

Machine Learning Approaches for Automated Contract Risk Assessment

Assistant Professor Smita
Mangesh Junnarkar¹

Neeraj S. Dontual²

¹Department of Data Science, SIES College of Arts, Science and Commerce
(Empowered Autonomous), Sion (W), Mumbai

²Department of Data Science, SIES College of Arts, Science and Commerce
(Empowered Autonomous), Sion (W), Mumbai

Roll No.: TDS2627011 neerajdontual130@gmail.com

Research Paper

2026

Abstract

Commercial agreements distribute rights, duties, costs, and uncertainty across organizations. A reviewer must locate relevant provisions, understand them in context, compare them with policy and precedent, and decide which deviations deserve attention. Machine learning (ML) can help with the document-intensive parts of this work, but a clause label or a language-model answer is not itself a defensible assessment of legal or business risk. This paper presents a structured research review and a proposed human-supervised framework for automated contract risk assessment. It synthesizes work on legal-domain language models, expert-annotated contract datasets, document-level inference, clause classification, information extraction, explainability, and evaluation. It distinguishes risk detection from legal advice and from prediction of whether a party will litigate or lose a dispute.

The proposed framework combines document ingestion and segmentation; clause and entity extraction; policy-conditioned risk identification; evidence-linked severity and confidence estimates; and human review with feedback. It treats a risk as a contextual judgment about a clause in a particular transaction rather than a universal property of a phrase. The research design is a reproducible comparative evaluation using public benchmarks such as CUAD, ContractNLI, and LEDGAR, followed by a carefully governed, locally annotated evaluation set if permission and resources are available. Baselines include rules, conventional classifiers, general-purpose transformers, and legal-domain transformers. Evaluation emphasizes class-wise precision and recall, evidence-span quality, calibration, ranking utility, robustness, and review time rather than headline accuracy alone.

The central conclusion is that current ML is most defensible as an assistive triage system: it can surface likely relevant text, identify deviations from a stated policy, and organize reviewer attention. Reliable deployment requires traceable evidence, explicit policy scope, representative annotation, privacy protections, drift monitoring, and a meaningful human override. The paper does not report new experiments or claim that a model has been validated for legal practice; it specifies a research agenda and evaluation protocol that can support such validation.

Keywords: contract analytics; legal natural language processing; machine learning; risk assessment; clause extraction; explainable AI; human-in-the-loop review

Contents

1	Introduction	1
2	Conceptual foundations	3
3	Review of related literature	5
4	Proposed framework	8
5	Research methodology and evaluation plan	11
6	Expected results and discussion	15
7	Ethical, legal, and operational considerations	16
8	Limitations	18
9	Conclusion and future work	19
A	Illustrative risk-review rubric	20
B	Illustrative structured output	21
C	Reproducibility checklist	22
D	Example error-analysis categories	23

1 Introduction

1.1 Background and motivation

Contracts are operational instruments as well as legal documents. They define payment, delivery, confidentiality, intellectual property, remedies, renewal, termination, and responsibility for failure. The practical cost of a problematic term depends on the business relationship, the parties’ bargaining positions, the organization’s fallback language, applicable law, and the likelihood and impact of an adverse event. A broad indemnity clause may be acceptable in one procurement arrangement and unacceptable in another. A limitation of liability may appear balanced until its cap excludes the largest plausible loss. Accordingly, automated assessment must connect text to context and policy.

Review remains difficult at scale. Agreements are long, often assembled from templates and negotiated amendments, and may use inconsistent names, cross-references, and defined terms. Reviewers need to discover both present clauses and meaningful omissions. Relevant language may be distributed across a main agreement, schedules, order forms, and amendments. A word-level search can miss paraphrases; an isolated clause classifier can ignore an exception elsewhere; and a summary can omit the precise evidence necessary to verify a concern. Contract data also contains commercially sensitive and personal information, making indiscriminate transfer to external services inappropriate.

Natural language processing (NLP) and ML offer several useful capabilities: identifying clause boundaries, classifying provisions, extracting parties and dates, retrieving similar language, finding evidence for a hypothesis, and generating a concise explanation. Legal-domain pretraining has been studied because ordinary-language models may not represent specialized legal vocabulary and drafting conventions well [2]. Expert-labeled resources such as CUAD make clause review measurable, while ContractNLI frames questions over entire contracts and requires supporting evidence [4, 5]. These resources advance research, but their labels do not fully encode an organization’s business appetite or jurisdiction-specific legal judgment.

This paper therefore uses *automated contract risk assessment* to mean a decision-support process that prioritizes provisions for human examination against an explicit policy or rubric. It does not mean automatically determining that a contract is legally invalid, predicting a court’s decision, or replacing counsel. The distinction is central: a system can reliably find a clause and still be wrong about its importance; it can correctly detect a deviation while incorrectly calling that deviation unacceptable.

1.2 Problem statement

The research problem is to design and evaluate ML methods that identify and rank potentially material contract risks while preserving enough evidence and context for a qualified reviewer to verify the output. The system should handle explicit risks, policy deviations, missing protections, and interactions among provisions. It should communicate uncertainty, avoid unsupported legal conclusions.

1.3 Aim and objectives

The aim is to develop a research-grounded framework for human-supervised automated contract risk assessment. The objectives are:

1. review relevant ML, NLP, legal-domain modeling, datasets, and evaluation practices;
2. define a risk representation that separates clause type, policy deviation, severity, confidence, and evidence;
3. propose a modular pipeline for processing long contracts and their related documents;
4. specify a comparative evaluation design with realistic splits, metrics, and error analysis; and
5. examine privacy, fairness, accountability, security, and limits on appropriate use.

