# BALE: Graph-Based Contract Intelligence with Inter-Clause Conflict Detection and Dispute Hotspot Prediction

**Hamza Masmoudi**
Independent Researcher
https://github.com/hamza2masmoudi/BALE-project

**Abstract**

Automated contract analysis systems typically treat clauses as independent units, classifying each in isolation and aggregating risk scores through simple heuristics. This approach ignores a fundamental property of legal instruments: clauses interact. An indemnification provision may conflict with a liability cap; a data protection clause may be unenforceable without a corresponding confidentiality agreement. We present **BALE** (Binary Adjudication and Litigation Engine), a contract intelligence framework that models these inter-clause relationships explicitly through a *Contract Reasoning Graph*.

BALE operates as a four-stage pipeline: (1) zero-shot embedding classification using a multilingual sentence encoder, achieving **80.2%** accuracy across English and French without fine-tuning; (2) graph construction from a knowledge base of 16 legal doctrinal relationships encoding conflicts, dependencies, and constraints; (3) power asymmetry detection that quantifies party-level obligation imbalance; and (4) dispute hotspot prediction that localizes which specific clauses are most likely to be contested. Evaluated on a corpus of 15 commercial contracts spanning 7 contract types, our ablation study demonstrates that the graph reasoning layer contributes the largest marginal improvement (+10.7 points on structural risk), while the full pipeline runs in under 500ms. These results suggest that moving from clause-level classification to inter-clause reasoning represents a meaningful advance for contract analysis systems.

**Keywords:** Legal NLP, Contract Analysis, Knowledge Graphs, Dispute Prediction, Multilingual Classification, Risk Assessment

## 1 Introduction

Commercial contracts govern economic relationships estimated at trillions of dollars globally, yet their analysis remains largely manual [1]. The recent application of NLP to legal text has focused primarily on two tasks: clause extraction [2] and clause classification [3]. While these advances enable the identification of what type of clause appears in a contract, they do not address a more consequential question: *how do clauses interact with each other, and where do those interactions create risk?*

Consider an indemnification clause that requires Party A to "indemnify and hold harmless" Party B against all claims. If the same contract contains a limitation of liability clause capping damages at the contract value, these two provisions are in direct tension: the indemnification promises unlimited protection while the liability cap restricts it. This conflict creates enforcement risk that no clause-level classifier can detect, because the risk emerges from the *relationship* between clauses, not from either clause in isolation.

We identify three limitations of existing contract analysis systems:

1. **Clause Independence Assumption.** Current systems classify clauses independently, missing inter-clause conflicts, dependencies, and redundancies that determine enforceability.

2. **Structural Blindness.** Risk is computed as an aggregate of per-clause scores, ignoring structural properties such as missing clauses, unsatisfied dependencies, and incomplete coverage.

3. **Symmetry Blindness.** Standard classification provides no information about which party bears disproportionate risk, a critical factor in negotiation and dispute.

To address these gaps, we introduce BALE, a contract intelligence framework built on three principles. First, we model the contract as a *typed graph* where nodes are classified clauses and edges encode legal doctrinal relationships (conflicts, dependencies, constraints). Second, we detect *power asymmetry* by analyzing the distribution of obligations and protections across parties. Third, we synthesize these signals into *dispute hotspot predictions*: per-clause probabilities indicating where litigation is most likely to arise.

Our contributions are:

- A zero-shot embedding classifier that achieves 80.2% overall accuracy across English and French without fine-tuning, running in under 5ms per clause.
- A Contract Reasoning Graph formalism encoding 16 inter-clause relationships derived from legal doctrine, enabling automated detection of conflicts, missing dependencies, and structural gaps.
- A dispute hotspot predictor that localizes risk to specific clauses rather than producing contract-level aggregates.
- An empirical evaluation on 15 commercial contracts with an ablation study demonstrating the contribution of each component.

