

CONTRAVIS: Evidence-Grounded Visual Analytics for Contradiction Review in Legal Contracts

Luis Sante, Paula Lima, Mariana Rocha, and Jorge Poco
School of Applied Mathematics, Fundação Getulio Vargas (FGV)
{taipe.luis, paula.lima.1, mariana.rocha}@fgv.edu.br, jorge.poco@fgv.br

Abstract—Legal contracts are structurally complex documents in which contradictions may emerge across distant and interconnected provisions. Although large language models (LLMs) improve legal language understanding, contradiction analysis remains a human-centered and evidence-grounded review task. We present CONTRAVIS, a visual analytics system for human-in-the-loop contradiction analysis in legal contracts. The system models contracts as typed paragraph graphs that combine explicit contractual references with semantic relationships between paragraphs. This graph plays a dual role: it conditions LLM reasoning and serves as the interactive representation the analyst explores, keeping model context and human inspection aligned across coordinated views. In a controlled comparison, graph-conditioned reasoning recovered more injected contradictions than standalone LLM analysis as contract length grew, while surfacing additional candidates for analyst validation. A formative study with contract-domain lawyers indicated that in-context evidence comparison supported contradiction validation, and we distill design implications for evidence-grounded, LLM-assisted document review.

I. INTRODUCTION

Legal contracts are high-stakes documents where small inconsistencies can create major financial and operational risks. However, contract review remains largely manual and cognitively demanding, particularly for long agreements containing dense cross-references, repeated obligations, and domain-specific language [1], [2]. Recent benchmarks further show that even strong LLMs still struggle with reliable legal reasoning in risk-sensitive contractual settings [3]. Prior work has advanced contract understanding through clause extraction, legal question answering, metadata identification, and document-level inference [1], [4]–[6]. At the same time, contradiction-oriented approaches have explored inconsistency detection in long documents [7]–[9] and logical relation classification between clause pairs [10]. However, these approaches operate as end-to-end predictors over isolated clause pairs or retrieved chunks: the context that conditions a prediction is not exposed to the analyst, so a flagged pair cannot be traced back to the contractual structure that makes it relevant. In parallel, legal visualization systems demonstrate the value of structural exploration of legal corpora [11]–[13], but do not operationalize contradiction discovery in contracts [14].

We argue that contradiction analysis in legal contracts is a structure-aware analytical task rather than a clause-level classification problem. To support this workflow, analysts must inspect related provisions, compare evidence across distant

regions, and determine whether inconsistencies represent conflicts, exceptions, or refinements.

We present CONTRAVIS, a visual analytics system for human-in-the-loop contradiction analysis in legal contracts that combines graph-conditioned LLM reasoning with multiple views. The system represents contracts as typed paragraph graphs combining explicit references with semantic relationships between paragraphs. CONTRAVIS uses this graph as both a computational substrate for contradiction analysis and a visual structure for contextual inspection. Following LegalWiz [15], contradictions are typed as *Temporal*, *Numerical*, *Authority*, *Process*, *Policy Reversal*, or *Specificity*, and scoped as intra- or inter-paragraph (App. A). CONTRAVIS supports contract triage, paragraph-neighborhood exploration, verification, and evidence-grounded legal interpretation.

We frame this work as an early-stage design study [16]: its contributions lie at the problem-characterization and task-abstraction levels of visualization design rather than in new visual encodings. This work makes three contributions. First, we formulate contradiction review as an evidence-grounded, human-in-the-loop visual analytics task, grounded in lawyer-centered requirements and abstracted into analytical tasks, rather than automatic clause-pair classification. Second, we propose a typed paragraph graph representation with a dual role: the structure that conditions LLM reasoning is also the interactive representation the analyst explores, keeping model context and human inspection aligned—unlike standard RAG pipelines, where the conditioning context stays hidden from the analyst. Third, we present CONTRAVIS, an interactive system integrating document reading, evidence comparison, paragraph-neighborhood exploration, and grounded legal assistance, together with a formative expert evaluation and a controlled comparison showing that graph-conditioned reasoning recovers more injected contradictions as contracts grow, while surfacing additional candidates for analyst validation and design implications for evidence-grounded, LLM-assisted document review. Code, prompts, and evaluation material are available at <https://visualdslab.com/papers/ContraVis>.

II. RELATED WORK

A. Legal Contract Understanding with NLP and LLMs

Research on legal contract analysis evolved from clause extraction benchmarks and expert-annotated datasets toward broader reasoning and understanding tasks [1], [2]. ContractNLI showed that document-level inference over contracts

Table 1
SYSTEM REQUIREMENTS AND THE ANALYTICAL TASKS.

System Requirements	T1	T2	T3	T4	T5
R1: Structure-aware contradiction discovery in contracts	✓	✓	✓		
R2: Contextual inspection of potentially risky clauses	✓	✓		✓	
R3: Grounded legal interpretation and exploration				✓	✓
R4: Evidence-grounded validation of findings		✓	✓		✓

remains challenging for language models, particularly in long-document settings [6]. Subsequent work explored metadata extraction, legal question answering, summarization, and contract understanding applications [4], [5], [17]. Recent studies show that structural and formatting cues strongly influence legal LLM behavior and that preserving document structure aids evidence localization and legal reasoning [18]–[20]. Although these approaches improve legal language understanding, they operate as end-to-end predictors, offering analysts no structural context for inspecting evidence or validating contradictions.