1.4 Research questions

1. Which ML approaches are suitable for finding and classifying potentially risky contract provisions?
2. How can a system produce useful, verifiable risk assessments rather than unsupported labels?
3. Which evaluation measures best reflect the needs and costs of contract review?
4. What technical and governance safeguards are necessary for responsible use?

1.5 Scope and contribution

The focus is English-language commercial contracts and supervised or retrieval-assisted NLP. The discussion covers clause identification, document-level inference, extraction, ranking, and human review. It does not provide jurisdiction-specific legal advice, establish a universal definition of risk, or claim empirical superiority for any model. The contribution is a synthesis and a testable framework: a policy-conditioned, evidence-grounded workflow plus an evaluation plan that makes its assumptions visible. Where the paper discusses model behavior, it distinguishes published findings from proposed hypotheses.

2 Conceptual foundations

2.1 What counts as contract risk?

Risk is not simply negative sentiment. Contract language is intentionally conditional, and terms that sound severe may be routine when balanced by other provisions. For this research, a *risk candidate* is a text span or omission that may create a material exposure under a specified review policy. The policy may concern, for example, uncapped liability, broad indemnification, unilateral termination rights, weak confidentiality protections, automatic renewal, assignment, data processing, or ownership of deliverables. The applicable set depends on transaction type and organizational policy.

An assessment should separate at least five dimensions. First is the provision category: what function the text serves. Second is the observed condition: for example, a cap is absent, a cap is unusually high, or an exception is broad. Third is policy deviation: whether the condition conflicts with a defined fallback or control. Fourth is potential severity and likelihood, which may require structured business judgment. Fifth is evidence and uncertainty: where the relevant text appears, what surrounding language changes its meaning, and how confident the system is. Merging these dimensions into one opaque score hides the reasons for a recommendation. Figure 1 summarizes the five dimensions.

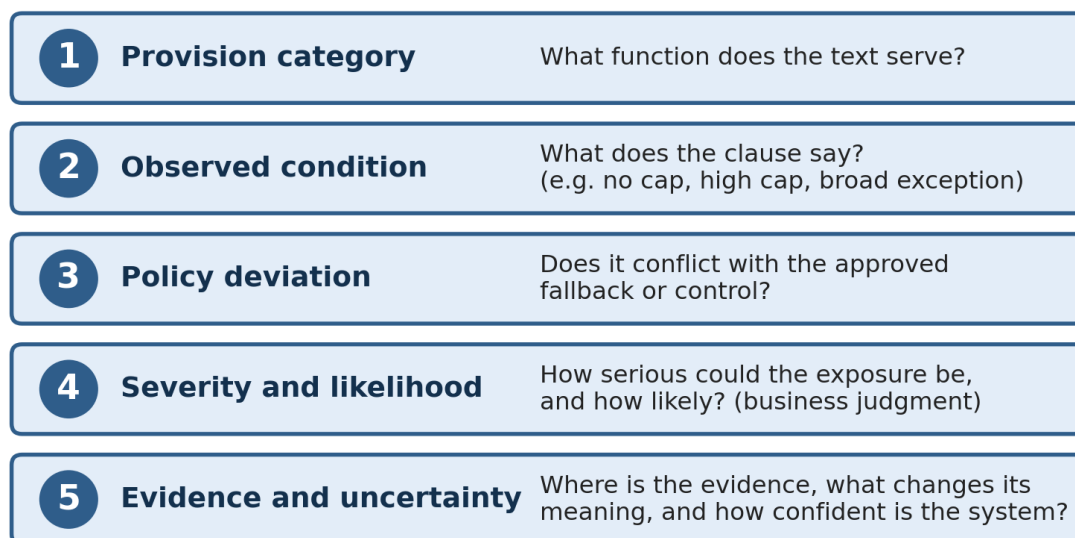


Figure 1: The five dimensions of a contract risk assessment, kept separate rather than merged into one score

Risk is relative to a reference. A system might compare a clause to approved language, a playbook, a transaction profile, or a reviewed corpus. Each reference has limitations. A template can encode outdated practice; historical contracts can reproduce prior bias; and a policy can be incomplete. Thus the system should identify the reference version and avoid treating frequency as proof of acceptability.

2.2 Tasks within an assessment system

The overall workflow comprises multiple NLP tasks:

- **Document parsing:** recover text, headings, tables, page locations, and document relationships from DOCX, PDF, and scanned files.
- **Segmentation:** identify sections, clauses, subclauses, and cross-references while retaining hierarchy.
- **Classification:** map provisions to one or more categories, including a label for uncertainty or other.
- **Extraction:** identify parties, dates, monetary limits, durations, governing-law terms, and other structured values.
- **Document-level inference:** test a policy hypothesis against the entire agreement, with evidence spans and treatment of entailment, contradiction, and omission.
- **Comparison and retrieval:** find relevant approved clauses, precedents, and deviations.
- **Triage:** rank candidates for review using severity policy, confidence, and expected review utility.

No one model necessarily performs all tasks well. A pipeline makes interfaces and errors inspectable, although errors can accumulate between stages.

2.3 Why context and evidence matter

Contract provisions refer to definitions, exceptions, schedules, and other sections. Negation, conditional language, carve-outs, and amendment precedence can reverse an apparent meaning. Document-level ContractNLI specifically illustrates the challenge of determining whether a hypothesis is entailed, contradicted, or unmentioned and locating the evidence [5]. Systems that classify short excerpts without recovering references risk false alerts and false reassurance. Evidence must therefore be presented as a navigable source location, with enough adjacent context to allow independent review.

