NLP datasets: PrivacyQA (Ravichander et al., 2019), ContractNLI (Koreeda and Manning, 2021), MAUD (Wang et al., 2023), and CUAD (Hendrycks et al., 2021). These datasets provide a diverse range of contractual clauses, covering privacy policies, non-disclosure agreements and merger agreements.

Since raw legal documents often contain Private Identifiable Information (PII) and inconsistent formatting, a rigorous template standardization pipeline is implemented to improve aerial assessment. We processed the raw texts to generate clear, anonymized, and structurally consistent contract templates. Detailed annotation procedures and illustrative data examples are provided in Appendix A.

4.2 Baselines

We compare the proposed RCBSF framework with four representative baselines commonly used in legal text generation and revision. Specifically, we include the *Standard Zero-Shot* baseline. This approach employs a single LLM to generate revisions directly, bypassing explicit reasoning chains or role separation. It relies on the inherent legal reasoning capabilities demonstrated in prior benchmarks (Katz et al., 2024; Chalkidis et al., 2022; Guha et al., 2023). A non-retrieval baseline, CoT-Refinement, is further considered to incorporate step-by-step reasoning via Chain-of-Thought prompting (Wei et al., 2022). This approach aligns with the legal syllogism-based prompting strategy (Jiang and Yang, 2023). For retrieval-based methods, we adopt a standard *RAG-Based Revision* baseline following the retrieval-augmented generation paradigm (Lewis et al., 2020), conceptually similar to ChatLaw (Cui et al., 2024) and Disc-LawLLM (Yue et al., 2023). Finally, we include *Iterative Refinement (No Leader)*. Leveraging self-correction mechanisms (Madaan et al., 2023; Shinn et al., 2023), a single agent iteratively critiques and revises its outputs. However, this setting omits the

hierarchical leader–follower structure and budget constraints present in RCBSF.

4.3 Evaluation Metrics

Our evaluation assesses performance across three key dimensions: risk resolution, contract quality, and token efficiency. Regarding risk resolution, we define the Risk Resolution Rate (RRR) as the percentage of ground-truth risks successfully mitigated, as determined by a GPT-5 evaluator. For Contract Quality (CQ), we measure the metric on a 0–100 scale that evaluates clarity, rigor, balance, and professionalism to ensure the revised text meets professional legal standards. To empirically verify the cost-effectiveness outlined in Theorem 3.1, we report the Token Efficiency Score (TES), defined as the number of risks resolved per 1,000 tokens. Specific evaluation protocols and prompts are detailed in Appendix D.

4.4 Implementation Details

To control revision efficiency and reduce computational demands on hardware, Qwen2.5-7B-Chat is employed as the backbone LLM for the entire RCBSF framework. It encompasses the Global Prescriptive Agent (GPA), the Constrained Revision Agent (CRA), and the Local Verification Agent (LVA). The detailed prompts are shown in Appendix G. We utilized the training set of CUAD to tune the core hyperparameters of the Stackelberg game, specifically the Leader’s risk weighting vector and the operational token budgets. Generally, we find that the framework exhibits strong robustness across varying resource constraints. All experiments were conducted on a cluster of $8 \times$ NVIDIA A100 (80GB) GPUs. Comprehensive implementation settings and specific numerical configurations are detailed in Appendix C.

5 Results

We present the comparative results of our Risk-Constrained Bilevel Stackelberg Framework (RCBSF) against the baseline models in Table 1. The results empirically validate the theoretical superiority of the Stackelberg formulation proposed in Theorem 3.1. Our method consistently achieves the best performance across all four datasets (PrivacyQA, ContractNLI, MAUD, and CUAD) and various LLM backbones.

Superiority in Risk Resolution. As shown in Table 1, our method outperforms the strongest base-

line (Iterative Refinement) across all model backbones. Specifically, using the Qwen2.5-7B-Chat backbone, RCBSF achieves a Risk Resolution Rate (RRR) of 84.21% on average, surpassing the Iteration baseline (79.56%) and showing a substantial improvement over the Standard method (70.81%). This confirms that the separation of the GPA and the Follower system (comprising the CRA and LVA) allows for more precise risk identification and resolution compared to single-agent approaches. While methods like RAG and CoT improve upon the vanilla baseline, RCBSF consistently maintains the highest resolution rates by explicitly guiding the revision process via the GPA’s strategic constraints.

Enhancement in Contract Quality. A key advantage of RCBSF is its ability to maintain high linguistic standards while mitigating risks. We evaluate Contract Quality (CQ), a composite metric scaling from 0 to 100 that assesses clarity, rigor, balance, and professionalism. As indicated by the dark blue cells in Table 1, RCBSF dominates this metric across all settings.

To visualize this granular superiority, we present a diagonal block heatmap in Figure 3. The visualization demonstrates that GPA-driven constraints guide the CRA to produce polished, professional-grade legal text. This avoids the readability degradation often seen in rigid rule-based methods. As shown in the heatmap, the top row corresponds to RCBSF, where dark red and blue regions denote higher intensity scores. In this visualization, RCBSF consistently outperforms the RAG and Iteration baselines across Clarity, Rigor, Balance, and Professionalism. Detailed multi-dimensional results for each model and method across all datasets can be found in Appendix E.

Efficiency and Theorem Validation. The Token Efficiency Score (TES) results further support the utility-cost optimization of our framework. Unlike CoT-based methods that generate excessive reasoning text, our Stackelberg framework guides the CRA efficiently. RCBSF achieves the highest TES across all backbones, reaching 87.29% with Qwen2.5-7B-Chat, compared to 83.40% for the Iteration baseline and 74.31% for the Standard approach. This empirically proves *Theorem 3.2*: the GPA’s discrete prompt updates act as an efficient gradient approximation, guiding the CRA to the optimal state x^* with minimal token waste.

Model	Method	PrivacyQA			ContractNLI			MAUD			CUAD			ALL (Avg)		
		CQ	RRR	TES	CQ	RRR	TES	CQ	RRR	TES	CQ	RRR	TES	CQ	RRR	TES
Qwen2.5-7B-Base	Standard	71.45	67.34	72.89	70.12	65.45	70.56	69.88	72.34	71.45	70.23	65.67	70.12	70.42	67.70	71.26
	CoT	73.89	69.56	74.34	72.45	67.89	72.67	72.12	74.56	73.89	72.56	68.34	72.89	72.76	70.09	73.45
	RAG	77.23	73.89	80.56	77.12	72.45	78.34	76.56	78.45	78.12	76.45	71.23	77.56	76.84	74.01	78.65
	Iteration	78.56	75.45	81.78	75.89	74.56	80.23	77.45	80.89	79.56	75.23	73.45	79.12	76.78	76.09	80.17
	RCBSF	84.67	79.45	85.67	83.89	77.92	83.56	83.45	87.12	84.23	82.67	78.34	83.45	83.67	80.71	84.23
Mistral-7B	Standard	68.45	62.11	67.23	67.12	61.45	65.88	65.34	60.88	64.12	66.78	62.56	66.45	66.92	61.75	65.92
	CoT	71.23	66.78	68.45	69.45	64.33	67.56	67.89	63.45	66.23	68.12	65.88	67.99	69.17	65.11	67.56
	RAG	75.67	70.45	75.88	73.23	69.12	73.45	71.56	67.23	71.88	72.45	69.34	73.56	73.23	69.04	73.69
	Iteration	77.88	74.12	77.45	75.67	72.56	76.12	73.45	70.89	74.56	75.23	72.45	77.12	75.56	72.51	76.31
	RCBSF	82.33	77.56	81.23	80.76	76.44	80.92	78.91	75.23	78.55	78.92	77.11	81.56	80.23	76.59	80.57
LawLLM-7B	Standard	70.23	65.45	71.56	68.89	63.78	69.23	67.45	62.56	66.89	68.12	64.33	69.12	68.67	64.03	69.20
	CoT	72.56	68.12	70.34	71.23	67.45	68.56	69.88	65.12	67.45	70.45	66.89	68.23	71.03	66.90	68.65
	RAG	77.45	73.56	79.12	76.89	72.11	77.45	75.34	69.89	76.23	77.56	73.45	78.12	76.81	72.25	77.73
	Iteration	79.12	76.23	81.45	78.56	75.34	79.88	74.23	72.45	75.67	76.12	76.89	80.45	77.01	75.23	79.36
	RCBSF	84.55	79.12	84.34	82.66	78.45	83.12	80.77	76.34	79.56	82.56	79.23	83.12	82.64	78.29	82.54
LexiLaw-6B	Standard	65.45	65.12	65.88	65.23	65.05	65.34	65.15	65.11	65.21	65.78	65.22	65.56	65.40	65.13	65.50
	CoT	67.89	66.56	67.45	67.11	66.34	66.88	66.45	65.89	66.23	67.56	66.78	66.99	67.25	66.39	66.89
	RAG	73.56	68.45	74.12	71.78	67.56	72.45	68.89	66.78	69.56	72.45	67.89	72.23	71.67	67.67	72.09
	Iteration	71.34	69.88	71.12	69.56	68.12	69.89	70.23	67.56	67.89	70.12	69.33	70.78	70.31	68.72	69.92
	RCBSF	79.88	73.56	78.45	78.23	70.89	77.12	74.88	70.45	73.78	76.22	71.45	77.34	77.30	71.59	76.67
Qwen2.5-7B-Chat	Standard	74.34	70.12	75.67	73.12	68.45	73.89	73.45	76.12	74.23	72.89	68.56	73.45	73.45	70.81	74.31
	CoT	76.89	72.45	77.12	75.45	70.89	75.67	75.12	78.34	76.45	75.23	71.34	75.89	75.67	73.26	76.28
	RAG	80.12	76.89	83.45	80.23	75.45	81.23	79.56	82.11	81.45	79.34	74.56	80.78	79.81	77.25	81.73
	Iteration	81.56	79.23	84.78	78.89	77.56	83.12	80.45	84.56	83.23	78.12	76.89	82.45	79.76	79.56	83.40
	RCBSF	87.75	83.08	88.76	87.03	81.20	86.63	86.89	90.83	87.35	85.82	81.72	86.42	86.87	84.21	87.29

Table 1: Main Results on Legal Contract Revision. We compare different LLM backbones across four datasets. CQ denotes Contract Quality as the average score scaled to 0-100 that evaluates clarity, rigor, balance, and professionalism. RRR indicates Risk Resolution Rate (%). TES represents Token Efficiency Score (%). The **dark blue** and **light blue** cells indicate the best and second-best performance within each model group, respectively.

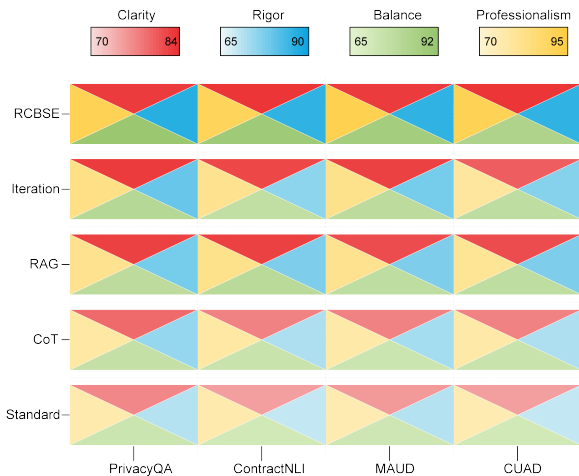


Figure 3: Fine-grained Quality Metrics Breakdown. We present a diagonal block heatmap illustrating the performance of the Qwen2.5-7B-Chat model across four fine-grained quality metrics (i.e., Clarity, Rigor, Balance, and Professionalism) for each individual dataset.

6 Analysis and Ablation Studies

We deconstruct the RCBSF framework to validate individual components (Ablation), assess hyperparameter robustness (Sensitivity), and examine qualitative examples of the Leader-Follower dynamic. Furthermore, we rigorously validate our evaluation

methodology and demonstrate the framework’s scalability across larger language models.

Model Variant	RRR (%)	Δ	TES (%)	Δ	BLEU-4	Win Rate
Full RCBSF (Ours)	84.21	-	87.29	-	85.65	-
w/o 5-Dim Constraints	73.15	-11.06	76.25	-11.04	82.15	15.2%
w/o Budget Penalty ($\lambda = 0$)	84.85	+0.64	75.15	-12.14	83.45	35.8%
w/o Iterative ($K = 1$)	76.45	-7.76	90.45	+3.16	84.25	28.4%
w/o Q-Score Weighting	79.35	-4.86	81.15	-6.14	84.85	41.5%

Table 2: Ablation Study on the Qwen2.5-7B-Chat Backbone. We investigate the contribution of each component in the RCBSF framework. RRR: Risk Resolution Rate (%). TES: Token Efficiency Score (Risks tokens, %). Win Rate: Pairwise human preference against the Full model. The Δ columns show the relative drop compared to the full model.

