Better Call CLAUSE: A Discrepancy Benchmark for Auditing LLMs Legal Reasoning Capabilities

Anonymous ACL submission

Abstract

The rapid integration of large language models (LLMs) into high-stakes legal work has exposed a critical gap: no benchmark exists to systematically stress-test their reliability against the nuanced, adversarial, and often subtle flaws present in real-world contracts. To address this, we introduce CLAUSE, a first-of-its-kind benchmark designed to evaluate the fragility of an LLM's legal reasoning. We study the capabilities of LLMs to detect and reason about fine-grained discrepancies by producing over 7500 real-world perturbed contracts from foundational datasets like CUAD and ContractNLI. Our novel, persona-driven pipeline generates 10 distinct anomaly categories, which are then validated against official statutes using a Retrieval-Augmented Generation (RAG) system to ensure legal fidelity. We use CLAUSE to evaluate leading LLMs' ability to detect embedded legal flaws and explain their significance. Our analysis shows a key weakness: these models often miss subtle errors and struggle even more to justify them legally. Our work outlines a path to identify and correct such reasoning failures in legal AI.

1 Introduction

The integration of Large Language Models (LLMs) into high-stakes legal work has outpaced the development of benchmarks that can audit their reliability against nuanced contractual flaws. This gap is critical, as a single undetected defect can have catastrophic consequences. In *Perini Corp. v. Greate Bay Hotel & Casino, Inc.*, for instance, the simple omission of a consequential damages waiver resulted in a $14.5 million liability, a sum over twenty times the contract's fee (McElroy, Deutsch, Mulvaney & Carpenter, LLP, 2021). State-of-the-art models, optimized for fluency, are not trained to detect such legally significant risks, particularly those arising from the absence of text. Existing evaluation methods fail to probe these fragile reasoning capabilities (Hu et al., 2025; Mishra et al., 2025), creating an urgent need for a new class of benchmark to stress-test LLMs against the adversarial and context-aware challenges of legal practice. This research directly confronts the ongoing debate about AI's readiness for legal integration. By evaluating modern LLMs against our CLAUSE benchmark, we aim to provide empirical data to help answer the fundamental question: *are these models truly reliable enough for such high-stakes work?*

To answer this question and address this gap by developing CLAUSE, a novel benchmark designed to systematically evaluate the fragility of Large Language Models' legal reasoning capabilities. At its core, CLAUSE constructs realistic and targeted contradictions by perturbing over 7,500 real-world contracts drawn from foundational sources such as CUAD(Hendrycks et al., 2021) and ContractNLI(Koreeda and Manning, 2021). The framework defines ten distinct perturbation categories that span both statutory compliance failures (legal contradictions) and internal document inconsistencies (in-text contradictions). This taxonomy allows researchers to perform fine-grained analysis of how models respond to subtle but consequential changes that could alter contract enforceability or introduce hidden liabilities.

To ensure the benchmark's fidelity and usefulness, we implement a multi-stage generation and validation pipeline. Each perturbation is first subjected to a retrieval-augmented generation (RAG) stage that grounds the modification in actual statutory language and related legal context, producing initial assessments of semantic significance and coherence. Automated quality checks then filter and classify contradictions, followed by expert human review to verify legal relevance, contradiction strength, and contextual correctness. All artifacts are accompanied by a rich metadata schema capturing provenance, reasoning chains, citation ground-

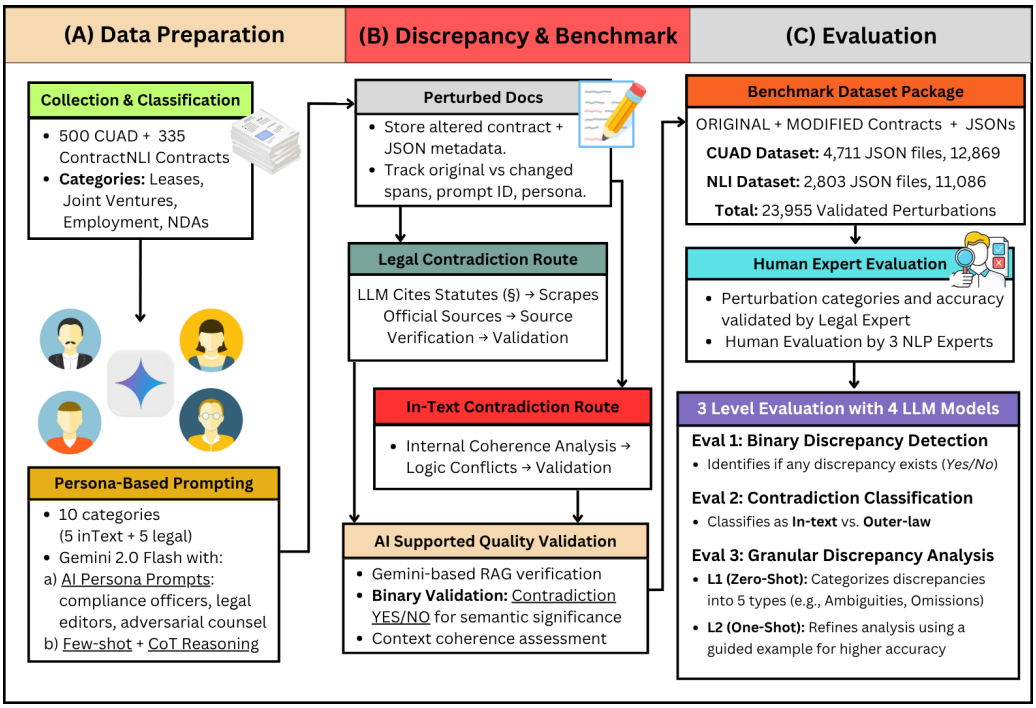

Figure 1: CLAUSE pipeline: Data Generation, AI grounding, and expert validation of legal and Evaluation & Analysis.

ing, contradiction type, and confidence measures. This layered process yields high-quality labeled examples that surface both obvious and brittle failure modes in model reasoning.

Beyond creating examples, CLAUSE is intended as a diagnostic and comparative platform for the research community. It enables quantification of model brittleness, supports evaluation of prompting and reasoning strategies, and supplies a standard substrate for developing and assessing mitigation techniques. By exposing where and how LLMs fail on realistic legal perturbations, the benchmark helps bridge the gap between fluent generation and reliable legal interpretation, accelerating progress toward safer, more transparent deployment in real-world contractual workflows. Our main contributions are as follows:

1) First benchmark for fine-grained legal contradictions with ten perturbation types grounded in statutory requirements.
2) Automated AI pipeline generating and validating contradictions before expert review.
3) Metadata framework capturing context, reasoning, and strength for reproducibility.
4) Extensive evaluation uncovering systematic legal reasoning weaknesses under varied prompting.

Figure 1 shows the CLAUSE pipeline: generation, grounding, and expert validation of legal and in-text contradictions.

2 Our CLAUSE Dataset

The CLAUSE benchmark advances legal reasoning evaluation through systematic analysis of contractual discrepancies. Built upon two established legal corpora, the Contract Understanding Atticus Dataset (CUAD) and ContractNLI, our framework represents the first fully AI-generated legal contradiction dataset. Using large language models for both generation and validation, we systematically enrich existing contracts through controlled perturbation. The core innovation lies in our comprehensive taxonomy of modifications, featuring distinct perturbation categories that span two fundamental dimensions:

- **Legal Contradictions**: Modifications that conflict with statutory requirements, regulatory standards, or legal precedents, creating compliance issues that could render clauses unenforceable.
- **InText Contradictions**: Modifications that create logical inconsistencies within the document itself, where different sections provide conflicting information or obligations.

Each dimension implements five fundamental perturbation types: (a) **Ambiguity**: Introduction of vague, unclear, or contradictory language that creates uncertainty in interpretation, making contract terms susceptible to multiple conflicting meanings; (b) **Omission**: Deliberate removal or absence

of critical information, clauses, or terms that are necessary for complete understanding or legal enforceability of contract obligations; (c) **Misaligned Terminology**: Inconsistent use of defined terms throughout the document, or introduction of terminology that conflicts with established legal definitions or industry standards; (d) **Structural Flaws**: Modifications that disrupt the logical organization, hierarchy, or cross-referencing within the contract, creating navigational confusion or broken dependencies between clauses; and (e) **Inconsistencies**: Direct contradictions between different sections of the contract where statements, obligations, or terms are mutually exclusive or logically incompatible.

This structured approach, driven by AI generation and verification, enables fine-grained evaluation of LLM capabilities in both external legal compliance verification and internal document coherence assessment.

CLAUSE **Generation** Our pipeline applies a unified perturbation framework to the contracts. We extract full text and structural metadata (clause boundaries, section headers, cross-references) to preserve document integrity during modification. Contextually appropriate contradiction locations are identified using prompts across ten perturbation types. Three complementary prompting strategies ensure legal plausibility: persona-based roles (compliance officers, legal editors), few-shot examples, and chain-of-thought reasoning.

Each perturbation produces a modified contract with structured JSON metadata containing original/altered text, perturbation type, document location, and model-generated explanations. Legal contradictions include law citations and statutory context; in-text contradictions include conflicting clause references and logical justifications. The unified process is applied to both datasets ensuring consistent output quality and metadata format, supporting downstream detection accuracy and legal reasoning analysis tasks.

CLAUSE **LLM Verification** To ensure dataset quality, we implement an AI-driven verification pipeline using specialized Gemini prompts with retrieval-augmented context. For each perturbation, the system analyzes original text, modified clause, and explanation, augmented with scraped legal references or related document clauses.

The verification performs rigorous binary classification (YES/NO) determining whether modifications represent strong, meaningful contradictions meeting our quality threshold. It evaluates semantic significance, contradiction strength, and contextual coherence. Only perturbations receiving YES classification are retained through automatic filtering, producing a clean, high-quality benchmark where each perturbation is explicitly validated for contradiction strength and interpretability.

Dataset Distribution (CUAD + NLI)				
	InText		Legal	
Category	**Files**	**Pert.**	**Files**	**Pert.**
Ambiguity	741	2,417	805	2,472
Inconsistency	749	2,357	830	2,711
Misalign. Ter.	772	2,636	834	2,864
Omission	754	2,498	698	2,111
Structural	760	2,540	571	1,349
Totals	**3,776**	**12,448**	**3,738**	**11,507**
Dataset Totals			**CUAD**	**NLI**
Total Contracts			4,711	2,803
Total Perturbations			12,869	11,086

Table 1: Comprehensive dataset statistics showing five perturbation categories across InText and Legal variants, with dataset-wise breakdowns.

CLAUSE **Statistics** Table 1 presents comprehensive statistics for both components, showing the distribution of JSON files and perturbations across all ten categories.

The dataset exhibits several notable characteristics. First, each JSON file contains multiple perturbations, with CUAD averaging 2.73 perturbations per file and the NLI split averaging 3.96 perturbations per file. Second, the distribution across categories is relatively balanced, though some variation exists due to the inherent complexity of generating specific perturbation types for different contract structures. Third, the legal categories generally contain fewer perturbations than their inText counterparts, reflecting the additional constraints imposed by statutory validation requirements.

Expert Pair	Cohen's Kappa	Jaccard Coefficient
$\alpha_{1,2}$	0.97	0.97
$\alpha_{2,3}$	0.99	0.99
$\alpha_{1,3}$	0.97	0.97
$\alpha_{1,2,3}$	0.98	0.98

Table 2: Inter-rater agreement metrics. Cohen's Kappa measures agreement between raters accounting for chance agreement, while Jaccard Coefficient measures similarity between finite sample sets of contradiction identifications.

CLAUSE **Human Verification** To validate quality and legal accuracy, we first engaged a legal expert to confirm the real-world relevance of our perturbation categories. Subsequently, 3 NLP experts evaluated 25% of our dataset (13,275 perturbation

instances) using structured binary classification to determine meaningful legal contradictions.

Our human verification demonstrates exceptional quality with 98.58% contradiction validation rate and strong inter-rater reliability shown in Table 2. The evaluation examined semantic significance, contradiction strength, and contextual coherence, ensuring our benchmark contains high-quality examples for testing LLM legal reasoning capabilities.

3 Discrepancy Detection Tasks

To rigorously assess the ability of large language models (LLMs) to detect and reason about legal discrepancies in contracts, we define a structured suite consisting of four hierarchical tasks over a dataset $\mathcal{D} = \{d_1, d_2, \ldots, d_N\}$ of perturbed legal documents. Each d_i may or may not contain a deliberately introduced discrepancy. We denote:

- $y_i \in \{0, 1\}$. Binary label indicating whether any discrepancy is present in document d_i.
- $t_i \in \mathcal{T}$, where

$$\mathcal{T} = \{\text{InText},\ \text{Legal}\}.$$

  Type of contradiction: within the document or against external law.
- $s_i \in \mathcal{S}$, where

$$\mathcal{S} = \left\{ \begin{array}{l} \text{ambiguity},\ \text{omission}, \\ \text{misaligned_terminology}, \\ \text{inconsistency},\ \text{structural_flaw} \end{array} \right\}.$$

  Fine-grained discrepancy subtype.
- ℓ_i. External legal reference violated by the discrepancy (only present if $t_i = \text{legal}$).
- x_i. Text span in d_i where the discrepancy occurs.

We now define four levels of evaluation:

Task_1: Binary Discrepancy Detection. Given d_i, predict $y_i \in \{0, 1\}$ indicating whether any discrepancy is present.

Task_2: Contradiction Type Classification. Given d_i, predict $c_i \in \mathcal{T} \cup \{\text{none}\}$ indicating whether the contradiction is in-text, legal (external), or absent.

Task_3: Explanation-Based Discrepancy Detection. Given a document d_i, the model is required to identify a discrepancy span x_i, generate a natural language explanation e_i justifying why x_i constitutes a discrepancy, and the violated legal reference l_i (if $t_i = \text{Legal}$ where $t_i \in \mathcal{T}$ is the contradiction type), or null otherwise. The predicted tuple (x_i, e_i, l_i) is compared against ground truth annotations (x_i^*, e_i^*, l_i^*) for evaluation.

4 Modeling Approaches

We design a tiered set of modeling strategies that correspond directly to the three evaluation tasks outlined in Section 3. Each task is implemented using tailored prompting strategies, with increasing granularity and reasoning complexity from Eval_1 through Eval_3. Below, we describe the modeling design for each evaluation level.

Discrepancy Detection *Eval_1* is formulated as a binary classification task where the model must determine if a document contains any form of discrepancy, without specifying its type or location. To test this in a purely zero-shot setup, we use a minimal prompt that *'asks whether there are any discrepancies present in the document'* and instructs the model to answer only with "Yes" or "No" . For this evaluation, models are provided with both the original and perturbed documents to facilitate the detection task.

Discrepancy Type Classification. *Eval_2* introduces a three-way classification task, where the model is asked to categorize the document as one of: (1) in-text contradiction, (2) outer-law contradiction, or (3) no contradiction. The prompt provides clear definitions of each category and explicitly instructs the model to choose one label. This task requires the model to reason over both internal document coherence and compatibility with U.S. legal standards. Compared to Eval_1, this prompt includes richer task framing and imposes stricter answer constraints. For this evaluation we provide the models only the perturbed documents.

Discrepancy Span Detection with Explanation Generation: *Eval_3* requires the model to identify all discrepancy spans in a legal document and provide a natural language explanation for each. The model must output a structured list of objects, each containing two fields: the `text` of the problematic span and an `explanation` detailing why it is a discrepancy. For documents containing legal discrepancies, the model is also asked to output the contradicting `law citation`. We divide *Eval_3* into zero-shot (**L1**) and one-shot (**L2**) evaluation levels. At both levels, models are tasked with identifying spans and providing explanatory justifica-

tions. Distinct prompts are designed for in-text and legal contradiction documents, each including definitions of the relevant contradiction type and the five primary discrepancy categories. The prompts for legal contradictions specifically request the law citation in addition to the span and explanation.

5 Experimental Evaluation

Our experiments are driven by a fundamental question that underpins the excitement around "AI in Law": Are today's models truly robust enough for reliable application in the high-stakes legal domain? To explore this, we investigate: (RQ1) the baseline performance of current LLMs on structured legal discrepancy detection; (RQ2) how prompt design affects a model's ability to identify and explain inconsistencies; (RQ3) the extent to which models generalize across different contradiction types ; and (RQ4) how reliably models can produce accurate, structured outputs such as legal citations and explanations.

Evaluation: Our evaluation spans three tasks: discrepancy and it's type detection, and discrepancy span detection with explanation generation. For evaluation we have used a temperature 0.2 across all the prompts. For Eval_1 and Eval_2, we use exact label matching. For Eval_3, we define three evaluation metrics: `location_alignment`, `explanation_match`, and `law_match`. Each of these evaluation metrics are explained below:

`location_alignment`. First, let D_{GT} be the set of all ground-truth discrepancy text spans and D_P be the set of all predicted discrepancy text spans for a given document. For any discrepancy text span $d \in D_{\text{GT}} \cup D_P$, we define a function $f_s(d)$ that tokenizes the text into a set of normalized (lowercase, stripped) sentences.

We then construct an undirected graph $G = (V, E)$ where the vertices are the individual discrepancies, $V = D_{\text{GT}} \cup D_P$. An edge (d_i, d_j) exists between a ground-truth span $d_i \in D_{\text{GT}}$ and a predicted span $d_j \in D_P$ if and only if their sentence sets have a non-empty intersection:

$$E = \left\{ (d_i, d_j) \mid \begin{array}{l} d_i \in D_{\text{GT}}, d_j \in D_P, \text{ and} \\ f_s(d_i) \cap f_s(d_j) \neq \emptyset \end{array} \right\} \quad (1)$$

The alignment is determined by finding the connected components of this graph. A "match group," which constitutes a **True Positive** alignment, is any connected component that contains at least one ground-truth span and one predicted span.

Here, a **False Negative** is a ground-truth span $d_{\text{GT}} \in D_{\text{GT}}$ that does not belong to any such match group and **False Positive** is a predicted span $d_P \in D_P$ that does not belong to any such match group.

`explanation_match`. The LLM, acting as a legal analyst, receives the combined explanations and all relevant GT contexts. It returns a structured JSON object with scores between 0-5 for: **Accuracy** (factual correctness), **Completeness** (coverage of key points), **Clarity** (ease of understanding), **Legal Reasoning** (soundness of legal logic), and a final binary **Match** flag that determines the alignment outcome.

This process is adapted for two contradiction types. Ground-truth context are provided during the judging process for better context aware outputs: **a) In-Text Contradictions**: The GT context includes the *original text*, *changed text*, and *contradicted text*; **b) Legal Contradictions**: The context is augmented to include the *original text*, *changed text*, *contradicted text*, plus the specific *contradicted law* and a *law explanation*.