3 Review of related literature

3.1 Machine learning and legal text

Earlier NLP systems relied on rules, dictionaries, and conventional classifiers. These methods can be attractive where a policy is explicit: a rule can flag a missing defined cap or a phrase in a list of prohibited terms. Their weakness is brittleness to word-ing, structure, and exceptions. Feature-based methods such as support vector machines and logistic regression can improve generalization over exact matching, but depend on representation and labeled data.

Transformer models, beginning with architectures such as BERT, learn contextual representations and can be fine-tuned for classification, question answering, and sequence labeling [3]. Legal-BERT examined strategies for adapting BERT to legal text, including continued domain pretraining and training with legal corpora, and emphasized that adaptation choices should be evaluated rather than assumed [2]. This supports testing legal-specific and general-purpose models as competing baselines. Domain pretraining can improve familiarity with legal terminology, but it does not provide legal authority, current law, or organization-specific risk preferences. Legal-language models should likewise be assessed across clearly defined tasks rather than treated as a single general measure of legal competence.

Long contracts pose an input-length problem. Splitting documents into chunks can make them fit within model limits, but naive chunking may separate a clause from a definition, exception, or referenced schedule. Hierarchical approaches can first encode sections or chunks and then aggregate representations at document level. Retrieval can select relevant passages for a query or policy. A robust design records chunk boundaries and source offsets, and it checks whether retrieved evidence omits a limiting provision elsewhere.

3.2 Expert-annotated contract datasets

CUAD (Contract Understanding Atticus Dataset) contains expert-annotated commercial contracts and 41 clause types designed to support contract-review research [4]. It provides a demanding span-identification setting: some answers are absent; spans vary in length; and some categories require locating several relevant phrases. CUAD enables evaluation of extraction and classification, but a clause type is not equivalent to an organization-specific risk rating. Its coverage reflects annotation design and source data, so performance on CUAD cannot by itself establish performance on other industries or jurisdictions. The dataset’s project documentation should also be consulted for the current files and permitted uses.

LEDGAR provides a large-scale collection of labeled legal provisions derived from contracts filed with the U.S. Securities and Exchange Commission [7]. Its scale and naturally occurring labels make it useful for provision classification. At the same time, labels derived from headings can be noisy, and filing contracts may differ from private agreements. Data leakage is a particular concern when related provisions, contract versions,

or repeated boilerplate appear in both training and test sets. Splitting at the document or organization level is more credible than random sentence splitting.

ContractNLI adds document-level inference, with labels for entailment, contradiction, and neutral, together with evidence annotation [5]. It captures the need to answer policy-like questions over lengthy agreements rather than classify a pre-isolated sentence. Its authors report that existing models struggle, highlighting exception handling and long-document context as continuing challenges. The dataset is suitable for evaluating inference and evidence localization, though its hypotheses and annotation scheme do not cover every form of business risk.

Other resources contribute complementary tasks. The MAUD dataset focuses on merger and acquisition agreement understanding, with expert annotations for provisions relevant to M&A review [6]. Public corpora can provide breadth and reproducibility, while private organizational corpora may better reflect a live workflow. Their use demands stronger governance and clear permission. Combining sources should not obscure their different populations, label definitions, and legal settings.

3.3 Clause classification and extraction

Clause classification assigns one or more functional labels to spans or provisions. Single-label classification assumes a clause has one purpose; multi-label classification allows a provision to concern several topics. Class imbalance is common: a dataset may contain many confidentiality examples and relatively few unusual liability clauses. Overall accuracy can then mask poor recall on rare but important classes. Macro-averaged measures, per-class results, and risk-weighted review are therefore necessary.

Extraction asks the model to return the precise span or value. Exact match is stringent and interpretable, while token-level F1 rewards partial overlap. Neither tells whether the extracted text is sufficient to support the reviewer. A complete evaluation should examine boundary errors, evidence completeness, and whether the highlighted span includes relevant exceptions. For structured values such as liability caps, extraction should normalize currency, units, time periods, and relation to fees without discarding the quoted source text.

3.4 Legal language models and generative methods

Legal-domain pretraining can be adapted to classification and extraction. Legal-BERT is an influential family of legal-domain models studied across legal NLP tasks [2]. Models such as Longformer address longer input sequences through sparse attention, offering one architectural path for long documents [1]. Nonetheless, a longer context window does not guarantee accurate reasoning across every clause and reference.

Generative language models can summarize, answer questions, and draft explanations. General-purpose foundation models offer broad capabilities, but they also carry risks when adapted to particular tasks. Their fluent output can be useful for navigation but

can also obscure unsupported inferences. Plausible-sounding model answers can still be false, so high-stakes review should constrain generated statements to retrieved text, point to exact evidence, and check whether the evidence supports each claim. Structured outputs and citation validation can help, but do not eliminate hallucination or missed context. Generative approaches should be compared against simpler extractive baselines and judged on factual support, not readability alone.

3.5 Human-centered legal technology

Legal technology research recognizes that task performance is not the only success criterion. A system can reduce search time while increasing verification burden or automation bias. Human reviewers need to understand why an item was flagged, inspect the source, reject or revise an assessment, and see which policy version was applied. Feedback can improve a model only if it is reviewed for label quality and used with a documented update process. Professional responsibility and human oversight remain relevant even when tools are technically capable.

3.6 Synthesis and research gap

The literature offers useful components: datasets for span extraction and inference, domain-adapted encoders, long-document techniques, and evaluation protocols for particular NLP tasks. A gap remains between benchmark performance and a complete risk workflow. Benchmarks tend to specify labels, whereas operational risk depends on a policy, transaction context, omissions, interactions, reviewer costs, and evidence quality. This paper addresses that gap by proposing an assessment architecture and an evaluation design that measure these elements separately.