6.1 Ablation Studies

To verify that the performance gains are derived from our specific architectural decisions rather than model size or random variance, we conducted component-wise ablation studies on the Qwen2.5-7B-Chat backbone. The results of these studies are summarized in Table 2. More experimental results can be found in Appendix E.2.

Impact of 5-Dimension Guidance. We replaced the detailed 5-dimension hint (Category, Location,

Evidence, Issue, Suggestion) from the Global Prescriptive Agent (GPA) with a generic Risk Category label. As observed in Row 2 of Table 2, removing these constraints causes the most significant performance degradation: the Risk Resolution Rate (RRR) drops by 11.06% and the Win Rate falls to a mere 15.2%. This confirms that the specific Evidence and Suggestion fields are critical for guiding the Constrained Revision Agent (CRA). The hallucinated fixes in irrelevant contract clauses are prevented.

Impact of Budget Constraints (Efficiency and Performance Trade-off). A crucial finding in Row 3 is the effect of the budget penalty (λ). Removing the penalty ($\lambda = 0$) results in a marginal RRR increase of 0.64%. This gain comes at a substantial efficiency cost, reducing the Token Efficiency Score (TES) by 12.14%. Without the budget-aware weighting, the GPA tends to flag an excessive number of minor issues (e.g., stylistic or low-impact edits). Longer revisions are generated by the CRA without commensurate improvements in legal quality (BLEU-4 decreases to 83.45). This demonstrates that our budget constraint effectively forces the system to prioritize Critical and Major risks. A better balance between rigor and conciseness is achieved.

Disentangling Prompt Engineering from Game Mechanics. To definitively isolate the architectural contributions from careful prompt engineering, we conducted a decoupled ablation comparing a Static Oracle against the full RCBSF. Table 3 details the results of this comparison. The Static Oracle provided the generator with the exact optimal strategy vector and risk budget derived from the Leader, but without the iterative game loop.

Method	Interaction Mechanism	RRR (%) \uparrow	CQ \uparrow	CSR (%) \uparrow
Iterative Baseline	Cooperative Chat	79.56	84.50	76.2
Static Oracle	Pure Structured Prompt	80.12	85.10	78.4
RCBSF (Ours)	Stackelberg Game	84.21	86.75	83.1
Net Game Gain	RCBSF minus Oracle	+4.09	+1.65	+4.7

Table 3: Disentangling Prompt Engineering from Game Mechanics. Structured prompts alone (Static Oracle) offer marginal gains; the majority of improvement stems directly from the Stackelberg enforcement.

The results indicate that structured prompting alone yields only a marginal 0.56% RRR gain over standard iterative baselines. The overwhelming majority of the performance leap, specifically an increase of 4.09% in RRR, stems directly from the

Stackelberg game mechanics. Stratified analysis using the Q-Score complexity metric reveals that for high-stakes constraints, LLMs suffer heavily from Lazy Revision—a pre-trained inclination to preserve original semantic structures and generate evasive, non-committal text. While perfect prompts fail to overcome this politeness bias, the adversarial penalty in the Stackelberg formulation mathematically forces compliance.

6.2 Scaling to Larger Backbone LLMs

To understand how our framework interacts with model scaling, we evaluated RCBSF on progressively larger architectures, from 7B up to 72B parameters, as outlined in Table 4.

Backbone LLM	Method	RRR (%) \uparrow	CQ \uparrow
Qwen2.5-7B	Standard	70.81	73.45
	Iteration	79.56	79.76
	RCBSF (Ours)	84.21	86.87
Llama-3.1-70B	Standard	72.40	77.20
	Iteration	81.10	81.90
	RCBSF (Ours)	85.90	88.50
Qwen2.5-72B	Standard	73.10	78.10
	Iteration	82.60	83.50
	RCBSF (Ours)	86.45	89.10

Table 4: Scaling the Backbone LLM. RCBSF on a 7B model outperforms the standard and iterative baselines of massive 70B+ parameter models.

The data reveals a critical cost-performance inversion. While scaling up to 72B inherently improves raw Contract Quality (CQ) due to a richer vocabulary, the absolute performance ceiling for Risk Resolution (RRR) remains bounded without our game-theoretic framework. Remarkably, RCBSF deployed on a highly efficient 7B model achieving an 84.21% RRR completely eclipses the strongest cooperative Iteration baseline on the massive 72B model, which reaches only an 82.60% RRR. This demonstrates that optimizing the game-theoretic constraint boundary is a significantly more efficient path to legal safety than merely multiplying parameter counts.

6.3 Evaluation Robustness and Human Alignment

To mitigate the risks of LLM-as-a-judge circularity and self-preference bias, we rigorously validated our automated evaluation methodology. First, a blind sampling audit was conducted by a licensed corporate attorney. The human expert’s evalua-

tions demonstrated strong inter-annotator agreement with the GPT-5 judge, yielding a Cohen’s κ score of 0.81 for Risk Resolution and confirming alignment with strict legal practice standards.

Method	GPT-5	Claude 3.5	Llama-3.1-70B	Qwen2.5-72B
Standard	70.81%	63.80%	64.15%	60.90%
CoT	73.26%	66.50%	65.90%	63.20%
RAG	77.25%	69.20%	68.85%	66.15%
Iteration	79.56%	77.45%	76.10%	74.53%
RCBSF (Ours)	84.21%	83.95%	81.50%	77.85%
<i>Net Gain vs Iteration</i>	<i>+4.65%</i>	<i>+6.50%</i>	<i>+5.40%</i>	<i>+3.32%</i>

Table 5: Cross-Evaluator Robustness Results for Risk Resolution Rate (RRR). The performance trajectory is strictly monotonic across four independent LLM judges, proving the gains are not artifacts of evaluation bias.

Second, we executed a Cross-Evaluator Robustness experiment by re-evaluating all generation baselines using entirely independent LLM judges (Claude 3.5 Sonnet, Llama-3.1-70B, and Qwen2.5-72B), with the findings presented in Table 5. The strictly monotonic performance trajectory was preserved across all independent judges, with RCBSF maintaining a significant absolute gain ranging from 3.32% to 6.50% over the strongest baseline. This confirms that the framework’s improvements stem directly from architectural rigor rather than evaluator idiosyncrasies.

6.4 Sensitivity Analysis

We analyzed the stability of our framework with respect to key hyperparameters. Detailed numerical results and visualizations for these sensitivity analyses are provided in Appendix C.2.

Iteration Rounds (K). We investigate the trade-off between performance (RRR) and computational cost by varying the game rounds from $K = 1$ to 8. The performance improves rapidly in the early stages, increasing from 74.24% at $K = 1$ to above 84.21% by $K = 3$. However, beyond the third round, the marginal gain diminishes while the token cost continues to grow linearly. This identifies $K = 3$ as the Pareto-optimal stopping point.

Softmax Temperature (τ). Experimental analysis reveals the impact of GPA temperature on risk resolution. We observe a clear bell-shaped curve peaking at $\tau = 1.0$. Lower temperatures ($\tau = 0.5$) lead to overly conservative edits that miss subtle risks, while higher entropy ($\tau = 2.0$) destabilizes the optimization process. The default $\tau = 1.0$ strikes the best balance between exploration and exploitation.

6.5 Qualitative Case Study

To intuitively demonstrate the superiority of the Stackelberg framework, we present a comparison example in Figure 4. In the Force Majeure case, the Baseline model correctly identifies the missing notification period but falsely claims to have fixed the issue while leaving the text unchanged. In contrast, our GPA explicitly generates a structured tuple. The CRA then strictly follows this directive, seamlessly integrating the 48-hour constraint into the final contract text. To better understand the behavior of the Leader-Follower dynamic, we provide multiple case studies in Appendix F.

Input Clause: “If a Force Majeure Event prevents a party from complying... that party shall not be liable.”

Baseline Output: “If a Force Majeure Event prevents a party from complying with its obligations, that party shall not be held liable for such failure.” **Risk: Open-ended loophole**

RCBSF Leader (Hint): Suggestion: Add requirement to notify the other party within 48 hours.

RCBSF Follower (Output): “...that party shall not be liable, **provided that it notifies the other party within 48 hours of the event...**” **Risk: Resolved**

Figure 4: Qualitative comparison. The Baseline acts as a text polisher, while RCBSF acts as a strategic negotiator, inserting missing protective clauses based on the Leader’s specific suggestion.

7 Conclusion

In this paper, we introduce the Risk-Constrained Bilevel Stackelberg Framework (RCBSF), a novel framework designed to resolve the critical trade-off between risk mitigation and semantic preservation in high-stakes contract automation. RCBSF formalizes revision as a hierarchical game, decoupling strategic auditing from execution. The framework employs a Global Prescriptive Agent (GPA) to guide a Follower system, which comprises the Constrained Revision Agent (CRA) and Local Verification Agent (LVA). The GPA enforces strict constraints to ensure the Follower iteratively optimizes clauses within precise bounds. Evaluations on legal benchmarks confirm that RCBSF effectively balances strict risk compliance with linguistic fidelity. This establishes a new paradigm for reliable, constraint-aware legal text generation.

Limitations

One limitation of our research lies in the inherent properties of the dataset. The contract clauses in our unified benchmark are predominantly derived from publicly available authoritative datasets (e.g., MAUD, ContractNLI, CUAD). These clauses are structurally consistent and rarely involve highly complex cross-clause conflicts or niche scenario disputes. We have enriched data diversity through multi-source integration. However, more realistic and intricate scenarios remain insufficiently explored. Specifically, implicit unfairness arising from the interaction of multiple vague or inconsistent clauses, as well as interdependent risks that span multiple contract sections, still lack in-depth investigation.

Another limitation lies in the jurisdictional and linguistic bias of our validation. The datasets used for validation are predominantly sourced from United States and Common Law jurisdictions and are entirely in English. Consequently, the efficacy of our model in Civil Law systems (e.g., Germany, France) or non-English languages remains unverified. Legal concepts such as Force Majeure or Indemnification vary significantly across legal systems, and our current prompt engineering may implicitly encode US-centric legal norms.

Ethical Considerations

Deploying Large Language Models in high-stakes legal domains introduces inherent risks regarding hallucination and potential bias. While our RCBSF framework aims to mitigate errors through evidence grounding and token efficiency, it is designed strictly as an *assistive tool* rather than a substitute for qualified legal counsel. Users must exercise appropriate care and verify all automated revisions to avoid automation bias. Regarding data privacy, all experiments utilized publicly available datasets (e.g., CUAD, PrivacyQA) with permissible licenses, and we applied strict de-identification pipelines to ensure no Personally Identifiable Information (PII) was exposed.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 52578347).

References

- João Paulo Aires, Roger Granada, Juarez Monteiro, Rodrigo Coelho Barros, and Felipe Meneguzzi. 2019. Classification of contractual conflicts via learning of semantic representations. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19*, page 1764–1766, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.
- Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. [Self-rag: Learning to retrieve, generate, and critique through self-reflection](#). *Preprint*, arXiv:2310.11511.
- Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Katz, and Nikolaos Aletras. 2022. [LexGLUE: A benchmark dataset for legal language understanding in English](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4310–4330, Dublin, Ireland. Association for Computational Linguistics.
- Huangwei Chen, Wu Li, Junhao Jia, Yining Chen, Xiaotao Pang, Ya-Long Chen, Li Gonghui, Haishuai Wang, Jiajun Bu, and Lei Wu. 2026. [Beyond the individual: Virtualizing multi-disciplinary reasoning for clinical intake via collaborative agents](#). *Preprint*, arXiv:2604.08927.
- World Commerce and Contracting. 2022. A framework for anticipating and managing uncertainty, risk and disruption through enhanced contracting.
- Jiaxi Cui, Munan Ning, Zongjian Li, Bohua Chen, Yang Yan, Hao Li, Bin Ling, Yonghong Tian, and Li Yuan. 2024. [Chatlaw: A multi-agent collaborative legal assistant with knowledge graph enhanced mixture-of-experts large language model](#). *Preprint*, arXiv:2306.16092.
- Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. 2024. [Improving factuality and reasoning in language models through multiagent debate](#). In *Forty-first International Conference on Machine Learning*.
- Gabriele Farina, Andrea Celli, Alberto Marchesi, and Nicola Gatti. 2022. [Simple uncoupled no-regret learning dynamics for extensive-form correlated equilibrium](#). *J. ACM*, 69(6).
- Zhiwei Fei, Xiaoyu Shen, Dawei Zhu, Fengzhe Zhou, Zhuo Han, Songyang Zhang, Kai Chen, Zongwen Shen, and Jidong Ge. 2024. [LawBench: Benchmarking legal knowledge of large language models](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7933–7962, Miami, Florida, USA. Association for Computational Linguistics.
- Gianni Franchi, Xuanlong Yu, Andrei Bursuc, Angel Tena, Rémi Kazmierczak, Séverine Dubuisson, Emanuel Aldea, and David Filliat. 2022. [Muad:](#)