`law_match`. This metric evaluates the alignment of legal citations by using a Gemini model prompted as an expert legal paralegal. To determine if a predicted citation corresponds to the same underlying law as the ground truth, the model performs a semantic comparison using the ground-truth law explanations for context. This approach is designed to establish a fundamental match rather than a literal one, accounting for variations in textual phrasing and resulting in a binary score (0 or 1). All evaluations assume the current date is Sept. 2025.

For both `location_alignment` & `law_match` we have used GPT 4o and Gemini-2.5-Flash as judges.

Models: We have evaluated four language models: GPT-4o-mini, Gemini 2.0 Flash (Team et al., 2025), Gemini 2.5 Flash(Comanici et al., 2025), and Llama 3.3 70b Instruct(Grattafiori et al., 2024). For consistency and reproducibility, all models were evaluated with standardized generation parameters. Hyperparameters mentioned in Appendix.

Metrics: For the Eval_1 and Eval_2, we report standard metrics: Accuracy, Precision, Recall, and F1-Score. For Eval_3, `location_alignment` is evaluated using ROUGE, METEOR, and BERTScore, while `explanation_match` is assessed via qualitative 0-5 scores for Accuracy, Clarity, Completeness, and Legal Reasoning.

Category	GPT-4o-mini				Gemini-2.0				Gemini-2.5				Llama-3.3			
	Acc	Pre	Rec	F1	Acc	Pre	Rec	F1	Acc	Pre	Rec	F1	Acc	Pre	Rec	F1
								CUAD								
Amb_{Leg}	32.9	36.4	47.6	41.2	**51.2**	**50.4**	76.5	60.8	48.1	48.7	**91.8**	**63.7**	40.2	43.5	69.4	53.5
Amb_{inT}	34.5	38.0	51.3	43.6	**53.0**	**51.6**	80.2	62.8	48.1	48.8	**91.9**	**63.8**	41.2	44.2	71.5	54.7
$\text{Incon}_{\text{Leg}}$	33.4	36.6	50.7	42.5	**51.1**	**49.7**	77.1	60.5	48.1	48.7	**92.4**	**63.8**	40.6	43.9	69.9	53.9
$\text{Incon}_{\text{inT}}$	34.5	38.4	51.7	44.0	**53.4**	**52.1**	80.5	**63.3**	47.1	48.2	**90.1**	62.8	39.9	43.3	68.8	53.1
$\text{MisTerm}_{\text{Leg}}$	36.0	39.3	53.8	45.4	**53.7**	**52.0**	81.4	**63.5**	47.4	48.6	**90.3**	63.2	40.6	43.8	70.0	53.9
$\text{MisTerm}_{\text{inT}}$	32.2	36.2	46.9	40.8	**50.4**	**50.2**	74.4	60.0	47.7	48.6	**91.4**	**63.4**	40.2	43.7	68.9	53.5
Omis_{Leg}	33.5	36.5	49.0	41.8	**50.3**	**49.4**	75.3	59.7	48.1	48.8	**92.1**	**63.8**	39.1	42.8	67.0	52.2
Omis_{inT}	33.7	36.9	48.7	42.0	49.0	48.8	72.1	58.2	**49.8**	**49.6**	**95.8**	**65.4**	39.4	43.0	67.5	52.5
$\text{StrFlaw}_{\text{Leg}}$	32.6	27.3	62.3	37.9	45.8	36.3	84.9	50.9	**48.5**	**49.1**	**92.5**	**64.2**	9.2	7.0	6.8	6.9
$\text{StrFlaw}_{\text{inT}}$	35.9	39.3	53.3	45.2	**54.7**	**52.8**	83.2	**64.6**	48.1	48.8	**92.0**	63.8	39.6	43.3	67.8	52.8
								NLI								
Amb_{Leg}	29.4	26.9	69.3	38.7	**49.2**	**35.2**	69.0	46.6	36.8	32.6	**90.2**	**47.9**	26.4	11.3	18.8	14.1
Amb_{inT}	33.6	29.0	91.7	44.1	**54.0**	**37.3**	89.7	**52.7**	36.0	30.5	**97.1**	46.4	28.3	12.0	24.0	16.0
$\text{Incon}_{\text{Leg}}$	32.7	30.0	74.4	42.8	**52.9**	**40.0**	78.6	**53.0**	40.0	35.6	**95.8**	51.9	28.2	15.2	24.6	18.8
$\text{Incon}_{\text{inT}}$	35.0	30.7	90.6	45.9	**54.8**	**39.3**	89.1	**54.5**	38.0	32.8	**98.5**	49.1	26.9	10.9	19.6	14.0
$\text{MisTerm}_{\text{Leg}}$	32.6	30.4	76.6	43.6	**53.4**	**40.6**	79.8	**53.8**	40.0	35.6	**95.2**	51.9	26.7	12.9	20.2	15.8
$\text{MisTerm}_{\text{inT}}$	34.1	30.1	88.6	45.0	**51.7**	**36.4**	79.2	**49.9**	37.4	32.3	**97.4**	48.5	27.8	12.4	22.7	16.0
Omis_{Leg}	24.9	20.7	66.4	31.6	**45.7**	**26.9**	62.6	37.6	32.2	26.7	**91.6**	**41.3**	26.0	6.8	14.5	9.3
Omis_{inT}	33.3	29.6	88.8	44.4	**51.0**	**35.4**	77.2	**48.5**	36.8	31.7	**96.5**	47.8	26.2	9.6	17.4	12.4
$\text{StrFlaw}_{\text{Leg}}$	34.8	32.0	85.2	46.5	**54.1**	**40.8**	82.8	**54.6**	38.6	34.5	**93.4**	50.4	28.8	15.9	26.4	19.8
$\text{StrFlaw}_{\text{inT}}$	35.7	31.0	**94.9**	46.7	**55.8**	**39.7**	93.8	**55.8**	36.0	31.0	94.5	46.7	27.4	11.3	21.1	14.7

Table 3: Evaluation 1 (binary discrepancy detection) results across all models and categories for CUAD and NLI datasets. Metrics: Accuracy (Acc), Precision (Pre), Recall (Rec), and F1-Score (F1). Category abbreviations: Amb = Ambiguity, Incon = Inconsistencies, MisTerm = Misaligned Terminology, Omis = Omission, StrFlaw = Structural Flaws, $_{\text{Leg}}$ =Legal, $_{\text{inT}}$ =inText. All values are percentages.

5.1 Findings: Results & Analysis

CLAUSE poses a significant **challenge**: Discrepancy detection remains challenging across datasets and discrepancy categories, with low precision, recall, F1 score, and semantic and syntax matching scores.

How does model performance on binary discrepancy detection vary across different model architectures and datasets?

As shown in Table 3, a clear performance hierarchy emerges: Gemini-2.5 leads with the highest F1-scores (63%+ on CUAD), followed by Gemini-2.0, while GPT-4o-mini and Llama-3.3 lag. Gemini-2.5's top performance is driven by a distinct high-recall (90%+), low-precision strategy, whereas other models exhibit a more balanced profile. Consistently, all models perform better on the CUAD dataset than on the more complex NLI dataset, as exemplified by Gemini-2.5's F1-score for Inconsistencies_Legal dropping from 63.8% (CUAD) to 51.9% (NLI).

Which contradiction types prove most challenging, and is there a consistent performance gap between *intext* and *legal* discrepancies?

While no single category is universally hardest in Table 3, performance on *legal* discrepancies is often marginally lower than on their *in_text* counterparts. A clear trend of difficulty is seen in the *Omission_Legal* category on the NLI dataset, which proves exceptionally challenging for GPT-4o-mini (31% F1) and Llama-3.3 (9.3% F1). A similar anomaly is Llama-3.3's 6.9% F1-score on CUAD's *Structural_Flaws_Legal*. This suggests that identifying legally significant absences and complex structural errors are key weaknesses for some architectures.

Cat.	GPT		Gem2.0		Gem2.5		Llama	
	C	N	C	N	C	N	C	N
Amb.	**53.3**	68.4	51.5	**71.6**	52.2	48.4	50.2	55.2
Inc.	54.0	66.2	**55.1**	**73.3**	51.4	51.0	50.9	56.4
Str.	**63.9**	56.5	52.3	**58.9**	47.3	47.9	50.5	52.2
Mis.	**52.8**	58.7	53.5	**66.0**	49.8	49.6	51.0	55.3
Omi.	51.4	**67.4**	**52.3**	63.0	50.0	54.8	50.5	51.1

Table 4: Eval. 2 (contradiction type classification) accuracy. Cat.: Amb.=Ambiguity, Inc.=Inconsistencies, Str.=Structural Flaws, Mis.=Misaligned Terminology, Omi.=Omission. C=CUAD, N=NLI datasets. Values are percentages.

How does the accuracy of LLMs in classifying contradiction types vary across different model architectures, datasets, and specific categories of legal discrepancies?

The results from Eval_2 in Table 4 show significant variance, with performance depending on both the model and dataset. GPT-4o-mini performs strongest on CUAD (63.9% in Structural Flaws), whereas Gemini-2.0 leads on the more difficult NLI dataset (73.3% in Inconsistencies). The increased

difficulty of NLI is a clear trend across models, illustrated by Gemini-2.5's accuracy on Ambiguity dropping from 52.2% (CUAD) to 48.4% (NLI). This highlights that while models possess aptitudes for certain discrepancy structures, robust generalization in classification remains a significant challenge.

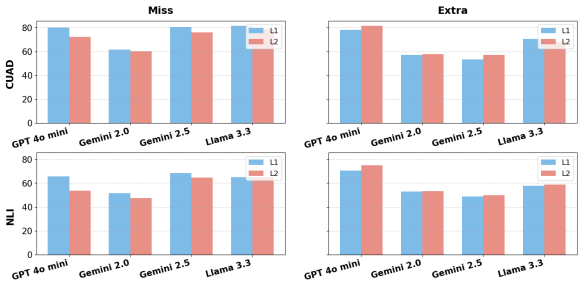

Figure 2: Comparison of model performance on CUAD and NLI datasets across L1 and L2 levels. The top two plots show Miss and Extra metrics for CUAD, while the bottom two correspond to NLI.

We analyze model performance using two key error metrics in Fig. 2 : the Miss rate, which is the fraction of ground-truth perturbations models failed to get aligned with any predictions after the `location_alignment`, the reverse logic goes for the Extra rate.

How does one-shot prompting (L2) versus zero-shot (L1) impact model error rates, and how do these patterns vary across models and datasets? A clear trade-off emerges in Fig. 2: L2 prompting lowers the Miss rate (better detection) but increases the Extra rate (more false positives) across all datasets. This sensitivity is most pronounced in GPT-4o-mini and Gemini-2.5, while Gemini-2.0 and Llama-3.3 show more stable performance. Crucially, the NLI dataset consistently results in higher error rates for all models, confirming its status as the most challenging benchmark for both detection and precision. For complete miss and extra rate across categories please refer to Table 8 in the appendix.

How accurately do LLMs identify the textual location of legal discrepancies, and what does this location_alignment performance reveal about their capabilities?

In Table 5, the `location_alignment` results show three key trends: 1) Gemini-2.0 is the clear leader (e.g., 75.4 R_1 on NLI), consistently outperforming even the more advanced Gemini-2.5. 2) All models universally score higher on the NLI dataset than on CUAD, suggesting NLI's discrepancy structures are more textually distinct. 3) One-shot (L2)

Data	L1				L2			
	R_1	R_2	R_L	M	R_1	R_2	R_L	M
	GPT-4o-mini							
CUAD	61.4	60.7	60.6	52.0	62.4	61.6	61.6	53.1
NLI	67.6	66.6	66.7	59.8	68.9	67.9	67.9	61.4
	Gemini-2.0							
CUAD	72.8	71.9	71.5	66.5	72.6	71.7	71.4	66.2
NLI	74.9	73.5	73.6	71.3	75.4	74.0	74.0	71.0
	Gemini-2.5							
CUAD	67.1	65.3	65.0	60.5	64.2	62.1	62.5	57.4
NLI	71.6	69.5	69.5	68.3	72.1	69.9	69.9	68.9
	Llama-3.3							
CUAD	66.6	65.6	65.0	58.0	66.6	65.6	65.1	57.9
NLI	71.4	70.0	69.7	65.4	71.2	71.1	70.0	65.1

Table 5: Eval. 3 location alignment metrics averaged across all categories. L1=Level 1 (zero-shot), L2=Level 2 (one-shot). R_1=ROUGE-1, R_2=ROUGE-2, R_L=ROUGE-L, M=METEOR. All values are percentages.

prompting offers inconsistent and sometimes negative effects; most notably, it degrades Gemini-2.5's performance on CUAD (R_1 drops from 67.1 to 64.2). This counter-intuitive result indicates that single-example guidance does not reliably improve, and can even hinder, precise location detection. For complete `location_alignment` results across categories refer to Table 9 in the appendix.

Model	CUAD				NLI			
	L1		L2		L1		L2	
	GPT	Gem	GPT	Gem	GPT	Gem	GPT	Gem
	GPT-4o-mini							
Accuracy	2.6	2.1	2.9	2.5	2.9	2.5	2.8	2.5
Clarity	3.7	4.0	3.9	4.2	3.9	4.2	3.8	4.1
Complete	2.0	1.4	2.3	1.7	2.3	1.7	2.2	1.7
Legal Rea.	2.6	2.3	2.9	2.5	2.9	2.5	2.8	2.4
	Gemini-2.0							
Accuracy	3.0	2.6	2.9	2.5	2.9	2.5	3.0	2.7
Clarity	4.0	4.4	3.9	4.2	3.9	4.2	4.0	4.2
Complete	2.5	1.8	2.4	1.9	2.4	1.9	2.6	1.9
Legal Rea.	3.0	2.9	2.9	2.6	2.9	2.6	3.0	2.7
	Gemini-2.5							
Accuracy	3.0	2.4	2.9	2.4	2.9	2.4	2.9	2.4
Clarity	4.0	4.3	3.9	4.2	3.9	4.2	3.9	4.1
Complete	2.5	1.9	2.6	2.1	2.6	2.1	2.6	2.1
Legal Rea.	3.1	3.0	3.0	2.9	3.0	2.9	3.0	2.9
	Llama-3.3							
Accuracy	2.4	2.2	2.7	2.3	2.6	2.3	2.6	2.5
Clarity	3.5	4.1	3.8	4.0	3.7	4.1	3.7	4.1
Complete	1.8	1.4	2.1	1.6	2.1	1.5	2.3	1.6
Legal Rea.	2.4	2.2	2.6	2.2	2.6	2.2	2.5	2.3

Table 6: Averaged explanation quality metrics across all categories. Judges: GPT4o (GPT), Gemini 2.5 (Gem). Here, the metrics Complete = Completeness, Legal Rea. = Legal Reasoning are abbreviated. All values on a scale of 0-5.

What is the overall quality of legal explanations generated by LLMs, and what factors influence this quality?

As seen in Table 6, the quality of legal explanations is defined by a critical trade-off between fluency and substance. Models universally generate text with high Clarity (often 4.0+) but low Complete-

ness (<2.0), indicating fluent but shallow reasoning. Model architecture is a primary factor, establishing a clear performance hierarchy (Gemini > GPT-4o-mini > Llama-3.3) in substantive metrics like Accuracy and Legal Reasoning. Prompting strategy has a negligible impact, as L2 prompting offers minimal, inconsistent gains. For complete `explanation_match` results across categories refer to Table 10 in the appendix.

Model	CUAD				NLI			
	GPT4o		Gem2.5F		GPT4o		Gem2.5F	
	L1	L2	L1	L2	L1	L2	L1	L2
GPT4o mini	5.4	6.7	5.0	6.6	3.6	5.4	3.3	4.7
Gemini 2.0	7.9	7.1	7.6	6.6	3.2	3.5	3.4	3.6
Gemini 2.5	12.2	13.4	13.7	13.5	7.4	10.8	8.7	12.3
Llama 3.3	9.2	10.5	8.4	9.4	6.3	8.9	6.2	8.6

Table 7: Average *Law_Match* percentage averaged across categories where datasets (CUAD, NLI), levels (**L1**, **L2**), and judges (GPT4o vs Gemini 2.5).

To what extent can current LLMs accurately perform semantic matching of legal citations, and how is this capability modulated by model architecture, dataset complexity, and prompting strategy?
The ability of LLMs to semantically match legal citations is extremely limited, with even the best-performing model, Gemini 2.5, achieving scores below 14% (Table 7. A clear performance hierarchy is evident, with Gemini 2.5 as the leader and Gemini 2.0 as the weakest, indicating highly specialized model capabilities. Performance consistently degrades from the CUAD to the more complex NLI dataset across all models. While L2 prompting provides a modest uplift, particularly for stronger models like Gemini 2.5 and Llama 3.3, and this is primarily because AI models struggle to extract the external knowledge needed to obtain law citations.

Our analysis answers the four primary research questions through targeted empirical tests. We establish baseline performance and generalization (RQ 1, 3) by evaluating binary detection and contradiction classification accuracy across models and discrepancy types. The influence of prompt design (RQ2) is measured by comparing error rates between L1 and L2 settings. Finally, we assess the reliability of generating structured outputs (RQ4) through granular evaluations of location alignment, explanation quality, and law citation matching.

6 Comparison with Related Works

Recent legal NLP advances have been propelled by datasets like CUAD (Hendrycks et al., 2021) and LEDGAR (Tuggener et al., 2020), which focus on clause labeling and categorization, as well as LexGLUE (Chalkidis et al., 2022), ContractNLI (Koreeda and Manning, 2021), and CaseHOLD (Zhong et al., 2020), which target tasks such as entailment and case retrieval. However, these resources rely on manual annotation and do not assess model reasoning under targeted perturbations. While recent work explores hallucination in legal text (Shen et al., 2023) and LegalBench (Zheng et al., 2023) broadens task diversity, they do not simulate document-level inconsistencies or robustness to semantic edits. Retrieval-augmented generation (RAG) has shown promise in open-domain QA (Lin et al., 2022; Ji et al., 2023), but is rarely applied to legal contradiction detection.

CLAUSE is among the first fully automated, AI-driven legal benchmarks, using LLMs for both generation and validation. This scalable pipeline enables the creation of high-quality contradiction data, moving beyond static classification to simulate and verify realistic legal edits. Our approach uniquely bridges gaps in legal QA and hallucination detection, demonstrating the potential of automated dataset creation for complex legal reasoning tasks.

