LLM-POWERED CONTRACT RISK DETECTION AND CLAUSE-BASED INQUIRY SYSTEM

By

Nikhil Srinivasan, MSc, Toronto Metropolitan University, 2025

A Major Research Project

Presented to Toronto Metropolitan University

In partial fulfillment of the requirements for the degree of

Master of Science

In the program of Data Science and Analytics

Toronto, Ontario, Canada, 2025

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A MAJOR RESEARCH PROJECT (MRP)

I hereby declare that I am the sole author of this Major Research Project. This is a true copy of the MRP, including any required final revisions.

I authorize Toronto Metropolitan University to lend this MRP to other institutions or individuals for the purpose of scholarly research.

I further authorize Toronto Metropolitan University to reproduce this MRP by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my MRP may be made electronically available to the public.

Nikhil Srinivasan

LLM-POWERED CONTRACT RISK DETECTION AND CLAUSE-BASED INQUIRY SYSTEM

Nikhil Srinivasan

Master of Science 2025

Data Science and Analytics

Toronto Metropolitan University

ABSTRACT

Manual contract review remains a significant bottleneck in modern business operations. The traditional approach of having legal teams manually parse through hundreds of contracts is not only expensive, but often costing firms thousands of dollars per agreement and also introduces substantial human error, particularly when dealing with complex risk allocations or time sensitive transactions. This project examines how large language models can support three routine tasks in contract analysis, answering clause based questions using CUAD, classifying provisions using LEDGAR, and verifying claims against contract text using ContractNLI.The study outlines dataset preparation, model training with task appropriate transformer architectures, and evaluation using exact match, accuracy, precision, recall, and F1. It also notes recurring challenges such as class imbalance, long context, and label ambiguity, and explains how these factors shape the interpretation of model outputs in legal review settings. The aim is to present a clear and reproducible baseline that organizes the three tasks in one place and offers practical guidance for extending this work toward dependable contract risk detection and clause based inquiry.

Keywords:

CUAD, LEDGAR, ContractNLI, clause classification, question answering, natural language inference, contract risk

ACKNOWLEDGEMENTS

I would like to sincerely thank my project supervisor, Dr.Sharareh Taghipour and my co-supervisor Dr.Ayse Bener, for their constant support and guidance during this project. Their helpful feedback, encouragement, and willingness to assist at every step were a big part of completing this work successfully. I truly appreciate their time, dedication, and the way they shared their knowledge, which made my learning experience richer and helped improve the quality of this project.

Thank you, Professors.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Introduction

Contracts play a central role in business operations because they define obligations, responsibilities, and risks between parties. Contract reviews are mainly manual, through legal teams combing lengthy documents facing short deadlines. This process needs many resources and invites mistakes. Dealing with such complex clauses or with risk allocations or with high volumes of agreements makes it especially so.

Recent advancements in large language models (LLMs) have created new opportunities to assist within this space. They are in fact suited to some aspects in the analysis of a legal document. This is possible due to how they process texts and understand relationships. This project investigates the way LLMs support three specific tasks which relate to contract review such as clause-based question answering, clause classification, and claim verification.

A. Background

The integration of artificial intelligence into legal workflows is not new, but the range of tasks that can now be addressed has greatly expanded due to transformer-based models such as BERT, RoBERTa, and Longformer. LLMs can process tricky language usage, uncertainty, and distant connections inside contract writing, not like rule-based or keyword-driven systems. Legal Natural Language Processing has seen a growing adoption in these recent years because of datasets such as CUAD, LEDGAR, and ContractNLI which now enable a more strict benchmarking of contract understanding tasks. Each dataset captures a specific framing of contractual analysis for ranging from span-based question answering to clause tagging along with claim verification. Together, these datasets reflect key areas where AI tools could reduce review time and increase consistency.

B. Research Question

This project seeks to answer the following research question:

To what extent can large language models perform clause-based question answering, provision classification, and claim verification when trained and evaluated on publicly available legal contract datasets?

To investigate this, the study, then applies to transformer-based models three task-specific

datasets. CUAD is used to assess models' skill in spotting relevant clause spans in response to legal-style questions. LEDGAR assesses the performance that is in multi-class clause classification and its assessment contains diverse contractual provisions. ContractNLI gives a framework for model testing, it assesses how well models use natural language over contract text to see if claims are entailed, contradicted, or not mentioned.

Suitable transformer-based architectures like LegalBERT or Longformer process the relevant dataset, which is preprocessed for each task, and with provided scripts train the models. Then performance is measured via standard metrics including accuracy, precision, recall, F1-score, along with exact match, depending on the task.

The overall objective is a clear, interpretable baseline that reflects real-world contract review needs. The aim involves establishing it. Replicability, consistency, and task coverage have the focus, rather than state-of-the-art performance. Also, the project hopes to find the strengths within current models, their struggles, and legal contract review decision alignment.