B. Contradiction Detection and Retrieval-Augmented Analysis

Contradiction analysis has been explored across factual, regulatory, and legal domains [21]. In legal contracts, prior work formulates it as logical relation classification between clause pairs using natural language inference [10]. Recent studies investigated self-contradiction detection [7], reference-aware contradiction discovery [8], chunk-based inconsistency analysis [9], and multi-agent contradiction workflows [15]. In parallel, graph-aware retrieval and reranking show that modeling document connectivity improves evidence discovery and contextual reasoning [22]–[24]. CONTRAVIS draws on both lines but differs in what the analyst sees: whereas these approaches predict over isolated clause pairs or chunks with opaque retrieved context, our candidates remain traceable to verbatim evidence spans and inspectable graph neighborhoods, supporting expert validation rather than only detection.

C. Visual and Interactive Analysis of Legal Documents

Prior visualization and visual analytics systems show that structural relationships, citation networks, and argument connections matter for legal reasoning workflows [11], [13], [14]. More recent research in legal HCI advocates moving beyond static document-centric workflows toward systems that support exploration, interpretation, and interactive reasoning over legal content [25]. Visual analytics systems have also begun coupling graph structure with LLM reasoning to keep model-generated explanations inspectable [26]. Whereas these systems support structural exploration of legal corpora, CONTRAVIS couples the visual structure to the reasoning process: the typed paragraph graph shown to the analyst is the same structure that conditions LLM inference, aligning model context with human inspection.

III. SYSTEM OVERVIEW

A. System Requirements and Analytical Tasks

Requirements were elicited through semi-structured discussions with 2 contract-domain lawyers, who described

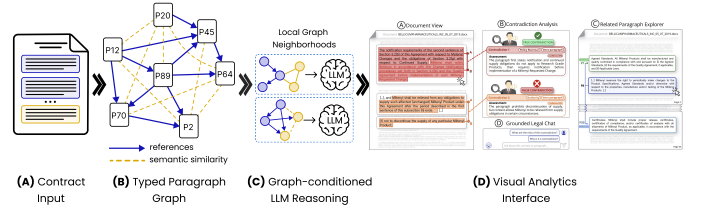


Figure 1. Human-in-the-loop workflow: from a contract (A), CONTRAVIS builds a typed paragraph graph (B), performs graph-conditioned LLM reasoning to surface candidates with supporting evidence (C), which experts validate through coordinated views (D).

their review practice and the difficulty of tracing obligations across distant provisions. From these we distilled four lawyer-centered requirements (R1–R4), abstracted into five analytical tasks (T1–T5) mapped in Table I: **T1**: triage contradiction-prone regions; **T2**: inspect structure-aware neighborhood context; **T3**: verify contradiction evidence and scope; **T4**: explore grounded legal questions and interpretations; and **T5**: interpret findings and explore corrective reasoning.

B. Workflow

Fig. 1 summarizes the human-in-the-loop workflow: from a legal contract (A), CONTRAVIS constructs a typed paragraph graph (B), retrieves local neighborhoods as contextual evidence for graph-conditioned reasoning (C), and surfaces contradiction candidates with supporting evidence, which analysts inspect through four coordinated views (D, Sec. V).

IV. GRAPH-CONDITIONED CONTRADICTION ANALYSIS

This section describes the graph representation, controlled contradiction-generation process, and graph-conditioned reasoning strategy used in CONTRAVIS.

A. Graph Representation

Given a contract $\mathcal{C} = \{p_1, \dots, p_n\}$, we construct a typed paragraph graph $\mathcal{G} = (V, E_r, E_s)$, where each node $v_i \in V$ represents a paragraph p_i .

Referential edges (E_r). Edges connecting paragraphs through explicit contractual cross-references, identified with rule-based patterns over legal identifiers (e.g., “Section 3.2”).

Semantic edges (E_s). Edges connecting semantically related paragraphs whose all-MiniLM-L6-v2 embeddings exceed a cosine similarity of $\tau = 0.80$.

Referential links preserve the document’s explicit legal structure, whereas semantic links capture latent topical relationships. Construction rules, preprocessing, and parameter settings are described in App. B.

B. Controlled Evaluation Dataset

To study contradiction inspection under controlled conditions, we construct synthetic contradiction-analysis scenarios from contracts in the CUAD corpus [2]. Starting from the typed paragraph graph $\mathcal{G}$, structurally related paragraph neighborhoods are provided as contextual input to GPT-4.1, which generates controlled inconsistencies using a taxonomy-conditioned prompt derived from LegalWiz [15] (App. E). The