Table 1: Representative public resources and their appropriate role

Resource	Main task	Strength	Important limitation
CUAD	Clause span extraction across 41 categories	Expert annotation and contract-review focus	Does not supply a universal risk score
LEDGAR	Provision classification	Large corpus and broad label inventory	Source and heading labels may not match private practice
ContractNLI	Document-level inference with evidence	Tests context and evidence localization	Limited hypothesis and risk-policy coverage
MAUD	M&A agreement understanding	Transaction-specific expert annotations	Specialized transaction domain

4 Proposed framework

4.1 Design principles

The framework follows six principles. The system must be policy-conditioned rather than claim to infer universal risk. It must retain source evidence and document location. It must separate detection from severity. It must communicate uncertainty and permit abstention. It must allow human correction with an audit trail. Finally, it must protect confidential material throughout ingestion, inference, storage, and evaluation.

4.2 Pipeline overview

The proposed pipeline contains the following stages, shown in Figure 2:

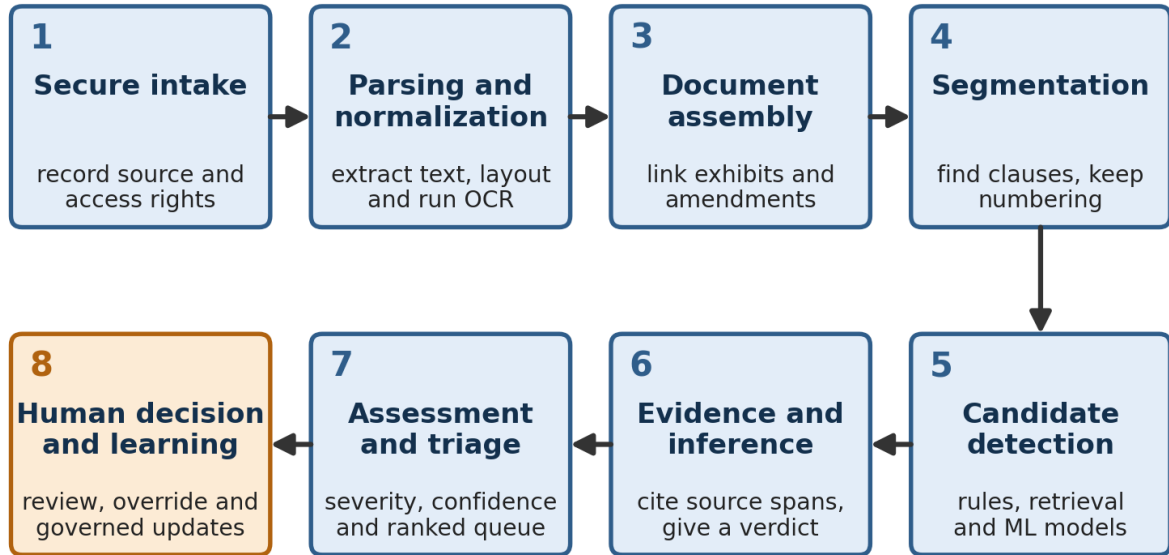


Figure 2: Proposed pipeline for human-supervised contract risk assessment

1. **Secure intake**: record document identity, source, version, access permissions, and processing authorization. Scan files for malware and isolate them from external transmission unless explicitly approved.
2. **Parsing and normalization**: extract text and layout, use OCR where needed, preserve page and paragraph positions, and identify tables, headers, footers, and signature blocks.
3. **Document assembly**: link the agreement to exhibits, schedules, amendments, and order forms; establish version and precedence relationships when available.
4. **Segmentation**: detect headings and nested clauses, retain numbering, resolve defined terms, and create overlapping context windows for long provisions.

5. **Candidate detection:** combine rules, retrieval, and ML classifiers to find expected topics, unusual formulations, missing provisions, and policy-specific deviations.
6. **Evidence and inference:** evaluate each candidate against explicit hypotheses; return one or more source spans, nearby context, cross-references, and an entailment/contradiction/unknown state.
7. **Assessment and triage:** attach a policy-based severity band and confidence separately, then rank the review queue using expected utility and reviewer capacity.
8. **Human decision and learning:** provide review controls, capture decisions and reasons, and incorporate adjudicated corrections into governed model updates.

4.3 Risk representation

Each assessment record should include: document and version identifier; clause identifier and text span; clause category; policy requirement; observed condition; evidence spans; cross-references; reviewer-facing explanation; severity band; model confidence; uncertainty reason; model and policy versions; and final reviewer disposition. The schema distinguishes “not found” from “not present.” A system may fail to retrieve a provision even when it exists, so absence should be reported only after a documented search procedure and with an uncertainty flag.

Severity can be represented as an ordinal rubric such as low, moderate, high, and critical. The rubric should define examples, not merely labels. It may incorporate financial exposure, operational interruption, regulatory impact, reversibility, and likelihood, but those components should be supplied or approved by accountable business and legal stakeholders. A model’s confidence in classification is not a probability that the organization will suffer loss. This distinction should be explicit in the interface and records.

4.4 Model families

The comparative design should include four families:

1. **Rules and lexicons:** transparent patterns for explicit thresholds, dates, named entities, and prohibited terms. These establish a simple baseline and may be suitable for deterministic checks.
2. **Classical supervised models:** TF-IDF with logistic regression or linear SVM for clause labels. They are computationally inexpensive and provide a useful benchmark.
3. **Transformer encoders:** a general encoder and a legal-domain encoder fine-tuned for classification, token labeling, and NLI. Legal-BERT is an example of a domain-adapted baseline.