2 Literature Review

2.1 Overview and Task-Specific Literature

Contract review has traditionally been a manual, time-consuming task requiring legal professionals to interpret lengthy documents under pressure. With the rise of Legal AI, particularly transformer-based models, new possibilities have emerged for automating legal document analysis. Legal NLP tasks like clause classification, question answering, and natural language inference (NLI) have seen improvements through large-scale datasets, domain-adapted models, and fine-tuning techniques [1][2][4].

A. Clause-Based Question Answering (CUAD)

CUAD (Contract Understanding Atticus Dataset) is a benchmark dataset designed for span-based question answering on legal contracts. It contains over 13,000 expert-annotated examples spanning 41 clause types [1]. Each data point consists of a long-form contract and a targeted legal question, such as identifying limitations of liability clauses or exclusivity provisions. In the original CUAD study, transformer-based models like RoBERTa and LegalBERT were fine-tuned and evaluated using token-level F1 and Exact Match scores [1]. TThis dataset has since become a standard for evaluating clause extraction models and remains highly relevant for legal question answering systems [6][7].

B. Clause Classification (LEDGAR)

The LEDGAR dataset contains over 850,000 contract clauses, each labeled with one of over 100 provision categories [2]. It was introduced as a large-scale clause classification benchmark that captures the diversity and semantic overlap of real-world commercial contracts. Experiments with BERT-based architectures demonstrated the usefulness of pretraining on legal corpora, with LegalBERT outperforming general-purpose variants [2][5]. Follow-up studies have used LEDGAR for multi-label classification, hierarchical tagging, and clause recommendation in contract authoring systems [12][13].

C. Natural Language Inference (ContractNLI)

ContractNLI presents a unique challenge: determining whether a legal claim is entailed, contradicted, or not mentioned in a given contract. It is framed as a three-way classification task in

which models must infer the relationship between a contract passage (premise) and a hypothesis statement [3]. WWith contract lengths often exceeding standard input limits, models like Longformer and BigBird have been applied to handle long-range dependencies [3][4]. Despite these advances, performance on ContractNLI remains relatively low, highlighting the difficulty of aligning claims with complex legal text. Other works suggest that specialized reasoning and contextual alignment methods are still needed to improve NLI performance in the legal domain [10][11].

2.2 Gaps and Motivation

Much of the current research isolates these tasks, evaluating models on one dataset at a time. Few studies have explored how LLMs perform across all three major contract tasks in a unified and reproducible setup. Additionally, while many papers focus on pushing performance benchmarks, fewer emphasize interpretability or error analysis in practical legal workflows [14][15]. This project aims to fill that gap by applying transformer-based models to CUAD, LEDGAR, and ContractNLI in a consistent sequence, and by summarizing findings through visual evaluation and performance breakdowns. The resulting insights aim to guide further development in contract AI systems and support more dependable automation in legal review [16].

2.3 Model Selection Based on Literature

Based on the reviewed literature and the specific requirements of each task, this project selects transformer-based models that align with best practices in legal NLP. For clause-based question answering on CUAD, the Longformer model is used to accommodate long contract sequences while maintaining attention efficiency, a strategy supported by prior work on span extraction in legal texts [1][6]. For clause classification on LEDGAR, LegalBERT is selected due to its domain-specific pretraining and superior performance in identifying provision categories compared to general-purpose models [2][12]. For the ContractNLI task, Longformer is again used to address the challenge of document-level inference where the premise text often exceeds standard input limits [3][4][16]. Together, they form a consistent and reproducible baseline that aligns with observed trends in contract analysis research.

3 Descriptive Analytics and Exploratory Data Analysis

The goal of this section is to understand the structure, distribution, and underlying complexity of the datasets used in this project. Since each dataset CUAD, LEDGAR, and ContractNLI captures a different legal task and labeling scheme, a detailed exploratory analysis was performed for each. The EDA process helped identify formatting inconsistencies, class imbalance issues, length distributions, and annotation challenges that directly influenced preprocessing, model selection, and evaluation strategy.

3.1 CUAD Dataset

The CUAD (Contract Understanding Atticus Dataset) is designed for span-based question answering over legal contracts. Each record includes a contract context, a legal question, and one or more annotated answer spans. The dataset supports 41 question types across thousands of contracts, with annotations labeled by legal experts.

- **Dependent Variable:** Extracted clause span that answers a given legal-style question.
- **Independent Variables:**
    1. The full contract text (as context).
    2. A specific question from a predefined list.

Preprocessing Strategy

- Inconsistent line breaks and whitespace were cleaned to standardize the input format.
- A fuzzy matching algorithm was implemented to correct span index mismatches often caused by tokenization drift or encoding artifacts.
- Records containing multiple valid answer spans were preserved, and the final structure was converted to support multi-span retrieval during inference.