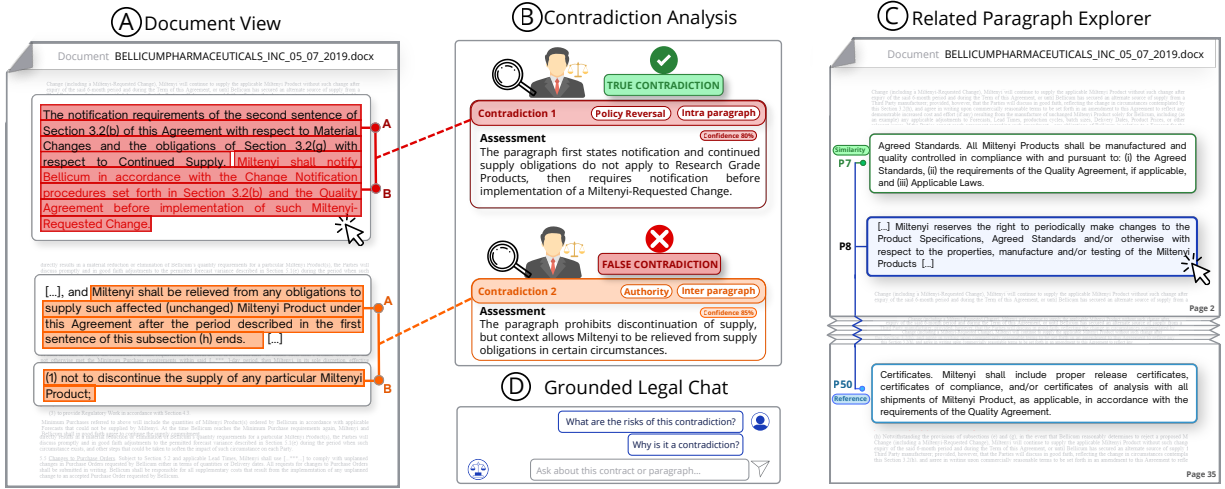


Figure 2. CONTRAVIS interface. (A) Document View highlights conflicting evidence spans in the contract text. (B) Contradiction Analysis lists candidates with category, scope, assessment, confidence, and evidence. (C) Related Paragraph Explorer brings distant but related clauses into a shared comparison space, distinguishing referential and semantic links. (D) Grounded Legal Chat supports explanation and follow-up reasoning grounded in the selected context.

generated contradictions are inserted into compatible locations of the original contract while preserving the remaining document structure, producing a synthetic contradicted contract C^* . Each candidate is kept only when its NLI contradiction score satisfies $\gamma \geq 0.70$ and its per-word perplexity increase satisfies $\Delta PPL_{\text{norm}} < \delta = 0.01$ (App. C); accepted candidates are then manually reviewed by the researchers for contextual plausibility, consistency with the intended category, and alignment with the contractual regions used during generation. Contradiction locations are known to the experimenters but hidden from both the system and analysts during inspection. For each contract, five contradictions are targeted. Candidates are filtered with NLI and perplexity, manually reviewed, and rejected ones are regenerated until five validated contradictions are obtained.

C. Structure-Aware Analytical Reasoning

During analysis, the system reconstructs the typed paragraph graph G^* from the input contract and retrieves local graph neighborhoods for contextual inspection (App. D). Neighborhoods are limited to direct (1-hop) graph connections, combining all referentially linked paragraphs and the top-5 semantically similar neighbors associated with the selected paragraph. Contextualizing each paragraph through both referential and semantic relationships lets candidates be surfaced from structurally connected regions together with evidence spans, contradiction categories, and contextual relationships (App. E). Rather than ranking candidates as final decisions, CONTRAVIS presents them as analytical evidence for expert inspection, validation, and interpretation.

V. CONTRAVIS SYSTEM

Fig. 2 presents the four coordinated views that support contradiction-centered contract review in CONTRAVIS; each view is annotated below with the analytical tasks it supports.

A. Document View

The *Document View* (Fig. 2.A) supports close reading and triage (T1) in long contracts. Because the contract text is the

legal source of truth, the document—rather than a derived abstraction—is the primary representation in which analysts read and judge provisions. The view presents the contract as an ordered sequence of paragraphs linked to the typed paragraph graph; contradiction candidates and related paragraphs are highlighted in the text via linked highlighting, and selecting a paragraph updates the graph neighborhood, contradiction evidence, and assistant context across the other views.

B. Contradiction Analysis

The *Contradiction Analysis* panel (Fig. 2.B) supports evidence inspection and validation (T3, T5), exposing candidate inconsistencies together with supporting evidence rather than final predictions—reflecting the legal nature of the task, in which potential conflicts must be interpreted in context before they count as meaningful inconsistencies. For each candidate, the panel presents: (1) the contradiction category, (2) a brief assessment, (3) contradiction scope (intra- or inter-paragraph), (4) the model-reported confidence score (0–100), and (5) highlighted evidence spans associated with the candidate. Evidence snippets are synchronized with the *Document View*, letting analysts cross-check conflicting provisions in their original contractual context. Users can mark each candidate as a true contradiction, a false contradiction, or a contextual exception/refinement, and the panel connects to the *Grounded Legal Chat* to request explanations, inspect potential risks, and explore revision strategies for the selected candidate.

C. Related Paragraph Explorer

The *Related Paragraph Explorer* (Fig. 2.C) supports neighborhood exploration (T2) by externalizing the graph neighborhood of a selected paragraph v_i , distinguishing referential edges E_r from semantic edges E_s . We adopt a node-link representation because neighborhoods are small in practice (1-hop referential links plus top-5 semantic neighbors): at this scale the diagrams remain legible, preserve per-paragraph coordination with the *Document View*, and map edge types to

Table II
TESTED DOCUMENTS AND CONTRADICTION OUTCOMES.

Cont.	Pages	$ V $	Inj.	LLM		CONTRAVIS	
				TP	FP	TP	FP
C_1^*	15	208	5	3	1	4	2
C_2^*	31	270	5	3	2	4	4
C_3^*	47	517	5	1	1	3	4
C_4^*	55	403	5	1	2	4	3

$|V|$: number of paragraphs; Inj.: injected contradictions; TP: true positives; FP: false positives.

link encodings, which matrices would obscure. A key interaction of this view is proximity-based comparison, motivated by the pairwise nature of contradiction validation: conflicting provisions must be read together yet may lie far apart. When analysts select a paragraph or contradiction candidate, CONTRAVIS brings distant but structurally related paragraphs into a shared comparison space, enabling side-by-side inspection of evidence without navigating across contract sections.