4. **Retrieval-assisted language models:** retrieve relevant contract passages and approved policy clauses, then produce constrained structured outputs with evidence citations. These require separate factuality and citation checks.

An ensemble may combine deterministic rules with calibrated model scores, but the combination must be evaluated independently. Adding a model should not obscure which signal caused an alert.

4.5 Long-document and cross-reference handling

Contracts may exceed a model’s context limit. The proposed approach first segments according to document structure, then adds overlapping windows around clause boundaries. A hierarchical model can aggregate clause representations to section and document levels. For a policy question, retrieval selects candidate passages; a second-stage inference model examines those passages together with definitions, exceptions, and relevant linked sections. The system should retain a map from every model token back to the original page and character offsets.

Cross-reference resolution is not a trivial string-matching task. A clause may refer to “Section 9.2” or to a defined term whose meaning changes through an amendment. The system should maintain links with confidence and expose unresolved references rather than silently assume their content. Multi-document precedence is a legal and operational issue, so automated assembly can suggest relationships but should not decide priority where the documents conflict.

4.6 Risk ranking

The review queue should not simply sort by raw model confidence. A high-confidence low-impact label may be less urgent than a moderate-confidence potential uncapped exposure. A transparent triage score can be designed as a function of policy severity, calibrated probability of a policy deviation, expected loss category, and reviewer capacity. For example, a decision-maker may define a review utility for each risk class and choose thresholds that meet a minimum recall on critical classes. Such values are policy inputs, not learned truths.

The system should support abstention. It can route low-confidence, out-of-distribution, or conflicting-evidence cases to manual review without assigning a definitive label. Missingness, OCR quality, unresolved cross-references, and disagreement among models are meaningful reasons to abstain. A lower alert volume is not automatically preferable if it is achieved by suppressing rare high-severity issues.

5 Research methodology and evaluation plan

5.1 Research design

This paper proposes a reproducible experimental study rather than reporting completed experiments. The research design is comparative and task-oriented. It evaluates detection, classification, extraction, document-level inference, and reviewer utility as distinct outcomes. Public benchmarks support initial reproducibility; a domain-specific evaluation set is needed before making claims about a particular organization’s workflow. Figure 3 gives an overview of the evaluation design.

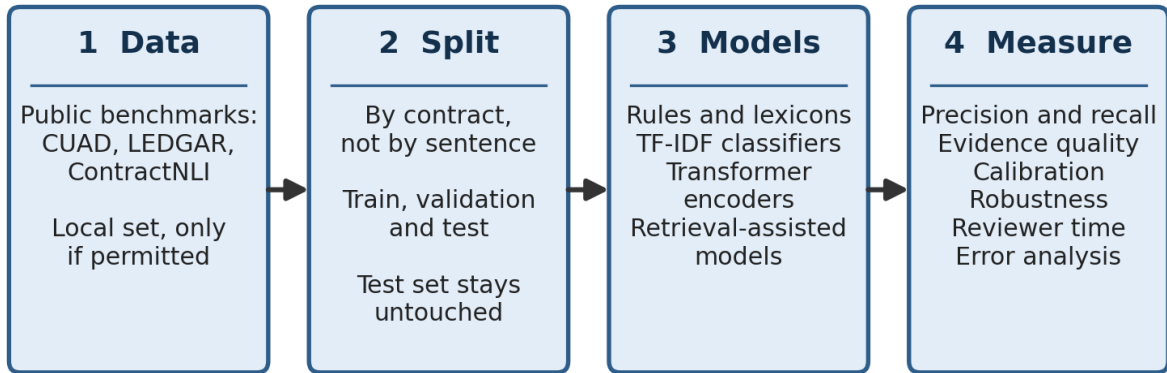


Figure 3: Overview of the proposed evaluation design

5.2 Data selection and governance

The initial benchmark phase can use CUAD for clause extraction, LEDGAR for provision classification, and ContractNLI for document-level hypothesis testing. Each dataset should be processed under its license and terms, with versions and preprocessing scripts recorded. The study should report the precise subset, label mapping, exclusions, and data split. Public data must not be presumed representative of all commercial agreements.

If a local evaluation is conducted, the organization should define an annotation protocol with practicing reviewers. Annotators should label clause type, policy deviation, evidence boundaries, severity according to a written rubric, and uncertainty. Each sample should receive independent annotation by at least two reviewers, followed by adjudication of disagreements. The study should report inter-annotator agreement and retain disagreement as information about ambiguity, rather than forcing every case into false consensus. Sensitive documents require authorization, minimization, access control, secure storage, retention limits, and a documented deletion process.

5.3 Sampling and splitting

Training, validation, and test data should be separated at the contract level. Where feasible, all documents from the same template family, organization, or matter should remain in a single split. A temporal split can test whether a model generalizes to newer drafting. Stratification can preserve rare classes, but must not break related-document grouping. Duplicate and near-duplicate detection should be performed before splitting. All test data and policy examples must remain unavailable during model selection.

For a local set, sampling should include contract types, lengths, counterparties, negotiation states, and document quality encountered in intended use. The annotation set should deliberately include hard negatives, missing clauses, exceptions, and amendments. If only one contract type is available, claims must be limited to that type. Sensitive subgroup analysis should use legally and ethically appropriate characteristics and avoid inferring protected characteristics from contract text.