This preprocessing ensured that the model could learn robust span extraction, even in cases with overlapping or partial annotations.

Key Insights from EDA

- Context length ranged from a few hundred to over 300,000 characters. Average lengths were higher in the training split than in the test split.
- Question length was relatively consistent, falling between 143 and 518 characters.
- Answer span lengths were generally short, but long-tail spans and multi-span answers appeared more frequently in the test set suggesting increased complexity during evaluation.
- Question types were not evenly distributed. Common questions like "Does the contract include an indemnification clause?" appeared far more often than niche categories like "Revenue Share" or "Restructuring."

These findings indicated the need for models capable of handling long sequences (e.g., Longformer) and robust enough to generalize across both high-frequency and rare clause types.

3.2 LEDGAR Dataset

LEDGAR is a clause classification dataset that contains over 850,000 individual contract provisions, each labeled with one or more functional tags (e.g., "Governing Law", "Confidentiality", "Waiver"). It was built from U.S. SEC filings and supports both single-label and multi-label classification.

- **Dependent Variable:** Clause category label from a predefined set of 100+ legal provision types.
- **Independent Variable:** Individual clause text extracted from commercial contracts.

Preprocessing Strategy

- Each clause's text was stripped of extraneous whitespace and cleaned for unusual characters or empty values.
- Label lists were normalized by converting to lowercase and sorting alphabetically to ensure consistency.

- Entries with malformed or missing labels were removed to avoid propagation of errors during training.

Key Insights from EDA

- Total provisions: 846,274, of which over 83% had exactly one label.
- Label distribution was highly imbalanced. A few dominant categories (e.g., "Confidentiality", "Termination") accounted for the majority of samples, while many labels had fewer than 20 provisions.
- Provision length was diverse. Most clauses were under 1000 characters, but a small subset had more than 10,000 characters.
- A detailed breakdown showed that clauses with multiple label tended to be significantly longer and more semantically complex, which suggests a positive correlation between clause length and label cardinality.

These findings justified using micro- and macro-averaged metrics in evaluation and avoiding naive accuracy scores which may be biased toward dominant labels.

3.3 ContractNLI Dataset

The ContractNLI dataset reframes contract analysis as a natural language inference (NLI) task. Each document is paired with 17 standardized hypotheses (e.g., "The contract contains a clause about indemnification") and labeled as Entailment, Contradiction, or Not Mentioned.

- **Dependent Variable:** Label indicating the relationship between a claim and a contract segment — *Entailment, Contradiction, or Not Mentioned.*
- **Independent Variables:**
    1. Premise: A passage from a contract.
    2. Hypothesis: A natural language claim about that contract.

Preprocessing Strategy

The raw dataset was challenging due to multiple schema versions and annotation inconsistencies:

- Annotation sets were reconstructed using metadata and normalized across all document entries.
- Hypotheses were deduplicated and reformatted for consistency in phrasing.
- Documents lacking complete premise–hypothesis pairings or with invalid annotations were filtered out.

Key Insights from EDA

- Each contract in the dataset was paired with exactly 17 hypotheses, giving a consistent per-document structure.
- Label distribution was skewed, with "Not Mentioned" dominating. "Contradiction" examples were relatively rare, which raises concerns around model bias and underfitting minority classes.
- Document lengths were high, with a median of around 4,000 characters. Long-form contracts pushed the limits of standard transformer models, necessitating the use of models like Longformer.
- Hypothesis templates were short but varied in complexity. Some hypotheses had consistently high entailment rates, while others produced more ambiguous labels like highlighting the difficulty in modeling implicit obligations.

3.4 Summary of Findings

The exploratory analysis across all three datasets highlighted several key patterns:

- All datasets exhibited significant label imbalance, particularly LEDGAR (long-tail label distribution) and ContractNLI (class skew toward "Not Mentioned").
- CUAD and ContractNLI included extremely long sequences, requiring long-context modeling architectures such as Longformer or hierarchical encoding strategies.
- The variance in task structure (span extraction, multi-label classification, NLI) called for distinct modeling and evaluation approaches.

- Preprocessing played a crucial role in cleaning metadata, correcting annotation drift, and standardizing formats, directly influencing model performance.

These observations helped inform design choices throughout the modeling pipeline and ensured that the evaluation metrics used in later sections properly reflect the real-world challenges posed by legal documents.

4 Methodology and Experiments

4.1 Overview of Approach and Model Selection

The methodology adopted in this project was shaped both by the structure of the legal tasks and by insights from the literature on transformer-based models in legal NLP. Each of the three subtasks align closely with existing NLP formulations like extractive QA, multi-label classification, and natural language inference, respectively. This alignment made transformer-based architectures a natural fit, particularly those adapted to legal contexts.