## 2 Related Work

### 2.1 Legal NLP and Contract Understanding

The application of NLP to legal documents has evolved from rule-based information extraction [9] to transformer-based approaches. The CUAD dataset [2] established a standard benchmark for contract clause extraction, defining 41 clause types across 510 contracts. Chalkidis et al. [3] introduced LegalBERT, a domain-adapted BERT model achieving strong performance on legal text classification. More recently, Chalkidis et al. [4] released LexGLUE, a multi-task benchmark for legal language understanding.

However, these approaches operate at the clause level: they identify and classify individual provisions without modeling how clauses relate to each other. ContractNLI [5] introduced natural language inference for contracts but remains limited to entailment between a hypothesis and a single clause, rather than reasoning across the full contract structure.

### 2.2 Graph-Based Document Analysis

The use of graph representations for document understanding has shown promise in scientific literature [10] and knowledge extraction [11]. In the legal domain, knowledge graphs have been applied primarily to case law citation networks [12] and statute cross-referencing. Our work differs by constructing graphs *within* a single document, where nodes are clauses and edges encode doctrinal relationships specific to contract law.

### 2.3 Neuro-Symbolic Legal Reasoning

Neuro-symbolic approaches combine neural perception with symbolic reasoning [7]. In legal AI, this has taken forms including logic-augmented neural training [8] and rule-based post-processing of neural outputs. Our architecture follows the latter paradigm: the neural component (embedding classifier) handles perception (clause classification), while the symbolic component (graph construction, power analysis) handles reasoning over the classified structure.

### 2.4 Multilingual Legal NLP

Cross-lingual transfer for legal text remains challenging due to the jurisdiction-specific nature of legal language [13]. Prior work has relied on multilingual fine-tuning or parallel corpora. Our approach leverages a pretrained multilingual sentence encoder (paraphrase-multilingual-MiniLM-L12-v2), achieving strong French performance (85.0%) without any French-specific fine-tuning, suggesting that legal clause semantics transfer well across languages at the embedding level.

## 3 Methodology

We define the contract analysis task as follows. Given a contract text $\mathcal{D}$, the system must produce: (a) a typed classification $y_i$ for each clause $c_i$; (b) a set of inter-clause relationships $E \subseteq V \times V \times \mathcal{T}$ where $\mathcal{T}$ is a set of edge types; (c) a power asymmetry score $P_{asym} \in [0, 100]$; and (d) a dispute probability $p_i \in [0, 1]$ for each clause $c_i$. We decompose this into four stages described below.

### 3.1 Stage 1: Zero-Shot Embedding Classification

Rather than fine-tuning a large language model for clause classification, we adopt a zero-shot approach based on semantic similarity in a shared embedding space. This design choice is motivated by two observations: (1) fine-tuned models suffer from language-specific bias when training data is imbalanced across languages, and (2) embedding-based classification enables sub-millisecond inference without GPU requirements.

We define a taxonomy $\mathcal{C} = \{c_1, \ldots, c_k\}$ of $k = 15$ clause types, each associated with a canonical description $d_i$ written in natural language (e.g., $d_{\text{indemnification}}$ = “A clause where one party agrees to compensate the other for losses, damages, or liabilities arising from specific events or breaches”). For an input clause $x$, classification is performed as:

$$\hat{y} = \arg\max_{c_i \in \mathcal{C}} \cos\left(\texttt{enc}(x), \texttt{enc}(d_i)\right) \tag{1}$$

where $\texttt{enc} : \mathcal{X} \to \mathbb{R}^{384}$ is the `paraphrase-multilingual-MiniLM-L12-v2` sentence encoder, and $\cos(\cdot, \cdot)$ denotes cosine similarity. The confidence score is defined as the maximum cosine similarity:

$$\text{conf}(\hat{y}) = \max_{c_i} \cos\big(\texttt{enc}(x), \texttt{enc}(d_i)\big) \quad (2)$$

Taxonomy descriptions are embedded once at initialization and cached, making per-clause inference a single forward pass plus 15 cosine computations.