5.4 Annotation protocol

The annotation handbook should state the unit of analysis, label definitions, treatment of multiple labels, evidence boundary rules, missing-provision rules, and severity anchors. Annotators should see enough context to follow definitions and exceptions. They should record whether a decision required another document or external legal knowledge. A label should distinguish an objectively observable property (for example, a clause states a cap of a specified amount) from a normative judgment (for example, that cap is unacceptable under a particular policy).

Before full annotation, a pilot should reveal ambiguous categories and estimate annotation effort. If annotation resources are limited, an active-learning strategy may help select uncertain or underrepresented examples for additional labeling, but the final test set must remain representative and untouched during model development. Reviewers can discuss disagreements, revise definitions, and then annotate a fresh calibration batch. Agreement should be reported with a measure suited to label type, such as Cohen’s kappa for two annotators on categorical labels, Krippendorff’s alpha for multiple annotators or missing judgments, and span-level overlap for extracted evidence. Agreement statistics do not prove validity; examples and disagreement analysis remain important.

5.5 Baselines and training

The proposed baseline suite is shown in Table 2. The same split and label mapping should be used where tasks permit fair comparison. Hyperparameters should be selected using validation data only. The study should report model size, tokenizer, maximum sequence length, chunking strategy, training seed, compute environment, and inference latency. Any data augmentation must be disclosed and must not place test text or labels into training.

Table 2: Proposed evaluation tasks and model comparisons

Task	Input and output	Comparisons	Main measures
Clause detection	Contract to spans and categories	Rules, TF-IDF, general BERT, Legal-BERT	Span F1, class F1
Policy inference	Contract plus hypothesis to label and evidence	NLI baseline, legal encoder, retrieval-assisted model	Macro F1, evidence F1
Value extraction	Provision to normalized value and quoted span	Pattern rules, token tagger, encoder QA	Exact match, token F1
Risk triage	Candidate list to prioritized queue	Fixed policy rules, calibrated scores, human review	Recall at k , precision at k , review time

5.6 Evaluation metrics

For classification, report precision, recall, and F1 per class, along with macro and micro averages. Macro F1 weights classes equally and can expose weak rare-category performance; micro F1 reflects aggregate instance performance. Accuracy should be included only with class distribution and a majority-class baseline. For multi-label outputs, specify the averaging method and threshold selection procedure.

For span extraction, report exact match and token-level precision, recall, and F1, with boundary-error analysis. Evidence sufficiency should be rated on a sample by reviewers: does the cited material actually support the output, and is relevant qualifying text omitted? For inference, report macro F1 across entailment, contradiction, and neutral, plus evidence localization. For triage, report recall among the top k reviewed candidates, precision at k , and risk-weighted recall for critical items. Calibration can be assessed with reliability diagrams, Brier score, or expected calibration error, with care about sample size and class imbalance.

Operational evaluation should measure time to first useful finding, total review time, corrections per document, reviewer agreement, and missed high-severity items. A controlled user study can compare manual review with model-assisted review using randomized task order and representative reviewers. It should account for learning effects and measure both speed and error. A speed improvement is not beneficial if material issues are missed more often.

5.7 Statistical reporting and error analysis

Results should include confidence intervals, preferably obtained by resampling at the contract level rather than treating correlated clauses as independent. Model comparisons should use paired evaluation where appropriate. Ablations should test the value of legal-domain pretraining, context windows, retrieval, evidence training, and policy conditioning. Error analysis should categorize false positives, false negatives, boundary errors,

missing clauses, cross-reference failures, OCR failures, and unsupported explanations. Examples should be anonymized and quoted only with permission.

The study should include robustness checks for paraphrases, formatting changes, OCR noise, unusual numbering, negation, exceptions, and unseen templates. Distribution shift can be assessed by contract type, date, language quality, and source. Tests should include adversarial or accidental prompt-like text embedded in documents if generative models are used. A system must treat contract contents as untrusted data rather than executable instructions.

5.8 Research hypotheses

The following are hypotheses for future testing, not results of this paper:

1. Domain-adapted encoders will outperform general encoders on some legal clause tasks, particularly when the training and test distributions are similar.
2. Evidence supervision and cross-reference-aware context will improve reviewer trustworthiness and reduce unsupported risk explanations.
3. A hybrid system will provide better high-risk recall at a fixed review burden than a single general-purpose classifier.
4. Calibration and abstention will improve the usefulness of the review queue under distribution shift, even if they reduce automatic coverage.

These hypotheses require experimental evidence and may not hold across datasets or domains.

6 Expected results and discussion

Because no experiment is conducted in this paper, this section describes expected patterns as research expectations rather than empirical findings. A legal-domain model may help with terminology and clause function, while a general encoder may remain competitive when the task is simple or training labels are abundant. Rules may outperform learned models on explicit, stable checks. More complex language models may produce readable explanations but require more stringent evidence validation and cost analysis.

The likely bottleneck is not only classification accuracy. It is whether the system can find the right passage, include the relevant exception, distinguish absence from retrieval failure, and connect the observed text to a policy. A strong benchmark score on clause detection could coexist with poor severity judgments. Likewise, a high-confidence answer can be dangerous if confidence is not calibrated to the deployment distribution. The design should therefore report a vector of performance measures rather than one aggregate number.

Human review is central to interpreting any expected benefit. A useful interface should show the original text, relevant neighboring context, policy clause, reason for the flag, and links to referenced provisions. It should make uncertainty visible and support correction. A reviewer should be able to say that a provision is correctly identified but acceptable, or that the model’s evidence is incomplete. These outcomes should be tracked separately from model classification correctness.

The framework also clarifies deployment boundaries. It is best suited to first-pass search, consistency checks, and structured triage. It is less suited to autonomous acceptance, rejection, or negotiation of a contract. A recommendation must not be represented as legal advice or as a guarantee that no risk exists. The organization remains responsible for selecting the policy, validating data, and determining when professional review is required.