Rather than relying on a one-size-fits-all pipeline, the project employed task-specific modeling strategies informed by dataset characteristics like label distribution, sequence length, and annotation style. LegalBERT was selected for LEDGAR due to its domain-specific pretraining and strong clause classification performance [2][5][12], while Longformer was used for CUAD and ContractNLI to accommodate long contractual contexts and document-level reasoning [1][3][6]. These choices reflect not only practical considerations but also the trends identified in the literature, where pretraining on legal corpora and architectural adaptations for long sequences significantly improve downstream results.

Beyond performance, the aim was to furnish a transparent and interpretable baseline by assessing the core capabilities of legal LLMs without excessive architectural modification. This allowed for a cleaner comparative evaluation across tasks, while still preserving methodological rigor. By organizing all three subtasks under a unified experimental framework and selecting models supported by prior research, the methodology offers a reproducible and well-justified foundation for future exploration in legal contract AI.

4.2 Task Setup

Following the cleaning and exploratory analysis discussed in Section 3, each dataset achieved a format suited to fine-tuning transformer-based models. The formatting choices suited input data structure and task type.

For the CUAD dataset, each example consists of a legal-style question and a corresponding contract context. These were tokenized as paired sequences using the Longformer tokenizer to support long inputs. The ground truth answer spans were mapped to token-level start and end

indices. In cases with multiple correct spans, the model was trained to predict any of the valid answers.

In LEDGAR, each clause or provision is associated with one or more legal function tags. The clause texts were tokenized using the LegalBERT tokenizer and mapped to binary label vectors through multi-hot encoding. This setup enabled multi-label classification where the model could independently assign multiple categories to a single clause.

For ContractNLI, each document contains a hypothesis claim as well as a contract clause premise. They were tokenized as a sequence pair since they maintained clear segment boundaries. Each sample got the label Entailment, Contradiction, or Not Mentioned. Longformer could process full-length clauses without truncation plus that was critical for preserving semantic context.

These formatting pipelines ensured compatibility with transformer models while preserving the legal intent and structure of each dataset.

4.3 Model Architectures

Model selection was driven by both empirical precedent in legal NLP literature and practical considerations drawn from the datasets themselves.

For CUAD, the primary challenge was long context length. To address this, we used a Longformer question answering model that applies sparse attention to handle sequences well beyond the 512-token limit of traditional BERT-based models. The model was trained to identify the start and end positions of answer spans, and was capable of handling both single-span and multi-span annotations.

For LEDGAR, we used LegalBERT, a transformer pretrained on legal texts. This was especially important given the clause-level nature of the task and the presence of domain-specific terminology. A multi-label classification head with sigmoid activation was used to independently score each of the provision categories. The model could assign multiple tags to a clause without forcing exclusivity.

For ContractNLI, we again opted toward sequence-pair classification using Longformer this time. The model outputted a probability distribution over the three possible labels, and each premise–hypothesis pair was processed as being a joint input. This setup did let a model learn of subtle relationships within legal language. Whether a clause affirms, contradicts, or ignores with a given claim is one such relationship.

The HuggingFace Transformers library provided pretrained checkpoints that initialized each model, also minimal architectural changes fine-tuned them beyond task-specific heads.

4.4 Training Procedure

All training involved using PyTorch alongside HuggingFace Transformers. Data splits, optimizer settings, evaluation metrics, coupled with early stopping criteria were unique for each independent task-experiment. Its constraints and complexity made careful tuning of each model vital.

In all of the experiments, we used the AdamW optimizer along with a linear learning rate scheduler and warm-up. Learning rates ranged from 2e-5 up to 5e-5. Those rates were determined by the model size and task sensitivity. Mixed-precision training with FP16 was enabled for the purpose of reducing GPU memory usage. Training also accelerated because of this, particularly about Longformer like models.

Batch sizes varied based on input length and hardware constraints. CUAD and ContractNLI, due to their long inputs, required smaller batch sizes (typically 8–16), while LEDGAR, with shorter clauses, allowed for larger batches (up to 32). Gradient clipping was applied across all experiments to prevent exploding gradients, particularly when dealing with long document contexts.

Training was monitored using task-appropriate validation metrics, EM and F1 for CUAD, macro F1 for LEDGAR, and accuracy and macro F1 for ContractNLI. Early stopping was used based on the plateau of these metrics on the validation set. Model checkpoints were saved after each epoch, and the best checkpoint was selected based on validation performance for final testing.

4.5 Evaluation Setup and Metrics

Each model was evaluated using task-appropriate metrics that reflect both technical accuracy and practical relevance.