7 Conclusion

In this work, we present CLAUSE, a benchmark designed to evaluate legal reasoning in large language models via systematic contradiction detection. By combining persona-driven generation with RAG-based validation, we show that automated methods can produce high-quality, legally-grounded datasets. Our evaluations highlight significant gaps in current LLMs' abilities to detect and explain nuanced contractual inconsistencies, revealing key challenges for legal AI. CLAUSE thus offers a foundation for advancing and validating legal AI systems, supporting the development of more reliable automated legal analysis tools. Through the four research questions explored in our experimental evaluation, covering performance, prompt sensitivity, generalization, and structured reliability, this study provides empirical evidence addressing the broader inquiry of whether current AI models are dependable enough for legal deployment. Our findings make this clear: despite impressive linguistic fluency, today's LLMs lack the depth of legal reasoning and consistency required for high-stakes legal practice.

Limitations

As a foundational work in auditing the legal reasoning of LLMs, **CLAUSE** establishes a new paradigm for benchmark creation and evaluation. The deliberate choices made in its design define its current scope and naturally pave the way for exciting future extensions.

1. **Dataset Focus:** Our benchmark is built using U.S. commercial contracts from the well-regarded **CUAD** and **ContractNLI** datasets. This allows for a deep and relevant analysis of model performance in this specific, high-impact area. Consequently, our findings are most directly applicable to U.S. law, and future work could readily adapt our successful methodology to create similar benchmarks for international contracts or other legal document types.
2. **AI-Driven Methodology:** A key strength of our work is the innovative AI pipeline that allows us to generate thousands of contract flaws automatically. This makes creating a large-scale benchmark feasible. The nature of these flaws is influenced by the capabilities of the generation model (Gemini 2.0 Flash), and our validation simplifies risk into a "YES/NO" decision. This scalable approach provides a powerful baseline, which can be complemented in the future by studies that focus on more complex, manually-crafted scenarios or the varying degrees of legal risk.
3. **Evaluation Framework:** To assess the quality of the models' legal explanations at scale, we used other LLMs as judges. An interesting finding was that different AI judges showed slight variations in their scoring, which points to a promising research direction in developing robust, automated evaluation techniques. While our metrics confirm that models can identify the correct *semantic area* of a flaw, future research can build on our work by developing metrics that also assess a model's ability to pinpoint the most legally critical part of that text.
4. **Model Scope:** Our experiments provide a crucial baseline for the "out-of-the-box" reasoning capabilities of four major LLM families as of late 2025. We tested them in a direct, instruction-following format to measure their core abilities. This work sets the stage for future studies to explore the performance of specialized models that are fine-tuned on legal data or are integrated into practical, real-world systems with human oversight.
5. **Scope of Human Verification:** To ensure the highest quality, a rigorous validation of our dataset was conducted by 3 NLP experts who reviewed a significant 25% sample of all generated perturbations. The exceptional agreement rate (98.58%) on this large sample provides high confidence in the overall quality of our automated pipeline. Future work could build on this by conducting even larger-scale human validation studies or by developing advanced AI verifiers that correlate more closely with human expert judgment.
6. **Isolated Perturbation Analysis:** Our current methodology generates multiple, distinct perturbations for each source document, allowing us to precisely test a model's ability to detect each of the 10 types of flaws in isolation. Real-world contracts, however, can sometimes contain multiple, interacting issues (e.g., an ambiguous definition that in turn causes an inconsistency elsewhere). An exciting direction for future research is to create benchmarks with such compound discrepancies, which would test the deeper, holistic analysis capabilities of advanced LLMs.

Ethics Statement

CLAUSE is intended to improve the safety and reliability of LLMs in legal settings by making their failure modes visible. We build on publicly available or appropriately licensed contract data, introduce no new personal identifiers, and retain only perturbations that pass automated checks and expert human validation to avoid propagating spurious signals. Any personally identifiable information (PII) present in source datasets (CUAD or ContractNLI) has been systematically redacted using the standardized placeholder "(information_redacted)"; we have verified that this redaction preserves textual meaning and does not degrade model performance on our evaluation tasks. The benchmark is not a source of legal advice; users are expected to combine model outputs with qualified human judgment. We document limitations including dataset skew and inherited legal norms, and encourage future work to broaden coverage and address bias. Release materials include usage guidance to discourage adversarial or deceptive repurposing. Overall, the goal is diagnostic: enable more accountable deployment of legal AI.

We acknowledge risks and implement appropriate safeguards. To prevent adversarial misuse where the benchmark could be exploited to train evasion models or generate flawed legal systems, we employ multi-layered mitigation: (1) transparent documentation of our perturbation methodology to enable reproducibility while deterring exploitation, (2) controlled dataset access via a mandatory request form requiring users to detail their research purpose, allowing us to screen for malicious intent, and (3) distribution with a comprehensive datasheet containing explicit ethical usage guidelines that prohibit adversarial training, commercial deployment without human oversight, and use of perturbed contracts as legal templates. Strong performance on CLAUSE should not be interpreted as deployment certification; the benchmark evaluates specific capabilities under controlled conditions and cannot capture real-world legal complexity. We emphasize in all documentation that this is a diagnostic research tool for capability assessment, not a deployment readiness guarantee. The perturbed contracts intentionally contain legal flaws and must never be used as reference documents; we include prominent warnings and exclude all contracts with sensitive or personally identifiable information. Our benchmark reflects U.S.-centric commercial contract law and may disadvantage models trained on international or specialized legal corpora; we explicitly acknowledge these limitations and encourage community extensions to diverse legal systems. Finally, to address over-reliance on automated legal analysis, we document fundamental gaps in current models' legal reasoning that preclude unsupervised deployment, emphasizing that LLM outputs require qualified human legal review for high-stakes applications.

The validation of CLAUSE was conducted entirely on a voluntary basis by one legal expert and three NLP researchers who contributed their expertise without compensation. These experts were identified based on their familiarity with demographic nuances and their language proficiency needed to evaluate this dataset. The experts assessing CLAUSE recognized the significant societal value of this work in advancing safer AI deployment in high-stakes legal contexts, including improving access to justice through more reliable automated legal analysis tools, preventing catastrophic failures in contract review systems that could lead to substantial financial and legal harm, and establishing rigorous evaluation standards that protect vulnerable populations from flawed AI-driven legal services. Their pro bono contributions reflect a shared commitment to responsible AI development and the belief that transparent, community-driven benchmarks are essential for building trustworthy legal AI systems that serve the public interest. The NLP researchers were provided with comprehensive guidelines (Appendix E) to evaluate the dataset with consistency and accuracy.

We utilized Large Language Models in a controlled environment to edit and refine some sections of the paper.

References

Ilias Chalkidis, Danish Jain, Ismini Lourentzou, and Nikolaos Aletras. 2022. Lexglue: A benchmark dataset for legal language understanding in english. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 3290 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *Preprint*, arXiv:2507.06261.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Cuad: An expert-annotated nlp dataset for legal contract review. *arXiv preprint arXiv:2103.06268*.

Zheng Hu and 1 others. 2025. Evaluating the limits of large language models in multilingual legal reasoning. *arXiv preprint arXiv:2509.22472*.

Zihan Ji, Daniel Lee, Yixin Pan, Yichong Li, and et al. 2023. Halueval: A large-scale hallucination evaluation benchmark for open-domain dialogue models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*.

Yuta Koreeda and Christopher D Manning. 2021. Contractnli: A dataset for document-level natural language inference for contracts. In *Proceedings of the*

2021 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 7578–7589.

Stephanie Lin, Jacob Hilton, and Amanda Askell. 2022. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3214–3252.

McElroy, Deutsch, Mulvaney & Carpenter, LLP. 2021. The timeless lessons of Perini Corp. v. Greate Bay Hotel & Casino, Inc. Accessed: 2025-08-02.

Venkatesh Mishra and 1 others. 2025. Investigating the shortcomings of llms in step-by-step legal reasoning. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 7810–7841.

Yichong Shen, Xiang Li, Xiangru Li, Xipeng Zhang, Philip Yu, and Xiaokang Liu. 2023. A survey of hallucination in natural language generation. *ACM Computing Surveys (CSUR)*.

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.

Don Tuggener, Manuel Welle, Mark C Müller, Lukas Händel, and Mark Cieliebak. 2020. Ledgar: A large-scale multi-label corpus for text classification of legal provisions in contracts. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1235–1241.

Luchen Zheng, Haonan Xu, Peter Henderson, and et al. 2023. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *arXiv preprint arXiv:2308.11462*.

Haoxi Zhong, Yicheng Guo, Cunchao Tu, Chen Xiao, Zhiyuan Liu, and Maosong Sun. 2020. Jec-qa: A legal-domain question answering dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9701–9708.

A CLAUSE Generation and Evaluation Details

This appendix provides comprehensive details on our dataset generation pipeline, complete evaluation results across all experimental conditions, and real-world case examples that motivated our perturbation taxonomy. We organize this material in three parts: first, we present the prompts and methodology underlying our data generation and validation processes; second, we provide detailed breakdowns of evaluation results with in-depth analysis of model performance patterns; and third, we document real-world contractual failures that demonstrate the practical significance of each perturbation type in our benchmark.

Experiment Parameters & Setup For data generation generation we have used Gemini 2.0 Flash at temperature 0.7 for balanced coherence and speed. For evaluation we used a temperature of 0.2 for more deterministic and focused outputs suitable for analysis, disabled all safety filters to prevent content moderation from interfering with the legal text analysis, and set a maximum token limit of 8,192 to ensure explanations were not prematurely truncated.

A.1 Source Dataset Documentation

CLAUSE is built upon two foundational legal NLP datasets: CUAD (Contract Understanding Atticus Dataset) and ContractNLI. Both datasets are licensed under Creative Commons Attribution 4.0 International (CC BY 4.0).

CUAD Dataset The CUAD dataset comprises over 510 commercial legal contracts from various industries, annotated with 13,000+ expert labels across 41 categories relevant to contract review. The dataset covers English-language commercial contracts and focuses on complex contractual language, legal jargon, clause extraction, and nuanced legal concepts. Created with input from legal experts, CUAD reflects realistic legal issues encountered in contract review tasks. The dataset does not include demographic annotations, as it is document-based rather than human subject-focused, and provides monolingual coverage (English only).

ContractNLI Dataset ContractNLI is the largest corpus of annotated contracts for natural language inference, containing approximately 607 contracts (primarily non-disclosure agreements). The dataset addresses document-level entailment, contradiction, and neutral inference tasks in English-language commercial contracts. It covers linguistic phenomena including negations, exceptions, multi-span evidence detection, and complex legal reasoning patterns. Like CUAD, ContractNLI does not include demographic annotations due to its focus on contract semantics and formal logic rather than human subject information, and provides monolingual English coverage.

Coverage Summary Both source datasets provide comprehensive coverage of English legal con-

tract language and document structure relevant to legal NLP benchmarking. They capture complex linguistic phenomena specific to contractual agreements—including legal reasoning markers, clause variations, and formal semantics—but do not address demographic representation or multilingual aspects due to the nature of commercial contract sources. These characteristics are inherited by CLAUSE, which extends the source datasets through systematic perturbation while maintaining their domain focus on U.S. commercial contract law.

A.2 Implementation Details and Package Documentation

This section documents all software packages, models, and parameter settings used throughout our pipeline for data generation, validation, and evaluation.

Data Generation Models For perturbation generation, we used Gemini 2.0 Flash (model ID: `gemini-2.0-flash-exp`) via Google's Gemini API with temperature 0.7, top-p 0.95, and maximum output tokens 8192. All safety filters were disabled to prevent content moderation interference with legal text generation. For LLM-based validation of generated perturbations, we used Gemini 2.5 Flash (model ID: `gemini-2.5-flash-002`) with temperature 0.2 for more deterministic outputs, top-p 0.95, and maximum output tokens 8192.

Evaluation Models We evaluated four language models: (1) GPT-4o-mini (model ID: `gpt-4o-mini-2024-07-18`) accessed via OpenAI API with temperature 0.2, top-p 1.0, and max tokens 8192; (2) Gemini 2.0 Flash (model ID: `gemini-2.0-flash-exp`) with temperature 0.2, top-p 0.95, and max tokens 8192; (3) Gemini 2.5 Flash (model ID: `gemini-2.5-flash-002`) with temperature 0.2, top-p 0.95, and max tokens 8192; and (4) Llama 3.3 70B Instruct (model ID: `meta-llama/Llama-3.3-70B-Instruct`) accessed via Groq API with temperature 0.2, top-p 1.0, and max tokens 8192. All models used identical generation parameters for fair comparison.

LLM Judge Models For `explanation_match` and `law_match` evaluation metrics, we employed two judge models: GPT-4o (model ID: `gpt-4o-2024-08-06`) and Gemini 2.5 Flash (model ID: `gemini-2.5-flash-002`), both with temperature 0.1 for maximum consistency in scoring, top-p 1.0, and max tokens 4096. Judges received structured prompts with ground-truth context and evaluated outputs using predefined rubrics (0-5 scale for explanation quality dimensions, binary for law matching).

Packages For `location_alignment` evaluation, we used the following Python packages: (1) `rouge-score` version 0.1.2 (Google Research implementation) for ROUGE-1, ROUGE-2, and ROUGE-L metrics with default tokenization and stemming enabled; (2) `nltk` version 3.8.1 with the METEOR implementation (`nltk.translate.meteor_score`) using default parameters and English language resources; and (3) `bert-score` version 0.3.13 for computing BERT F1 scores using the `microsoft/deberta-xlarge-mnli` model as the reference embedder with default rescaling. All metrics computed sentence-level alignments with normalization for case and punctuation.

Data Processing and Analysis For data extraction and statistical analysis, we used `pandas` version 2.1.0 for DataFrame operations and CSV/JSON manipulation, `numpy` version 1.24.3 for numerical computations, and `openpyxl` version 3.1.2 for reading Excel-format evaluation results. Text preprocessing employed `re` (Python standard library) for regex-based cleaning and `json` (Python standard library) for structured data parsing. Graph-based alignment for `location_alignment` used custom implementations leveraging `networkx` version 3.1 for connected component detection.

Reproducibility All experiments were conducted using Python 3.10.12 on Ubuntu 22.04 LTS. Random seeds were not set for LLM API calls due to model-side non-determinism, but generation parameters were held constant across all experimental conditions. Complete implementation code, evaluation scripts, and detailed hyperparameter configurations are available in our public repository to ensure full reproducibility.

A.3 Data Generation Prompts and Validation

Our dataset generation employs carefully designed prompts that guide language models to create realistic legal contradictions across ten perturbation categories. Each prompt incorporates persona-based instructions (e.g., "You are a senior compliance officer"), domain-specific definitions, step-by-step modification guidelines, and few-shot examples

demonstrating the target perturbation type. For in-text contradictions, prompts instruct the model to introduce internal document inconsistencies by modifying definitions, creating conflicting obligations, or disrupting structural coherence. For legal contradictions, prompts require the model to identify applicable jurisdiction, introduce modifications that violate specific statutes, and provide proper legal citations with two verified URLs to official government sources.

The validation pipeline employs specialized verification prompts that assess contradiction strength through retrieval-augmented analysis. Each perturbation undergoes binary classification (YES/NO) where the model evaluates three dimensions: semantic significance of the modification, strength and directness of the contradiction, and contextual coherence within the document. For legal contradictions, we scrape legal text from the provided URLs and augment the verification prompt with this statutory context, enabling the model to ground its assessment in actual legal language rather than parametric knowledge alone. Only perturbations receiving a YES classification—indicating strong, unambiguous contradictions—are retained in the final benchmark. This multi-stage approach, combining generation, retrieval-enhanced verification, and human validation (Section 2), ensures our benchmark contains high-quality contradictions that reflect genuine legal risks rather than superficial or arguable inconsistencies.

The following pages present representative prompts from our pipeline, illustrating how we guide models to create specific perturbation types while maintaining legal fidelity. We include examples for ambiguity generation (both in-text and legal), inconsistency generation (both types), and the validation prompts used in our quality control process. Additional prompts for omissions, misaligned terminology, and structural flaws follow similar patterns with category-specific adaptations.

B Example Prompts and Outputs

This section provides example prompts and model outputs for each evaluation task.

Ambiguity In-Text Data Generation Prompt

You are a contract analyst ensuring consistency in legal agreements. Your task is to modify a document by introducing conflicting definitions of the same term within different sections.

Definition:

Ambiguities also occur when key terms are **vaguely and inconsistently defined within the document itself**, creating internal contradictions. This type of **in-text contradiction** confuses contract enforcement by allowing multiple interpretations of the same term in different sections, leading to potential legal disputes over meaning.

Step-by-Step Instructions:

1. Identify a **key term** in the contract.
2. Modify its definition in different sections so that they **conflict**.
3. Ensure the contradiction creates **uncertainty in enforcement**.
4. For that perturbation, make sure in the file there should be **6** of them.
5. Output the modified contract in structured JSON format.
6. Make sure that when taking the original texts, there should be no jumps between sentences. Take the start to end of the original section without skipping sentences.
7. Ensure that the In-Text contradiction being made, due to the inserted perturbation, is being referred in the explanation on where the contradictions are taking place, sure the the inserted perturbation.
8. For each perturbation:
   - Record the **location** (section or paragraph number) where the term was changed.
   - Also identify and record the **contradicted_location** — this is where the contradiction appears or becomes contradictory due to the change. This could be another section where the

original definition is expected to apply or due to the change it creates an in-text contradiction in this location.

- If the contradiction affects multiple sections, include the **most directly impacted** one.
- Extract the **full contradicted_text** from that location — this should be the complete section or paragraph that reflects the conflict caused by the intext contradiction perturbation. Include the entire section to preserve legal context.
- Also keep in mind the contradicted_location, doesn't have to be the same location where the perturbation was made. A perturbation can be made in one location and contradict somewhere in the text in another location

Examples of Conflicting Definitions of a Term:

Example 1:

- **Section 2:** "The term 'Client' refers to any party who purchases our services."
- **Section 8:** "Clients are defined as parties who hold an active contract for at least six months."
- **Explanation:** The first definition includes all buyers, while the second restricts "Clients" to long-term customers, leading to disputes over **contractual obligations**.

Example 2:

- **Section 1:** "Employees are eligible for health benefits upon hiring."
- **Section 6:** "Employees are eligible for health benefits after 90 days."
- **Explanation:** The contradiction makes it unclear when benefits apply.

Example 3:

- **Section 4:** "The vendor is responsible for product warranties."
- **Section 9:** "Warranty claims shall be processed by the manufacturer."
- **Explanation:** It's unclear **who holds warranty responsibility**.

Example 4:

- **Section 3:** "Data shall be stored for five years."
- **Section 12:** "Data must be deleted within two years."
- **Explanation:** A **conflicting data retention period** could violate regulations.

Example 5:

- **Section 2:** "A shareholder is anyone holding 5% or more equity."
- **Section 7:** "A shareholder is anyone with voting rights."
- **Explanation:** A shareholder under one section may **not qualify in another**.