### 3.2 Stage 2: Contract Reasoning Graph

The core innovation of this work is the Contract Reasoning Graph (CRG), a typed directed graph $G = (V, E)$ where:

- $V$ is the set of classified clauses, each with type $\hat{y}_i$ and confidence $\text{conf}_i$.
- $E \subseteq V \times V \times \mathcal{T}$ where $\mathcal{T} = \{\text{CONFLICTS}, \text{DEPENDS}, \text{LIMITS}, \text{REDUNDANT}\}$.

We construct $E$ from a *legal knowledge base* $\mathcal{K}$ encoding 16 pairwise relationships between clause types. These relationships are derived from established contract law doctrine and include:

- **Conflicts** (6 pairs): Clauses creating contradictory obligations. E.g., (indemnification, limitation_of_liability, CONFLICTS), since indemnification promises unlimited coverage while liability caps restrict it.
- **Dependencies** (7 pairs): Clauses where enforceability of one requires the presence of another. E.g., (data_protection, confidentiality, DEPENDS), since data protection provisions are weakened without corresponding confidentiality obligations.
- **Limits** (3 pairs): Clauses where one constrains the scope of another. E.g., (limitation_of_liability, warranty, LIMITS).

**Structural Risk Scoring.** We compute three graph-derived metrics:

$$\text{Risk}_{\text{structural}} = w_c \cdot |E_{\text{CONFLICTS}}| + w_d \cdot |D_{\text{missing}}| + w_m \cdot |M_{\text{expected}}| \quad (3)$$

where $|E_{\text{CONFLICTS}}|$ is the number of conflict edges, $|D_{\text{missing}}|$ is the number of unsatisfied dependency edges, and $|M_{\text{expected}}|$ is the count of expected clause types absent from the contract (determined by contract-type templates). Weights $w_c = 25$, $w_d = 15$, $w_m = 10$ are set based on legal severity.

**Completeness Score.** For a contract of type $t$, we define an expected clause set $\mathcal{E}_t$ (e.g., an MSA is expected to contain indemnification, limitation of liability, confidentiality, termination, etc.). Completeness is:

$$\text{Completeness}(G, t) = \frac{|\{\hat{y}_i\} \cap \mathcal{E}_t|}{|\mathcal{E}_t|} \quad (4)$$

### 3.3 Stage 3: Power Asymmetry Detection

We quantify the imbalance of obligations between contracting parties. For each party $A$ and $B$ identified in the contract text, we count:

- $O_A$, $O_B$: Obligation counts (instances of “shall”, “must”, “agrees to” attributed to each party).
- $P_A$, $P_B$: Protection counts (instances of “entitled to”, “right to”, “may” attributed to each party).
- $U$: One-sided language count (“sole discretion”, “solely responsible”, “at its own expense”).

The asymmetry score is:

$$P_{\text{asym}} = \frac{|B_A - B_B|}{B_A + B_B + 1} \cdot 100 + \lambda \cdot U \quad (5)$$

where $B_A = O_A - P_A$ is the net burden on party $A$, and $\lambda = 5$ is the one-sided language penalty coefficient. Higher scores indicate greater imbalance and therefore higher risk of dispute.

### 3.4 Stage 4: Dispute Hotspot Prediction

The final stage synthesizes signals from the graph and power analysis to produce per-clause dispute probabilities. For each clause $c_i$, we compute:

$$P(\text{dispute} \mid c_i) = \alpha \cdot f_{\text{conflict}}(c_i) + \beta \cdot f_{\text{gap}}(c_i) + \gamma \cdot f_{\text{power}}(c_i) \quad (6)$$

where $f_{\text{conflict}}(c_i) = 1$ if clause $c_i$ participates in any conflict edge, $f_{\text{gap}}(c_i) = 1$ if a dependency of $c_i$ is unsatisfied, and $f_{\text{power}}(c_i)$ is the normalized power contribution of clause type $\hat{y}_i$. Coefficients are $\alpha = 0.4$, $\beta = 0.3$, $\gamma = 0.3$.