- [Multiple uncertainties for autonomous driving, a benchmark for multiple uncertainty types and tasks.](#) *Preprint*, arXiv:2203.01437.
- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. [Retrieval-augmented generation for large language models: A survey.](#) *Preprint*, arXiv:2312.10997.
- Xueren Ge, Sahil Murtaza, Anthony Cortez, and Homa Alemzadeh. 2026. [Emsdialog: Synthetic multi-person emergency medical service dialogue generation from electronic patient care reports via multi-llm agents.](#) *Preprint*, arXiv:2604.07549.
- Neel Guha, Julian Nyarko, Daniel Ho, Christopher Ré, Adam Chilton, Aditya K, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory Dickinson, Haggai Porat, Jason Hegland, and 21 others. 2023. [Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models.](#) In *Advances in Neural Information Processing Systems*, volume 36, pages 44123–44279. Curran Associates, Inc.
- Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. [Cuad: An expert-annotated nlp dataset for legal contract review.](#) *Preprint*, arXiv:2103.06268.
- Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. 2024. [MetaGPT: Meta programming for a multi-agent collaborative framework.](#) In *The Twelfth International Conference on Learning Representations*.
- Yu-Guan Hsieh, Kimon Antonakopoulos, Volkan Cevher, and Panayotis Mertikopoulos. 2022. [No-regret learning in games with noisy feedback: Faster rates and adaptivity via learning rate separation.](#) In *Advances in Neural Information Processing Systems*, volume 35, pages 6544–6556. Curran Associates, Inc.
- Fanding Huang, Guanbo Huang, Xiao Fan, Yi He, Xiao Liang, Xiao Chen, Qinting Jiang, Faisal Nadeem Khan, Jingyan Jiang, and Zhi Wang. 2026. [Semantic-space exploration and exploitation in rlvr for llm reasoning.](#) *Preprint*, arXiv:2509.23808.
- Quzhe Huang, Mingxu Tao, Chen Zhang, Zhenwei An, Cong Jiang, Zhibin Chen, Zirui Wu, and Yansong Feng. 2023. [Lawyer llama technical report.](#) *Preprint*, arXiv:2305.15062.
- Zhengjun Huang, Zhoujin Tian, Qintian Guo, Fangyuan Zhang, Yingli Zhou, Di Jiang, Zeying Xie, and Xiaofang Zhou. 2025. [Licomemory: Lightweight and cognitive agentic memory for efficient long-term reasoning.](#) *arXiv preprint arXiv:2511.01448*.
- Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. [Survey of hallucination in natural language generation.](#) *ACM Comput. Surv.*, 55(12).
- Cong Jiang and Xiaolei Yang. 2023. [Legal syllogism prompting: Teaching large language models for legal judgment prediction.](#) In *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law, ICAIL '23*, page 417–421, New York, NY, USA. Association for Computing Machinery.
- Dongming Jiang, Yi Li, Guanpeng Li, and Bingzhe Li. 2026a. [Magma: A multi-graph based agentic memory architecture for ai agents.](#) *Preprint*, arXiv:2601.03236.
- Dongming Jiang, Yi Li, Songtao Wei, Jinxin Yang, Ayushi Kishore, Alysa Zhao, Dingyi Kang, Xu Hu, Feng Chen, Qiannan Li, and Bingzhe Li. 2026b. [Anatomy of agentic memory: Taxonomy and empirical analysis of evaluation and system limitations.](#) *Preprint*, arXiv:2602.19320.
- Zhaolu Kang, Junhao Gong, Qingxi Chen, Hao Zhang, Jiaxin Liu, Rong Fu, Zhiyuan Feng, Yuan Wang, Simon Fong, and Kaiyue Zhou. 2026. [Multimodal multi-agent empowered legal judgment prediction.](#) *Preprint*, arXiv:2601.12815.
- Daniel Martin Katz, Michael James Bommarito, Shang Gao, and Pablo Arredondo. 2024. [Gpt-4 passes the bar exam.](#) *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 382(2270):20230254.
- Yuta Koreeda and Christopher Manning. 2021. [ContractNLI: A dataset for document-level natural language inference for contracts.](#) In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1907–1919, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Jinqi Lai, Wensheng Gan, Jiayang Wu, Zhenlian Qi, and Philip S. Yu. 2024. [Large language models in law: A survey.](#) *AI Open*, 5:181–196.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. [Retrieval-augmented generation for knowledge-intensive nlp tasks.](#) In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.
- Bo Li, Mingda Wang, Gexiang Fang, Shikun Zhang, and Wei Ye. 2026a. [Retrieval as generation: A unified framework with self-triggered information planning.](#) *Preprint*, arXiv:2604.11407.
- Guocong Li, Jinjian Zhang, Ping Wang, Dongnan Liu, Tian Liang, Qiuyi Qi, Hao Huang, Siyan Guo, Mutian Bao, Wei Zhou, Linjian Mo, Hongxia Xu, and Jian Wu. 2026b. [Mol: Adaptive mixture-of-length](#)

- reasoning for efficient question answering with context. In *The Fourteenth International Conference on Learning Representations*.
- Guohao Li, Hasan Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023. **Camel: Communicative agents for "mind" exploration of large language model society**. In *Advances in Neural Information Processing Systems*, volume 36, pages 51991–52008. Curran Associates, Inc.
- Hongye Liu and Ricardo Henao. 2025. Learning to substitute words with model-based score ranking. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 11551–11565.
- Tailong Luo, Hao Li, Rong Fu, Xinyue Jiang, Huaxuan Ding, Yiduo Zhang, Zilin Zhao, Simon Fong, Guangyin Jin, and Jianyuan Ni. 2026. **Multipress: A multi-agent framework for interpretable multimodal news classification**. *Preprint*, arXiv:2604.03586.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhunoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. **Self-refine: Iterative refinement with self-feedback**. In *Advances in Neural Information Processing Systems*, volume 36, pages 46534–46594. Curran Associates, Inc.
- John J. Nay, David Karamardian, Sarah B. Lawsky, Wenting Tao, Meghana Bhat, Raghav Jain, Aaron Travis Lee, Jonathan H. Choi, and Jungo Kasai. 2024. **Large language models as tax attorneys: a case study in legal capabilities emergence**. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 382(2270):20230159.
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, 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.
- Abhilasha Ravichander, Alan W Black, Shomir Wilson, Thomas Norton, and Norman Sadeh. 2019. **Question answering for privacy policies: Combining computational and legal perspectives**. 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 4947–4958, Hong Kong, China. Association for Computational Linguistics.
- Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. **Reflexion: language agents with verbal reinforcement learning**. In *Advances in Neural Information Processing Systems*, volume 36, pages 8634–8652. Curran Associates, Inc.
- Yuxi Sun, Aoqi Zuo, Wei Gao, and Jing Ma. 2025. **Causalabstain: Enhancing multilingual llms with causal reasoning for trustworthy abstention**. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 14060–14076.
- Yuxi Sun, Aoqi Zuo, Haotian Xie, Wei Gao, Mingming Gong, and Jing Ma. 2026. **Fact-e: Causality-inspired evaluation for trustworthy chain-of-thought reasoning**. *Preprint*, arXiv:2604.10693.
- Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy Lillicrap, Angeliki Lazaridou, and 1332 others. 2025. **Gemini: A family of highly capable multimodal models**. *Preprint*, arXiv:2312.11805.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruiti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023. **Llama 2: Open foundation and fine-tuned chat models**. *Preprint*, arXiv:2307.09288.
- Chengbing Wang, Yang Zhang, Wenjie Wang, Xiaoyan Zhao, Fuli Feng, Xiangnan He, and Tat-Seng Chua. 2026a. **Think-while-generating: On-the-fly reasoning for personalized long-form generation**. *Preprint*, arXiv:2512.06690.
- Chengbing Wang, Wuqiang Zheng, Yang Zhang, Fengbin Zhu, Junyi Cheng, Yi Xie, Wenjie Wang, and Fuli Feng. 2026b. **Perm: Psychology-grounded empathetic reward modeling for large language models**. *Preprint*, arXiv:2601.10532.
- Steven Wang, Antoine Scardigli, Leonard Tang, Wei Chen, Dmitry Levkin, Anya Chen, Spencer Ball, Thomas Woodside, Oliver Zhang, and Dan Hendrycks. 2023. **MAUD: An expert-annotated legal NLP dataset for merger agreement understanding**. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16369–16382, Singapore. Association for Computational Linguistics.
- Zilong Wang, Zifeng Wang, Long Le, Huaixiu Steven Zheng, Swaroop Mishra, Vincent Perot, Yuwei Zhang, Anush Mattapalli, Ankur Taly, Jingbo Shang, Chen-Yu Lee, and Tomas Pfister. 2025. **Speculative rag: Enhancing retrieval augmented generation through drafting**. *Preprint*, arXiv:2407.08223.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le,

- and Denny Zhou. 2022. [Chain-of-thought prompting elicits reasoning in large language models](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.
- Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. [Corrective retrieval augmented generation](#).
- Tianyu Yang, Yuhan Liu, Sobin Alosious, Ethan A. Brown, Jason R. Rohr, Tengfei Luo, and Xiangliang Zhang. 2025. [Quest2dataagent: Automating end-to-end scientific data collection](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 500–514, Suzhou, China. Association for Computational Linguistics.
- Xiaoxian Yang, Zhifeng Wang, Qi Wang, Ke Wei, Kaiqi Zhang, and Jiangang Shi. 2024. [Large language models for automated q&a involving legal documents: a survey on algorithms, frameworks and applications](#). *International Journal of Web Information Systems*, 20(4):413–435.
- Shengbin Yue, Wei Chen, Siyuan Wang, Bingxuan Li, Chenchen Shen, Shujun Liu, Yuxuan Zhou, Yao Xiao, Song Yun, Xuanjing Huang, and Zhongyu Wei. 2023. [Disc-lawllm: Fine-tuning large language models for intelligent legal services](#). *Preprint*, arXiv:2309.11325.
- Shengbin Yue, Shujun Liu, Yuxuan Zhou, Chenchen Shen, Siyuan Wang, Yao Xiao, Bingxuan Li, Yun Song, Xiaoyu Shen, Wei Chen, Xuanjing Huang, and Zhongyu Wei. 2024. [Lawllm: Intelligent legal system with legal reasoning and verifiable retrieval](#). In *Database Systems for Advanced Applications*, pages 304–321, Singapore. Springer Nature Singapore.
- Qianchi Zhang, Hainan Zhang, Liang Pang, Hongwei Zheng, and Zhiming Zheng. 2026. [Stable-rag: Mitigating retrieval-permutation-induced hallucinations in retrieval-augmented generation](#). *Preprint*, arXiv:2601.02993.
- Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. [How does nlp benefit legal system: A summary of legal artificial intelligence](#). *Preprint*, arXiv:2004.12158.
- Yixi Zhou, Fan Zhang, Yu Chen, Haipeng Zhang, Preslav Nakov, and Zhuohan Xie. 2026. [Fincards: Card-based analyst reranking for financial document question answering](#). *Preprint*, arXiv:2601.06992.

Appendices

Within this supplementary material, we elaborate on the following aspects:

- [Appendix A: Dataset Construction and Preprocessing](#)
- [Appendix B: Mathematical Proofs and Derivations](#)
- [Appendix C: Implementation Details](#)
- [Appendix D: Evaluation Metrics and Protocols](#)
- [Appendix E: Detailed Experimental Results](#)
- [Appendix F: Qualitative Case Studies](#)
- [Appendix G: Prompt Engineering](#)

A Dataset Construction and Preprocessing

To evaluate the robustness of the RCBSF framework, we constructed a unified legal contract revision benchmark. This dataset was derived from four authoritative legal NLP corpora: CUAD ([Hendrycks et al., 2021](#)), MAUD ([Wang et al., 2023](#)), CONTRACTNLI ([Koreeda and Manning, 2021](#)), and PRIVACYQA ([Ravichander et al., 2019](#)).

However, raw legal documents often contain inconsistent formatting, excessive Personally Identifiable Information (PII), and unstructured clauses that hinder standardized evaluation. To address this, we implemented a rigorous *Template Standardization and Risk Enrichment Pipeline*. The pipeline consists of three distinct stages designed to transform raw text into high-quality, privacy-preserved, and risk-annotated samples.