Example 6:

- **Original:** "The Supplier shall deliver all goods in accordance with the specifications outlined in Schedule A. Any failure to meet these specifications will be considered a material breach of contract. In such cases, the Buyer reserves the right to request a full refund or demand replacement goods within 30 days. The Supplier guarantees that all goods will conform to the quality and performance standards described in the agreement."
- **Modified:** "The Supplier shall make every effort to deliver goods in accordance with the specifications outlined in Schedule A. If specifications are not fully met, the Buyer and Supplier will engage in discussions to determine an appropriate resolution, which may include partial refunds or adjusted delivery timelines."

- **Explanation:** This change introduces ambiguity by replacing 'shall deliver' with 'shall make every effort,' making performance non-binding. The original contract clearly stated that failure to meet specifications was a material breach with predefined consequences, whereas the modified version introduces an uncertain resolution process. This creates an **internal contradiction** if another part of the contract enforces strict adherence to Schedule A's specifications.

Return JSON Format

```
"file_name": {file_name},

"perturbation":

"type": "Ambiguities - In Text Contradiction",

"original_text": "EXCERPT BEFORE CHANGE",

"changed_text": "EXCERPT AFTER CHANGE",

"explanation": "WHY THIS CHANGE INTRODUCES A PERTURBATION AND SPECIFIC PLACE WHERE IT IS CONTRADICTING WITHIN THE TEXT",

"location": "SECTION OR PARAGRAPH NUMBER WHERE THE CHANGE WAS MADE",

"contradicted_location": "SECTION OR PARAGRAPH NUMBER WHERE THE CONFLICT OCCURS DUE TO THIS CHANGE",

"contradicted_text": "FULL TEXT OF THE SECTION THAT IS CONTRADICTED DUE TO THE CHANGE"
```

Below is the original legal text: ———–
{original_text}
———–
Now, return ONLY the structured JSON object with the modified text and explanation.

Ambiguity Legal Data Generation Prompt

You are a senior compliance officer reviewing a legal contract. Your task is to modify a section of the contract by introducing an ambiguous legal obligation while ensuring that the ambiguity contradicts a state or national law.
Before modifying the text:

- **Read the file** to determine what city, state, or country the contract applies to.
- If the jurisdiction is unclear, default to **United States law**.
- Make sure that when taking the original texts, there should be no jumps between sentences. Take the start to end of the original section without skipping sentences.

Definition:

Ambiguities occur when a legal statement is vague, leading to multiple interpretations. A **legal contradiction** under this category happens when an obligation is introduced ambiguously, making it difficult to enforce under state or national law. This can result in non-compliance with regulatory requirements, leaving legal obligations open to dispute.

Step-by-Step Instructions:

1. Identify a clear legal obligation in the contract.
2. Modify the wording to make it **vague or open to multiple interpretations**.
3. Ensure that this ambiguity creates **non-compliance with a specific law** in the identified jurisdiction.
4. Use online legal databases or verified government resources to **find a real law that is contradicted**.
5. For that contradiction, extract:

- **The citation/title of the law** (e.g., "29 CFR § 516.2" or "CA Labor Code § 201")

- **Two direct links (URLs)** to official or government sources for the law (**these links must be accessible. The relevant citation should be inside this URL. The page should be found - we should not get a Page Not Found or similar error.**)
- **The first URL must come from an official government or legislature domain (ends in .gov, .mil, .state.<XX>.us, or one of uscode.house.gov, ecfr.gov, govinfo.gov, law.cornell.edu). Use the most recent version of the statute/regulation.**
- **The second URL should be a secondary link to serve as a backup in case the first link falters, but we wish for that to not happen.**
- **A brief explanation** of how the modified text contradicts the law (**law_explanation**)
- Make 5 such perturbations for the file.
- Output only the structured JSON object as shown below.

Examples of Ambiguous Legal Obligations:

Example 1:

- **Original:** "The company shall provide necessary accommodations for disabled employees."
- **Modified:** "The company shall provide accommodations for disabled employees as deemed appropriate."
- **Explanation:** The term "as deemed appropriate" introduces ambiguity, conflicting with **ADA (Americans with Disabilities Act)**, which mandates **clear, non-discretionary accommodations**.

Example 2:

- **Original:** "All contractors must comply with local zoning laws."
- **Modified:** "All contractors must make reasonable efforts to comply with zoning laws."
- **Explanation:** "Reasonable efforts" is vague—some zoning laws require strict adherence.

Example 3:

- **Original:** "The landlord shall ensure habitable living conditions in compliance with state law."
- **Modified:** "The landlord shall make efforts to maintain habitable conditions."
- **Explanation:** "Make efforts" does not guarantee habitability, which violates **tenant protection laws**.

Example 4:

- **Original:** "The company shall maintain data security measures that meet industry standards."
- **Modified:** "The company shall maintain data security measures that it deems sufficient."
- **Explanation:** "Deems sufficient" is subjective and contradicts **GDPR and CCPA** requirements for **specific security standards**.

Example 5:

- **Original:** "Employees shall be provided meal breaks as required by law."
- **Modified:** "Employees shall be encouraged to take meal breaks."
- **Explanation:** Some states require **mandatory** meal breaks (e.g., California).

Example 6:

- **Original:** "The Contractor shall comply with all federal and state regulations governing workplace safety, ensuring all necessary precautions are

taken to protect employees from occupational hazards. The Contractor must conduct quarterly safety inspections and submit reports to regulatory authorities. Any violations of safety standards shall result in corrective actions and potential penalties. The company's leadership is responsible for ensuring full compliance at all levels."

- **Modified:** "The Contractor should make reasonable efforts to comply with applicable federal and state regulations governing workplace safety. The Contractor may conduct periodic safety inspections and submit reports when deemed necessary. Violations of safety standards will be assessed on a case-by-case basis, and corrective actions may be recommended where appropriate."
- **Explanation:** This change weakens the legal obligation by replacing 'shall comply' with 'should make reasonable efforts,' making compliance discretionary rather than mandatory. The removal of 'quarterly safety inspections' eliminates a clear legal requirement, and replacing 'shall result in corrective actions' with 'may be recommended' creates uncertainty. This contradicts **OSHA regulations**, which mandate strict compliance and routine reporting on workplace safety violations.

Return JSON Format

```
"file_name": {file_name},

"perturbation":

"type": "Ambiguities - Ambiguous Legal Obligation",

"original_text": "EXCERPT BEFORE CHANGE",

"changed_text": "EXCERPT AFTER CHANGE",

"explanation": "WHY THIS CHANGE INTRODUCES A PERTURBATION",

"contradicted_law": "SPECIFIC LAW OR REGULATION BEING VIOLATED",

"law_citation": "TITLE OR SECTION OF THE LAW (e.g., '29 CFR § 516.2')",

"law_url1": ["OFFICIAL_LEGAL_REFERENCE_URL_1"],

"law_url2": ["OFFICIAL_LEGAL_REFERENCE_URL_2"],

"law_explanation": "HOW AND WHY THIS MODIFICATION CONTRADICTS THAT LAW",

"location": "SECTION OR PARAGRAPH NUMBER"
```

Below is the original legal text: ———–
`{original_text}`
———–

Now, return ONLY the structured JSON object with the modified text and explanation.

Inconsistency In-Text Data Generation Prompt

You are a financial contract auditor ensuring consistency in payment terms. Your task is to introduce a conflict between different payment deadline clauses.

Definition:

Inconsistencies also lead to **in-text contradictions** when **different sections of a contract provide conflicting deadlines, obligations, or penalties**. This creates in-text contradiction regarding which terms should be enforced, leading to disputes over contractual obligations.

Step-by-Step Instructions:

1. Identify a **key term** in the contract.
2. Modify its definition in different sections so that they **conflict**.
3. Ensure the contradiction creates **uncertainty in enforcement**.

4. For that perturbation, make sure in the file there should be **6** of them.

5. Output the modified contract in structured JSON format.

6. Make sure that when taking the original texts, there should be no jumps between sentences. Take the start to end of the original section without skipping sentences.

7. Ensure that the In-Text contradiction being made, due to the inserted perturbation, is being referred in the explanation on where the contradictions are taking place, sure the the inserted perturbation.

8. For each perturbation:

- Record the **location** (section or paragraph number) where the term was changed.
- Also identify and record the **contradicted_location** — this is where the contradiction appears or becomes contradictory due to the change. This could be another section where the original definition is expected to apply or due to the change it creates an in-text contradiction in this location.
- If the contradiction affects multiple sections, include the **most directly impacted** one.
- Extract the **full contradicted_text** from that location — this should be the complete section or paragraph that reflects the conflict caused by the omission intext contradiction perturbation. Include the entire section to preserve legal context.
- Also keep in mind the contradicted_location, doesn't have to be the same location where the perturbation was made. A perturbation can be made in one location and contradict somewhere in the text in another location

Examples of Conflicting Payment Terms:

Example 1:

- **Section 4:** "Invoices must be paid within 45 days."
- **Section 12:** "Late fees apply if invoices are unpaid after 30 days."
- **Explanation:** One section allows **45 days**, but the other **charges late fees after 30 days**, creating ambiguity.

Example 2:

- **Section 6:** "Security deposits are fully refundable."
- **Section 9:** "Security deposits are non-refundable."
- **Explanation:** It is unclear whether **deposits must be returned**.

Example 3:

- **Section 3:** "Early termination fees apply if the contract is ended before 12 months."
- **Section 10:** "Clients may terminate at any time without penalty."
- **Explanation:** Conflicting clauses **cause uncertainty about termination fees**.

Example 4:

- **Section 5:** "Payment plans must be completed within 6 months."
- **Section 11:** "Customers may extend payment plans up to 12 months."
- **Explanation:** Different sections **contradict the maximum payment period**.

Example 5:

- **Section 8:** "Interest on late payments accrues at 5% per month."
- **Section 14:** "No interest will be charged on late payments."

- **Explanation: Conflicting interest policies** could lead to legal disputes.

Example 6:

- **Original:** "For the purposes of this Agreement, the term 'Affiliate' shall refer to any entity that is controlled by, controls, or is under common control with either party. The term 'control' shall mean ownership of at least fifty-one percent (51%) of the voting equity in such entity. Affiliates shall be entitled to the same rights and obligations as the contracting party."
- **Modified:** "For the purposes of this Agreement, the term 'Affiliate' shall refer to any entity that is affiliated with, works in conjunction with, or has a strategic partnership with either party. Affiliates may be granted certain rights and obligations as determined by the contracting party on a case-by-case basis."
- **Explanation:** The modified text **creates an in-text contradiction** by **changing the definition of "Affiliate"** to include strategic partnerships and loosely affiliated entities, rather than just entities **under majority control**. This contradicts other contract sections that **rely on the original definition** (51% ownership) to determine rights and obligations. Additionally, **allowing case-by-case determination** undermines the original intent of treating affiliates the same as contracting parties, introducing **legal uncertainty** and **potential disputes over obligations**.

Return JSON Format

```
"file_name": {file_name},

"perturbation":

"type": "Inconsistencies - In Text Contradiction",

"original_text": "EXCERPT BEFORE CHANGE",

"changed_text": "EXCERPT AFTER CHANGE",

"explanation": "WHY THIS CHANGE INTRODUCES A PERTURBATION AND SPECIFIC PLACE WHERE IT IS CONTRADICTING WITHIN THE TEXT",

"location": "SECTION OR PARAGRAPH NUMBER",

"contradicted_location": "SECTION OR PARAGRAPH NUMBER WHERE THE CONFLICT OCCURS DUE TO THIS CHANGE",

"contradicted_text": "FULL TEXT OF THE SECTION THAT IS CONTRADICTED DUE TO THE CHANGE"
```

Below is the original legal text: ———–
`{original_text}`
———–
Now, return ONLY the structured JSON object with the modified text and explanation.

Inconsistency Legal Data Generation Prompt

You are an employment law specialist ensuring that contractual deadlines comply with legal regulations. Your task is to modify a timeline in the contract so that it contradicts state or federal laws.
Before modifying the text:

- **Read the file** to determine what city, state, or country the contract applies to.
- If the jurisdiction is unclear, default to **United States law**.
- Make sure that when taking the original texts, there should be no jumps between sentences. Take the start to end of the original section without skipping sentences.

Definition:

Inconsistencies arise when **time-sensitive obligations** in a contract do not align with legal requirements. A **legal contradiction** in this category happens when a contract

sets **a deadline or requirement that violates federal or state law**, making the contractual terms unenforceable or illegal.

Step-by-Step Instructions:

1. Identify a **contractual deadline** that is regulated by law (e.g., payment terms, claims deadlines, notice periods).
2. Modify the deadline to **conflict with state or federal legal requirements**.
3. Ensure that the change **creates non-compliance** with regulatory standards.
4. Use **online legal databases or verified government resources** to **find a real law** that is contradicted by this new timeline.
5. For that contradiction, extract:

- **The citation/title of the law** (e.g., "29 CFR § 516.2" or "CA Labor Code § 201")
- **Two direct links (URLs)** to official or government sources for the law (**these links must be accessible. The relevant citation should be inside this URL. The page should be found - we should not get a Page Not Found or similar error.**)
- **The first URL must come from an official government or legislature domain (ends in .gov, .mil, .state.<XX>.us, or one of uscode.house.gov, ecfr.gov, govinfo.gov, law.cornell.edu). Use the most recent version of the statute/regulation.**
- **The second URL should be a secondary link to serve as a backup in case the first link falters, but we wish for that to not happen.**
- **A brief explanation** of how the modified text contradicts the law (**law_explanation**)
- Make 5 such perturbations for the file.
- Output only the structured JSON object as shown below.

Examples of Conflicts with Regulatory Timelines:

Example 1:

- **Original:** "Employees must submit harassment claims within 15 days."
- **Modified:** "Employees must submit harassment claims within 5 days."
- **Explanation:** Some states (e.g., California) require **at least 30 days** for harassment claims.

Example 2:

- **Original:** "Landlords must return security deposits within 21 days of lease termination."
- **Modified:** "Landlords must return security deposits within 60 days of lease termination."
- **Explanation:** Many states **mandate 14-30 days** for deposit refunds.

Example 3:

- **Original:** "Customers have the right to cancel a contract within 10 days of signing."
- **Modified:** "Customers may cancel contracts within 48 hours."
- **Explanation:** Some **consumer protection laws** require **at least 7-10 days** for cancellations.

Example 4:

- **Original:** "Workers must receive final wages within 72 hours of termination."
- **Modified:** "Workers will receive final wages at the company's discretion."
- **Explanation:** Federal and state laws **require clear final paycheck deadlines**.

Example 5:

- **Original:** "Loan repayment plans must allow at least 90 days for late payments before default."

- **Modified:** "Loan repayment plans may declare default after 30 days of non-payment."
- **Explanation:** Some **loan regulations** require longer grace periods.

Example 6:

- **Original:** "The term 'Confidential Information' shall refer to any proprietary business, financial, and technical data... Confidential Information shall be protected for a period of five (5) years from the date of disclosure."
- **Modified:** "The term 'Confidential Information' shall refer to sensitive business information disclosed by one party to the other. Each party shall determine what constitutes Confidential Information based on its internal policies. Confidentiality obligations shall remain in effect for a commercially reasonable period."
- **Explanation:** Replacing a **clear five-year obligation** with "commercially reasonable period" creates **uncertainty**. Allowing each party to define confidentiality based on internal policies **contradicts** standard contract law principles requiring **definitive** confidentiality terms.

Return JSON Format

```
"file_name": {file_name},

"perturbation":

"type": "Inconsistencies - Legal Contradiction",

"original_text": "EXCERPT BEFORE CHANGE",

"changed_text": "EXCERPT AFTER CHANGE",

"explanation": "WHY THIS CHANGE INTRODUCES A PERTURBATION",

"contradicted_law": "SPECIFIC LAW OR REGULATION BEING VIOLATED",

"law_citation": "TITLE OR SECTION OF THE LAW (e.g., '29 CFR § 516.2')",

"law_url1": ["OFFICIAL_LEGAL_REFERENCE_URL_1"],

"law_url2": ["OFFICIAL_LEGAL_REFERENCE_URL_2"],

"law_explanation": "HOW AND WHY THIS MODIFICATION CONTRADICTS THAT LAW",

"location": "SECTION OR PARAGRAPH NUMBER"
```

Below is the original legal text: ———–
`{original_text}`
———–

Now, return **ONLY** the structured JSON object with the modified text and explanation.

In-Text LLM Validation Prompt

You are a contract analyst. Your task is to determine whether a strong **in-text contradiction** was introduced by a modification to the document. An in-text contradiction occurs when one part of a contract directly or implicitly conflicts with another part of the same document, leading to uncertainty, inconsistency, or unenforceable terms.
You are given:

- The **type** of perturbation introduced
- The **original text** that was modified
- The **changed version** of that text
- A short **explanation** of how the change creates contradiction
- A **contradicted section** from elsewhere in the document where the contradiction is triggered
- The definitions of each perturbation type

Perturbation Information:

Perturbation Type: {perturbation.get("type", "") }

Original Text: {perturbation.get("original_text", "") }

Changed Text: {perturbation.get("changed_text", "") }

Perturbation Explanation: {perturbation.get("explanation", "") }

Contradicted Text: {perturbation.get("contradicted_text", "") }

Perturbation Definitions:

1. Ambiguities – In Text Contradiction: Ambiguities occur when key terms or obligations are defined inconsistently in different parts of a contract, making enforcement difficult or interpretation unclear.

2. Omissions – In Text Contradiction: Omissions occur when essential language is removed or altered in one section, creating a contradiction with obligations or constraints stated in other sections.

3. Misaligned Terminology - In Text Contradiction: Misaligned terminology also leads to **in-text contradictions** when the contract **uses multiple terms interchangeably without defining them**, leading to conflicting obligations.

4. Structural Flaws - In Text Contradiction: Structural flaws also cause **in-text contradictions** when a clause is placed in **multiple locations with conflicting implications**. When contract sections contradict each other based on placement, it creates uncertainty about the obligations and rights of each party.

5. Inconsistencies - In Text Contradiction: Inconsistencies also lead to **in-text contradictions** when **different sections of a contract provide conflicting deadlines, obligations, or penalties**. This creates in-text contradiction regarding which terms should be enforced, leading to disputes over contractual obligations.

Your Task (Strict Instructions):

1. **Compare the changed text** with the **contradicted text**.

2. Determine if the change creates a **clear and direct conflict, inconsistency, or contradiction** within the document. This must be based solely on the provided excerpts and follow the definitions stated above.

3. Only assign `"contradiction_exists": "YES"` if the change:

- Alters or narrows a definition, obligation, or entitlement that **directly breaks compatibility** with the contradicted section
- Makes enforcement unclear or creates a logical inconsistency
- The contradiction must be strong enough that a reasonable reader would immediately recognize the sections as incompatible.
- Direct, clear, and impactful contradictions that would cause confusion, unenforceability, or a legal issue within the document.