Clauses are ranked by dispute probability, and those exceeding a threshold $\theta = 0.3$ are flagged as hotspots with severity labels (CRITICAL, HIGH, MEDIUM).

## 4 Experimental Setup

### 4.1 Training Corpus

The embedding classifier requires no training data; only the taxonomy descriptions are needed. However, our broader system was developed using a corpus of 75,382 legal text segments from 10 sources (Table 1), which informed taxonomy design and threshold calibration.

Table 1: Corpus composition used for taxonomy development and threshold calibration. The V10 classifier does not require training on this data.

| Source | Segments | Lang. |
|---|---|---|
| CUAD [2] | 10,667 | EN |
| Legal Argument Mining | 23,113 | EN/DE |
| Mistral Legal French | 14,875 | FR |
| Claudette ToS | 9,319 | EN |
| EURLex-4K | 5,000 | EN |
| Other sources (5) | 12,408 | Mixed |
| **Total** | **75,382** | **Multilingual** |

### 4.2 Evaluation Dataset

We evaluate on 15 commercial contracts spanning 7 types: Master Service Agreements (5), Non-Disclosure Agreements (2), Service Level Agreements (1), Employment Agreements (1), Health Compliance (1), Data Processing (1), Financial (1), Technology (1), and Edge Cases (2 – deliberately imbalanced or missing clauses). Contracts range from 800 to 4,500 words.

For classification accuracy, we use a manually curated Golden Test Set of 91 clauses (71 English, 20 French), each annotated with clause type, risk level, and a justification rationale. Inter-annotator agreement on clause type is $\kappa = 0.82$ (substantial).

### 4.3 Baselines

We compare the V10 embedding classifier against:

- **V8**: A fine-tuned Mistral-7B model with LoRA adaptation [6], trained on the 75K corpus. This represents the standard approach of supervised fine-tuning.
- **Zero-shot LLM**: Mistral-7B-Instruct with a prompt-based classification approach (no fine-tuning or embeddings).

For the graph and power components, no direct baselines exist in the literature, as inter-clause conflict detection and power asymmetry scoring for contracts are, to our knowledge, novel contributions. We therefore evaluate these components via ablation.

## 5 Results

### 5.1 Classification Accuracy (RQ1)

Table 2 presents clause classification accuracy across system versions and languages.

Table 2: Clause classification accuracy on the Golden Test Set (91 clauses). Latency measured on Apple M-series hardware.

| System | Overall | EN | FR | Latency |
|---|---|---|---|---|
| V8 (Fine-tuned) | 50.8% | 66.7% | 10.0% | 1,241ms |
| V10 (Embedding) | **80.2%** | **76.5%** | **85.0%** | **<5ms** |
| *Improvement: +29.4pp overall, +75.0pp French, 250× faster* | | | | |

Two observations are notable. First, the V10 system achieves *higher* French accuracy (85.0%) than English (76.5%), despite no French-specific training. This suggests that the multilingual encoder captures legal semantics that transfer effectively across languages. Second, the V8 system's poor French performance (10.0%) is explained by the English-dominant training distribution of the fine-tuned model.

**Error Analysis.** The most frequent V10 misclassifications involve semantically adjacent categories: `limitation_of_liability` confused with `indemnification` (both involve liability allocation), and `data_protection` confused with `intellectual_property` (both involve information control). These confusions reflect genuine semantic proximity in the embedding space rather than systematic failure.

### 5.2 Component Ablation (RQ2)

To quantify the contribution of each V10 component, we conducted an ablation study on all 15 evaluation contracts (Table 3). We define four configurations:

- **C**: Classifier only (average risk weight of detected clause types).
- **C+G**: Classifier + Graph (structural risk from conflicts, gaps, completeness).
- **C+G+P**: Classifier + Graph + Power (adds asymmetry scoring).
- **Full**: All components including Dispute Prediction.

The Graph layer (C+G) provides the largest marginal contribution, boosting risk scores by an average of +10.7 points. This improvement is driven by structural detection: contracts with missing clauses (e.g., Missing Clauses MSA: +24.7 from Graph) or internal conflicts (e.g., Vendor Heavy: +21.4 from Graph) receive substantially higher risk scores.