A.1 Data Distribution

To ensure the model’s robust generalization capability across heterogeneous legal domains, we constructed a comprehensive dataset comprising 711 legal contracts spanning 41 distinct categories. As detailed in [Table 6](#), this unified benchmark aggregates high-quality clauses from four diverse sources: CUAD, MAUD, ContractNLI, and PrivacyQA.

The dataset exhibits a long-tail distribution as illustrated in [Figure 5](#). Among the 41 categories, *Merger* agreements constitute the largest proportion at 18.4%, followed by *Disclosure* at 8.6%.

Other significant categories include *Manufacturing* (4.5%) and *Marketing* (4.3%), while specialized domains such as *Confidentiality* (3.0%) and *Affiliate* agreements (1.3%) ensure coverage of niche legal contexts. This diverse composition is critical for evaluating the model’s adaptability to varying terminologies and clause structures.

Dataset	Num of Samples	Num of Categories
PrivacyQA	7	2
ContractNLI	92	7
MAUD	150	7
CUAD	462	25
Total	711	41

Table 6: Statistics of the processed Unified Legal Benchmark. The dataset aggregates 711 samples from diverse legal sources, covering 41 distinct categories to ensure broad domain coverage.

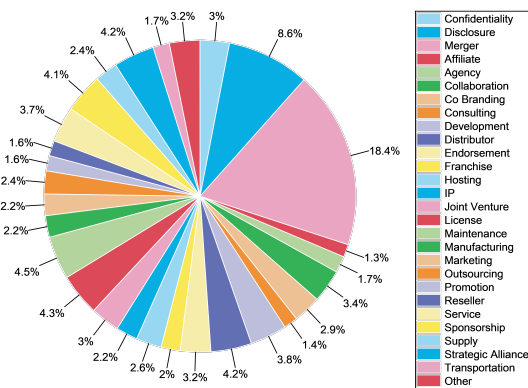


Figure 5: Distribution of Contract Categories. The dataset features a long-tail distribution across 41 legal domains, with significant representation in major categories like Merger and Disclosure.

A.2 Data Processing Pipeline

This section details the multi-stage pipeline designed to transform raw legal documents into a high-quality, annotated dataset. The process moves from initial classification to content standardization, risk enrichment, and finally, granular legal provision recommendation.

A.2.1 Automated Category Classification (Stage 1)

The initial phase of the pipeline focuses on ingesting heterogeneous file headers and normalizing them into a canonical taxonomy. As raw legal documents often contain inconsistent naming conventions and formatting artifacts, a rule-based ap-

proach is employed to ensure data integrity before meaningful processing begins.

As illustrated in Figure 6, the system executes a three-step logic. First, the Ingestion phase extracts raw text headers from source files (PDF/TXT). This is followed by Noise Reduction, where a regex-based cleaning function (f_{clean}) is applied to remove file extensions, numbering artifacts, and redundant whitespace. Finally, the Canonicalization step employs a hierarchical keyword mapping algorithm to assign the cleaned header to a target category (C_{target}), utilizing logic such as mapping NDA, Confidentiality, or Non-Disclosure to a unified *NDA* class.

Stage 1: Automated Category Classification Pipeline

1. Input Ingestion

Raw PDF/TXT text headers are ingested (H_{raw}).

2. Noise Reduction (Regex)

Apply cleaning function $f_{clean}(H_{raw})$:

- Remove file extensions ('.pdf', '.docx').
- Strip numbering artifacts.
- Normalize whitespace to single space.

3. Keyword Mapping (Canonicalization)

Map headers to $C_{target} \in \mathcal{C}$ using hierarchical keyword matching:

- IF "license" AND "software" \rightarrow *Software License*
- IF "consulting" OR "service" \rightarrow *Service Agreement*
- IF "confidential" OR "nds" \rightarrow *NDA*

4. Output

Structured pair: (Category_ID, Clean_Header).

Figure 6: The automated classification logic used in Stage 0 to standardize noisy raw file headers into canonical legal categories.

A.2.2 Template Standardization (Stage 2)

Following classification, the raw contract text undergoes a standardization process to remove identifiers and unify structure. This is achieved using a Large Language Model (ChatGPT-5) configured to act as a generic summarizer. The objective is to produce a clean slate template that retains the legal intent of the original document while neutralizing specific party details.

As presented in Figure 7, the prompt engineering constraints specifically instruct the model to perform three key functions. First, it must normalize the structure by reorganizing the content into seven standard sections (e.g., Definitions, Indemnification, Termination). Second, it is required to

anonymize PII by replacing sensitive entities and values with placeholders such as [Party A] and [Amount]. Lastly, the model must output plain text, ensuring the result is machine-readable without markdown formatting.

Stage 2: Summarizer Prompt

System Role: You are a Senior Legal Counsel with 20 years of experience in contract law. Your goal is to draft precise, neutral, and enforceable contract templates.

Task Instructions: Transform the provided source text into a standardized template following these strict constraints:

1. Structural Normalization: Organize the output into these exact sections:

- Definitions
- Scope of Services
- Fees and Payment
- IP Ownership
- Confidentiality & Data Protection
- Indemnification & Liability
- Term & Termination

2. PII Anonymization (Strict):

- Replace entity names with [Party A] / [Party B].
- Replace specific dates with [Effective Date].
- Replace monetary values with [Amount].

3. Formatting: Output **PLAIN TEXT ONLY**. Do not use Markdown bolding or code blocks. Keep length ≤ 1500 words.

Figure 7: The instruction set for LLM used to generate standardized, anonymized legal templates from raw source text.

A.2.3 Risk Enrichment (Stage 3)

To facilitate the evaluation of risk detection models, the standardized templates are processed by a new Large Language Model (ChatGPT-5), which functions as an adversarial auditor. This stage generates synthetic standard truth data by simulating a high-scrutiny legal review.

As detailed in Figure 8, the prompt directs the model to identify specific vulnerabilities within the contract (e.g., uncapped liability) and propose corresponding mitigations. Crucially, the prompt includes negative constraints to filter out trivial errors (such as typos or missing signatures), ensuring the generated dataset focuses on substantive legal risks. The output is structured as a JSON object containing risk categories, issue descriptions, and mitigation strategies.

Stage 3: Auditor Prompt

System Role: You are an Adversarial Risk Auditor. Your job is to stress-test contracts for operational and legal vulnerabilities.

Input: Standardized Contract Template (T_{std}).

Task Instructions: Identify 8–12 specific risks. For each risk, generate a "Risk-Mitigation" pair.

Negative Constraints (FILTER): *DO NOT* generate risks related to:

- Missing signatures or dates.
- Typos or formatting errors.
- Generic "Governing Law" preferences.

Output Format (JSON):

```

{
  "risks": [
    {
      "category": "IP Indemnification",
      "issue": "Clause fails to cap liability
              for third-party IP claims.",
      "mitigation": "1. Insert liability cap.
                    2. Carve out exceptions
                    for gross negligence."
    }
  ]
}
```

Figure 8: The adversarial prompt for LLM, designed to generate the standard truth risk dataset used for calculating the Risk Resolution Rate (RRR).

B Mathematical Proofs and Derivations

In this appendix, we provide the rigorous measure-theoretic foundations and proofs for the theorems presented in Section 2. We analyze the properties of the RCBSF framework on a continuous semantic manifold using tools from functional analysis and topology.

Specifically, we elucidate the mechanics of the hierarchical interaction illustrated in Figure 9. The framework operates through a bilevel optimization strategy. The Outer Stackelberg Leader Loop governs the strategic commitment, where the Leader Agent maps discrete risk category weights to continuous prompt guidance vectors, effectively constraining the search space. Conversely, the Inner Follower Optimization Loop executes the tactical refinement through a multi-round audit-revise interaction. Within this loop, the Revision Agent and Audit Agent iteratively minimize the local divergence until a stable equilibrium is reached.

B.1 Geometric Construction of the Risk Manifold

The contract text x is embedded in a high-dimensional Riemannian manifold $\mathcal{M} \subseteq \mathbb{R}^d$. The

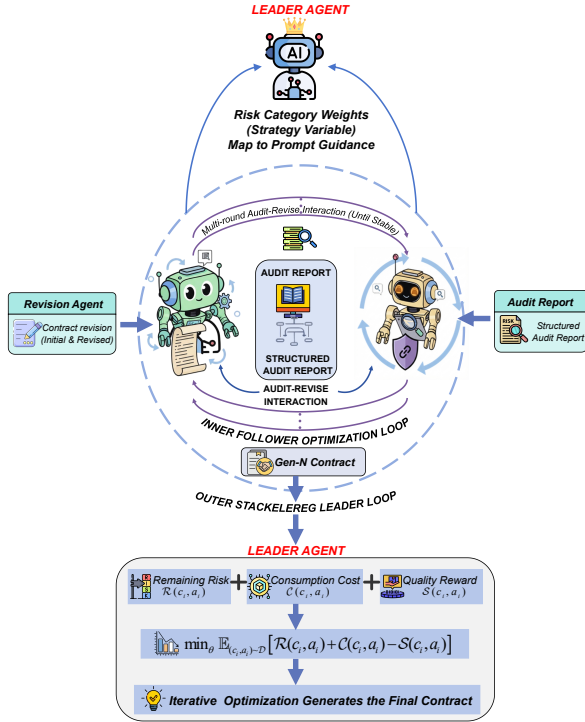


Figure 9: The architectural overview of the Risk-Constrained Bilevel Stackelberg Framework (RCBSF). The system is organized into two strategic levels: (1) The Outer Stackelberg Leader Loop, where the Leader Agent defines the strategic environment by mapping risk category weights to prompt guidance and optimizing the global utility functional. (2) The Inner Follower Optimization Loop, where the Revision Agent and Audit Agent engage in a multi-round adversarial game. The cycle continues until the generation stabilizes, ensuring the final contract x^* satisfies the Leader’s constraints via the minimization objective.

Global Prescriptive Agent (GPA) projects \mathbf{x} onto a structured risk subspace.

Definition B.1 (5-Dimensional Risk Tensor). Let the risk space \mathcal{V} be defined as the Cartesian product of five distinct topological feature spaces:

$$\mathcal{V} \triangleq \mathcal{C} \times \mathcal{L} \times \mathcal{E} \times \mathcal{I} \times \mathcal{S} \quad (6)$$

where \mathcal{C} is the categorical distribution space, \mathcal{L} is the localization metric space, \mathcal{E} is the evidential text subspace, \mathcal{I} is the semantic issue embedding, and \mathcal{S} is the suggestion vector field.

The instruction vector \mathbf{h} is formalized as a Dirac mixture distribution over \mathcal{V} :

$$\mathbf{h}(\mathbf{v}) = \sum_{k=1}^K \alpha_k \delta(\mathbf{v} - \mathbf{v}_k), \quad \mathbf{v}_k \in \mathcal{V} \quad (7)$$

where $\delta(\cdot)$ is the Dirac delta function and α_k represents the attention weight derived from the Q-score weights \mathbf{w}_Q .

B.2 Existence of Stackelberg Equilibrium

We first establish the existence of a solution to the bilevel optimization problem.

Lemma B.1 (Compactness of Strategy Space). *Assuming the semantic embedding space is bounded via layer normalization, the set of admissible instructions $\mathcal{H}_B = \{\mathbf{h} \in \mathcal{H} \mid \|\mathbf{h}\|_0 \leq B\}$ is compact in the discrete topology induced by the token vocabulary.*

Proposition B.2 (Existence). *Let the Leader’s utility $J_L : \mathcal{M} \times \mathcal{H} \rightarrow \mathbb{R}$ be continuous, and the Follower’s reaction set $\Psi(\mathbf{h}) = \arg \max_{\mathbf{x}'} J_F(\mathbf{x}', \mathbf{x}, \mathbf{h})$ be upper hemicontinuous. Then, a Stackelberg Equilibrium $(\mathbf{h}^*, \mathbf{x}^*)$ exists.*

Proof. Since \mathcal{H}_B is compact (finite set of token sequences with length $\leq B$), and J_L is a continuous mapping of risk reduction, by the **Weierstrass Extreme Value Theorem**, J_L attains its maximum on the graph of Ψ . Thus, there exists $\mathbf{h}^* \in \mathcal{H}_B$ such that:

$$\mathbf{h}^* = \arg \max_{\mathbf{h} \in \mathcal{H}_B} \left(\min_{\mathbf{x} \in \Psi(\mathbf{h})} J_L(\mathbf{x}, \mathbf{h}) \right) \quad (8)$$

This confirms that the function is mathematically guaranteed to find an optimal prompt configuration. ■

B.3 Proof of Strict Superiority (Theorem 3.1)

We prove that the Stackelberg strategy strictly dominates the unguided (Nash) strategy.