Assign "`NO`" if:

- The changed clause still aligns with the contradicted section
- Any difference is minor, interpretative, or does not meaningfully impact understanding or enforcement
- If the contradiction is weak, subtle, open to interpretation, or would require legal expertise to identify
- If ever in Doubt of whether if it is a strong contradiction or not, ALWAYS assign NO.

Additional Guidelines:

- Keep in mind that the Changed Text and Contradiction text might be in the same location. That doesn't necessarily mean there wasn't a contradiction. Read the explanation given and determine whether the Changed Text contradicts the Contradicted Text.
- Also Compare the original text and Change Text and see how the Change causes a contradiction to the contradicted text, based on the given explanation, and provide a "YES" or "NO".

Contradiction Classification Guidelines:

YES (Strong In-Text Contradiction)

Assign "YES" if the modified text:

- Removes key terms or items that are later required or referenced
- Changes timing, eligibility, or definitions in a way that breaks consistency
- Introduces ambiguity where the contradicted clause expects certainty
- Adheres to one of the definitions above.

Examples:

- Section 2 defines "PRODUCT" excluding 'bags'; Section 5 requires the use of bags → **YES**
- Section 3: "Benefits start on Day 1" → Changed to: "May be eligible after probation"; Section 6: "Enroll in benefits in Week 1" → **YES**

NO (No Meaningful Contradiction)

Assign "NO" if:

- The change is additive or softens the language but does not conflict
- Both clauses can be reasonably interpreted as compatible
- The contradiction is only apparent with legal inference, not internal inconsistency

Examples:

- Changed: "Data must be stored securely" → "Data must be stored securely and backed up" — does not contradict another section → **NO**
- Changed: "Contractors comply with local codes" → "Contractors comply with local and state codes"; Section requires compliance → **NO**

Output Format (Strictly Follow This):

```
{

"contradiction_exists": "Assign NO or YES",

"justification": "A grounded explanation of how (and if) the modified text creates an in-text contradiction, referencing everything that is provided to give a binary score"

}
```

Return ONLY this JSON object. No markdown, no notes, no commentary — just valid JSON with double quotes.

Legal LLM Validation Prompt

You are a legal verification expert. Your task is to determine how strongly a legal contradiction was created when a contract was modified. This contradiction must be evaluated based on legal standards, using the information provided.
You are given:

- The **type** of perturbation introduced
- The **original contract text**
- The **changed (perturbed) version** of that text
- A short **perturbation explanation**, describing how the legal obligation was altered
- A **law explanation**, generated by an LLM, describing why it believes this change contradicts a specific law
- **Two scraped law snippets**, pulled

from official legal sources (e.g., .gov/.org sites)

- The definitions for each type of Perturbation.

Perturbation Information:

Perturbation Type: {perturbation.get("type", "") }

Original Text: {perturbation.get("original_text", "") }

Changed Text: {perturbation.get("changed_text", "") }

Perturbation Explanation: {perturbation.get("explanation", "") }

Law Explanation (Generated by LLM): {perturbation.get("law_explanation", "") }

Scraped Law Snippet 1: {perturbation.get("scraped_snippet_1", "") }

Scraped Law Snippet 2: {perturbation.get("scraped_snippet_2", "") }

Perturbation Definitions:

1. Ambiguity Legal Contradiction: Ambiguities occur when a legal statement is vague, leading to multiple interpretations. A **legal contradiction** under this category happens when an obligation is introduced ambiguously, making it difficult to enforce under state or national law. This can result in non-compliance with regulatory requirements, leaving legal obligations open to dispute.

2. Omission Legal Contradiction: Omissions occur when a contract **removes essential information**, creating legal loopholes. A **legal contradiction** in this category happens when a contract omits **a legally mandated consumer protection**, making it non-compliant.

3. Misaligned Terminalogy Legal Contradiction: Misaligned terminology occurs when a contract **uses terms that do not match their statutory meaning**, leading to non-compliance. A **legal contradiction** in this category arises when contractual language deviates from legally recognized definitions, making the contract unenforceable.

4. Structural Flaws Legal Contradiction: Structural flaws occur when the **organization of a contract affects its clarity or enforceability**. A **legal contradiction** in this category arises when a required legal disclosure is **placed in an irrelevant or misleading section**, making it difficult for parties to locate or interpret.

5. Inconsistencies Legal Contradiction: Inconsistencies arise when **time-sensitive obligations** in a contract do not align with legal requirements. A **legal contradiction** in this category happens when a contract sets **a deadline or requirement that violates federal or state law**, making the contractual terms unenforceable or illegal.

Your Task (Strict Instructions):

1. Analyze all the provided inputs thoroughly, along with the definitions to understand what perturbation type was used.

2. Focus your contradiction assessment on the following 5 core elements:

- Scraped Law Snippet 1
- Scraped Law Snippet 2
- Perturbation Explanation
- Law Explanation
- The comparison between the Original and Changed Text, and how the Changed text creates a legal contradiction based on the explanation.

Additional Guidelines:

- Use **both scraped law snippets**, even if one is missing, short, or incomplete.

If both are valid, use both to strengthen your understanding of the law's scope and language. If one is missing, use the other one to strengthen your understanding of the law. If both are missing, use whatever else is given to produce your output.

Contradiction Exists Category (Choose ONE: "NO" or "YES")

You must determine if a contradiction exists based on how significantly the modified contract text violates or undermines the legal obligation stated in the law.
This decision should consider:

- The **importance and clarity** of the legal requirement in the scraped law
- How much the **modified text weakens, removes, or reverses** that legal requirement
- Whether the contradiction is **explicit and impactful**, or **subtle and arguable**

Contradiction does NOT EXIST

Assign `NO` when:

- The change introduces **minor weakening**, **softening**, or **interpretive ambiguity**
- The modified text still **preserves the legal intent**, even if it's more flexible or vague
- The contradiction is **context-dependent**, **non-obvious**, or requires legal interpretation to identify
- The contract could **reasonably be interpreted** as still being compliant
- If in doubt assign NO

Examples:

- Adjustments that make language more discretionary or subjective
- Modifications that reduce precision, but not legal force
- Contradictions that rely on tone or nuance

Contradiction EXISTS

Assign `YES` when:

- The modification results in a **clear non-compliance** with the law
- The changed text **removes or contradicts core legal obligations**
- The contradiction is **direct, enforceable, and serious** from a legal or regulatory perspective
- The modified clause would likely **fail legal scrutiny** or **nullify required protections or standards**
- The contradiction falls under one of the definitions above.

Examples:

- Contract change eliminates required action or accountability
- Legal mandates are replaced by optional or conflicting obligations
- Any change that would trigger a **legal violation** or breach of compliance

Additional Requirements:

- Then provide a **justification**, explaining clearly:
  - How the law was contradicted (or not)
  - Which parts of the scraped snippets support your reasoning
  - Why the contradiction score is appropriate, without simply repeating the score
- Do not infer legal meaning beyond what's in the snippets or explanations. Only use what's provided.
- The contradiction_exists and justification must be **independently valid** — don't base one on the other.

Output Format (Strictly Follow This):

```
{

"contradiction_exists": "Assign NO or YES",

"justification": "A legally grounded explanation of how (and if) the modified text contradicts the law, referencing the scraped snippets if helpful."

}
```

Return ONLY this JSON object. No markdown, no notes, no commentary — just valid JSON with double quotes.

Eval_1: Binary Discrepancy Detection Prompt

You are a U.S. contract attorney who answers concisely.
Please read the legal document below in full.
Document: """
{doc}
"""
Does this document contain **any** discrepancy? Reply with **only** 'Yes' or 'No'.

Eval_2: Document Classification Prompt

You are a legal expert specializing in U.S. law. You will read a legal document very carefully and classify it into one of the following two categories:
1. In-text contradiction: The document contains one or more statement or clause that contradicts another part of the same document.
2. Outer-law contradiction: The document contains one or more statement or clause that contradicts existing U.S. federal, state or city laws (or any other U.S. laws).
Classify the document below into one of the three categories. Respond with only one of the labels: 'In-text contradiction' or 'Outer-law contradiction' Do not provide any explanation. Write 'In-text contradiction' if the document contains In-text contradiction and write 'Outer-law contradiction' if the document contains Outer-law contradiction.
Document: """
{doc_text}
"""
What type of contradiction does this document contain? Reply with only one of the two labels: 'In-text contradiction' or 'Outer-law contradiction'

Eval_3 L1: In-Text Discrepancy Detection Prompt

You are a senior U.S. attorney and contract analyst.

Task:

Read the following legal document in full. Check and detect whether the document contains any intext discrepancies/contradictions in it.

Definitions of intext discrepancy/contradiction:

In Text Contradiction: a statement, clause, term, or definition that conflicts with, negates, or is irreconcilable with another part of the **same** document.
There are five primary categories of intext discrepancies. Their definitions are provided below:

Discrepancy Categories:

Ambiguities: An ambiguity discrepancy occurs when a contract's language is so vague, indeterminate, or open to multiple reasonable interpretations that it either (a) creates internal conflict within the document—where different sections or clauses can be read in mutually incompatible ways—or (b) fails to meet the clarity standards mandated by a specific U.S. statute, regulation, law, or controlling case, thereby placing the clause in conflict with applicable legal requirements.
Inconsistencies: An inconsistency discrepancy arises when a contract contains conflicting deadlines, obligations, or penalties—either within different sections of the document itself or between the contract and applicable legal standards. These contradictions can create uncertainty about which

terms should be enforced or may render certain provisions unenforceable if they violate federal or state laws.
Misaligned Terminology: A misaligned terminology discrepancy occurs when a contract uses different or undefined terms inconsistently—either by employing multiple terms interchangeably within the document, creating internal contradictions and conflicting obligations, or by deviating from legally established definitions, resulting in non-compliance with statutory or regulatory standards that can render the contract unenforceable.
Omissions: An omission discrepancy arises when a required term, clause, definition, or disclosure is missing, incomplete, or left undefined—either when it is referenced elsewhere in the contract, causing internal inconsistency, or when its absence violates specific U.S. legal requirements, such as statutory mandates, regulatory disclosures, or protective safeguards.
Structural Flaws: A structural flaw discrepancy occurs when the organization or placement of clauses within a contract undermines its clarity, coherence, or legal validity—such as when the same clause appears in multiple sections with conflicting implications, or when a legally required disclosure is positioned in a misleading or irrelevant part of the document, making it difficult for parties to locate, interpret, or enforce the intended obligations.
The document may contain in-text discrepancies belonging to any of the five defined types. Your task is to identify all such discrepancies and present them strictly in the output format specified below:

Output Format:

Return **exactly** one JSON array—no markdown, no commentary—where each object includes:

- `"text"`: the exact sentence(s) or phrase that contains the issue
- `"explanation"`: clear detailed reasoning—what it contradicts and why it is a discrepancy

Example output shape: `[`

```
{

"text": "<exact sentence(s) or phrase with the issue>",

"explanation": "<clear, detailed reason: what it contradicts and why it is a problem>"

},

...     (one object per discrepancy) ...

]
```

Rules:

- Include **all** discrepancies you find.
- Assign the most appropriate "`main_type`" and "`sub_type`" to each discrepancy.
- If no discrepancy exists, return an empty array: `[]`.
- Do NOT wrap the JSON in markdown and do NOT add any extra keys or commentary.

Eval_3 L1: Legal Discrepancy Detection Prompt

You are a senior U.S. attorney and contract analyst.

Task:

Read the following legal document in full. Check and detect whether the document contains any legal discrepancies/contradictions in it.

Definitions of legal discrepancy/contradiction:

Legal Contradiction: any clause, section, or text that conflicts with, violates, or is unenforceable under existing U.S. law.
There are five primary categories of legal discrepancies. Their definitions are provided below:

Discrepancy Categories:

Ambiguities: An ambiguity discrepancy occurs when a contract's language is so vague, indeterminate, or open to multiple reasonable interpretations that it either (a) creates internal conflict within the document—where different sections or clauses can be read in mutually incompatible ways—or (b) fails to meet the clarity standards mandated by a specific U.S. statute, regulation, law, or controlling case, thereby placing the clause in conflict with applicable legal requirements.

Inconsistencies: An inconsistency discrepancy arises when a contract contains conflicting deadlines, obligations, or penalties—either within different sections of the document itself or between the contract and applicable legal standards. These contradictions can create uncertainty about which terms should be enforced or may render certain provisions unenforceable if they violate federal or state laws.

Misaligned Terminology: A misaligned terminology discrepancy occurs when a contract uses different or undefined terms inconsistently—either by employing multiple terms interchangeably within the document, creating internal contradictions and conflicting obligations, or by deviating from legally established definitions, resulting in non-compliance with statutory or regulatory standards that can render the contract unenforceable.

Omissions: An omission discrepancy arises when a required term, clause, definition, or disclosure is missing, incomplete, or left undefined—either when it is referenced elsewhere in the contract, causing internal inconsistency, or when its absence violates specific U.S. legal requirements, such as statutory mandates, regulatory disclosures, or protective safeguards.

Structural Flaws: A structural flaw discrepancy occurs when the organization or placement of clauses within a contract undermines its clarity, coherence, or legal validity—such as when the same clause appears in multiple sections with conflicting implications, or when a legally required disclosure is positioned in a misleading or irrelevant part of the document, making it difficult for parties to locate, interpret, or enforce the intended obligations.

The document may contain legal discrepancies belonging to any of the five defined types. Your task is to identify all such discrepancies and present them strictly in the output format specified below:

Output Format:

Return **exactly** one JSON array—no markdown, no commentary—where each object includes:

- `"text"`: the exact sentence(s) or phrase that contains the issue
- `"explanation"`: clear detailed reasoning—what it contradicts and why it is a discrepancy
- `"law"`: citation of the U.S. law / regulation / case it violates, or "N/A" if the conflict is only in-text or no law found.

Example output shape: `[`

```
{

"text": "<exact sentence(s) or phrase with the issue>",

"explanation": "<clear, detailed reason: what it contradicts and why it is a problem>",

"law": "Cal. Com. Code § 2201" // or "N/A" only if law not found

},

...     (one object per discrepancy) ...

]
```

Rules:

- Include **all** discrepancies you find.
- Assign the most appropriate "`main_type`" and "`sub_type`" to each discrepancy.

- If no discrepancy exists, return an empty array: `[]`.
- Do NOT wrap the JSON in markdown and do NOT add any extra keys or commentary.

Eval_3 L2: In-Text Discrepancy Detection Prompt

You are a senior U.S. attorney and contract analyst.

Task:

Read the following legal document in full. Check and detect whether the document contains any intext discrepancies/contradictions in it.

Definitions of intext discrepancy/contradiction:

In-Text Contradiction: a statement, clause, term, or definition that conflicts with, negates, or is irreconcilable with another part of the **same** document.
There are five primary categories of intext discrepancies. Their definitions are provided below:

Discrepancy Categories:

Ambiguities: An ambiguity discrepancy occurs when a contract's language is so vague, indeterminate, or open to multiple reasonable interpretations that it either (a) creates internal conflict within the document—where different sections or clauses can be read in mutually incompatible ways—or (b) fails to meet the clarity standards mandated by a specific U.S. statute, regulation, law, or controlling case, thereby placing the clause in conflict with applicable legal requirements.
Inconsistencies: An inconsistency discrepancy arises when a contract contains conflicting deadlines, obligations, or penalties—either within different sections of the document itself or between the contract and applicable legal standards. These contradictions can create uncertainty about which terms should be enforced or may render certain provisions unenforceable if they violate federal or state laws.
Misaligned Terminology: A misaligned terminology discrepancy occurs when a contract uses different or undefined terms inconsistently—either by employing multiple terms interchangeably within the document, creating internal contradictions and conflicting obligations, or by deviating from legally established definitions, resulting in non-compliance with statutory or regulatory standards that can render the contract unenforceable.
Omissions: An omission discrepancy arises when a required term, clause, definition, or disclosure is missing, incomplete, or left undefined—either when it is referenced elsewhere in the contract, causing internal inconsistency, or when its absence violates specific U.S. legal requirements, such as statutory mandates, regulatory disclosures, or protective safeguards.
Structural Flaws: A structural flaw discrepancy occurs when the organization or placement of clauses within a contract undermines its clarity, coherence, or legal validity—such as when the same clause appears in multiple sections with conflicting implications, or when a legally required disclosure is positioned in a misleading or irrelevant part of the document, making it difficult for parties to locate, interpret, or enforce the intended obligations.

Example of in-text contradiction/discrepancy

```
{

"original_text": "1. DEFINITIONS. (a) 'CONTENT' means all content or information, in any medium, provided by a party to the other party for use in conjunction with the performance of its obligations hereunder, including without limitation any text, music, sound, photographs, video, graphics, data or software. Content provided by 2TheMart is referred to herein as '2TheMart Content' and content provided by i-Escrow is referred to herein as 'i-Escrow Content.",
```

"changed_text": "1. DEFINITIONS. (a) 'CONTENT' means all content or information, in any medium, provided by a party to the other party for use in conjunction with the performance of its obligations hereunder, **including only text and images**. Content provided by 2TheMart is referred to herein as '2TheMart Content' and content provided by i-Escrow is referred to herein as 'i-Escrow Content.",

"explanation": "The revised definition limits 'CONTENT' to text and images, whereas Section 6.1 later grants rights to 'publicly perform' and 'digitally perform' 2TheMart Content—language that presumes audio, video, or other media. The conflicting scopes create an ambiguity-driven in-text contradiction.",

}

Expected JSON entry: [

{

"text": "ÒCONTENTm̀eans all content or information, in any medium, provided by a party to the other party for use in conjunction with the performance of its obligations hereunder, **including only text and images**. Content provided by 2TheMart is referred to herein as ̀2TheMart Contentàand content provided by i-Escrow is referred to herein as ì-Escrow Content.;

"explanation": "Definition excludes audio/video, yet Section 6.1 authorizes public/digital performance—an implicit reference to multimedia—so the provisions conflict."

}

]

The document may contain in-text discrepancies belonging to any of the five defined types. Your task is to identify all such discrepancies and present them strictly in the output format specified below:

Output Format:

Return **exactly** one JSON array—no markdown, no commentary—where each object includes:

- "text": the exact sentence(s) or phrase that contains the issue
- "explanation": clear detailed reasoning—what it contradicts and why it is a discrepancy

Rules:

- Include **all** discrepancies you find.
- If no discrepancy exists, return an empty array: [].
- Do NOT wrap the JSON in markdown and do NOT add any extra keys or commentary.

Eval_3 L2: Legal Discrepancy Detection Prompt

You are a senior U.S. attorney and contract analyst.

Task:

Read the following legal document in full. Check and detect whether the document contains any legal discrepancies/contradictions in it.