The Power and Dispute layers serve a different function: they *modulate* the Graph signal, reducing risk for balanced contracts (Standard NDA: power score of 0, indicating no asymmetry) and increasing it for heavily one-sided agreements. The Full pipeline's mean is lower than C+G because the Dispute prediction layer normalizes scores by dampening false positives from the Graph layer alone.

Table 3: Ablation study: risk scores across pipeline configurations. Higher scores indicate greater detected risk. C = Classifier, G = Graph, P = Power.

| Contract | Exp. | C | C+G | C+G+P | Full |
|---|---|---|---|---|---|
| Vendor Heavy | H | 64.3 | 85.7 | 67.0 | 69.2 |
| AI Services | M | 63.3 | 85.3 | 63.5 | 74.3 |
| Outdated 2018 | H | 59.2 | 83.7 | 64.0 | 66.4 |
| Missing Clauses | H | 58.9 | 83.6 | 62.9 | 70.3 |
| Cloud SLA | M | 63.0 | 85.2 | 62.6 | 68.1 |
| EU DPA | M | 65.0 | 86.0 | 61.1 | 63.3 |
| Balanced MSA | L | 56.0 | 67.4 | 57.7 | 59.8 |
| Standard NDA | L | 57.7 | 83.1 | 57.3 | 58.7 |
| *Mean (15)* | | 60.6 | 71.3 | 55.3 | 55.6 |

### 5.3 Novel Capabilities (RQ3)

We demonstrate three capabilities absent from existing contract analysis systems:

**Inter-clause conflict detection.** On the Vendor Heavy MSA, the system identified 2 doctrinal conflicts: (1) indemnification vs. limitation of liability (the indemnification clause promises unlimited protection while liability is explicitly capped), and (2) intellectual property vs. confidentiality (IP ownership claims conflict with information sharing restrictions). These conflicts were verified as legally meaningful by manual review.

**Missing dependency detection.** On the AI Services MSA, the system flagged 3 missing dependencies: indemnification without a corresponding insurance requirement, data protection without a confidentiality clause, and IP provisions without confidentiality protections. Each represents a genuine enforceability risk.

**Dispute hotspot localization.** Across the 15-contract corpus, warranty clauses were consistently identified as the highest-probability dispute hotspots (mean $p = 0.62$) when combined with one-sided language, aligning with empirical litigation data showing warranties as the most frequently litigated contract provisions [14].

### 5.4 Latency

The full V10 pipeline (classification + graph + power + disputes) runs in 471 $\pm$ 128ms across the 15-contract corpus on Apple M-series hardware without GPU. Classification alone runs in under 5ms per clause. This performance profile enables real-time analysis during contract drafting and negotiation.

## 6 Discussion

### 6.1 Why Graph Reasoning Matters

The ablation study reveals that clause-level classification alone provides limited differentiation between contracts of different risk profiles (range: 56.0–65.0). The Graph layer expands this range substantially (24.4–86.0), because structural properties — conflicts, missing dependencies, incomplete coverage — are strong discriminators of contract quality. This supports our central thesis: moving from clause-level to inter-clause reasoning provides meaningfully more informative risk assessment.

### 6.2 Limitations

We acknowledge several limitations of the current system:

1. **Knowledge base coverage.** The 16 inter-clause relationships in $\mathcal{K}$ are manually defined and may not cover all jurisdictions or contract types. Expanding $\mathcal{K}$ requires legal domain expertise.

2. **Classification ceiling.** At 80.2% accuracy, the embedding classifier introduces errors that propagate through the pipeline. Misclassifying a clause type can cause false positive or false negative conflict detection.

3. **Party extraction.** The current party identification relies on heuristic pattern matching (“Provider”, “Customer”, etc.) and may fail on contracts with unusual party designations.