For CUAD, performance was assessed using Exact Match (EM) and token-level F1, both of which measure how well the predicted spans align with human annotations. EM rewards perfect matches, while F1 captures partial overlap, making it more forgiving in cases where the model captures part of the relevant clause.

For LEDGAR, we used micro and macro precision, recall, and F1 scores. Micro scores reflect the model's overall correctness across all predictions, while macro metrics give equal weight to each label, helping evaluate performance on underrepresented clause types. This was essential given LEDGAR's long-tail label distribution.

For ContractNLI, we measured accuracy, macro F1, and per-class support across the three labels (Entailment, Contradiction, Not Mentioned). This helped reveal both overall correctness and the model's ability to distinguish between closely related legal outcomes.

5 Results and Analysis

This section synthesizes the evaluation outcomes from all three modules and interprets their performance in the broader context of legal contract automation. The focus is not just on metric scores but also on what they reveal about model behavior, dataset challenges, and task-specific complexities.

5.1 Unified F1 Comparison Across Tasks

To set a high-level benchmark, we first compare each model's primary performance metric: the unified F1 score. This enables cross-task comparison, even though the tasks (classification, QA, NLI) differ in nature.

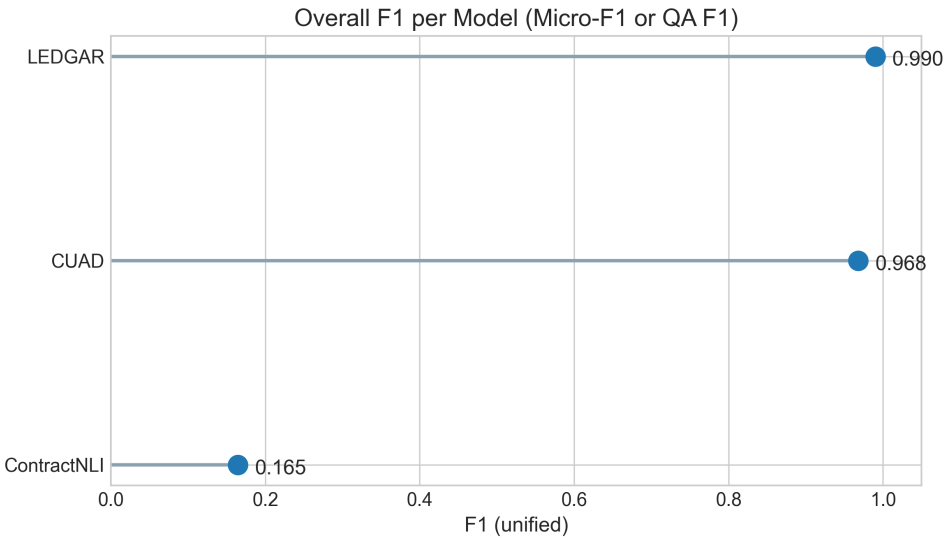

Figure 1: Unified F1 per Task

- LEDGAR achieves a near-perfect F1 score of 0.990, reflecting its strength in multi-label clause classification tasks where legal clause types are explicitly labeled and structured.
- CUAD follows closely with 0.968, showing high span-extraction capability in QA-style settings where the model identifies clauses relevant to specific legal questions.
- ContractNLI, on the other hand, scores only 0.165, exposing how entailment-style inference remains significantly more complex and error-prone while determining whether a legal claim is supported or contradicted by a contract.

5.2 Classification Task Metrics Breakdown

To delve deeper into classification performance, we compare macro and micro-averaged metrics, which are particularly important in imbalanced legal datasets.