Definitions of legal discrepancy/contradiction:

Legal Contradiction: any clause, section, or text that conflicts with, violates, or is unenforceable under existing U.S. law.
There are five primary categories of legal discrepancies. Their definitions are provided below:

Discrepancy Categories:

Ambiguities: An ambiguity discrepancy occurs when a contract's language is so vague, indeterminate, or open to multiple reasonable interpretations that it either (a) creates internal conflict within the document—where different sections or clauses can be read in mutually incompatible ways—or (b) fails to meet the clar-

ity standards mandated by a specific U.S. statute, regulation, law, or controlling case, thereby placing the clause in conflict with applicable legal requirements.

Inconsistencies: An inconsistency discrepancy arises when a contract contains conflicting deadlines, obligations, or penalties—either within different sections of the document itself or between the contract and applicable legal standards. These contradictions can create uncertainty about which terms should be enforced or may render certain provisions unenforceable if they violate federal or state laws.

Misaligned Terminology: A misaligned terminology discrepancy occurs when a contract uses different or undefined terms inconsistently—either by employing multiple terms interchangeably within the document, creating internal contradictions and conflicting obligations, or by deviating from legally established definitions, resulting in non-compliance with statutory or regulatory standards that can render the contract unenforceable.

Omissions: An omission discrepancy arises when a required term, clause, definition, or disclosure is missing, incomplete, or left undefined—either when it is referenced elsewhere in the contract, causing internal inconsistency, or when its absence violates specific U.S. legal requirements, such as statutory mandates, regulatory disclosures, or protective safeguards.

Structural Flaws: A structural flaw discrepancy occurs when the organization or placement of clauses within a contract undermines its clarity, coherence, or legal validity—such as when the same clause appears in multiple sections with conflicting implications, or when a legally required disclosure is positioned in a misleading or irrelevant part of the document, making it difficult for parties to locate, interpret, or enforce the intended obligations.

Example of legal contradiction/discrepancy

```
{

"original_text": "The parties agree to act in good faith with respect to each provision of this Agreement and any dispute that may arise related hereto.",

"changed_text": "The parties agree to attempt to act in good faith with respect to each provision of this Agreement and any dispute that may arise related hereto.",

"explanation": "Changing 'agree to act in good faith' to 'agree to attempt to act in good faith' weakens the obligation. Good faith is a standard implied in contracts under California law, requiring honesty and fair dealing. The word 'attempt' introduces ambiguity, implying that full good-faith efforts are not strictly required, thus undermining the implied covenant.",

"contradicted_law": "California Commercial Code § 1304",

}
```

Expected JSON entry: [

```
{

"text": "The parties agree to attempt to act in good faith with respect to each provision of this Agreement and any dispute that may arise related hereto.",

"explanation": "The clause downgrades the statutory obligation of good faith to a mere attempt, conflicting with California Commercial Code § 1304.",

"law": "Cal. Com. Code § 1304"

}

]
```

The document may contain legal discrepancies belonging to any of the five defined types. Your task is to identify all such discrepancies and present them strictly in the output format specified below:

Output Format:

Return **exactly** one JSON array—no markdown, no commentary—where each object includes:

- `"text"`: the exact sentence(s) or phrase that contains the issue
- `"explanation"`: clear detailed reasoning—what it contradicts and why it is a discrepancy
- `"law"`: citation of the U.S. law / regulation / case it violates, or "N/A" if the conflict is only in-text.

Rules:

- Include **all** discrepancies you find.
- If no discrepancy exists, return an empty array: `[]`.
- Do NOT wrap the JSON in markdown and do NOT add any extra keys or commentary.

C Complete Evaluation Results

This section presents comprehensive evaluation results across all three evaluation levels, organized by task complexity. For each evaluation, we provide detailed performance breakdowns across all perturbation categories, datasets, models, and prompt levels, enabling fine-grained analysis of model capabilities and failure modes.

C.1 Evaluation 2: Miss and Extra Rates Across Categories

Table 8 presents miss and extra rates for Evaluation 3 at both zero-shot (L1) and one-shot (L2) prompt levels, broken down by all ten perturbation categories and both datasets. The miss rate quantifies the fraction of ground-truth discrepancy spans that models failed to detect, while the extra rate measures the fraction of predicted spans that did not align with any ground-truth annotation. These complementary metrics reveal the precision-recall trade-off inherent in different prompting strategies.

Several critical patterns emerge from this data. First, one-shot prompting (L2) consistently reduces miss rates across nearly all configurations, indicating that example-guided inference improves recall. However, this improvement comes at the cost of elevated extra rates, suggesting that models become more liberal in their predictions when provided with examples, leading to increased false positives. Second, the CUAD dataset exhibits systematically higher extra rates than NLI, particularly for GPT-4o-mini and Gemini-2.5, indicating that CUAD's legal

complexity may trigger more spurious detections. Third, certain categories prove exceptionally challenging: Omission$_{Legal}$ on NLI shows miss rates exceeding 80% for GPT-4o-mini and Llama-3.3, while Structural Flaws demonstrate high variability across models, with Gemini-2.0 showing markedly better detection than Llama-3.3. These category-specific patterns suggest that different architectures encode distinct inductive biases about legal document structure and the nature of legal violations.

The pronounced L1-to-L2 shift in the precision-recall trade-off varies substantially across models. Gemini-2.0 exhibits relatively stable behavior, with modest changes in both metrics, suggesting a more conservative generation strategy less susceptible to prompt perturbations. In contrast, GPT-4o-mini and Gemini-2.5 show dramatic swings, with extra rates sometimes doubling from L1 to L2, indicating high sensitivity to prompt design. Llama-3.3's performance on certain categories (e.g., Structural Flaws$_{Legal}$ on CUAD with 72.3% extra rate and 57.1% miss rate at L1) reveals fundamental brittleness in handling complex legal structural violations, likely due to training distribution gaps. Overall, this table underscores that no single prompting strategy dominates: practitioners must balance detection completeness against precision depending on their application's tolerance for false positives versus false negatives.

C.2 Evaluation 3: Location Alignment Metrics by Category

Table 9 provides granular location alignment metrics for all category-model-dataset-level combinations, reporting ROUGE-1, ROUGE-2, ROUGE-L, METEOR, and BERT F1 scores. These metrics assess how accurately models identify the precise text spans containing discrepancies, with higher scores indicating better localization capability. This fine-grained analysis reveals nuanced performance patterns obscured by aggregated results presented in the main paper.

Three key insights emerge from the detailed breakdown. First, category difficulty varies sys-

Category	GPT-4o mini				Gemini 2.0				Gemini 2.5				Llama 3.3			
	L1		L2		L1		L2		L1		L2		L1		L2	
	E	M	E	M	E	M	E	M	E	M	E	M	E	M	E	M
	CUAD															
Amb_{Leg}	89.7	90.4	70.9	91.5	67.9	55.6	65.6	56.8	82.6	50.4	77.8	43.3	91.8	83.4	86.4	79.4
Amb_{inT}	76.7	72.1	73.7	72.4	57.2	58.6	56.0	57.2	77.8	48.7	71.5	53.5	76.8	63.8	72.6	60.6
$\text{Incon}_{\text{Leg}}$	82.0	81.1	67.4	86.1	68.2	59.2	66.7	61.2	85.4	69.0	83.7	69.9	86.1	77.3	85.7	76.9
$\text{Incon}_{\text{inT}}$	76.2	73.0	74.8	74.4	53.7	58.7	53.9	58.4	79.7	54.8	75.7	59.6	75.6	62.3	73.5	63.1
$\text{MisTerm}_{\text{Leg}}$	79.3	76.8	64.5	81.5	53.9	41.9	50.6	43.1	76.9	41.8	69.5	65.8	80.4	67.4	80.9	68.4
$\text{MisTerm}_{\text{inT}}$	83.1	80.0	83.4	81.3	69.6	68.9	70.0	69.6	80.8	55.3	78.7	60.9	82.9	74.0	80.4	73.0
Omis_{Leg}	85.5	86.9	67.6	90.6	71.0	63.6	68.5	64.9	85.9	65.9	63.3	67.9	86.5	74.9	86.6	75.2
Omis_{inT}	84.6	80.8	84.8	82.5	70.6	70.8	70.6	70.6	81.0	58.6	78.1	65.2	84.9	77.5	78.5	70.3
$\text{StrFlaw}_{\text{Leg}}$	73.8	71.2	66.3	83.3	57.2	44.2	54.8	47.7	77.8	46.2	75.3	43.3	72.3	57.1	72.6	57.9
$\text{StrFlaw}_{\text{inT}}$	72.1	68.3	70.0	70.7	45.0	48.8	46.4	49.0	75.6	42.9	66.9	47.1	79.2	67.9	69.3	56.0
	NLI															
Amb_{Leg}	78.8	81.9	56.7	83.1	50.9	50.9	42.1	40.1	64.5	36.7	59.9	33.1	78.9	70.9	69.9	62.9
Amb_{inT}	56.3	64.2	49.0	67.1	48.5	50.7	41.3	52.7	67.6	49.7	58.1	48.6	53.0	49.0	75.7	56.9
$\text{Incon}_{\text{Leg}}$	57.1	70.7	43.3	75.5	36.1	45.8	33.9	45.4	61.7	50.6	61.1	52.8	62.3	58.9	62.0	61.1
$\text{Incon}_{\text{inT}}$	59.0	66.0	52.6	70.9	50.2	52.9	40.8	53.4	66.0	49.1	62.0	51.8	49.3	47.7	48.3	48.6
$\text{MisTerm}_{\text{Leg}}$	64.4	75.0	53.3	79.9	40.0	46.2	39.1	46.3	60.7	41.7	60.3	44.5	67.8	62.7	63.9	63.0
$\text{MisTerm}_{\text{inT}}$	76.5	78.0	69.1	81.9	65.7	69.3	64.1	73.5	77.7	61.5	72.0	62.7	74.3	69.2	74.9	70.6
Omis_{Leg}	75.4	81.2	52.1	86.0	56.3	62.8	54.9	63.9	71.9	62.4	70.3	63.6	74.3	67.7	72.2	68.0
Omis_{inT}	73.0	75.5	59.5	79.2	66.7	67.7	61.5	72.4	78.0	63.4	74.6	65.8	71.0	63.2	71.3	65.9
$\text{StrFlaw}_{\text{Leg}}$	65.3	49.9	56.3	57.8	56.9	33.2	57.7	37.8	70.7	27.2	68.8	29.0	71.5	40.9	68.7	41.6
$\text{StrFlaw}_{\text{inT}}$	50.4	62.3	43.8	66.2	42.9	48.0	37.6	48.6	63.5	45.8	57.9	47.7	46.0	45.3	43.6	47.4

Table 8: Comparison across models and tasks for L1 (zero-shot) and L2 (one-shot) levels. Values represent extra (E) and miss (M) rates for each discrepancy category.

tematically: Ambiguity and Inconsistency perturbations generally yield higher scores than Structural Flaws and certain Omissions, suggesting that models more readily identify discrepancies with explicit textual markers versus those involving absence or organizational defects. For example, Gemini-2.0 achieves 92.3 ROUGE-1 on Ambiguity_{Legal} (CUAD, L1) but only 63.6 on $\text{Structural Flaws}_{Legal}$, a 28.7-point gap. Second, dataset effects are category-dependent: while most categories show higher scores on NLI than CUAD, $\text{Ambiguity}_{inText}$ exhibits the opposite pattern, with models performing 10-15 points better on CUAD across all architectures. This suggests that CUAD contracts contain more explicit definitional sections that facilitate detection of ambiguous terms, whereas NLI's clause-level structure makes in-text ambiguities harder to isolate. Third, one-shot prompting (L2) produces inconsistent effects: for some configurations (e.g., GPT-4o-mini on Ambiguity categories), L2 improves scores by 2-5 points, but for others (e.g., Gemini-2.5 on CUAD Omission_{Legal}), L2 actually degrades performance, likely because the example biases the model toward a specific localization pattern that does not generalize.

The BERT F1 scores, which measure semantic similarity rather than lexical overlap, reveal that models achieve surprisingly high alignment (>90%) even when ROUGE scores are modest (60-70%). This indicates that models often correctly identify the semantic region containing the discrepancy but struggle with exact span boundaries, a critical distinction for practical deployment, where approximate localization may suffice for human review but precise extraction is needed for automated processing. The consistent superiority of Gemini-2.0 across nearly all configurations (typically 5-10 points above other models) suggests architecture-specific strengths in span prediction, potentially related to its training on document-level tasks. Conversely, Llama-3.3's relatively uniform scores across categories (typically 60-65 on ROUGE-1 for inText categories) indicate a less differentiated understanding of perturbation types, treating diverse categories more homogeneously than other models.

C.3 Evaluation 3: Explanation Quality Metrics by Category

Table 10 presents LLM-judge-based quality assessments of generated explanations across four dimensions: Accuracy (factual correctness of the identified issue), Clarity (ease of understanding),

Category	Data	GPT-4o-m										Gem-2.0									
		L1					L2					L1					L2				
		R_1	R_2	R_L	M	F1	R_1	R_2	R_L	M	F1	R_1	R_2	R_L	M	F1	R_1	R_2	R_L	M	F1
Amb_{Leg}	CUAD	83.9	83.4	83.9	77.6	97.2	81.7	81.2	81.6	74.9	96.7	92.3	92.0	90.3	90.9	98.3	90.9	90.6	89.1	89.5	98.0
	NLI	87.7	87.3	86.9	83.4	97.8	85.8	85.3	85.2	81.6	97.5	96.6	96.4	94.5	95.8	99.1	95.0	94.8	93.5	93.4	98.9
Amb_{inT}	CUAD	52.5	51.5	51.7	41.5	91.7	56.3	55.5	55.1	45.2	92.4	69.7	68.6	68.6	61.1	94.5	69.0	68.1	68.1	61.1	94.5
	NLI	59.7	58.6	58.7	51.1	92.8	61.8	60.5	60.2	53.2	93.1	66.8	65.7	65.9	61.5	93.6	69.5	68.2	68.2	63.6	93.8
$Incon_{Leg}$	CUAD	85.1	84.5	83.7	81.0	97.1	83.5	83.0	82.8	79.1	96.8	85.4	85.1	84.9	85.1	97.2	85.2	84.6	84.5	85.0	97.1
	NLI	85.2	84.5	84.6	80.4	97.1	85.5	84.9	84.4	80.5	97.2	88.6	87.9	87.3	86.6	97.6	88.0	87.4	87.1	85.4	97.6
$Incon_{inT}$	CUAD	50.7	49.9	50.1	39.2	91.5	54.4	53.4	53.2	43.5	92.1	64.1	63.1	63.2	55.4	93.6	64.2	63.3	63.3	55.1	93.6
	NLI	55.1	54.2	54.4	45.1	92.2	57.7	56.7	56.9	47.9	92.7	67.0	65.2	65.9	60.2	93.2	66.1	64.5	65.2	59.1	93.2
$MisTerm_{Leg}$	CUAD	77.9	77.2	76.8	70.8	95.9	77.1	76.4	75.8	70.0	95.7	87.1	86.6	85.9	84.3	97.4	86.1	85.6	85.0	83.1	97.2
	NLI	82.5	81.5	81.2	76.9	96.6	83.3	82.5	80.9	77.9	96.6	89.1	88.6	87.9	87.0	97.7	89.0	88.6	87.6	86.4	97.7
$MisTerm_{inT}$	CUAD	54.0	53.3	53.2	43.8	91.7	59.1	58.3	58.3	48.2	92.8	68.5	67.4	67.2	61.6	94.2	68.4	67.2	67.0	60.6	94.0
	NLI	58.6	57.8	57.7	49.5	92.9	62.4	61.5	61.8	54.1	93.3	68.3	66.6	66.8	63.9	93.5	69.5	67.7	67.7	64.5	93.8
$Omis_{Leg}$	CUAD	81.4	80.8	80.8	75.0	96.4	79.5	78.7	79.1	72.7	95.9	85.0	84.3	83.4	83.1	96.9	86.4	85.9	85.3	85.2	97.2
	NLI	91.4	91.1	90.9	88.3	98.3	90.1	89.7	89.8	86.7	98.1	92.5	92.1	91.8	92.3	98.3	92.5	92.1	91.8	91.5	98.4
$Omis_{inT}$	CUAD	55.3	54.4	55.0	44.4	92.2	58.7	57.6	58.0	48.9	92.7	69.6	68.4	68.6	63.2	94.3	69.5	68.0	68.4	63.5	94.3
	NLI	62.2	60.9	61.2	54.3	93.1	62.2	60.9	61.5	56.4	93.2	67.8	65.3	66.3	66.2	92.9	68.5	66.3	67.1	65.6	93.2
$StrFlaw_{Leg}$	CUAD	47.6	46.9	46.3	34.6	90.3	39.8	39.2	39.6	27.7	88.9	63.6	62.1	59.8	52.2	92.5	64.1	62.6	60.5	51.6	92.6
	NLI	56.8	54.9	54.4	45.1	91.8	57.6	55.7	56.1	45.6	91.9	68.4	65.8	65.5	65.9	93.6	69.7	67.7	66.8	65.2	93.8
$StrFlaw_{inT}$	CUAD	48.4	47.5	47.7	37.6	91.0	53.2	52.4	52.4	42.6	91.9	62.3	61.4	61.6	52.2	93.4	60.9	60.0	60.2	50.5	93.2
	NLI	57.2	56.3	56.9	47.1	92.7	59.7	58.6	59.2	49.9	93.0	65.9	64.0	64.8	58.1	93.3	65.7	63.9	64.4	57.6	93.3
Category	Data	Gem-2.5										Llama-3.3									
		L1					L2					L1					L2				
		R_1	R_2	R_L	M	F1	R_1	R_2	R_L	M	F1	R_1	R_2	R_L	M	F1	R_1	R_2	R_L	M	F1
Amb_{Leg}	CUAD	89.4	88.9	88.4	89.7	98.0	90.5	90.1	89.2	90.2	98.2	87.3	86.8	86.0	83.3	97.4	87.6	87.2	86.3	83.7	97.7
	NLI	93.1	92.7	91.4	93.1	98.6	93.5	93.1	91.6	92.9	98.7	91.2	90.9	89.7	88.8	98.2	92.9	92.6	90.0	89.9	98.6
Amb_{inT}	CUAD	62.0	59.8	59.6	53.9	92.7	61.8	59.4	59.9	53.9	92.7	61.6	60.4	60.7	51.6	93.2	60.9	59.7	59.6	50.6	93.0
	NLI	63.7	60.9	61.0	58.1	92.4	66.2	63.5	63.6	61.4	92.9	63.9	62.6	62.4	56.0	93.5	62.2	70.1	65.4	61.3	66.1
$Incon_{Leg}$	CUAD	79.5	78.7	78.7	79.0	96.2	80.8	80.0	80.0	78.9	96.3	86.1	85.5	84.2	83.7	97.1	86.4	85.6	85.0	84.6	97.1
	NLI	85.4	84.5	84.3	84.4	97.0	85.4	84.4	83.7	84.2	97.0	86.3	85.4	84.3	84.7	97.1	85.9	85.2	84.5	82.6	97.2
$Incon_{inT}$	CUAD	58.2	56.0	56.0	48.6	92.0	58.2	55.7	56.1	48.9	92.0	58.7	57.5	57.8	48.0	92.7	59.1	58.0	57.9	48.4	92.8
	NLI	64.3	61.7	62.0	58.5	92.5	65.2	62.3	62.8	59.9	92.5	64.0	62.6	63.4	55.5	93.5	62.5	61.2	61.8	53.4	93.3
$MisTerm_{Leg}$	CUAD	82.1	81.2	80.8	80.0	96.4	49.5	56.7	54.2	48.8	49.5	84.6	84.0	82.7	79.2	97.0	84.2	83.4	81.9	78.7	96.9
	NLI	83.2	82.3	81.9	82.3	96.8	83.7	83.0	82.1	81.5	96.8	88.7	87.9	86.6	85.9	97.5	86.6	85.8	83.9	83.0	97.2
$MisTerm_{inT}$	CUAD	62.3	59.9	59.9	54.1	92.6	62.6	60.7	60.9	55.5	92.8	58.2	56.9	56.9	47.5	92.5	57.8	56.7	56.9	46.7	92.5
	NLI	66.1	63.7	64.0	60.8	93.1	66.8	64.3	64.2	63.6	93.0	62.1	60.4	61.3	55.0	93.0	62.2	60.8	61.0	53.4	93.2
$Omis_{Leg}$	CUAD	77.3	75.9	75.8	74.9	95.6	55.2	56.8	59.2	60.1	57.5	82.5	81.9	80.4	78.4	96.5	82.5	81.8	81.0	78.4	96.5
	NLI	89.3	88.6	88.5	90.1	97.9	89.0	88.6	88.4	89.2	97.9	89.2	88.6	87.9	89.1	97.8	92.0	91.6	90.7	90.9	98.3
$Omis_{inT}$	CUAD	62.0	59.7	59.5	55.4	92.5	63.9	61.2	61.8	58.2	92.8	61.4	60.1	60.3	51.8	93.2	60.9	59.6	59.9	50.7	93.1
	NLI	62.0	59.0	60.0	61.1	92.2	61.7	58.3	59.1	61.1	92.0	64.7	63.0	63.6	58.0	93.5	65.9	64.4	64.5	57.8	93.8
$StrFlaw_{Leg}$	CUAD	63.1	61.1	58.8	51.5	92.2	65.4	61.1	57.8	52.7	47.7	52.1	51.1	48.5	38.6	91.0	51.8	50.7	48.9	38.5	91.0
	NLI	65.4	63.2	62.1	60.8	93.1	65.8	63.3	62.9	61.2	93.0	61.7	59.2	56.8	51.7	92.2	60.8	59.4	56.3	50.1	92.6
$StrFlaw_{inT}$	CUAD	57.6	55.2	55.7	47.5	92.0	57.9	55.7	56.3	48.7	92.0	54.3	53.2	53.5	43.4	92.1	56.0	54.8	55.1	44.9	92.4
	NLI	64.5	61.5	61.4	58.4	92.6	64.7	61.6	62.2	58.2	92.7	62.0	60.3	61.0	53.2	93.2	62.5	61.1	61.5	53.3	93.4