7 Ethical, legal, and operational considerations

7.1 Privacy and confidentiality

Contracts can contain trade secrets, personal data, financial terms, and security details. Processing should follow data minimization: retain only the information required for the review task, control access by role, encrypt data in transit and at rest, and define retention and deletion schedules. Before using a hosted model, an organization must understand where inputs are processed, whether they are retained, and whether they can be used for provider training. Redaction can reduce exposure but may remove information required to interpret a clause.

7.2 Bias and representativeness

Training data may overrepresent particular industries, jurisdictions, company sizes, drafting styles, and bargaining positions. A model that treats common historical language as acceptable can reproduce unequal practices. Rare classes may have poor recall. Performance should be audited across relevant populations and contract types, with the limits of available data reported. Fairness analysis should consider both who bears false alarms and who bears missed-risk consequences.

7.3 Explainability and accountability

An explanation should be a concise statement grounded in cited text and a named policy, not a post-hoc story about hidden model reasoning. Evidence links, policy versions, model versions, timestamps, and reviewer decisions support auditability. Model cards and dataset datasheets offer useful documentation patterns for recording intended use, evaluation scope, and known limitations. Responsibility for accepting the recommendation should remain with a designated human role. Organizations should define escalation rules for critical, uncertain, conflicting, or out-of-scope cases.

7.4 Security and misuse

Contract text may contain malicious instructions intended to manipulate generative systems. The processing pipeline should isolate document text from system instructions, disable unintended tool execution, and validate structured outputs. Access logs, secure document stores, and output handling are needed to reduce unauthorized disclosure. Organizations can use risk-management guidance such as ISO/IEC 42001 to organize governance processes [8]. Privacy controls should also follow a documented lifecycle approach. Models should not be allowed to send contracts to external tools or counterparties without explicit authorization.

7.5 Automation bias and professional practice

Reviewers may over-trust a confident system or defer to a ranked queue. Interface design should encourage independent verification, show evidence before generated commentary, and make overrides easy. Training should explain what the model can and cannot establish. Users should be able to understand, correct, and safely control automated system behavior. Periodic blind audits of unflagged material are needed because users otherwise see only the system’s selected examples. The intended role is assistance, with legal and business professionals retaining decision authority.

8 Limitations

This paper is a synthesis and proposed research design, not an empirical benchmark study. No new contract corpus was annotated, no models were trained, and no performance values are claimed. The cited public datasets and model studies have specific task definitions, language coverage, source populations, and dates; results on them would not demonstrate universal applicability. The paper focuses on English-language commercial agreements and does not evaluate multilingual or civil-law contexts. Cross-lingual legal classification is a distinct research problem; success on English contracts should not be assumed to transfer to other languages.

The meaning of risk depends on organizational priorities and transaction context. A general model cannot choose those priorities legitimately. Severity rubrics may be difficult to annotate consistently, and the labels may change when policy changes. Public benchmark annotations may not include all cross-document relationships. OCR and document parsing errors can corrupt downstream predictions. Even evidence-linked systems can miss a relevant provision or fail to resolve an exception.

The proposed metrics cannot fully capture legal quality or commercial outcomes. Measuring dispute incidence would require long follow-up, reliable outcome labels, and controls for confounders; such data is not readily available and would not establish that a clause caused a dispute. Reviewer time studies can also be affected by participant experience and task design. These limitations make local validation and cautious claims essential. Calibration should be evaluated under dataset shift because confidence estimates that appear reliable on a benchmark may deteriorate on a new contract population.

9 Conclusion and future work

Machine learning can support contract review by finding provisions, extracting structured terms, testing policy hypotheses, and organizing reviewer attention. The literature provides useful foundations in expert-annotated extraction, provision classification, document-level inference, and legal-domain language modeling. Yet automated contract risk assessment is broader than a benchmark NLP label. It requires a policy reference, document context, an explicit distinction between detection and judgment, and evidence that a human can verify.

This paper proposed a modular framework with secure intake, structure-aware parsing, clause analysis, policy-conditioned inference, evidence-linked triage, and human adjudication. It also specified an evaluation plan that compares rules, classical models, legal encoders, and retrieval-assisted systems using task-specific metrics, realistic document-level splits, calibration, robustness, and reviewer utility. The key research direction is to evaluate complete workflows, not just model outputs, and to determine when systems should abstain.

Future work should build a carefully governed, multi-annotator dataset tied to a clearly defined policy; compare public benchmark performance with prospective local performance; evaluate evidence completeness and cross-reference resolution; and conduct controlled studies with reviewers. Research should also test distribution shift, multilingual agreements, and privacy-preserving learning. Until such evidence is available, systems should be used as auditable decision support under qualified human supervision.

Author note

This manuscript is written as a research review and proposed methodology. It should be adapted to the course's required citation style, supervisor guidance, and institutional submission rules. Any empirical study, data access, or claims of results must be completed and verified before submission.

A Illustrative risk-review rubric

The following example is a research instrument, not legal advice and not a universal policy. Before use, an organization should adapt every definition to its agreements, business model, and jurisdiction. The purpose is to show how a model output can be decomposed into observable text, a policy comparison, severity, and review action.