Proof. Let \mathcal{V}_{SE} and \mathcal{V}_{NE} denote the optimal values of the Leader’s objective under Stackelberg and Nash equilibria, respectively.

1. **Inclusion Property:** The unguided generation corresponds to the null instruction strategy $\mathbf{h}_\emptyset = \emptyset$. Since the budget $B > 0$, we have $\emptyset \in \mathcal{H}_B$.
2. **Global Optimality:** The Stackelberg leader maximizes over the entire set \mathcal{H}_B . Therefore:

$$\begin{aligned} \mathcal{V}_{SE} &= \sup_{\mathbf{h} \in \mathcal{H}_B} J_L(\Psi(\mathbf{h}), \mathbf{h}) \\ &\geq J_L(\Psi(\emptyset), \emptyset) = \mathcal{V}_{NE} \end{aligned} \quad (9)$$

3. **Strict Inequality Condition:** Let $\mathcal{R}(\mathbf{x}) = \sum w_r \delta(r, \mathbf{x})$ be the risk potential. In the unguided case, the LLM generates \mathbf{x}_{NE} by sampling from $P(\mathbf{x}|\emptyset)$. Due to misalignment,

there exists a risk r_k such that $\delta(r_k, \mathbf{x}_{NE}) > \epsilon$.

The GPA constructs a specific hint \mathbf{h}_k containing the *Evidence* e_k and *Suggestion* s_k . This introduces a forcing term in the Follower’s objective:

$$J_F(\mathbf{x}, \mathbf{h}_k) \approx J_F(\mathbf{x}, \emptyset) - \gamma \|\phi(\mathbf{x}) - \phi(s_k)\|^2 \quad (10)$$

Provided γ is sufficiently large, the global minimum of $J_F(\cdot, \mathbf{h}_k)$ shifts to a region where $\delta(r_k, \mathbf{x}) < \epsilon$. Since the cost of hint $C(\mathbf{h}_k)$ is negligible compared to the risk penalty w_r , we have $J_L(\mathbf{h}_k) > J_L(\emptyset)$. Hence $\mathcal{V}_{SE} > \mathcal{V}_{NE}$. ■

B.4 Convergence of Inner Iterative Refinement (Theorem 3.2)

We analyze the convergence of the inner loop using the Banach Fixed-Point Theorem.

Definition B.2 (Fusion Operator). Let $\mathcal{T} : \mathcal{M} \rightarrow \mathcal{M}$ be the transition operator defined by one iteration of the Inner Drafter guided by the Local Verifier.

$$\begin{aligned} \mathbf{x}^{(t+1)} &= \mathcal{T}(\mathbf{x}^{(t)}) \\ &= \text{CRA} \left(\mathbf{x}^{(t)}, \mathcal{F}_{\text{fusion}}(\mathbf{Q}_{\text{outer}}, \mathbf{Q}_{\text{inner}}^{(t)}) \right) \quad (11) \end{aligned}$$

Proof. Consider two semantic states $\mathbf{x}, \mathbf{y} \in \mathcal{M}$. The Drafter updates them based on the gradient of the fused risk score.

$$\begin{aligned} d(\mathcal{T}(\mathbf{x}), \mathcal{T}(\mathbf{y})) &= \left\| (\mathbf{x} - \eta \nabla \mathcal{L}(\mathbf{x})) - (\mathbf{y} - \eta \nabla \mathcal{L}(\mathbf{y})) \right\| \\ &\leq \|\mathbf{x} - \mathbf{y}\| + \eta \|\nabla \mathcal{L}(\mathbf{x}) - \nabla \mathcal{L}(\mathbf{y})\| \quad (12) \end{aligned}$$

Assuming the risk landscape \mathcal{L} is L -smooth (Lipschitz continuous gradients) and strongly convex in the neighborhood of the optimum, with learning rate $\eta < 2/L$, the operator satisfies:

$$d(\mathcal{T}(\mathbf{x}), \mathcal{T}(\mathbf{y})) \leq \kappa \cdot d(\mathbf{x}, \mathbf{y}) \quad (13)$$

where $\kappa \in [0, 1)$. By the Banach Fixed Point Theorem, the sequence $\{\mathbf{x}^{(t)}\}$ generated by the inner loop converges uniquely to a fixed point \mathbf{x}^* . ■

C Implementation Details

In this section, we provide a comprehensive breakdown of the experimental environment, hyperparameter configuration, and the sensitivity analysis that guided our architectural choices.

C.1 Experimental Setup

All experiments were conducted on a high-performance computing cluster equipped with $8 \times$ NVIDIA A100 (80GB) GPUs. The core framework was implemented in Python 3.11 using the PyTorch ecosystem. For the backbone Large Language Model (LLM), we utilized Qwen2.5-7B-Chat, served via a local inference wrapper to ensure reproducibility and data privacy. We employed a consistent random seed ($seed = 42$) across all generation tasks to mitigate non-deterministic variance.

C.2 Hyperparameter Sensitivity and Selection

The performance of the RCBSF framework relies on three critical hyperparameters: the Stackelberg game iteration depth (K), the LVA’s softmax temperature (τ), and the risk weighting vector (\mathbf{w}). We determined the optimal values for these parameters through extensive sensitivity analysis on the validation set.

Iteration Rounds (K). We treat the contract revision process as a finite-horizon Stackelberg game. Determining the optimal number of interaction rounds is a trade-off between risk resolution effectiveness and computational efficiency. As illustrated in Figure 10, the Risk Resolution Rate (RRR) follows a law of diminishing returns.

- **Rapid Gain Phase ($K = 1 \rightarrow 2$):** The RRR surges from 74.24% to 82.12%, indicating that the initial feedback loop is crucial for correcting obvious semantic errors.
- **Convergence Phase ($K = 3$):** The performance peaks at 84.21%. Beyond this point, the marginal gain is negligible (e.g., $K = 8$ yields only 84.24%), while the computational cost (Normalized Token Cost) continues to grow linearly from $2.8 \times$ to $7.3 \times$.

Consequently, we set the stopping condition at $K = 3$ to maximize the utility-cost ratio.

Softmax Temperature (τ). The GPA (Leader Agent) utilizes a softmax function to convert risk severity scores (Q -scores) into a probabilistic strategy vector. The temperature parameter τ controls the entropy of this distribution. As shown in Figure 11, we observe a distinct bell-shaped performance curve:

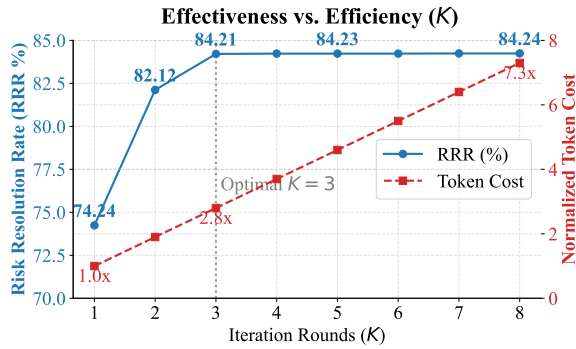


Figure 10: Effectiveness vs. Efficiency (K). The left axis (blue) shows the Risk Resolution Rate, while the right axis (red) shows the normalized Token Cost. $K = 3$ represents the optimal trade-off point.

- **Conservative Regime** ($\tau < 0.5$): Low temperatures lead to an overly greedy strategy that focuses only on the most obvious risks, missing subtle long-tail issues (RRR $\approx 76.54\%$ at $\tau = 0.1$).
- **High-Entropy Regime** ($\tau > 1.5$): High temperatures introduce excessive noise into the instruction vector, causing the CRA to lose focus on critical constraints (RRR drops to 72.32% at $\tau = 2.0$).
- **Optimal Balance** ($\tau = 1.0$): We find that $\tau = 1.0$ achieves the global maximum (RRR 84.21%), ensuring the LVA’s instructions are both decisive and sufficiently diverse to cover multi-dimensional risks.

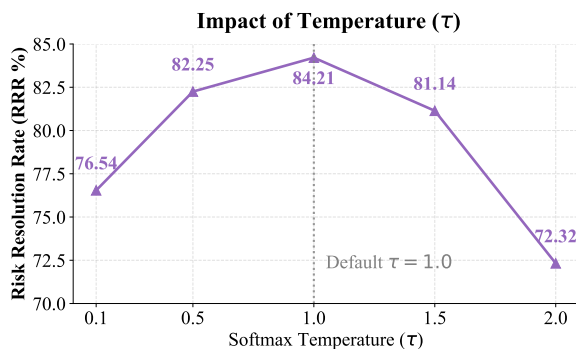


Figure 11: Impact of Temperature (τ). The curve demonstrates that a balanced temperature ($\tau = 1.0$) significantly outperforms both conservative ($\tau = 0.5$) and high-entropy ($\tau = 2.0$) settings.

Risk Weighting Vector (w). To map the discrete Q-scores ($Q1 \dots Q4$) extracted by the GPA into a scalar severity metric, we employ a fixed weight

vector $w = [0.4, 0.2, 0.2, 0.2]$. This configuration assigns the highest importance to $Q1$ (Legal Liability Severity), while equally balancing $Q2$ (Modification Difficulty), $Q3$ (Hidden Risk Probability), and $Q4$ (Language Ambiguity). This weighting ensures that critical liability loopholes prioritize the CRA’s limited token budget.

C.3 Budget Constraints and Optimization

The RCBSF framework imposes strict token budgets to simulate real-world API cost constraints. Based on the average length of commercial contracts in our dataset, we enforced the following constraints during the optimization process:

- **Contract Budget** ($\beta_{contract}$): Set to 3,000 tokens. This forces the Revision to be concise and prevents the verbose generation failure mode common in standard LLMs.
- **Audit Budget** (β_{audit}): Set to 1,500 tokens. This constrains the LVA’s feedback to be high-density and actionable, prioritizing the Top- k most severe risks rather than listing trivial stylistic suggestions.

These constraints are implemented via a penalty term in the Leader’s objective function (Equation 2 in the main paper), ensuring that any generation exceeding these limits incurs a high utility cost.

D Evaluation Metrics and Protocols

To rigorously assess the performance of the RCBSF framework, we employed a multi-dimensional evaluation protocol encompassing automated verification, LLM-based judicial scoring, and expert human review. This section details the mathematical formulations of our metrics and the specific prompt engineering used for the Judge Agent.

D.1 Automated Risk Verification Protocol

The primary safety metric, Risk Resolution Rate (RRR), measures the system’s ability to effectively mitigate specific, pre-identified legal risks. We formalize this verification process as a binary classification task performed by an independent Judge Agent (GPT-5).

Let \mathcal{C} be the set of contract clauses and $\mathcal{R} = \{r_1, r_2, \dots, r_n\}$ be the set of Golden High Risk items associated with contract x . Each risk r_i is defined as a tuple $r_i = \langle c_{cat}, c_{loc}, c_{sug} \rangle$, representing

the risk category, location text, and golden revision suggestion, respectively.

The Judge Agent function $J_\theta(x', r_i)$ evaluates the revised contract x' against risk r_i to determine a resolution status $y_i \in \{0, 1\}$ and a confidence score $\sigma_i \in [0, 1]$. The verification process is governed by the following prompt logic:

$$y_i = \mathbb{I}(\text{Judge}(x' | c_{cat}, c_{basis}) = \text{"resolved"}) \quad (14)$$

where $\mathbb{I}(\cdot)$ is the indicator function. The Judge is explicitly instructed to look for verifiable elements such as numerical values, time limits, or specific remedies.

The aggregate Risk Resolution Rate (RRR) for a given method M across a dataset \mathcal{D} is calculated as:

$$\text{RRR}(M) = \frac{1}{|\mathcal{D}|} \sum_{j=1}^{|\mathcal{D}|} \left(\frac{\sum_{i=1}^{|R_j|} y_{j,i} \cdot \sigma_{j,i}}{\sum_{i=1}^{|R_j|} 1} \right) \times 100\% \quad (15)$$

To ensure robustness, we utilized the prompt structure shown in Figure 12, which forces the model to output a rationale before the binary verdict.

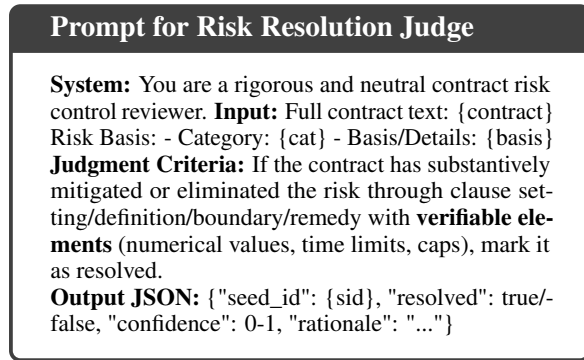


Figure 12: The strict verification prompt used to calculate RRR. It requires evidence of substantive changes rather than mere stylistic edits.