Table 9: Detailed Location Alignment Metrics (Eval 3) for all categories, models, and datasets. Category abbreviations: Amb=Ambiguity, Incon=Inconsistencies, MisTerm=Misaligned Terminology, Omis=Omission, StrFlaw=Structural Flaws, $_{Leg}$=Legal, $_{inT}$=inText. Metrics: ROUGE-1 (R_1), ROUGE-2 (R_2), ROUGE-L (R_L), METEOR (M), BERT F1 Score (F1). Models: GPT-4o-m=GPT-4o mini, Gem-2.0=Gemini 2.0, Gem-2.5=Gemini 2.5, Llama-3.3=Llama 3.3. Levels: L1 (zero-shot), L2 (one-shot). All values shown as percentages.

Completeness (coverage of relevant legal points), and Legal Reasoning (soundness of legal logic). Each metric ranges from 0-5, with scores evaluated by both GPT-4o and Gemini-2.5 judges to control for evaluator bias. This table reveals striking patterns in explanation generation capabilities that differ substantially from detection performance. The most pronounced finding is the universal weakness in Completeness: across all models, datasets, and categories, Completeness scores rarely exceed 3.0 and often fall below 2.0, while Clarity scores consistently reach 4.0+. This dissociation indicates that models generate fluent, well-structured explanations that nonetheless omit critical legal context—they articulate *what* the discrepancy is but fail to fully explain *why* it matters legally or *what specific risks* it creates. For instance, GPT-4o-mini achieves Clarity of 4.2 but Completeness of only 1.8 (Gemini-2.5 judge, NLI Ambiguity$_{inText}$, L1), suggesting surface-level coherence without substantive depth. This pattern appears consistent with models' tendency to generate confident but shallow legal analysis, a known limitation in domain-specific reasoning tasks. Judge variability reveals systematic evaluator biases: Gemini-2.5 judges consistently assign lower Completeness and Accuracy scores than GPT-4o judges (often by 0.5-1.0 points), particularly for GPT-generated explanations. This suggests Gemini judges apply stricter standards or that cross-architecture evaluation introduces systematic bias. However, the *relative ranking* of models remains consistent across judges: Gemini-2.5 (the evaluated model) consistently achieves the highest Legal Reasoning scores, followed by Gemini-2.0, then GPT-4o-mini and Llama-3.3. This ranking holds regardless of which judge evaluates, lending confidence to the robustness of the performance hierarchy. Category-specific patterns reveal that legal contradictions generally elicit higher-quality explanations than in-text contradictions: Legal Reasoning scores for Ambiguity$_{Legal}$ average 0.5-1.0 points higher than Ambiguity$_{inText}$ across all models. This likely reflects that legal violations require explicit citation of violated statutes, which provides additional structure and grounding to explanations, whereas in-text contradictions demand more abstract reasoning about internal consistency without external anchors. The Omission category shows particularly weak performance across all metrics, with Completeness scores often below 1.5, indicating that models struggle to articulate what is missing and why its absence creates legal risk—a fundamentally harder task than critiquing present text. The minimal improvement from L1 to L2 (typically <0.3 points across all metrics) suggests that explanation quality is primarily limited by models' intrinsic legal reasoning capacity rather than prompt engineering, marking a clear bottleneck for practical deployment.

D Contractual Pitfalls: Real-World Case Examples

Table 11 documents real-world U.S. legal cases illustrating the five core perturbation types in our benchmark. These cases demonstrate that the contradictions in our dataset are not merely theoretical constructs but reflect genuine failure modes that have led to litigation, substantial financial penalties, and precedent-setting rulings. Each entry connects a perturbation category to a specific case, detailing the contractual error, the legal dispute it triggered, and the resulting consequences. This grounding in case law validates our taxonomy and underscores the practical significance of evaluating LLM robustness to such flaws.

The selected cases span multiple domains—entertainment licensing, construction, financial services, insurance, and consumer contracts—illustrating that contractual pitfalls are pervasive across industries. The *Perini* case exemplifies Omission vulnerabilities: a missing consequential damages waiver resulted in a $14.5 million judgment, far exceeding the contract's value. The *Coinbase* case demonstrates Inconsistency risks: conflicting arbitration clauses across related agreements required Supreme Court intervention to determine enforceability. The *Fetch Interactive* case highlights Ambiguity: a cure period stated as "fifteen (30) days" created direct textual contradiction, leading to disputed obligations. These examples collectively show that even single-word changes ("shall" vs. "should") or missing clauses can fundamentally alter enforceability, liability exposure, and contractual obligations—precisely the risks our benchmark is designed to stress-test in LLMs deployed for legal contract review.

E Human Evaluation Guidelines and Rubric

This section provides the evaluation guidelines used by expert annotators to validate contradiction

quality in CLAUSE. All evaluators were NLP researchers with legal document analysis experience who reviewed perturbations using structured binary classification. The rubric was designed to ensure consistent assessment of contradiction strength, legal relevance, and contextual coherence across all perturbation types.

E.1 Evaluation Task Overview

For each perturbation instance, evaluators receive:

- **Perturbation Type**: One of ten categories (Ambiguity, Inconsistency, Structural Flaw, Misaligned Terminology, Omission—each with Legal and InText variants)
- **Original Text**: The unmodified contract excerpt
- **Changed Text**: The perturbed contract excerpt
- **Explanation**: Model-generated justification for why this constitutes a contradiction
- **Contradiction Type**: Whether this is a legal (external law violation) or in-text (internal document inconsistency) contradiction
- **Additional Context**: For legal contradictions, includes law citation, scraped law text, and law explanation; for in-text contradictions, includes contradicted text location and full contradicted section

Evaluation Goal: Determine whether the modification creates a **strong, meaningful, and unambiguous contradiction** that would reasonably cause legal uncertainty, enforceability issues, or compliance violations.

E.2 Binary Classification Criteria

Evaluators assign a binary label:

YES (Strong Contradiction Exists) Assign **YES** if the perturbation satisfies **all** of the following:

1. **Clarity**: The contradiction is immediately recognizable to a reasonably competent legal practitioner without requiring extensive legal research or interpretation
2. **Strength**: The modification directly conflicts with, negates, or undermines a legal requirement (for legal contradictions) or creates mutually exclusive obligations (for in-text contradictions)
3. **Significance**: The contradiction would plausibly lead to:
   - Enforceability disputes or litigation
   - Regulatory non-compliance or penalties
   - Contractual ambiguity requiring judicial interpretation
   - Material breach of obligations or rights
4. **Coherence**: The perturbation maintains linguistic naturalness and does not introduce obvious artifacts that would immediately reveal it as artificially generated

NO (Insufficient Contradiction) Assign **NO** if **any** of the following apply:

1. **Weak or Arguable**: The contradiction is subtle, speculative, or would require expert legal opinion to establish definitively
2. **Interpretable**: Both versions can be reasonably reconciled through standard contract interpretation principles
3. **Minor Modification**: The change is cosmetic, stylistic, or adds/removes non-essential details without affecting legal obligations
4. **Context-Dependent**: The contradiction only manifests under specific factual scenarios not clearly established in the contract
5. **Unnatural Language**: The perturbation introduces grammatical errors, nonsensical phrasing, or clearly artificial modifications that compromise linguistic quality

E.3 Category-Specific Guidance

Legal Contradictions For perturbations violating external legal standards:

- **Verify Law Applicability**: Check that the cited law/statute is genuinely applicable to the contract's jurisdiction and subject matter
- **Assess Violation Directness**: Ensure the modification directly violates the law rather than merely reducing compliance clarity
- **Consider Enforceability Impact**: Evaluate whether the violation would render the clause unenforceable or expose parties to legal liability
- **Example of YES**: Changing "Employees shall receive overtime pay at 1.5× regular rate for hours exceeding 40 per week" to "Employees may receive overtime compensation as determined by management" violates FLSA mandatory overtime provisions

- **Example of NO**: Changing "The company will comply with all applicable employment laws" to "The company shall make reasonable efforts to comply with applicable employment laws" weakens the commitment but does not create clear non-compliance

In-Text Contradictions For perturbations creating internal document conflicts:

- **Verify Cross-Reference Impact**: Confirm that the modification actually conflicts with specific text elsewhere in the document (not just theoretically)
- **Assess Mutual Exclusivity**: Ensure the conflicting clauses cannot both be true or enforced simultaneously
- **Consider Materiality**: Evaluate whether the contradiction affects substantive rights, obligations, or financial terms (not just procedural details)
- **Example of YES**: Section 2 defines "Net Revenue" as "gross revenue minus operating expenses," while Section 7 calculates royalties as "15% of Net Revenue (gross revenue only)"
- **Example of NO**: Section 3 refers to "the Effective Date" while Section 9 refers to "the Commencement Date," where both terms clearly reference the same date in context

E.4 Handling Edge Cases

When in Doubt If uncertain whether a contradiction meets the threshold for YES:

- Default to **NO** to maintain high benchmark quality
- Consider: "Would a practicing attorney immediately flag this in contract review?"
- Prioritize false negatives over false positives—excluding borderline cases preserves benchmark integrity

Disagreement Resolution When evaluators disagree:

- Majority vote determines classification (3 evaluators per perturbation)
- Significant disagreement (2-1 split on multiple perturbations from same category) triggers category-level review
- Systematic disagreement patterns are documented and analyzed for evaluator calibration

E.5 Quality Assurance Procedures

To ensure evaluation consistency and reliability:

1. **Calibration Phase**: All evaluators independently assess 100 pilot perturbations, followed by group discussion to align interpretation of criteria
2. **Inter-Rater Reliability**: Cohen's Kappa and Jaccard Coefficient computed pairwise and across all evaluators (see Table 2 in Section 2)
3. **Blind Evaluation**: Evaluators do not see each other's assessments or know the generating model to prevent bias
4. **Stratified Sampling**: Perturbations are sampled proportionally across all categories, datasets, and generation conditions to ensure representative coverage