Table 3: Example severity anchors for an annotation handbook

Band	Illustrative meaning	Example observable condition	Suggested route
Low	Limited exposure or routine variance under the specified playbook	Minor notice-period variation with a clear termination right	Standard review
Moderate	A deviation with bounded impact or a compensating protection	A cap differs from fallback but remains linked to fees	Reviewer checks context and countervailing terms
High	Material potential exposure or an important missing protection	Broad indemnity or materially elevated cap, subject to confirmation	Escalate to responsible legal or business reviewer
Critical	Potentially severe or difficult-to-reverse exposure, or a policy prohibition	Uncapped liability in a category the approved policy treats as critical	Immediate specialist review; do not auto-approve
Unknown	Evidence, OCR, context, or policy is insufficient	Relevant schedule missing or conflicting amendment references	Obtain missing material or abstain

Annotators should assign a band only after identifying the policy and evidence. They should be able to record “not assessable” separately from low risk. If the clause is absent, the record should distinguish a verified omission from an incomplete or failed document search. If a provision is present but its meaning depends on another agreement, the annotation should say which material is required. These distinctions help prevent missing evidence from becoming an unjustified reassuring label.

B Illustrative structured output

The sample below uses fictional text and demonstrates the proposed output contract. It does not describe a real agreement or assert a legal conclusion.

```
document_id: sample-001
document_version: 3
policy_id: procurement-playbook-2026-04
topic: limitation_of_liability
observed_condition: cap_exceptions_may_be_broad
evidence:
  - page: 18
    section: 12.3(b)
    quote: "The foregoing limitation shall not apply to..."
  - page: 5
    section: 1.8
    quote: "Confidential Information includes..."
policy_deviation: possible
severity: high (policy rubric)
model_confidence: 0.74 (not a loss probability)
uncertainty: exception scope depends on referenced definition
review_action: inspect full clause and related indemnity section
model_version: legal-encoder-0.1
reviewer_disposition: pending
```

The important feature is not the particular score. It is the separation between the model's estimated confidence in its output and the policy's severity category. The output exposes evidence, source location, uncertainty, and the policy version, so a reviewer can verify or reject the proposed interpretation. A production interface would render the quotations as links to the exact passages and adjacent context, rather than asking the reviewer to rely on copied fragments.

C Reproducibility checklist

A reproducible report should state the dataset release and license; document counts and exclusion reasons; contract-level split strategy; duplicate-detection method; text extraction and OCR versions; normalization and chunking rules; label definitions; annotation process; model architecture and checkpoint; tokenizer and maximum input length; hyperparameters; seeds; compute environment; inference latency; calibration procedure; thresholds; and statistical uncertainty. It should also state whether prompts, policies, and retrieval corpora were fixed before evaluation.

The data statement should describe whether documents contain confidential information, what permission supports their use, whether any redaction occurred, and what access controls and retention period apply. Model cards should report intended uses, out-of-scope uses, evaluation populations, known limitations, and failure examples. Dataset documentation should explain the source population and labeling process. These reports make it possible to assess whether a result applies to a proposed deployment.

For a human evaluation, report reviewer experience, sample selection, task order, interface, training, blinding, time measurement, disagreement resolution, and adverse events such as missed critical provisions. Preserve a record of all policy and model changes during the study. If a model is updated, evaluate it against a fixed holdout and a fresh sample before replacing the deployed version.

D Example error-analysis categories

1. **Missed provision:** a relevant clause exists but the parser, retrieval stage, or classifier fails to surface it.
2. **False omission:** the model reports that a protection is missing although it appears under different terminology or in an exhibit.
3. **Scope error:** the correct clause is found, but an exception, definition, or amendment changes its effect.
4. **Boundary error:** the evidence span excludes language necessary to interpret the provision or includes unrelated text.
5. **Policy mismatch:** the text is described correctly, but the wrong playbook version or transaction profile is applied.
6. **Severity error:** the policy deviation is identified, but the assigned priority conflicts with the approved rubric.
7. **Unsupported explanation:** generated prose adds a factual or legal claim that the cited text does not support.
8. **Document quality failure:** OCR, table extraction, numbering, or cross-document assembly corrupts the input.

Error categories should be reported by frequency and severity, with representative examples where permitted. Aggregate metrics alone cannot show which stage needs correction. The categories also provide a basis for targeted annotation and system improvements.

References

- [1] Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The Long-Document Transformer. *arXiv preprint arXiv:2004.05150*. <https://arxiv.org/abs/2004.05150>.
- [2] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The Muppets straight out of Law School. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 2898–2904. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2020.findings-emnlp.261>.
- [3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT 2019*, pages 4171–4186. Association for Computational Linguistics. <https://doi.org/10.18653/v1/N19-1423>.
- [4] Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. CUAD: An Expert-Annotated NLP Dataset for Legal Contract Review. In *Advances in Neural Information Processing Systems 34: Datasets and Benchmarks Track*. <https://openreview.net/forum?id=7l1Ygs3Bamw>.
- [5] Yuta Koreeda and Christopher D. Manning. 2021. ContractNLI: A Dataset for Document-level Natural Language Inference for Contracts. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1907–1919. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2021.findings-emnlp.164>.
- [6] Steven H. Wang, Antoine Scardigli, Leonard Tang, Wei Chen, Dmitry Levkin, Anya Chen, Spencer Ball, Thomas Woodside, Oliver Zhang, and Dan Hendrycks. 2023. MAUD: An Expert-Annotated Legal NLP Dataset for Merger Agreement Understanding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16369–16382. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2023.emnlp-main.1019>.
- [7] Don Tuggener, Pius von Däniken, Thomas Peetz, 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. European Language Resources Association. <https://aclanthology.org/2020.lrec-1.155/>.
- [8] International Organization for Standardization. 2023. *ISO/IEC 42001:2023 Information technology—Artificial intelligence—Management system*. Geneva: ISO. <https://www.iso.org/standard/81230.html>.