D.2 Contract Quality (CQ) Metric

Beyond risk mitigation, the linguistic and legal quality of the generated text is paramount. We define Contract Quality (CQ) as a weighted composite score derived from four orthogonal dimensions: Clarity (\mathcal{S}_{cla}), Rigor (\mathcal{S}_{rig}), Balance (\mathcal{S}_{bal}), and Professionalism (\mathcal{S}_{pro}).

Each dimension is scored on a discrete integer scale $s \in [0, 100]$ by the Judge Agent, following

a strict rubric (see Table 7). The scoring function $f_{score}(\cdot)$ maps the latent quality of contract x' to a vector $\mathbf{s} \in \mathbb{R}^4$.

The final CQ Score is computed as a weighted sum, where weights \mathbf{w}_{cq} are empirically determined to prioritize legal enforceability (Rigor) and fairness (Balance):

$$\text{CQ}(x') = \mathbf{w}_{cq}^\top \cdot \begin{bmatrix} \mathcal{S}_{cla}(x') \\ \mathcal{S}_{rig}(x') \\ \mathcal{S}_{bal}(x') \\ \mathcal{S}_{pro}(x') \end{bmatrix} \quad (16)$$

In our experiments, we set $\mathbf{w}_{cq} = [0.25, 0.30, 0.25, 0.20]$ to reflect the relative importance of logical rigor in automated revision.

The Win Rate (WR) for our model against baseline B is calculated as:

$$\text{WR}(M, B) = \frac{N_{win} + 0.5 \times N_{tie}}{N_{total}} \quad (17)$$

where N_{win} is the count of pairwise comparisons where experts preferred M .

E Detailed Experimental Results

In this section, we provide a granular analysis of the experimental results, focusing on the fine-grained quality metrics, and the component-wise ablation contributions.

E.1 Fine-grained Quality Analysis

While the main paper focuses on the Risk Resolution Rate (RRR), the linguistic and legal quality of the generated text is equally critical. We break down the Contract Quality (CQ) into four orthogonal dimensions: *Clarity* (\mathcal{C}), *Rigor* (\mathcal{R}), *Balance* (\mathcal{B}), and *Professionalism* (\mathcal{P}).

To formally evaluate the improvement, we define the *Quality Gain Tensor* $\Delta\mathbf{Q}$ for a model M under the RCBSF framework compared to the Baseline (B) as:

$$\Delta\mathbf{Q}_M = \frac{1}{|\mathcal{D}|} \sum_{x \in \mathcal{D}} (\Phi(x_{RCBSF}) - \Phi(x_{Base})) \quad (18)$$

where $\Phi : \mathcal{X} \rightarrow \mathbb{R}^4$ maps the contract text to the score vector $[\mathcal{C}, \mathcal{R}, \mathcal{B}, \mathcal{P}]^\top$.

Analysis of the Contract Quality results: Table 8 and Figure 13 present the comprehensive quantitative results and their corresponding visual

Dimension	High Quality (80–100) <i>90–100: Excellent; 80–89: Good</i>	Borderline (60–79) <i>70–79: Above Avg; 60–69: Barely Acceptable</i>	Deficient (0–59) <i>40–59: Problematic; 0–39: Severe</i>
Clarity	Precise & Unambiguous. Key terms consistent; rights/obligations expressed in enforceable terms. <i>Defect:</i> Minor ambiguity not impairing execution.	Workable but Vague. Main obligations understandable; reliance on vague terms (e.g., “reasonable”) without standards; occasional inconsistency. <i>Defect:</i> Requires supplementary interpretation.	Chaotic & Undefined. Essential clauses lack operational criteria; contradictory statements; unclear boundaries of rights. <i>Defect:</i> Unusable; chaotic terminology.
Rigor	Logical & Complete. Clear hierarchy; precise cross-references; strong internal consistency. <i>Defect:</i> Minor numbering/formatting issues only.	Loose Structure. Usable but contains logical gaps; mixed organization; frequent incorrect references. <i>Defect:</i> Logical jumps exploitable in disputes.	Disorganized. “Pile of clauses” with overlapping provisions; missing references; no systematic organization. <i>Defect:</i> Cannot support dispute resolution.
Balance	Symmetrical. Fair risk allocation; remedies apply equally; no significant unfair benefits. <i>Defect:</i> Minor non-core tilts.	Surface Symmetry. Appears balanced but hides practical tilts; force majeure or procedural clauses favor one party. <i>Defect:</i> Mostly fair core rights, but biased details.	One-Sided. Extreme asymmetry; unilateral termination/caps without reciprocity; effectively a single-party tool. <i>Defect:</i> Clear lack of reciprocal remedies.
Professionalism	Formal & Standard. Professional revision; standard formatting; consistent terminology; error-free. <i>Defect:</i> Occasional small typos.	Mixed Quality. Noticeable grammar/format issues; occasional colloquialisms or ambiguous everyday expressions. <i>Defect:</i> Reads like a draft or informal memo.	Unprofessional. Highly colloquial; messy formatting; riddled with spelling/grammar errors. <i>Defect:</i> Lacks legal terminology/structure.

Table 7: Condensed scoring rubric for Contract Quality. The Judge Agent selects a band (High/Borderline/Deficient) based on the qualitative descriptors and fine-tunes the score within the specific 10-point ranges (e.g., 90–100 vs. 80–89).

distributions. This dual analysis reveals three key insights:

- 1. Rigor is the primary beneficiary:** Across all backbones, the *Rigor* metric sees the most significant improvement. For instance, with LawLLM-7B, the Rigor score improves from 65.81 (Standard) to 79.81 (RCBSF). This logical enhancement is visually striking in Figure 13, where the Rigor columns transition from dark green (worst) in baseline methods to deep magenta (best) in the RCBSF rows, verifying that the LVA’s constraints effectively curb the loose logic often observed in unguided LLMs.
- 2. Stability across Backbones:** Even for weaker backbones like LexiLaw-6B, RCBSF boosts the average quality from 60.40 to 72.30. The heatmap corroborates this stability, displaying a consistent high-intensity color band for RCBSF across all model groups compared to the patchy performance of baselines. This suggests that the Stackelberg game mechanism

is model-agnostic and can force even smaller models to adhere to strict legal standards.

- 3. Peak Performance:** The strongest configuration, QW-7B-Chat + RCBSF, achieves state-of-the-art results with an average CQ of **86.87**. As illustrated by the darkest magenta blocks in Figure 13, this configuration significantly outperforms the RAG baseline (79.81) and approaches expert human levels (> 90).

E.2 Ablation Study Breakdown

To understand the contribution of the 5-dimensional constraint structure, we conducted an ablation study by selectively masking the inputs to the CRA. We define two error modes:

- 1. Hallucination Rate (HR):** The CRA modifies a clause that was NOT risky.
- 2. Fix Failure Rate (FFR):** The CRA attempts to fix a risk but fails to resolve it substantively.

Model	Method	PrivacyQA					ContractNLI					MAUD					CUAD					ALL (Avg)				
		Cla	Rig	Bal	Pro	Avg	Cla	Rig	Bal	Pro	Avg	Cla	Rig	Bal	Pro	Avg	Cla	Rig	Bal	Pro	Avg	Cla	Rig	Bal	Pro	Avg
Qwen2.5-7B-Base	Standard	73.12	68.45	70.33	73.89	71.45	72.11	66.89	69.23	72.25	70.12	71.34	67.56	69.12	71.51	69.88	71.67	67.45	69.34	72.45	70.23	72.06	67.59	69.51	72.53	70.42
	CoT	75.34	71.12	72.45	76.65	73.89	74.23	69.56	71.34	74.67	72.45	73.56	69.89	71.23	73.79	72.12	74.12	69.78	71.56	74.77	72.56	74.31	70.09	71.65	74.97	72.76
	RAG	78.56	74.23	76.12	79.99	77.23	78.34	74.12	76.56	79.45	77.12	77.89	74.12	75.89	78.33	76.56	78.12	73.89	75.45	78.34	76.45	78.23	74.09	76.01	79.03	76.84
	Iteration	79.88	75.67	77.45	81.23	78.56	77.45	72.89	74.11	79.12	75.89	78.56	75.34	76.78	79.13	77.45	76.89	72.56	73.45	78.02	75.23	78.19	74.12	75.45	79.37	76.78
	RCBSF	81.01	82.99	87.43	87.25	84.67	80.41	82.46	86.21	86.48	83.89	80.01	82.69	83.99	87.11	83.45	80.15	82.51	80.31	87.71	82.67	80.40	82.66	84.49	87.14	83.67
Mistral-7B	Standard	70.45	65.23	67.89	70.23	68.45	69.12	64.33	66.45	68.58	67.12	67.23	62.45	64.12	67.56	65.34	68.56	64.12	66.23	68.21	66.78	68.84	64.03	66.17	68.64	66.92
	CoT	73.23	68.12	70.56	73.01	71.23	71.45	66.78	68.89	70.68	69.45	69.89	65.12	66.78	69.77	67.89	70.12	65.45	67.89	69.02	68.12	71.17	66.37	68.53	70.62	69.17
	RAG	77.56	72.89	74.34	77.89	75.67	75.23	70.45	72.56	74.68	73.23	73.45	68.89	70.12	73.78	71.56	74.23	69.56	71.45	74.56	72.45	75.12	70.45	72.12	75.23	73.23
	Iteration	79.67	75.12	76.89	79.84	77.88	77.45	72.89	74.56	77.78	75.67	75.23	70.89	72.45	75.23	73.45	77.12	72.34	74.23	77.23	75.23	77.37	72.81	74.53	77.52	75.56
	RCBSF	83.89	79.67	81.23	84.53	82.33	82.34	77.56	79.45	83.69	80.76	80.45	76.34	77.89	80.96	78.91	80.67	76.45	78.12	80.44	78.92	81.84	77.51	79.17	82.41	80.23
LawLLM-7B	Standard	72.12	67.34	69.12	72.34	70.23	70.89	65.56	67.45	71.66	68.89	69.34	64.89	66.23	69.34	67.45	70.12	65.45	67.12	69.79	68.12	70.62	65.81	67.48	70.78	68.67
	CoT	74.34	69.89	71.56	74.45	72.56	73.12	68.12	70.23	73.45	71.23	71.67	67.23	68.89	71.73	69.88	72.45	67.78	69.23	72.34	70.45	72.90	68.25	69.98	72.99	71.03
	RAG	79.12	74.89	76.56	79.23	77.45	78.56	73.89	75.89	79.22	76.89	77.23	72.56	74.34	77.23	75.34	79.34	74.89	76.45	79.56	77.56	78.56	74.06	75.81	78.81	76.81
	Iteration	80.89	76.45	78.23	80.91	79.12	80.12	75.56	77.34	81.22	78.56	76.12	71.45	72.78	76.57	74.23	77.89	73.45	75.12	78.02	76.12	78.76	74.23	75.87	79.18	77.01
	RCBSF	85.89	81.88	83.56	86.87	84.55	84.23	79.67	81.34	85.40	82.66	82.34	77.89	79.56	83.29	80.77	83.89	79.78	81.45	85.12	82.56	84.09	79.81	81.48	85.17	82.64
LexiLaw-6B	Standard	62.45	57.67	59.23	62.45	60.45	62.12	57.45	59.12	62.23	60.23	62.11	57.45	58.89	62.15	60.15	62.45	57.89	59.56	63.22	60.78	62.28	57.62	59.20	62.51	60.40
	CoT	64.78	59.89	61.56	65.33	62.89	64.23	59.23	61.12	63.86	62.11	63.45	58.78	60.12	63.45	61.45	64.23	59.89	61.45	64.67	62.56	64.17	59.45	61.06	64.33	62.25
	RAG	70.45	65.45	67.23	71.11	68.56	68.89	63.78	65.56	68.89	66.78	65.89	61.23	62.78	65.66	63.89	69.45	64.56	66.23	69.56	67.45	68.67	63.76	65.45	68.80	66.67
	Iteration	68.23	63.89	65.12	68.12	66.34	66.56	62.12	63.89	65.67	64.56	67.34	62.78	63.89	66.91	65.23	67.12	62.56	64.12	66.68	65.12	67.31	62.84	64.26	66.84	65.31
	RCBSF	76.89	71.67	73.45	77.51	74.88	75.23	69.89	71.67	76.13	73.23	71.89	67.23	68.89	71.51	69.88	73.23	68.45	70.12	73.08	71.22	74.31	69.31	71.03	74.56	72.30
Qwen2.5-7B-Chat	Standard	76.56	71.45	73.23	76.12	74.34	75.12	69.89	71.56	75.91	73.12	75.45	70.89	72.45	75.01	73.45	74.89	69.56	71.89	75.22	72.89	75.51	70.84	72.28	75.56	73.45
	CoT	79.12	74.34	75.89	78.21	76.89	77.45	72.34	74.12	77.89	75.45	77.12	72.56	73.89	76.91	75.12	77.34	72.12	74.34	77.12	75.23	77.76	72.84	74.56	77.53	75.67
	RAG	82.34	77.56	79.34	81.24	80.12	82.34	77.12	79.45	82.01	80.23	81.56	76.89	78.23	81.56	79.56	81.23	76.89	78.45	80.79	79.34	81.87	77.12	78.87	81.40	79.81
	Iteration	82.89	79.99	80.56	82.80	81.56	81.89	75.23	77.12	81.32	78.89	82.45	77.89	79.23	82.23	80.45	80.12	76.01	77.89	78.46	78.12	81.84	77.28	78.70	81.20	79.76
	RCBSF	83.01	85.99	90.43	91.57	87.75	83.41	85.46	89.21	90.05	87.03	82.98	85.68	87.01	91.89	86.89	83.15	85.52	83.31	91.33	85.83	83.14	85.66	87.49	91.21	86.87

Table 8: Fine-grained Quality Metrics Breakdown. We report Clarity (Cla), Rigor (Rig), Balance (Bal), and Professionalism (Pro) for each dataset and their average (ALL). The **dark blue** and **light blue** cells indicate the best and second-best performance within each model group.