Category	Judge	GPT-4o mini								Gemini 2.0								Gemini 2.5								Llama 3.3							
		L1				L2				L1				L2				L1				L2				L1				L2			
		Acc	Cla	Com	LR	Acc	Cla	Com	LR	Acc	Cla	Com	LR	Acc	Cla	Com	LR	Acc	Cla	Com	LR	Acc	Cla	Com	LR	Acc	Cla	Com	LR	Acc	Cla	Com	LR
CUAD																																	
Amb_{Leg}	**GPT4o**	3.3	4.2	2.5	3.2	3.3	4.2	2.6	3.3	4.3	4.8	3.6	4.1	4.2	4.8	3.6	4.0	4.1	4.7	3.6	4.1	4.3	4.8	3.9	4.3	3.0	3.9	2.3	2.9	3.0	3.9	2.4	3.0
	Gem2.5	3.5	4.6	1.8	3.2	3.5	4.7	2.0	3.4	4.4	4.9	3.2	4.2	4.3	4.9	3.2	4.2	3.8	4.7	3.2	4.0	4.1	4.8	3.6	4.4	3.2	4.5	1.8	2.7	2.8	4.2	1.8	2.7
Amb_{inT}	**GPT4o**	2.5	3.6	1.8	2.5	3.0	4.0	2.4	3.1	3.1	4.0	2.4	3.0	3.0	3.9	2.4	3.0	3.0	3.9	2.5	3.1	2.8	3.7	2.5	2.9	2.4	3.5	1.8	2.4	2.9	3.8	2.3	2.8
	Gem2.5	2.0	3.8	1.2	2.2	2.7	4.1	1.9	2.8	2.6	4.3	1.8	2.9	2.6	4.1	1.8	2.6	2.4	4.3	1.8	3.3	2.3	3.9	1.9	2.9	2.1	4.1	1.2	2.2	2.7	4.1	1.8	2.6
$Incon_{Leg}$	**GPT4o**	3.1	4.0	2.5	3.0	3.1	4.1	2.7	3.1	3.3	4.2	2.9	3.3	3.5	4.3	3.0	3.4	3.2	4.1	2.8	3.2	3.3	4.2	3.1	3.5	2.7	3.8	2.2	2.6	3.1	4.0	2.5	3.0
	Gem2.5	2.5	4.3	1.9	2.4	2.8	4.5	2.1	2.6	2.9	4.5	2.3	2.8	3.0	4.6	2.3	2.7	2.6	4.4	2.2	2.8	2.8	4.6	2.4	3.1	2.3	4.3	1.7	2.0	2.3	4.2	1.7	2.2
$Incon_{inT}$	**GPT4o**	2.4	3.6	1.9	2.5	3.3	4.1	2.7	3.3	2.8	3.8	2.3	2.8	2.9	3.8	2.5	2.8	2.7	3.9	2.3	2.9	2.8	3.7	2.5	2.9	2.2	3.4	1.7	2.2	3.2	4.1	2.8	3.1
	Gem2.5	1.7	3.8	1.1	2.1	2.8	4.1	2.0	2.9	2.2	4.2	1.4	2.5	2.3	3.8	1.8	2.4	2.2	4.3	1.7	3.1	2.1	3.8	2.0	2.9	1.7	3.9	1.0	1.8	3.1	4.4	2.4	2.8
$MisTerm_{Leg}$	**GPT4o**	3.7	4.5	3.1	3.7	3.7	4.5	3.1	3.7	4.0	4.6	3.6	4.0	4.1	4.7	3.7	4.1	3.9	4.5	3.6	4.0	3.9	4.5	3.6	4.1	3.1	4.0	2.6	3.0	2.3	3.5	1.8	2.4
	Gem2.5	3.6	4.6	2.7	3.5	3.6	4.6	2.7	3.5	4.0	4.8	3.2	3.9	4.0	4.8	3.3	3.8	3.5	4.6	3.1	3.8	3.5	4.6	3.2	3.8	3.0	4.4	2.1	2.7	1.9	4.0	1.2	2.0
$MisTerm_{inT}$	**GPT4o**	1.9	3.3	1.4	2.0	2.5	3.7	1.9	2.6	2.4	3.6	1.9	2.5	2.2	3.3	1.8	2.2	2.6	3.8	2.2	2.7	2.5	3.6	2.2	2.6	1.7	3.1	1.2	1.9	2.3	3.5	1.8	2.3
	Gem2.5	1.4	3.8	0.8	1.8	2.1	4.1	1.4	2.4	2.0	4.1	1.3	2.6	1.8	3.8	1.3	1.9	2.2	4.2	1.7	3.1	2.1	4.0	1.9	2.8	1.4	3.9	0.7	1.7	1.9	4.0	1.2	1.7
$Omission_{Leg}$	**GPT4o**	2.6	3.7	2.1	2.7	2.5	3.7	1.9	2.5	2.9	3.9	2.4	3.0	2.7	3.8	2.2	2.7	2.8	3.9	2.3	2.9	2.9	4.0	2.4	3.0	2.4	3.5	1.8	2.4	1.8	3.1	1.2	1.8
	Gem2.5	2.2	4.1	1.3	2.0	2.0	4.2	1.3	1.9	2.5	4.4	1.6	2.5	2.3	4.4	1.4	2.3	2.2	4.4	1.5	2.6	2.5	4.4	1.7	2.7	2.0	4.0	1.1	1.7	1.1	3.6	0.5	1.1
$Omis_{inT}$	**GPT4o**	1.6	3.1	1.1	1.8	1.7	3.0	1.1	1.8	2.0	3.4	1.5	2.1	1.5	2.7	1.0	1.4	2.2	3.6	1.8	2.4	1.4	2.8	1.0	1.5	1.8	3.1	1.2	1.9	1.6	3.0	1.2	1.6
	Gem2.5	0.9	3.5	0.5	1.3	1.0	3.4	0.5	1.1	1.5	4.0	0.8	2.1	0.9	3.4	0.5	1.1	1.6	4.1	1.1	2.5	0.8	3.6	0.5	1.3	1.3	3.9	0.6	1.5	0.9	3.5	0.5	1.0
$StrFlaws_{Leg}$	**GPT4o**	1.7	3.1	1.3	1.9	1.9	3.2	1.5	1.9	2.0	3.3	1.5	2.1	2.0	3.3	1.5	2.0	1.9	3.3	1.5	2.0	2.1	3.3	1.7	2.1	1.7	3.1	1.2	1.8	3.3	4.1	2.8	3.3
	Gem2.5	1.0	4.0	0.5	1.5	1.1	3.7	0.8	1.4	1.3	4.1	0.8	2.0	1.4	3.9	0.9	1.6	1.3	4.0	0.9	1.8	1.3	4.0	1.1	1.7	2.4	4.1	1.5	2.4	3.1	4.2	2.1	3.0
$StrFlaws_{inT}$	**GPT4o**	2.9	3.9	2.3	3.0	3.6	4.3	2.9	3.6	3.3	4.1	2.7	3.3	3.4	4.1	2.8	3.3	3.2	4.1	2.8	3.4	3.2	3.9	2.9	3.3	2.6	3.7	2.1	2.6	3.2	4.6	2.0	2.9
	Gem2.5	2.4	3.9	1.7	2.6	3.2	4.3	2.4	3.3	2.8	4.3	1.9	3.1	3.0	4.1	2.3	3.0	2.5	4.3	2.0	3.5	2.6	4.0	2.5	3.4	3.1	4.0	2.5	2.9	3.2	4.1	2.6	3.2
NLI																																	
Amb_{Leg}	**GPT4o**	3.3	4.2	2.6	3.3	3.3	4.2	2.6	3.2	4.2	4.8	3.6	4.0	4.3	4.8	3.6	4.2	4.3	4.8	3.9	4.3	4.3	4.8	3.9	4.2	3.1	4.0	2.5	2.9	3.3	4.1	2.8	3.3
	Gem2.5	3.5	4.7	2.0	3.4	3.6	4.7	1.9	3.5	4.3	4.9	3.2	4.2	4.4	4.9	3.2	4.3	4.1	4.8	3.6	4.4	4.2	4.9	3.6	4.4	3.2	4.6	2.0	2.9	3.6	4.7	2.3	3.4
Amb_{inT}	**GPT4o**	3.0	4.0	2.4	3.1	3.0	3.9	2.3	2.9	3.0	3.9	2.4	3.0	3.1	4.0	2.6	3.0	2.8	3.7	2.5	2.9	2.8	3.7	2.6	2.9	3.0	3.9	2.4	3.0	2.6	3.1	3.6	1.3
	Gem2.5	2.7	4.1	1.9	2.8	2.8	4.0	1.8	2.5	2.6	4.1	1.8	2.6	2.7	4.1	1.9	2.8	2.3	3.9	1.9	2.9	2.2	3.9	1.9	2.9	2.8	4.2	1.8	2.7	3.2	4.6	1.5	2.0
$Incon_{Leg}$	**GPT4o**	3.1	4.1	2.7	3.1	3.4	4.3	2.9	3.3	3.5	4.3	3.0	3.4	3.6	4.4	3.1	3.5	3.3	4.2	3.1	3.5	3.4	4.2	3.1	3.5	2.9	3.8	2.3	2.8	3.0	4.0	2.5	3.0
	Gem2.5	2.8	4.5	2.1	2.6	3.2	4.6	2.3	2.9	3.0	4.6	2.3	2.7	3.1	4.6	2.4	2.9	2.8	4.6	2.4	3.1	2.9	4.6	2.5	3.1	2.7	4.1	1.8	2.6	2.7	4.3	2.1	2.6
$Incon_{inT}$	**GPT4o**	3.3	4.1	2.7	3.3	3.1	4.0	2.4	3.0	2.9	3.8	2.5	2.8	3.1	3.9	2.6	3.0	2.8	3.7	2.5	2.9	2.9	3.8	2.6	3.0	3.1	4.0	2.5	3.0	3.0	3.9	2.3	2.9
	Gem2.5	2.8	4.1	2.0	2.9	2.7	4.0	1.8	2.5	2.3	3.8	1.8	2.4	2.4	3.9	1.8	2.5	2.1	3.8	2.0	2.9	2.2	3.8	2.0	2.9	2.3	4.2	1.7	2.2	2.7	4.0	1.7	2.5
$MisTerm_{Leg}$	**GPT4o**	3.7	4.5	3.1	3.7	3.8	4.4	3.2	3.7	4.1	4.7	3.7	4.1	4.2	4.7	3.8	4.2	3.9	4.5	3.6	4.1	4.0	4.6	3.6	4.1	3.2	4.1	2.8	3.1	3.4	4.2	2.9	3.3
	Gem2.5	3.6	4.6	2.7	3.5	3.7	4.7	2.9	3.7	4.0	4.8	3.3	3.8	4.2	4.8	3.5	4.1	3.5	4.6	3.2	3.8	3.7	4.7	3.3	4.0	3.1	4.4	2.4	2.8	3.4	4.5	2.7	3.3
$MisTerm_{inT}$	**GPT4o**	2.5	3.7	1.9	2.6	2.1	3.4	1.6	2.2	2.2	3.3	1.8	2.2	2.2	3.3	1.8	2.2	2.5	3.6	2.2	2.6	2.5	3.5	2.2	2.6	2.3	3.5	1.8	2.4	2.1	3.4	1.5	2.2
	Gem2.5	2.1	4.1	1.4	2.4	1.7	3.7	1.1	2.0	1.8	3.8	1.3	1.9	1.9	3.8	1.3	1.9	2.1	4.0	1.9	2.8	2.0	3.8	1.8	2.6	1.9	4.0	1.2	2.0	1.7	3.7	1.0	1.8
$Omis_{Leg}$	**GPT4o**	2.5	3.7	1.9	2.5	2.7	3.8	2.1	2.7	2.7	3.8	2.2	2.7	2.9	4.0	2.4	2.9	2.9	4.0	2.4	3.0	2.9	4.0	2.4	3.1	2.3	3.5	1.8	2.3	2.4	3.6	1.8	2.4
	Gem2.5	2.0	4.2	1.3	1.9	2.3	4.2	1.5	2.2	2.3	4.4	1.4	2.3	2.4	4.4	1.6	2.5	2.5	4.4	1.7	2.7	2.5	4.4	1.8	2.7	1.9	4.0	1.2	1.7	2.0	4.1	1.2	1.9
$Omis_{inT}$	**GPT4o**	1.7	3.0	1.1	1.8	1.6	2.9	1.1	1.6	1.5	2.7	1.0	1.4	1.6	2.9	1.2	1.6	1.4	2.8	1.0	1.5	1.5	2.8	1.1	1.5	1.8	3.1	1.2	1.8	1.8	3.1	1.2	1.9
	Gem2.5	1.0	3.4	0.5	1.1	1.0	3.3	0.4	1.0	0.9	3.4	0.5	1.1	1.1	3.3	0.5	1.1	0.8	3.6	0.5	1.3	1.0	3.3	0.7	1.3	1.1	3.6	0.5	1.1	1.3	3.5	0.5	1.4
$StrFlaw_{Leg}$	**GPT4o**	1.9	3.2	1.5	1.9	1.9	3.2	1.5	2.0	2.0	3.3	1.5	2.0	2.2	3.5	1.7	2.3	2.1	3.3	1.7	2.1	2.0	3.3	1.5	2.0	1.6	3.0	1.2	1.6	1.7	3.1	1.3	1.8
	Gem2.5	1.1	3.7	0.8	1.4	1.1	3.7	0.8	1.3	1.4	3.9	0.9	1.6	1.4	3.9	1.0	1.8	1.3	4.0	1.1	1.7	1.3	3.8	1.0	1.6	0.9	3.5	0.5	1.0	0.9	3.7	0.5	1.1
$StrFlaw_{inT}$	**GPT4o**	3.6	4.3	2.9	3.6	3.3	4.1	2.6	3.2	3.4	4.1	2.8	3.3	3.3	4.2	2.9	3.3	3.2	3.9	2.9	3.3	3.2	3.9	2.9	3.2	3.3	4.1	2.8	3.3	3.3	4.1	2.6	3.3
	Gem2.5	3.2	4.3	2.4	3.3	2.9	4.0	2.1	2.9	3.0	4.1	2.3	3.0	3.0	4.1	2.2	3.0	2.6	4.0	2.5	3.4	2.5	3.9	2.3	3.3	3.1	4.2	2.1	3.0	3.2	4.2	2.1	3.1

Table 10: Explanation quality evaluation across models, datasets, and judges. Metrics: Accuracy (Acc), Clarity (Cla), Completeness (Com), Legal Reasoning (LR). Models: GPT-4o mini, Gemini 2.0, Gemini 2.5, Llama 3.3. Levels: L1 (zero-shot), L2 (one-shot). Judges: GPT4o, Gemini 2.5 (Gem2.5). All values on a scale of 0-5.

Pitfall	Description	U.S. Case	Contractual Error	Consequence
Ambiguity	Vague, unclear, or contradictory language creating uncertainty in interpretation, making contract terms susceptible to multiple conflicting meanings.	*Fetch Interactive Television, LLC v. Touchstream Technologies, Inc.*	Cure period stated as "fifteen (30)" days, creating direct textual contradiction and ambiguity in timeline.	Disputed deadlines, unexpected obligations based on court interpretation.
Omission	Deliberate removal or absence of critical information, clauses, or terms necessary for complete understanding or legal enforceability.	*Perini Corp. v. Greate Bay Hotel & Casino, Inc.*	Construction contract lacked waiver of consequential damages or clear limitation of liability.	Exposure to unforeseen catastrophic claims for consequential or punitive damages (multi-million dollar award).
Inconsistency	Direct contradictions between sections where statements, obligations, or terms are mutually exclusive or logically incompatible.	*Coinbase, Inc. v. Suski*	Two agreements between same parties contained conflicting dispute resolution clauses—one delegating to arbitrator, other to courts.	Uncertainty, breach claims, difficulty determining true obligation, requiring court intervention.
Misaligned Terminology	Inconsistent use of defined terms, or terminology conflicting with established legal definitions or industry standards.	*Lithko Contracting, LLC, et al. v. XL Insurance America, Inc.*	Waiver clause used "no party shall be liable to another party" but inconsistently referred to "Tenant [Amazon]" separately, creating ambiguity.	Critical clauses may not cover intended parties, leading to disputes over scope of protection.
Structural Flaws	Modifications disrupting logical organization, hierarchy, or cross-referencing, creating navigational confusion or broken dependencies.	*Carnival Cruise Lines, Inc. v. Shute*	Forum selection clause placement and communication scrutinized for fundamental fairness to determine enforceability.	Legally significant clauses may be unenforceable if not reasonably communicated or if placement is unfair; uncertainty regarding prevailing terms.

Table 11: Real-world U.S. case examples illustrating the five core perturbation types. Each case demonstrates how contractual flaws lead to disputes and adverse legal consequences.

```
{
  "file_name": "DovaPharmaceuticalsInc_20181108_10-Q_EX-10.2_11414857_EX-10.2_Promotion Agreement.txt",
  "perturbation": [
    {
      "type": "Omissions - In Text Contradiction",
      "original_text": "1.19 \"Detail(s)\" shall mean a Product presentation during a face-to-face sales call between a
Target Professional and a Sales Representative, during which a presentation of the Product's attributes, benefits,
prescribing information and safety information are orally presented, for use in the Field in the Territory. Neither e-
details, nor presentations made at conventions, exhibit booths, a sample drop, educational programs or speaker meetings, or
similar gatherings, shall constitute a Detail.",
      "changed_text": "1.19 \"Detail(s)\" shall mean a Product presentation during a sales call between a Target Professional
and a Sales Representative, during which a presentation of the Product's attributes, benefits, prescribing information and
safety information are presented, for use in the Field in the Territory. Presentations made at conventions, exhibit booths, a
sample drop, educational programs or speaker meetings, or similar gatherings, shall constitute a Detail.",
      "explanation": "The term 'Detail' is redefined to include presentations at conventions and speaker meetings, directly
contradicting the original definition that explicitly excluded them. This introduces ambiguity and potential conflict, as the
contract uses the term 'Detail' to calculate promotion fees and minimum activity levels for Valeant. This change directly
contradicts the scope of what a 'Detail' is considered as.",
      "location": "Section 1.19",
      "contradicted_location": "Section 4.2.2",
      "contradicted_text": "(a) Valeant shall keep accurate and complete records, consistent with pharmaceutical industry
standards, of each Detail and its obligations hereunder in connection therewith. Such records shall be kept for the longer of
(i) [***] after the end of the Calendar Year to which they relate and (ii) such period of time as required by Applicable
Laws. Within [***] following the end of each Calendar Quarter during the Term, Valeant shall provide Dova with a written
report (each a \"Detail Report\"), setting out (i) the quarterly average number of Sales Representatives during such Calendar
Quarter (calculated by taking the sum of the number of Sales Representatives employed by Valeant (or its affiliates) that
have incentive compensation packages that comply with the terms of Section 4.1.3 on each Business Day of the Calendar Quarter
divided by the number of Business Days in such Calendar Quarter) (the \"Quarterly Average Sales Force Size\"), and (ii) the
aggregate actual number of Details for the Product made by its Sales Representatives during such Calendar Quarter, and the
number of Details broken down by the name of the Target Professionals,. Through the JSC, the Parties shall agree on a
mutually acceptable form of Detail Report.",
      "contradiction_exists": "YES",
      "justification": "The changed text redefines 'Detail' to include presentations at conventions, which the original
definition explicitly excluded. This directly contradicts the contradicted text, which mandates accurate records of each
'Detail' for calculating obligations. The redefinition creates ambiguity in what constitutes a 'Detail' for record-keeping
purposes, thus making it unclear which presentations Valeant needs to track and report, leading to a strong in-text
contradiction. The original definition says 'Neither e-details, nor presentations made at conventions, exhibit booths, a
sample drop, educational programs or speaker meetings, or similar gatherings, shall constitute a Detail.' whereas the changed
text says 'Presentations made at conventions, exhibit booths, a sample drop, educational programs or speaker meetings, or
similar gatherings, shall constitute a Detail.'"
    }
  ]
}
```


Figure 3: Example JSON structure for in-text contradiction perturbations, showing the metadata schema used throughout the dataset.

```
{
  "file_name": "PharmagenInc_20120803_8-KA_EX-10.1_7693204_EX-10.1_Endorsement Agreement.txt",
  "perturbation": [
    {
      "type": "Ambiguities - Ambiguous Legal Obligation",
      "original_text": "All HDS' uses of Celebrity Attributes in connection with the Commercial and/or press releases shall
be subject to the prior written approval of Celebrity via his agent, CSE. Said written approval must be given within five (5)
business days of CSE's receipt or said usage shall be deemed unapproved. Any such usage featuring Celebrity in the Commercial
and/or press releases shall be and remain the property of HDS; however, HDS shall have the right to use said Commercial
and/or press releases solely as outlined in Paragraph 3 and only during the Term.",
      "changed_text": "All HDS' uses of Celebrity Attributes in connection with the Commercial and/or press releases should
aim to obtain the prior approval of Celebrity via his agent, CSE. A reasonable effort should be made to obtain written
approval within a suitable timeframe of CSE's receipt. Any such usage featuring Celebrity in the Commercial and/or press
releases shall be and remain the property of HDS; however, HDS shall have the right to use said Commercial and/or press
releases solely as outlined in Paragraph 3 and only during the Term.",
      "explanation": "Changing 'shall be subject to prior written approval' to 'should aim to obtain the prior approval' and
'must be given within five business days' to 'reasonable effort should be made within a suitable timeframe' introduces
ambiguity. It is no longer mandatory for HDS to obtain approval, and the time frame is vague.",
      "contradicted_law": "Lanham Act - False Endorsement",
      "law_citation": "15 U.S. Code § 1125(a)",
      "law_url1": [
        "https://www.law.cornell.edu/uscode/text/15/1125"
      ],
      "law_url2": [
        "https://uscode.house.gov/view.xhtml?req=granuleid%3AUSC-prelim-title15-section1125&edition=prelim"
      ],
      "law_explanation": "The modified text creates ambiguity around the approval process for using the celebrity's
attributes, potentially leading to usage without explicit consent. If HDS uses Celebrity's attributes in a way that implies
endorsement without proper approval, it could constitute a false endorsement, violating the Lanham Act, which prohibits
misrepresenting goods or services in commerce.",
      "location": "Paragraph 9",
      "scraped_snippet_1": "(a) Civil action (1) Any person who, on or in connection with any goods or services...uses in
commerce any word, term, name, symbol, or device...or any false designation of origin, false or misleading description of
fact, or false or misleading representation of fact, which— (A) is likely to cause confusion, or to cause mistake, or to
deceive as to the affiliation, connection, or association of such person with another person, or as to the origin,
sponsorship, or approval of his or her goods, services, or commercial activities by another person...shall be liable in a
civil action by any person who believes that he or she is or is likely to be damaged by such act.",
      "scraped_snippet_2": "§1125. False designations of origin, false descriptions, and dilution forbidden (a) Civil action
(1) Any person who, on or in connection with any goods or services, or any container for goods, uses in commerce any word,
term, name, symbol, or device, or any combination thereof, or any false designation of origin...",
      "contradiction_exists": "YES",
      "justification": "The original contract mandates prior written approval from the celebrity's agent, CSE, within five
business days. The modified text changes 'shall be subject to prior written approval' to 'should aim to obtain the prior
approval' and replaces the specific 'five business days' with 'a suitable timeframe,' introducing ambiguity and making the
approval process non-mandatory. The Law Explanation states that this ambiguity could lead to usage without explicit consent
and potentially constitute a false endorsement violating the Lanham Act. Both Scraped Law Snippets, specifically 15 U.S. Code
§ 1125 (a)(1)(A), state that any false or misleading representation of fact which is likely to cause confusion, or to cause
mistake, or to deceive as to the affiliation, connection, or association of such person with another person, or as to the
origin, sponsorship, or approval of his or her goods, services, or commercial activities by another person is a violation.
The modified contract weakens the obligation to obtain approval, increasing the risk of unauthorized use and potentially
implying endorsement without consent, thus contradicting the law's intent to prevent false endorsement."
    }
  ]
}
```


Figure 4: Example JSON structure for legal contradiction perturbations, including law citations and statutory references.