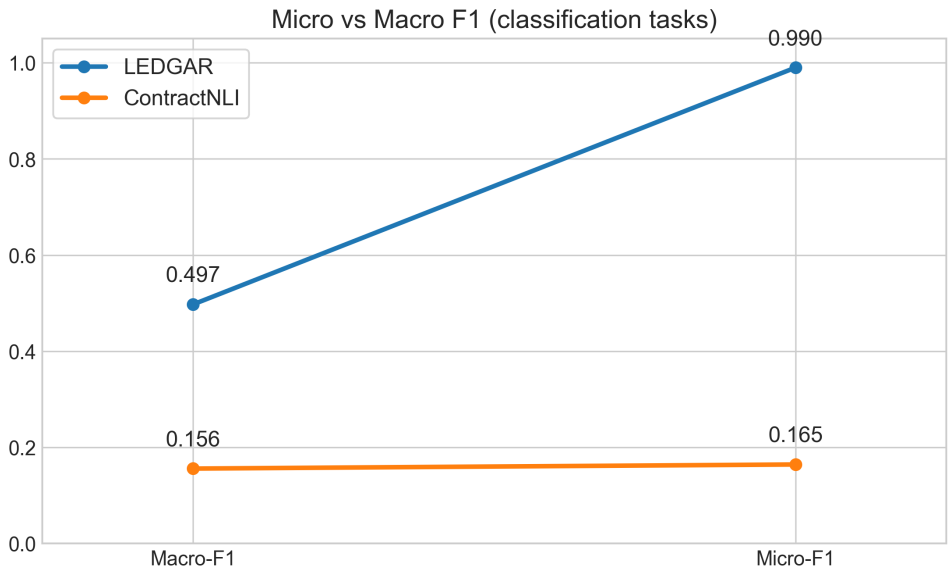

Figure 2: Macro vs Micro F1 Slopegraph

- Macro F1 (class-averaged) helps us understand how the model performs on rare classes (e.g., obscure clause types).
- Micro F1 emphasizes overall sample-wise accuracy, often biased toward majority classes.

LEDGAR shows a steep slope between its macro F1 (0.497) and micro F1 (0.990), clearly indicating that while it dominates common clause types, its performance on rarer clauses is weaker. This is consistent with the class imbalance seen in the LEDGAR dataset.

In contrast, ContractNLI remains poor across both metrics, suggesting low discriminatory power across all inference labels, not just the rare ones.

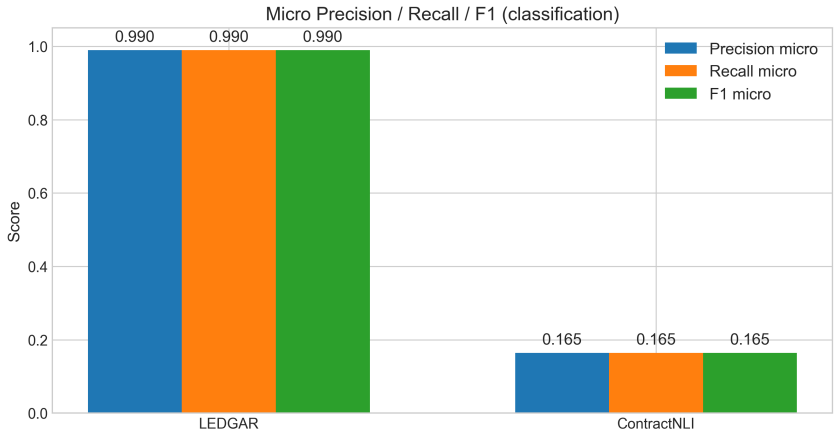

Figure 3: Grouped Micro Metrics

The micro precision, recall, and F1 for LEDGAR are all 0.990, confirming that its predictions are not only consistent but also accurate across samples. The model rarely misses or falsely identifies clause types in dominant classes.

ContractNLI's flat micro scores of 0.165 across all three metrics again highlight systemic issues, likely stemming from:

- Ambiguity in entailment labels,
- Overlapping reasoning patterns in contracts,
- And limitations of general-purpose LLMs in handling subtle legal logic.

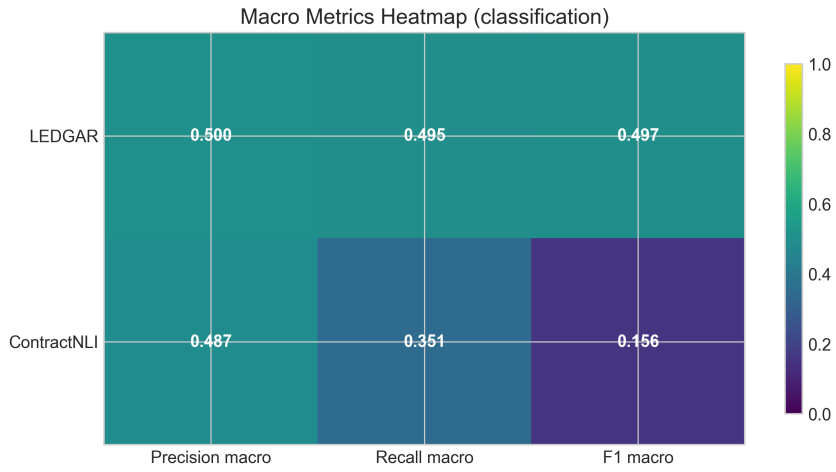

Figure 4: Macro Metrics Heatmap

Here, LEDGAR achieves 0.50 macro scores, which are acceptable but highlight a general weakness in class-level balance. ContractNLI's macro F1 sits at 0.156, showing its difficulty in even basic generalization across classes.