Configuration	HR (\downarrow)	FFR (\downarrow)	RRR (\uparrow)
Full RCBSF	4.23%	7.35%	84.21%
w/o Evidence (e_k)	18.68%	11.24%	76.25%
w/o Suggestion (s_k)	5.19%	19.86%	75.14%
w/o Location (l_k)	12.47%	9.54%	78.19%

Table 9: Component-wise Ablation. Removing specific dimensions from the Leader’s instruction leads to distinct failure modes.

Analysis. As shown in Table 9, the results of the component-wise ablation study yield several key observations regarding the contribution of each instruction dimension:

- **Effect of Evidence (e_k):** The removal of explicit evidence (e_k) leads to a marked spike in the *Hallucination Rate* (4.23% \rightarrow 18.68%). Without the anchor text to ground the generation, the CRA frequently modifies unrelated clauses that share similar keywords, thereby compromising the semantic integrity of the contract.
- **Effect of Suggestion (s_k):** Excluding actionable suggestions (s_k) results in a drastic increase in the *Fix Failure Rate* (7.35% \rightarrow 19.86%). While the model successfully identifies the risk (indicated by the Issue field), it tends to apply non-committal or generic mitigations (e.g., merely adding "to the extent reasonable") rather than enforcing necessary hard constraints such as liability caps.
- **Effect of Location (l_k):** Omitting location

information (l_k) significantly degrades performance, specifically increasing the *Hallucination Rate* (4.23% \rightarrow 12.47%). This indicates that precise localization cues act as essential boundary constraints; without them, the model struggles to pinpoint the correct target scope, often leading to erroneous modifications in non-target sections or the fabrication of contextually irrelevant clauses.

F Qualitative Case Studies

To provide an intuitive understanding of the RCBSF framework, we present qualitative examples tracing the full lifecycle of a contract revision. We illustrate how the *Global Prescriptive Agent (Leader)* enforces constraints and how the *Follower System (CRA + Inner LVA)* iteratively refines the text. Finally, we analyze typical failure modes to discuss current limitations.

F.1 Success Cases (End-to-End Flow)

In Figure 14 and Figure 15, we display two distinct negotiation scenarios.

Case 1: The Unlimited Liability Trap. This case (Figure 14) demonstrates the correction of a high-risk liability clause.

1. **Input:** The original text contains a "blanket indemnity" clause exposing the provider to unlimited risk.
2. **Leader Analysis:** The LVA identifies this as a **Category A** risk (Q2=A). It outputs a

Fine grained Quality Metrics Breakdown

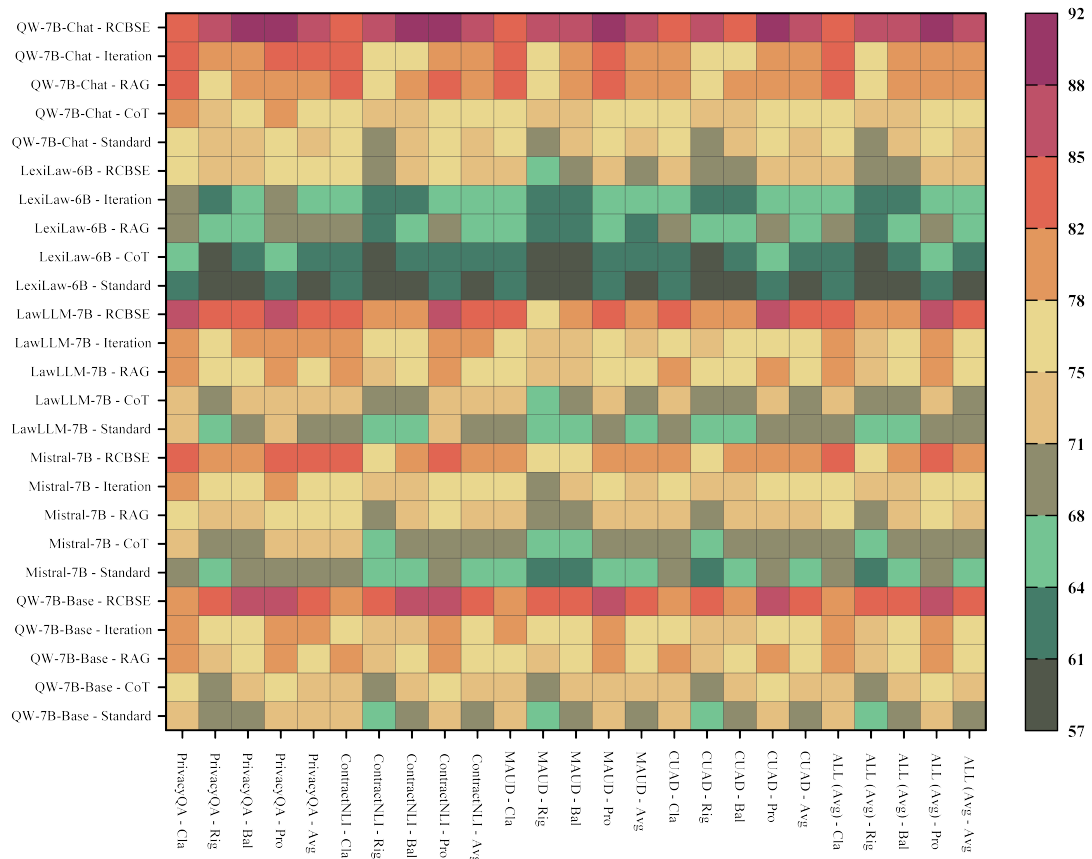


Figure 13: Heatmap of Fine-Grained Quality Metrics. Multi-model-group variants' performance across multi-tasks. Dark magenta is the best in group, dark green is the worst in group. Metric values range from 57–92.

structured tuple specifying the *Evidence* ("indemnify... from any and all claims") and a concrete *Suggestion* ("Insert a liability cap").

3. Revision Process:

- *Revision #1*: The CRA softens the language but forgets the specific cap.
- *Inner Audit*: The LVA flags that the "financial cap is still missing."
- *Final Revision*: The CRA incorporates the "12 months' fees" cap, satisfying the Leader's constraint.

Case 2: Multi-Objective Risk Resolution in Commercial Contracts. To illustrate the capability of RCBSF in handling complex commercial agreements, Figure 15 visualizes the revision trajectory of a *Software Development Services Agreement*. The input text contains three distinct high-stakes risks: (1) *Unilateral Termination*, (2) *Vendor IP Retention* (replacing standard liability issues

with a critical asset ownership risk), and (3) *Ambiguous Payment Terms*.

The revision process adheres to the hierarchical Stackelberg dynamics defined in our implementation:

1. **Global Prescription (Outer Loop):** The **Global Prescriptive Agent (GPA)** decomposes the contract text, extracting a 5-dimensional constraint vector for each of the three identified risks.
2. **Execution & Audit (Inner Loop):** The Follower system (CRA & LVA) engages in a 3-round iterative revision process.
3. **Trajectory Analysis:**
 - **Iteration #1:** The CRA successfully resolves the Payment and Termination risks but fails to address the IP ownership risks but fails to address the IP ownership risks, merely polishing the language.
 - **Fusion Feedback:** The LVA triggers a negative reward signal regarding the IP

Case Study 1: Mitigating Unlimited Liability Risks

1. Original Contract Segment (Input): "The Provider shall indemnify and hold the Client harmless from and against any and all claims, losses, damages, liabilities, and expenses arising out of or in connection with the Services, regardless of cause."

2. Leader Agent Analysis (Strategic Instruction):

- **Category:** EXCESSIVE LIABILITY
- **Location:** Section 9.1 (Indemnification)
- **Evidence:** "...indemnify... from any and all claims... regardless of cause."
- **Issue:** Creates unlimited exposure for the Provider; lacks a monetary cap or exclusion for indirect damages.
- **Suggestion:** Limit liability to the fees paid in the preceding 12 months and exclude consequential damages.
- **Q-Score:** {Q1: B, Q2: A (High), Q3: B, Q4: A}

3. Follower System Execution (Iterative):

[Revision #1 - Attempt]: "The Provider shall indemnify the Client for direct losses arising from the Provider's negligence..." → Rejected by Inner Audit (Missing Cap).

[Final Revision - Convergence]: "The Provider shall indemnify the Client for direct losses... provided that the Provider's total aggregate liability shall not exceed the total fees paid by the Client in the twelve (12) months preceding the claim. In no event shall Provider be liable for indirect or consequential damages."

Figure 14: A complete revision flow for a Liability Clause. The Leader's specific suggestion (highlighted in yellow) serves as a hard constraint that the Follower eventually satisfies in the final output (green).

clause. The Fusion module generates a specific hint: "Priority: Reassign Deliverable ownership to Client."

- **Iteration #3:** The CRA integrates all constraints, producing a finalized clause that balances fairness, clarity, and asset protection.

F.2 Failure Case Analysis

Despite the strong performance, RCBSF exhibits specific failure modes, primarily driven by context limitations and jurisdictional ambiguity.

Failure Mode A: The "Lost-in-the-Middle" Hallucination. In extremely long documents (> 15,000 tokens), the Leader Agent occasionally "hallucinates" risks by misattributing Evidence from one section to another.

- **Scenario:** A Merger Agreement where "Definition of Material Adverse Effect" (Page 5) excludes pandemics, but "Closing Conditions" (Page 80) references it.
- **Model Error:** The Leader claims "Missing Pandemic Exclusion" because it fails to re-

trieve the definition from Page 5 due to attention decay over long contexts.

- **Result:** The CRA adds a redundant exclusion clause, lowering the *Conciseness* score, although the legal risk is technically resolved.

Failure Mode B: Jurisdictional Overfitting.

The model is predominantly trained on US-centric English corpora (CUAD, MAUD), leading to bias when handling Civil Law contracts.

- **Input:** A contract governed by German Law (*BGB*).
- **Leader Suggestion:** "Add a clause explicitly stating that 'consideration' has been exchanged to ensure validity."
- **Analysis:** Under German law, "consideration" is not a requirement for contract validity (unlike in US/UK Common Law).
- **Outcome:** The system inserts a legally unnecessary "Consideration Clause," revealing a lack of jurisdiction-specific legal reasoning. This highlights the need for a "Jurisdiction Awareness" parameter in future iterations of the *h* vector.

G Prompt Engineering

To ensure reproducibility and transparency, we provide the exact system prompts used for the Global Prescriptive Agent (Leader), the Q-Score Assessment mechanism, and the Constrained Revision Agent (Follower). The design of these prompts is central to the RCBSF framework’s ability to balance risk mitigation with semantic preservation.

G.1 Global Prescriptive Agent (Leader)

The Leader Agent is responsible for the initial risk auditing. Unlike standard find and fix prompts, we enforce a strict 5-Dimensional Output Structure (see Figure 16) to ground the model’s reasoning:

- **Category:** Provides a high-level taxonomy for clustering risks.
- **Location:** Forces the model to perform "retrieval" within the context, mitigating hallucination by requiring pointer references.
- **Evidence:** Serves as a fact-checking mechanism. The model must quote the exact text causing the risk.
- **Issue:** Encodes the "Chain-of-Thought" reasoning, explaining *why* the evidence constitutes a risk.
- **Suggestion:** Provides the actionable instruction for the Follower, decoupling the "what to do" from the "how to do it."