D. Grounded Legal Chat

The *Grounded Legal Chat* (Fig. 2.D) supports grounded legal interpretation and follow-up reasoning (T4). Unlike a generic conversational assistant, the chat is conditioned on the selected paragraph, its graph neighborhood, and the current contradiction evidence—the same context that conditions detection—so conversational answers stay anchored to inspectable evidence rather than free-form model output. Analysts can ask global or targeted questions about clauses and contradiction candidates, supporting explanation, risk inspection, and revision-oriented interactions.

VI. CONTROLLED COMPARISON AND USAGE SCENARIO

A. Controlled Comparison with Standalone LLM Analysis

We compare CONTRAVIS against standalone LLM analysis—the same detection model and prompt structure operating on the contract text without graph conditioning—on four contracts of increasing length (15–55 pages; 208–517 paragraphs), each containing five injected contradictions (C^*). The comparison is deliberately small in scale: rather than a benchmark, it functions as an ablation that isolates the contribution of the structural component precisely where it matters most, long contracts. Table II reports per-document true and false positives; matching is automatic at the evidence-span level, with a surfaced candidate counting as a true positive when its evidence spans overlap an injected contradiction, and as a false positive otherwise.

On the shorter contracts, both approaches recover most injected contradictions (3–4 of 5). As length grows, however, standalone recovery degrades to 1 of 5, while graph-conditioned reasoning still recovers 3–4 of 5. The failure mode is consistent: when a contradiction depends on distant or semantically related provisions, the standalone model produced weakly contextualized candidates, whereas graph neighborhoods keep the relevant regions jointly visible to the model and, through the coordinated views, to the analyst—who can compare evidence spans, inspect related paragraphs,

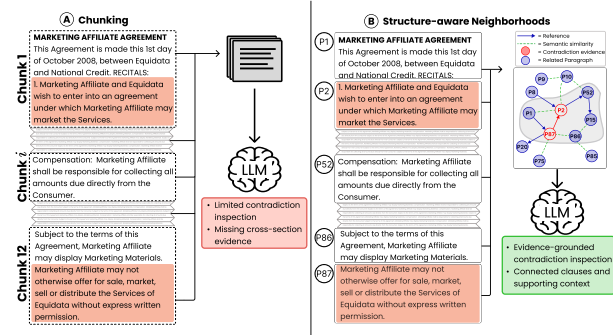


Figure 3. Fixed-size chunking vs. structure-aware neighborhoods. (A) Contradictory clauses far apart fall into different chunks, so cross-chunk contradictions are missed. (B) The typed paragraph graph reconnects them through referential and semantic links, enabling evidence-grounded analysis across sections.

and understand why a contractual region is relevant to a surfaced candidate.

Graph conditioning also surfaces more candidates beyond the injected ones (2–4 versus 1–2 per contract), reflecting the ambiguity of legal language and the deliberately high-recall detection bias (App. E). In legal review this trade-off is asymmetric: a missed contradiction is a silent legal risk, whereas an additional candidate costs bounded review effort. Consistent with the system’s human-in-the-loop design, CONTRAVIS does not treat these cases as final detection errors but exposes them as inspectable candidates (each with evidence spans and a confidence score) that analysts can validate, contextualize, or discard through coordinated visual inspection (Fig. 2.B).

B. Chunking Limitations and Structure-Aware Neighborhoods

Large contracts often contain dependencies distributed across distant sections of the document. Although fixed-size chunking reduces context-window overload during LLM processing, it fragments structural and semantic relationships that analysts must inspect when evaluating supporting evidence and potential contradictions. By preserving these relationships in the typed paragraph graph, CONTRAVIS supports graph-conditioned reasoning and interactive exploration, allowing analysts to compare evidence across distant contractual regions and inspect the context underlying contradiction candidates. Fig. 3 illustrates this contrast: fixed-size chunking may separate related clauses into independent contexts, whereas structure-aware neighborhoods reconnect them through referential and semantic relationships.

VII. FORMATIVE EXPERT EVALUATION

We conducted a formative expert evaluation with three contract-domain experts to assess whether CONTRAVIS supports meaningful contradiction inspection, evidence interpretation, and structure-aware exploration in legal-review workflows. Rather than aiming for statistically generalizable usability measures, the evaluation focused on collecting domain-informed feedback about the usefulness, interpretability, and limitations of the proposed visual analytics workflow. This

evaluation is appropriate for an early-stage design study because access to contract-specialized legal reviewers is limited and their qualitative feedback is critical for refining the design.

A. Participants

The study involved three contract-domain lawyers from a large Brazilian law firm. None of them had participated in the requirements-elicitation discussions (Sec. III), so the evaluation was not performed by the experts who informed the design. All participants had direct experience with contract drafting and review. P1 had 6–10 years of contract-related experience and worked with contracts several times per week. P2 and P3 had 3–5 years of experience; P2 worked with contracts several times per month, while P3 worked with contracts several times per week.

B. Procedure

The evaluation was conducted through live in-person sessions, each lasting approximately 30 minutes. Each session began with an introduction to CONTRAVIS, including the typed paragraph graph representation, contradiction-analysis workflow, and coordinated visual interface. Participants then interacted with the system through two study tasks (S1–S2, distinct from the analytical tasks T1–T5), thinking aloud about why each surfaced candidate was a correct or incorrect contradiction. Sessions were not recorded, but the experimenters noted participants’ feedback.

S1: Guided contradiction inspection. Participants explored a pre-selected C^* contract containing an injected contradiction involving late fees and bounced-check penalties. They inspected supporting evidence, compared contractual fragments, and explored relationships among the involved clauses.