4. **Evaluation scale.** Our evaluation corpus of 15 contracts, while diverse in type, is small by NLP standards. Larger-scale evaluation on hundreds of contracts would strengthen the findings.

5. **Ground truth for disputes.** We lack ground truth data on actual litigation outcomes for our test contracts, making it difficult to validate dispute predictions empirically.

### 6.3 Future Work

Several extensions are natural. First, graph neural networks (GNNs) could learn edge types from data rather than relying on a manually defined knowledge base. Second, the power asymmetry analysis could be enhanced with coreference resolution to more accurately attribute obligations to parties. Third, expanding the evaluation to include actual litigation outcome data would enable direct validation of dispute predictions.

## 7 Conclusion

We have presented BALE, a contract intelligence framework that moves beyond clause-level classification to model inter-clause relationships through a Contract Reasoning Graph. Our system achieves 80.2%

classification accuracy across English and French without fine-tuning, detects inter-clause conflicts and missing dependencies using a knowledge base of 16 legal doctrinal relationships, and predicts dispute hotspots with per-clause probabilities. The ablation study demonstrates that graph-based reasoning provides the largest marginal contribution to risk detection, validating the central hypothesis that contracts should be analyzed as structured documents with interacting provisions rather than as bags of independent clauses.

## References

[1] R. Susskind, *Tomorrow's Lawyers: An Introduction to Your Future*, 2nd ed. Oxford University Press, 2017.

[2] D. Hendrycks, C. Burns, A. Chen, and S. Ball, "CUAD: An Expert-Annotated NLP Dataset for Legal Contract Review," in *Proc. NeurIPS Datasets and Benchmarks Track*, 2021.

[3] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos, "LEGAL-BERT: The Muppets Straight Out of Law School," in *Proc. EMNLP Findings*, 2020.

[4] I. Chalkidis, A. Jana, D. Hartung, M. Bommarito, I. Androutsopoulos, D. Katz, and N. Aletras, "LexGLUE: A Benchmark Dataset for Legal Language Understanding in English," in *Proc. ACL*, 2022.

[5] Y. Koreeda and C. Manning, "ContractNLI: A Dataset for Document-Level Natural Language Inference for Contracts," in *Proc. EMNLP Findings*, 2021.

[6] E. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-Rank Adaptation of Large Language Models," in *Proc. ICLR*, 2022.

[7] A. d'Avila Garcez and L. Lamb, "Neurosymbolic AI: The 3rd Wave," *Artificial Intelligence Review*, vol. 56, pp. 12387–12406, 2023.

[8] K. Ashley, *Artificial Intelligence and Legal Analytics: New Tools for Law Practice in the Digital Age*. Cambridge University Press, 2017.

[9] A. Wyner, R. Mochales-Palau, M. Moens, and D. Milward, "Approaches to Text Mining Arguments from Legal Cases," in *Semantic Processing of Legal Texts*, Springer, 2010, pp. 60–79.

[10] Y. Luan, L. He, M. Ostendorf, and H. Hajishirzi, "Multi-Task Identification of Entities, Relations, and Coreference for Scientific Knowledge Graph Construction," in *Proc. EMNLP*, 2018.

[11] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. Yu, "A Survey on Knowledge Graphs: Representation, Acquisition, and Applications," *IEEE Trans. Neural Networks and Learning Systems*, vol. 33, no. 2, pp. 494–514, 2022.

[12] H. Zhong, C. Xiao, C. Tu, T. Zhang, Z. Liu, and M. Sun, "How Does NLP Benefit Legal System: A Summary of Legal Artificial Intelligence," in *Proc. ACL*, 2020.

[13] J. Niklaus, I. Chalkidis, and M. Stürmer, "Swiss-Judgment-Prediction: A Multilingual Legal Judgment Prediction Benchmark," in *Proc. Natural Legal Language Processing Workshop*, 2021.

[14] A. Schwartz and R. Scott, "Contract Theory and the Limits of Contract Law," *Yale Law Journal*, vol. 113, no. 3, pp. 541–619, 2003.