G.2 Q-Score Assessment Mechanism

To quantify the severity of each identified risk, we employ a dedicated quantization prompt. As shown in Figure 20, the model evaluates each risk across four specific orthogonal dimensions ($Q1 \dots Q4$). This quantization allows the framework to construct the weighted risk instruction vector h .

- **Q1 (Validity/Compliance):** Assesses the risk of contract invalidation or regulatory penalties.
- **Q2 (Liability Scope):** Measures the financial exposure (e.g., unlimited liability vs. capped).
- **Q3 (Control Allocation):** Evaluates the balance of power (e.g., unilateral termination rights).
- **Q4 (Remediability):** Estimates the cost and difficulty of fixing the issue if the risk materializes.

G.3 Constrained Revision Agent (CRA)

The Constrained Revision Agent (CRA) functions as the strategic follower and execution engine within the lower level of the Risk-Constrained Bilevel Stackelberg Framework (RCBSF). Unlike standard generative models that rely solely on semantic probability, the CRA operates as a rational optimizer in a non-cooperative game setting. Its primary objective is to generate a contract variant x' that minimizes the composite loss function defined in Eq. 19:

$$\min_{x'} \mathbb{E} [\mathcal{R}(x', a_i) + \lambda_1 \mathcal{C}(x', x) - \lambda_2 \mathcal{S}(x')] \quad (19)$$

where \mathcal{R} denotes the residual risk detected by the auditor, \mathcal{C} represents the consumption cost (edit distance from the original contract), and \mathcal{S} is the semantic quality reward.

As illustrated in Figure 18, the CRA receives a structured composite instruction designed to guide this optimization:

- **Global Constraints (OUTER_HINT):** Derived from the Leader Agent (GPA), this provides the high-level risk budget and mandatory clauses (e.g., governing law, liability caps).
- **Local Gradients (INNER_HINT):** Feedback from the Local Verification Agent (LVA) acting as an adversarial critic, pointing out specific residual risks in the previous iteration (Generation N).
- **Budget Control (FUSION_HINT):** Instructions to manage the trade-off between risk mitigation and text preservation to prevent excessive modification.

Handling Local Optima. A critical challenge in iterative LLM generation is the lazy revision phenomenon, where the model converges to a local optimum by returning the original text to avoid introducing new errors. To counteract this, we implement a Dynamic Force Rewrite Mechanism. As shown in Figure 19, when the edit distance between iterations is zero ($EditDistance(x_t, x_{t-1}) == 0$), the system injects a high-priority adversarial prompt to force a substantive modification, pushing the agent out of the saddle point.

G.4 Local Verification Agent (LVA)

The Local Verification Agent (LVA) serves as the inner-loop auditor within the Stackelberg game.

It is responsible for computing the residual risk $\mathcal{R}(c_j, a_i)$ at each iteration step. Unlike the global leader which sets the direction, the LVA performs a fine-grained, clause-level inspection of the *revised* contract (Gen- n Contract).

This agent operates in a Multi-Round Revision-Audit Interaction mechanism. In each epoch, the LVA analyzes the output of the CRA and generates a structured Gen- n Audit Report. This report maps detected issues to specific risk categories and weights, providing the necessary gradient signal for the CRA to adjust its next generation. This adversarial dynamic ensures that hidden risks are exposed and resolved dynamically, addressing the Incomplete Information and Hidden Risks issues prevalent in baseline methods.

The LVA's prompt (see Figure 17) strictly enforces a structured JSON output to facilitate automated parsing and weight mapping in the next optimization cycle.

Case Study 2: Simultaneous Resolution of Three Commercial Risks

Phase 1: Input Contract (Raw Text)

"The Company may terminate this Agreement at any time, effective immediately . Regarding intellectual property, Vendor shall retain all right, title, and interest in and to any software or deliverables created hereunder. Client shall pay all invoices submitted by Vendor ."

Phase 2: Global Prescriptive Agent (GPA) Analysis *The GPA generates a 5-dimensional risk budget for each detected issue:*

- **[Risk 1] Unilateral Termination (Fairness)**
 - *Evidence:* "...terminate... at any time, effective immediately."
 - *Issue:* Immediate termination disrupts project continuity and lacks procedural fairness.
 - *Suggestion:* Require 30-day prior written notice for convenience termination.
- **[Risk 2] IP Retention (Asset Ownership)**
 - *Evidence:* "Vendor shall retain all right, title, and interest..."
 - *Issue:* Client pays for development but fails to own the work product (Work Made for Hire).
 - *Suggestion:* Transfer ownership of Deliverables to Client upon payment.
- **[Risk 3] Ambiguous Payment (Financial)**
 - *Evidence:* "...pay all invoices submitted..."
 - *Issue:* Lacks specific payment terms (e.g., Net 30), creating cash flow uncertainty.
 - *Suggestion:* Specify payment due date (e.g., within 30 days of receipt).

Phase 3: Follower System Execution (Inner Loop Dynamics)

▷ **Iteration #1 (Initial Revision by CRA):** *The agent fixes Termination and Payment but misses the IP nuance.*

"The Company may terminate this Agreement upon 30 days' prior written notice . Vendor retains ownership of the software but grants Client a license to use it. Client shall pay invoices within thirty (30) days of receipt ."

→ **LVA Audit / Fusion Hint:** *"Risk 1 (Termination) & Risk 3 (Payment) Resolved (Q = A). Critical Failure on Risk 2 (IP): 'License to use' is insufficient for custom software. Instruction: Must assign full title to Client."*

▷ **Iteration #3 (Final Converged Output):** *The CRA applies the 'Work Made for Hire' logic mandated by the Fusion Hint.*

"The Company may terminate this Agreement without cause upon providing at least thirty (30) days' prior written notice . Vendor agrees that all Deliverables shall be considered 'Work Made for Hire' and hereby assigns all right, title, and interest to Client. Client shall pay all undisputed invoices within thirty (30) days of receipt."

Figure 15: Detailed visualization of the multi-risk resolution in a commercial software context. The **GPA** successfully identifies three disparate risks (Fairness, IP, Financial). The **Inner Loop** demonstrates the system's self-correction capability: while Iteration #1 fails to fully transfer IP rights, the LVA's audit feedback forces the CRA to execute a legally binding assignment in Iteration #3, satisfying all Leader constraints simultaneously.

Prompt for Global Prescriptive Agent (Risk Extraction)

System Instruction: You are a strictly logical contract risk auditor. Please review the contract text below and extract risk points. You must analyze from 5 dimensions for each risk:

1. **category:** Specific risk classification (At least ten words or more, describe the risk category as detailed as possible).
2. **location:** Where this risk appears (e.g., "Section 1.2").
3. **evidence:** Original text quote supporting this risk. If missing, state "Missing clause".
4. **issue:** Specific description of what is wrong (the defect, ambiguity, or unfairness).
5. **suggestion:** Actionable advice on how to modify/add clause text.

Output Requirements:

- Output must be a JSON object containing a list "risk_categories".
- If the text is very short, try to identify at least 8-15 potential risks/missing elements.
- Even for short texts or definitions, analyze strictly (e.g., Are definitions vague? Is the scope clear?).

Input Context: Contract text: «<CONTRACT»» {CONTRACT_TEXT} «<END»»
Please only output JSON.

Figure 16: The structured prompt used by the Leader Agent to extract the 5-dimensional risk tuple. This structure forces the model to ground its suggestions in specific evidence.

Prompt for Local Verification Agent (Adversarial Auditor)

[System Role Definition] You are the **Inner Auditor (LVA)**. Your function is to calculate the *Residual Risk* $\mathcal{R}(c_i, a_i)$ of the generated text. You must act logically and critically. Do not be lenient.

[Audit Logic] Compare the Revised_Contract against the Risk_Definition_List. For every clause, determine if the risk is: RESOLVED, PARTIALLY_RESOLVED, or UNRESOLVED.

[Structured Output Requirement] Return a strictly formatted JSON object. This will be used to calculate the Loss Function.

```
{
  "audit_report": [
    {
      "risk_id": "R_01 (Liability)",
      "status": "UNRESOLVED",
      "severity_weight": 0.9,
      "location_quote": "...liability shall not exceed...",
      "issue_description": "Cap is still ambiguous.",
      "gradient_feedback": "Specify exact dollar amount or fee multiplier."
    },
    ...
  ],
  "global_safety_score": 0.45
}
```

Figure 17: The LVA Prompt. By enforcing a structured JSON output containing severity weights and gradient feedback, the Auditor transforms qualitative legal analysis into quantitative signals for the next optimization round.

Prompt Construction for Constrained Revision Agent (CRA)

[System Role Definition] You are the **Strategic Follower** (Optimizer) in a Stackelberg game for legal contract revision. Your goal is not just to write text, but to solve a bilevel optimization problem:

$$\min \text{Total Loss} = \text{Residual Risk}(\mathcal{R}) + \text{Modification Cost}(\mathcal{C}) - \text{Quality Reward}(\mathcal{S})$$

You must strictly adhere to the constraints passed by the Leader (GPA) and the Auditor (LVA).

[Input 1: Global Strategic Constraints (from Leader)] *Source: Outer_Hint*

- **Risk Budget:** Strict. No high-risk clauses allowed.
- **Mandatory Requirements:** Governing Law must be "Delaware"; Liability Cap must be limited to "12 months fees".
- **Key Entity:** "Alpha Product" licensing.

[Input 2: Optimization Constraints (Budget)] *Source: Fusion_Hint*

- **Edit Distance Limit:** $\leq 15\%$ deviation from original structure.
- **Style:** Formal legal English, maintaining original definitions.

[Input 3: Gradient Feedback (from Auditor)] *Source: Inner_Hint (Previous Iteration N)* > Critical Risk Detected in Section 4.2: "Indemnification clause is missing IP infringement coverage." > Suggestion: Insert standard IP indemnification language.

[Execution Task] Output the full revised contract. Do not explain. Do not summarize.

Figure 18: The composite prompt mechanism for the CRA. It translates the abstract Stackelberg theoretical constraints into executable instructions, combining global strategy (Leader), resource budget (Fusion), and specific risk gradients (Auditor).

△ Dynamic Intervention: Anti-Lazy Mechanism

Trigger Condition: $\text{EditDistance}(x_t, x_{t-1}) < \epsilon$
(Indicator: Model has converged prematurely with no effective revisions)

[Adversarial Injection Prompt] "SYSTEM ALERT: DETECTED SADDLE POINT IN OPTIMIZATION."
The previous output is identical to the input – you are NOT optimizing the objective function.

MANDATORY OVERRIDE (Non-Negotiable):

1. You **MUST** rewrite the content in Section [Detected_Risk_Location].
2. Mere wording adjustments are insufficient: you **MUST** alter the logical framework to mitigate [Risk_Category].
3. Penalty consequence for non-compliance: Penalty Score = ∞ (immediate optimization failure).

Figure 19: The Force Rewrite injection mechanism. This acts as a perturbation vector to push the CRA out of local optima (i.e., lazy revisions) — triggered when the system detects zero effective edits despite unresolved risks.

Prompt for Q-Score Assessment (Abbreviated)

System Instruction: Below is a contract clause (or a risk point). Based solely on this content, please choose an option from A/B/C for each of the 4 dimensions.

Question 1: Nature of Legal Consequences (Validity / Compliance)

- **A (High):** Likely to invalidate the contract, trigger criminal liability, or result in license revocation.
- **B (Medium):** Clause partially invalid or subject to administrative fines/rectification.
- **C (Low):** Disputes over interpretation; does not affect validity or involve penalties.

Question 2: Scope and Limit of Liability (Exposure)

- **A (High):** No liability cap; covers indirect losses; "bear all consequences."
- **B (Medium):** Cap exists but is high (>100% price); includes some third-party claims.
- **C (Low):** Clear cap (\leq 100% price); direct losses only; excludes consequential damages.

Question 3: Allocation of Rights, Obligations, and Control

- **A (High):** Counterparty has unilateral authority (termination, pricing) with no veto/remedy.
- **B (Medium):** Unilateral rights exist but require notice, cure period, or objective conditions.
- **C (Low):** Rights are balanced; major matters require joint consent.

Question 4: Remediability and Long-Term Impact

- **A (High):** Difficult to restore state; permanent loss of IP or trade secrets.
- **B (Medium):** Remediable via substantial renegotiation or multi-layer approvals.
- **C (Low):** Remediable via simple supplementary agreement; short-term impact.

Output Format: Output **only one JSON object**: {"Q1": "A", "Q2": "B", "Q3": "C", "Q4": "C"}

Figure 20: The definition of the Q-Score metrics. Each dimension is graded on a 3-point scale to compute the final risk weight.