S2: Free exploration of long contracts. Participants selected contracts ranging from 15 to 55 pages and freely explored contradiction candidates surfaced by the system, focusing on cases they considered legally relevant.

Following the exploration tasks, participants answered usability, usefulness, and contradiction-analysis questions using a five-point Likert scale (App. F) and provided open-ended feedback about useful features, workflow limitations, and possible improvements.

C. Questionnaire Results

Responses to the usability items were uniformly positive (agree or strongly agree on US1–US3), as were the usefulness of the tool for supporting real contract review (UF1) and the value of the contradiction explanations for legal validation (UF2). Adoption intent, however, diverged: asked whether they would use CONTRAVIS in their own workflow (UF3), P1 agreed, P2 was neutral, and P3 disagreed, preferring to stay with an AI legal-assistance tool already integrated into their firm’s workflow (App. F). This spread is consistent with the experts’ overall assessment: they found the workflow useful for contradiction inspection, evidence interpretation, and structure-aware exploration, while noting that refinement is needed before deployment in professional legal settings.

D. Qualitative Findings

Finding 1: Visual evidence comparison supported contradiction validation. Participants consistently identified contradiction-centered visual exploration as one of the most useful aspects of CONTRAVIS. The ability to compare conflicting spans directly in context helped participants inspect and validate contradiction candidates more effectively than isolated textual outputs.

Finding 2: Structure-aware exploration supported contextual inspection, but legal relationship types require refinement. Participants found value in exploring relationships between distant contractual provisions. However, one participant noted that the system occasionally classified rule-exception relationships as contradictions. This observation is consistent with the additional candidates surfaced in the controlled comparison (Sec. VI) and suggests a concrete source for them: future versions should better distinguish genuine contradictions from legal exceptions, refinements, and scope restrictions.

Finding 3: Grounded explanations and revision support were useful, but require deeper legal integration. Participants valued the explanatory and corrective capabilities of the system, especially for understanding why candidates were flagged and how contractual language could be revised. However, these features should remain tightly grounded in evidence and legal context to avoid over-reliance on LLM-generated reasoning.

E. Design Implications

The evaluation suggests several implications for improving CONTRAVIS. First, visual evidence comparison should remain central to the workflow. Second, the relationship model should better distinguish contradictions from legal exceptions and scope refinements. Third, explanation and rewriting features should be strengthened while preserving explicit evidence grounding and expert control over interpretation. Fourth, the divergent adoption responses suggest that usefulness for inspection does not by itself translate into workflow adoption; integration with existing review practices deserves design attention of its own.

VIII. DISCUSSION, LIMITATIONS, AND FUTURE WORK

Discussion. Our observations suggest that contradiction analysis in long contracts should not be treated as a purely long-context prompting problem. Although modern LLMs reason effectively over localized regions, their behavior may degrade when contradictions depend on relationships distributed across distant sections of large agreements. Rather than uniformly increasing context size, preserving structure-aware neighborhoods appears more effective for evidence-grounded inspection and human validation. Our expert study suggests that coordinated evidence views, rather than model accuracy alone, drive trustworthy review: in-context comparison was how experts accepted, contextualized, or discarded candidates. The

per-candidate confidence score (0–100) (Sec. V-B) is self-reported by the model and not calibrated, so analysts should read it as a triage cue rather than a probability; calibrating it against expert validation decisions, and encoding graded signals such as edge similarities and reranking scores (App. E), remain directions for improving trust calibration.

Limitations. Our formative study involved three experts from a single Brazilian law firm, so the results do not support generalizable conclusions about usability or effectiveness. Our evaluation also relies on synthetic contradictions—no public benchmark of naturally occurring document-level contractual contradictions exists, and ContractNLI [6] targets clause-pair entailment—which, despite graph-conditioned generation, automatic validation, and manual review (App. C), may not fully capture the ambiguity of real disputes. The comparison covers four contracts and one baseline, isolating the effect of graph conditioning without characterizing it across corpora or models. Graph-conditioned reasoning also remains sensitive to paragraph segmentation, embedding quality, similarity thresholds, and prompt design. Finally, as an early-stage design study, CONTRAVIS has not been deployed in professional practice; the deployment and longitudinal-reflection stages of design-study methodology [16] remain open.

Future Work. Future work includes enriching the typed paragraph graph with contract-level entities and obligations, expanding toward broader contract-quality tasks such as ambiguity and redundancy inspection, and improving robustness against adversarial content. We also plan sensitivity analyses of key graph parameters (*e.g.*, threshold τ), component ablations of referential edges, semantic edges, and reranking, experiments on larger collections, and longitudinal studies with legal professionals assessing trust calibration, usability, task-based analyst efficiency and cognitive workload against text-only baselines, and naturally occurring inconsistencies.

IX. CONCLUSION

We presented CONTRAVIS, a visual analytics system for structure-aware contradiction analysis in legal contracts. The system models contracts as typed paragraph graphs whose dual role—conditioning LLM reasoning while serving as the interactive representation the analyst explores—keeps model context and human inspection aligned. In a controlled comparison, graph-conditioned reasoning recovered more injected contradictions than standalone LLM analysis as contract length grew, and our formative study with contract-domain lawyers indicated that in-context evidence comparison, rather than model output alone, is what supported expert validation. Overall, CONTRAVIS frames contradiction analysis as a human-in-the-loop workflow that combines automated reasoning, evidence-grounded visual exploration, and expert legal judgment—a design we expect to transfer to other long, densely interlinked documents such as regulations and policy corpora.