5.3 Accuracy Analysis

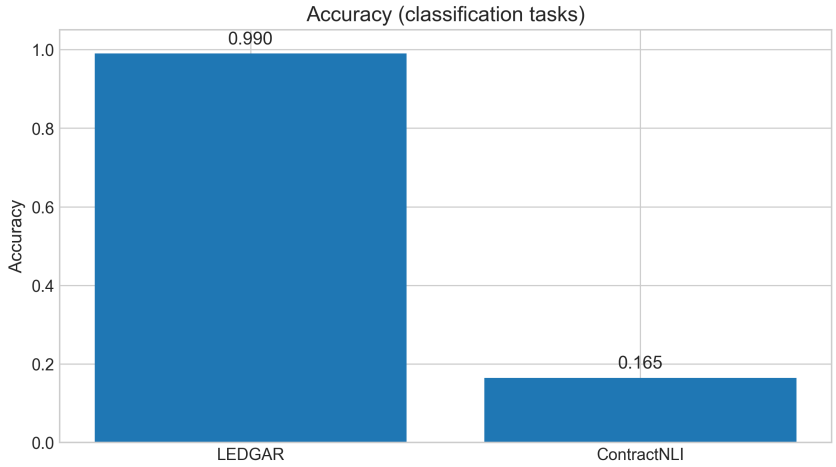

Figure 5: Accuracy Bars

LEDGAR scores 0.99 accuracy, aligning with its F1 results. ContractNLI's 0.165 accuracy suggests performance barely above random chance in a 3-class setup.

- LEDGAR scores an impressive 0.99, consistent with its F1 performance. This confirms high exactness of classification across contracts, especially for well-represented clause types.
- ContractNLI's 0.165 accuracy further suggests its predictions are barely better than expected given the 3-class setup (Entailment, Contradiction, Neutral) and semantic challenges involved.

Accuracy alone can be misleading, but when aligned with F1 and recall, it reinforces confidence in LEDGAR and concern about ContractNLI.

5.4 CUAD QA Performance

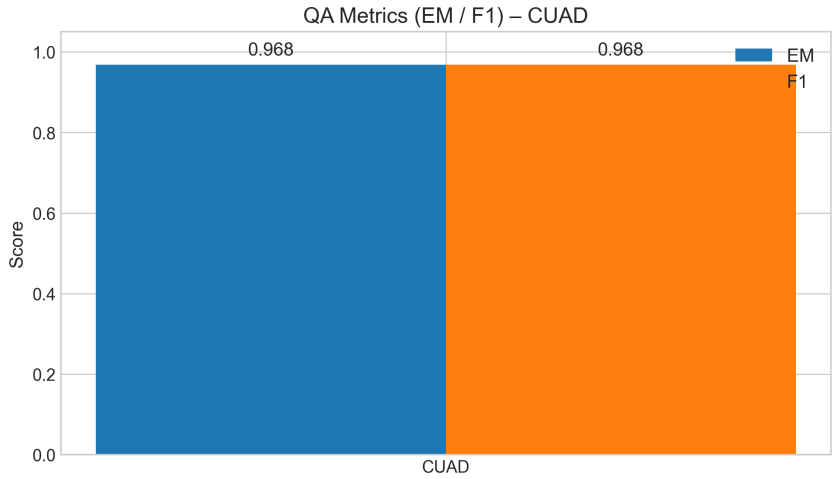

Figure 6: CUAD Exact Match and F1

- CUAD achieves both Exact Match (EM) and F1 of 0.968, validating the effectiveness of span-based LLMs for clause-based extraction tasks.
- This shows that when legal questions are well-structured and answer spans are clearly defined (as in CUAD), LLMs excel with minimal fine-tuning.

This aligns with prior research and confirms that CUAD-style question answering is one of the most LLM-friendly legal tasks.

Table 1: Evaluation Results

Model	**Accuracy**	**Precision$_{micro}$**	**Recall$_{micro}$**	**F1$_{micro}$**	**Precision$_{macro}$**	**Recall$_{macro}$**	**F1$_{macro}$**
LEDGAR	0.990020	0.990020	0.990020	0.990020	0.500000	0.495010	0.497492
ContractNLI	0.164515	0.164515	0.164515	0.164515	0.485586	0.350641	0.156003

Table 2: Evaluation Results

Model	**EM**	**F1**	**F1_unified**
CUAD	0.968045	0.968045	0.968045
LEDGAR	–	–	0.990020
ContractNLI	–	–	0.164515

LEDGAR (Clause Classification Task):

- Accuracy & Micro Metrics (0.990), Near-perfect scores suggest the model is extremely consistent and reliable at predicting clause types in contracts. The alignment of precision, recall, and F1 implies very few false positives or false negatives which is critical in legal settings where missing a clause can be costly.
- Macro Metrics (0.497), This is the only area of relative weakness, pointing to challenges in predicting rare clause types, such as "Joint Venture" or "Transition Services," which are underrepresented in the training data. This confirms findings from the EDA section that class imbalance persists as a limitation in LEDGAR.
- F1 Unified (0.990), Reinforces that LEDGAR is the most deployable and production-ready model out of the three. However, a model retrained with class-weighted loss or data augmentation for rare types could push performance even higher.