ACKNOWLEDGMENT

This work was supported by CNPq (#313454/2026-4), and FGV. The opinions, hypotheses, conclusions, and recommen-

dations expressed in this material are the responsibility of the authors and do not necessarily reflect the views of CNPq and FGV.

REFERENCES

- [1] A. Singh, A. Joshi, J. Jiang, and H.-y. Paik, “A survey of classification tasks and approaches for legal contracts,” *Artificial Intelligence Review*, vol. 58, no. 12, p. 380, 2025.
- [2] D. Hendrycks, C. Burns, A. Chen, and S. Ball, “Cuad: An expert-annotated nlp dataset for legal contract review,” 2021.
- [3] S. Liu, Z. Li, R. Ma, H. Zhao, and M. Du, “ContractEval: Benchmarking LLMs for clause-level legal risk identification in commercial contracts,” in *NLLP Workshop*, 2025, pp. 291–291.
- [4] A. Nawar, M. Rakib, S. A. Hai, and S. Haq, “An open source contractual language understanding application using machine learning,” in *LT4FISS @ LREC*, 2022, pp. 42–50.
- [5] C. Han and S. Jalagam, “Metadata extraction leveraging large language models,” 2025.
- [6] Y. Koreeda and C. Manning, “ContractNLI: A dataset for document-level natural language inference for contracts,” in *Findings of EMNLP*, 2021.
- [7] J. Li, V. Raheja, and D. Kumar, “ContraDoc: Understanding self-contradictions in documents with large language models,” in *NAACL-HLT*, 2024, pp. 6509–6523.
- [8] Y. Chen, Y. Lyu, S. Liu, C. Zhang, J. Lv, and T. Xu, “Think wider, detect sharper: Reinforced reference coverage for document-level self-contradiction detection,” in *Proc. EMNLP*, 2025, pp. 1273–1288.
- [9] B. Lattimer, P. H. Chen, X. Zhang, and Y. Yang, “Fast and accurate factual inconsistency detection over long documents,” in *EMNLP*, 2023.
- [10] A. Y. Ichida and F. Meneguzzi, “Detecting logical relation in contract clauses,” 2021.
- [11] L. E. Resck, J. R. Ponciano, L. G. Nonato, and J. Poco, “Legalvis: Exploring and inferring precedent citations in legal documents,” *IEEE TVCG*, vol. 29, no. 6, pp. 3105–3120, 2023.
- [12] L. Resck, F. A. Moreno-Vera, T. Veiga, G. Paucar, E. Fajreldines, G. Klafke, L. G. Nonato, and J. Poco, “Legalanalytics: Bridging visual explanations and workload streamline in brazilian supreme court appeals,” *Artificial Intelligence and Law*, 2025.
- [13] F. Sperrle, R. Sevastjanova, R. Kehlbeck, and M. El-Assady, “Viana: Visual interactive annotation of argumentation,” in *VAST*, 2019.
- [14] N. R. Carvalho and L. S. Barbosa, “Transforming legal documents for visualization and analysis,” in *ICEGOV 2018*, 2018.
- [15] A. Mantravadi, S. Dalmia, O. Pospelova, A. Mukherji, N. Dave, and A. Mittal, “Legalwiz: A multi-agent generation framework for contradiction detection in legal documents,” 2025.
- [16] M. Sedlmair, M. Meyer, and T. Munzner, “Design study methodology: Reflections from the trenches and the stacks,” *IEEE TVCG*, vol. 18, no. 12, pp. 2431–2440, 2012.
- [17] A. Sancheti, A. Garimella, B. Srinivasan, and R. Rudinger, “What to read in a contract? party-specific summarization of legal obligations, entitlements, and prohibitions,” in *EMNLP*, 2023.
- [18] C. Braun, A. Lilienbeck, and D. Mentjukov, “The hidden structure – improving legal document understanding through explicit text formatting,” 2025.
- [19] R. Yu, T. Wang, R. Liu, and Y. Wang, “Sf-rag: Structure-fidelity retrieval-augmented generation for academic question answering,” 2026.
- [20] N. Pipitone and G. H. Alami, “Legalbench-rag: A benchmark for retrieval-augmented generation in the legal domain,” 2024.
- [21] G. Schumann and J. M. Gómex, “Detection of conflicts, contradictions and inconsistencies in regulatory documents: A literature review,” in *IDSTA*, 2024, pp. 81–88.
- [22] M. Glass, G. Rossiello, M. F. M. Chowdhury, A. Naik, P. Cai, and A. Gliozzo, “Re2G: Retrieve, rerank, generate,” in *NAACL-HLT*, 2022.
- [23] J. Dong, B. Fatemi, B. Perozzi, L. F. Yang, and A. Tsitsulin, “Don’t forget to connect! improving rag with graph-based reranking,” 2024.
- [24] M. Mesgar, L. F. R. Ribeiro, and I. Gurevych, “A neural graph-based local coherence model,” in *Findings of EMNLP*, 2021.
- [25] Z. Huang, R. Kar, H. Sundaram, and T. August, “Living contracts: Beyond document-centric interaction with legal agreements,” in *CHI*, 2026.
- [26] J. Heredia, L. Estrada-Rayme, J. Matos-Cangalaya, and J. Poco, “Interactive exploration and explanation of spatiotemporal anomalies with graph-llm integration,” in *SIBGRAPI*, 2025.

A. Contradiction Taxonomy

Following the contradiction taxonomy proposed in LegalWiz [15], the generated contradictions cover six categories: (i) **Temporal**, involving dates, deadlines, durations, or event ordering; (ii) **Numerical**, concerning monetary values, percentages, or quantities; (iii) **Authority**, involving the responsible entity or authoritative source; (iv) **Process**, concerning operational procedures or execution conditions; (v) **Policy Reversal**, representing direct negation of previously established rules or obligations; and (vi) **Specificity**, involving differences in scope or granularity.

B. Graph Construction Details

Preprocessing. Non-contractual and low-information paragraphs (section headers, repetitive footnotes) are removed using structural heuristics. Each remaining paragraph becomes a node, $V = \{v_1, \dots, v_n\}$.

Referential edges. A directed edge $(v_i, v_j) \in E_r$ is added when p_i explicitly references p_j . Extraction proceeds in three steps: (1) regular expressions capture legal identifiers over section, article, clause, and subclause numbering schemes (e.g., “Section 3.2”, “Article 5”, “Section 4.1(a)"); (2) each identifier is resolved to its target paragraph by matching against the numbering recovered during preprocessing; and (3) contextual filtering discards self-references and structural heading matches.

Semantic edges. Each paragraph is embedded as $e_i = \phi(p_i)$, where $\phi(\cdot)$ is the all-MiniLM-L6-v2 SentenceTransformer model. An edge $(v_i, v_j, s_{ij}) \in E_s$ is added when $s_{ij} = \cos(e_i, e_j) \geq \tau$, with $\tau = 0.80$.

The resulting hybrid graph combines explicit legal structure (E_r) with latent topical structure (E_s).

C. Synthetic Contradiction Generation and Validation

Starting from the preprocessing graph $\mathcal{G} = (V, E_r, E_s)$, we generate controlled contradictions through the four-stage pipeline summarized in Algorithm 1.

Stages. (1) *Seed selection* picks the m highest-degree nodes of the hybrid graph. (2) *Neighborhood reranking* scores each candidate neighbor with the cross-encoder ms-marco-MiniLM-L-6-v2 and keeps the top K reference and semantic neighbors, duplicates removed. (3) *Generation* prompts GPT-4.1 with the seed, its reranked neighborhood, and a target category to produce an *intra-* or *inter-paragraph* contradiction together with its insertion region (App. E). (4) *Validation* accepts a contradiction only if it is both semantically valid and fluent.

Validation criteria. Contradiction consistency is scored by the NLI model nli-deberta-v3-base as $\gamma(s, c) = P_{\text{NLI}}(\text{contradiction} \mid s, c)$, and linguistic plausibility by the per-word perplexity shift $\Delta\text{PPL}_{\text{norm}} = (\text{PPL}(p^*) - \text{PPL}(p))/|c|$ under GPT-2, where p and p^* are the paragraph before and after inserting the contradiction c , and $|c|$ is

Algorithm 1 Synthetic contradiction generation and validation.

Require: preprocessing graph $\mathcal{G} = (V, E_r, E_s)$

Ensure: synthetic contradicted contract $\mathcal{C}^*$

```

1:  $H \leftarrow \text{Top}_m\{d(v) : v \in V\}$  {seed selection,  $m = 5$ }
2: for all seed  $v \in H$  do
3:   for all neighbor  $n \in N(v)$  do
4:      $\text{score}[n] \leftarrow \text{CrossEncoder}(v, n)$  {pairwise reranking}
5:   end for
6:    $N_K(v) \leftarrow \text{TopK}(\text{score}, K)$  { $K = 5$ }
7:    $(s, c) \leftarrow \text{LLM}(v, N_K(v))$  {taxonomy-conditioned prompt}
8:    $\gamma \leftarrow P_{\text{NLI}}(\text{contradiction} \mid s, c)$ 
9:    $\Delta\text{PPL}_{\text{norm}} \leftarrow (\text{PPL}(p^*) - \text{PPL}(p))/|c|$  {per-word normalized}
10:  if  $\gamma \geq \alpha$  and  $\Delta\text{PPL}_{\text{norm}} < \delta$  then
11:    insert  $c$  into its target paragraph in  $\mathcal{C}^*$  { $\alpha = 0.70$ ,  $\delta = 0.01$ }
12:  end if
13: end for
14: expert manual review of accepted contradictions

```

the number of words in c . A contradiction is kept only if $\gamma \geq \alpha$ and $\Delta\text{PPL}_{\text{norm}} < \delta$; accepted samples are additionally reviewed manually.

D. Graph-Conditioned Detection Details

During detection, the input is the contradicted contract $\mathcal{C}^*$ with contradiction locations unknown; the typed paragraph graph $\mathcal{G}^*$ is reconstructed in real time (App. B) and serialized into a structured payload of paragraph nodes with their referential and semantic relations. This payload is passed to the LLM in a single inference step, enabling document-level reasoning over interconnected graph neighborhoods rather than isolated paragraph pairs.

The detector returns structured predictions following the output schema in App. E; the parsed fields drive the coordinated views: evidence spans are highlighted in the *Document View*, involved paragraphs are surfaced in the *Related Paragraph Explorer*, and category, evidence, and justification populate the *Contradiction Analysis* panel. Importantly, the detector is blind to insertion positions: only the experimenters know where contradictions were injected, so performance reflects true discovery rather than position-aware matching. Fig. 4 shows the CONTRAVIS prototype interface.

Computational cost. Typed graph construction remained lightweight (2.3–4.7 s for contracts of 208–517 paragraphs), while graph-conditioned detection with GPT-4.1 cost between \$0.21 and \$0.82 per contract. Larger graph neighborhoods nevertheless increase inference cost and latency for long contracts.

E. Prompt Templates and Output Schema

The full verbatim prompts for all LLM-driven steps are released with the source code; we summarize their structure and the detection output schema here.

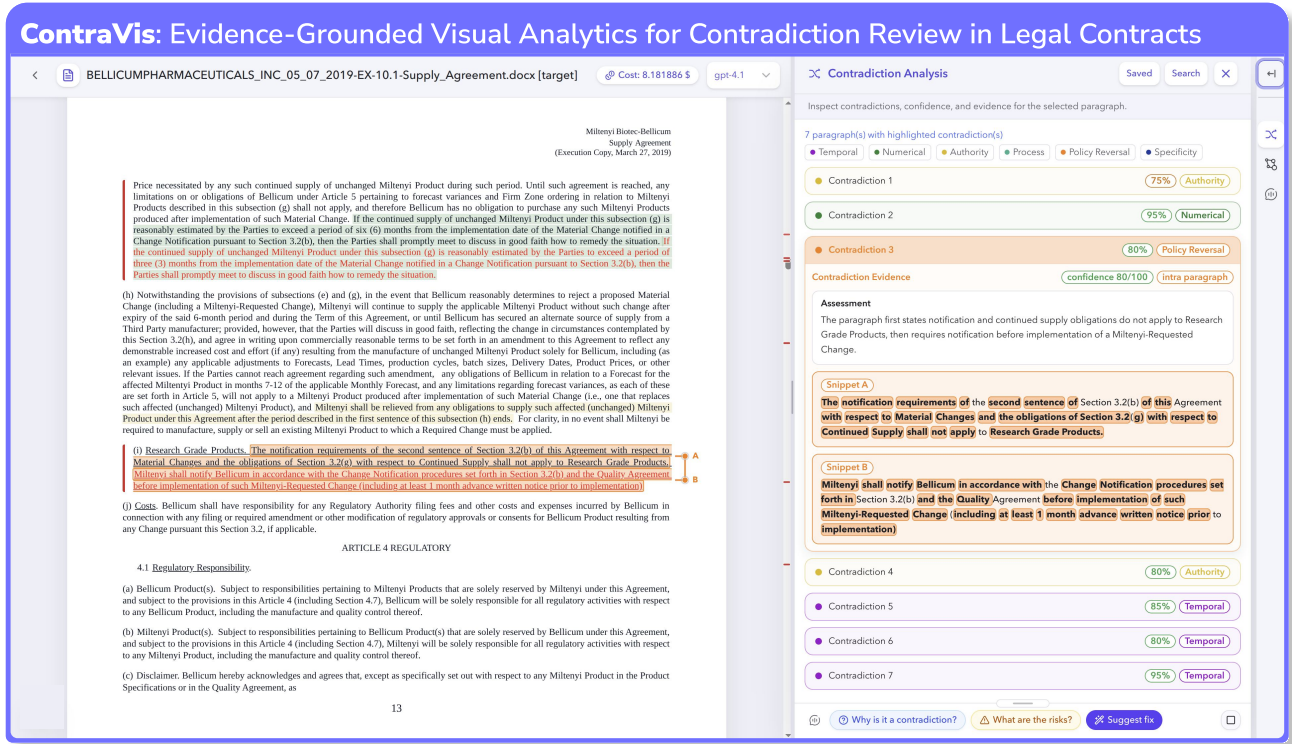


Figure 4. CONTRAVIS prototype interface on a CUAD contract: the *Document View* (left) highlights contradiction evidence, and the *Contradiction Analysis* panel (right) lists candidates with their category and confidence score, expanding one into its assessment and conflicting snippets (A, B).

Contradiction detection. A fixed system prompt configures the model as a legal contradiction classifier with a high-recall bias and requires evidence snippets to be exact verbatim substrings of the contract. The user prompt provides the serialized graph payload—the target paragraph and its typed reference/semantic neighbors—and instructs an intra- and an inter-paragraph pass. Responses follow the schema:

```
{
  "paragraph_results": [
    {
      "paragraph_id": "string or integer",
      "contradictions": [
        {
          "confidence": integer from 0 to 100,
          "contradiction_type": "temporal | numerical |
                                authority |
                                process | policy_reversal |
                                specificity",
          "brief_reason": "one short sentence",
          "evidence": {
            "snippet_a": "exact excerpt #1", "snippet_b": "
                                exact excerpt #2",
            "source_a": "paragraph | context | unknown",
            "source_b": "paragraph | context | unknown"
          }
        }
      ]
    }
  ]
}
```

Synthetic contradiction generation. The generation prompt embeds the taxonomy of App. A (a definition and example per category) and instructs the model to select a clause, produce a contradictory sentence for insertion into the target paragraph, and label its type

Table III
USABILITY AND USEFULNESS QUESTIONNAIRE RESULTS (FIVE-POINT LIKERT SCALE).

Question	P1	P2	P3
US1	SA	SA	A
US2	SA	A	A
US3	A	A	A
UF1	A	A	A
UF2	A	SA	A
UF3	A	N	D

SD: strongly disagree, D: disagree, N: neutral, A: agree, SA: strongly agree.

and scope, returning {statement, contradiction, type_contradiction, scope_contradiction}.

F. User Evaluation Questionnaire

After the exploration tasks, participants answered the following items on a five-point Likert scale. Table III reports the individual responses.

Usability Items “Was the interface easy to learn?” (US1); “Was the navigation between paragraphs, relationship structure, and contradiction analysis panel clear?” (US2); “Was the presented evidence easy to interpret?” (US3).

Usefulness Items “Is the tool useful for supporting real contract review?” (UF1); “Did the contradiction explanations help with my legal validation?” (UF2); “Would I use CONTRAVIS in my workflow?” (UF3).