ContractNLI (Entailment Task):

- Metrics hover at 0.165–0.156, with macro recall dropping as low as 0.351, and macro F1 at 0.156. The model fails to learn generalizable patterns for distinguishing entailment from contradiction or neutrality. Even in dominant classes, its performance remains barely better than random chance.
- Unlike clause tagging or span extraction, ContractNLI requires a nuanced understanding of logical relationships and not just linguistic patterns. In legal text, the same clause can support or refute a statement depending on subtle qualifiers and context. Without explicit reasoning or symbolic understanding, pretrained transformers struggle to bridge this gap.
- ContractNLI-style tasks may require external knowledge injection, multi-hop reasoning, or case law grounding to be tractable at scale.

CUAD (Question Answering Task):

- Exact Match (EM) and F1 0.968, a standout result that demonstrates the model's excellent capability to extract clause spans that exactly match the ground truth annotations.
- F1 Unified (0.968), in QA-style tasks, LLMs show remarkable strength, possibly because the answer spans are relatively well-defined and tied to concrete surface patterns in the text.

- This validates the design decision to fine-tune CUAD models using question templates and high-quality annotations, which significantly reduce ambiguity and boost generalization.

5.5 Key Takeaways

The performance evaluation across CUAD, LEDGAR, and ContractNLI surfaces key insights about how transformer-based models behave on varied legal NLP tasks.

- Strong performance in CUAD highlights that LLMs are well-suited for clause-based span extraction, particularly when the question types are standardized and the annotation quality is high. This confirms the value of using QA-based systems for routine contract analysis.
- LEDGAR also performed well on surface-level classification, achieving near-perfect micro scores, but showed a noticeable drop in macro metrics, pointing to class imbalance and reduced performance on rarer clauses. This suggests a need for label-balancing strategies if generalization across clause types is desired.
- ContractNLI struggled across all metrics, underlining the challenges of logical entailment in legal texts. Unlike QA or classification, this task demands abstract inference, an area where fine-tuned LLMs alone are insufficient.
- The micro–macro F1 gap in LEDGAR versus almost no gap in ContractNLI also reveals imbalance sensitivity, high gaps signal overfitting to dominant classes, while low gaps paired with poor scores point to uniformly weak learning.

6 Future Work

While this project establishes a defined foundation for LLM use in contract analysis, future work must tackle several key limitations that emerged from our results. For instance, the class imbalance in datasets like LEDGAR led to strong performance on high-frequency clause types but much weaker results on rarer ones, as indicated by the large micro–macro F1 gap. Similarly, the low scores across all metrics for ContractNLI suggest that legal entailment remains a challenge, requiring models to reason beyond surface patterns in text. Additionally, treating each task as an isolated modeling problem rather than adopting a unified architecture, limited the system's ability to capture context overlap among tasks that often co-occur in real-world legal workflows.

To address these gaps, future studies may benefit from incorporating class-balancing strategies, such as focal loss or synthetic oversampling, particularly for low-resource clause types. Inference models like ContractNLI may also see improvements through hybrid pipelines that combine LLMs with symbolic logic, retrieval-augmented generation (RAG), or external fact-checking systems. Multi-task learning frameworks that jointly model clause extraction, classification, and entailment could further unlock the shared legal semantics across these subtasks.

Beyond modeling techniques, future work should broaden the scope of contract types under evaluation. The current project focused primarily on general commercial agreements such as NDAs, employment clauses, and liability statements, as found in CUAD, LEDGAR, and ContractNLI. However, legal practice spans a wider domain including Sales, Distribution & Licensing Agreements, Service & Maintenance Contracts, and Regulatory & Financial Documents. While large language models demonstrate some degree of generalization, their robustness in these domains remains largely untested. Expanding evaluation to these categories could ensure broader applicability of LLMs in contract review systems and help identify new domain-specific challenges.

Finally, future directions should also include systematic error analysis and calibration studies to better understand where models fail and how confident they are in high-stakes scenarios. Incorporating legal expert feedback during validation or using human-in-the-loop evaluation could make model outputs more trustworthy and interpretable for actual legal workflows.

7 Conclusion and limitations

This study explored how transformer-based language models perform across a diverse set of legal tasks, also the study offered a practical snapshot of their strengths and limitations in contract review settings. Instead of one end-to-end pipeline, the project modeled three core subtasks each aligned with legal workflows in the real world.

The evaluation showed models to perform reliably within structured well-annotated tasks like question answering and clause classification, where legal language often follows repeatable patterns. Pretrained LLMs were able to generalize effectively here with minimal adaptation. However, the model performed greatly worse on the ContractNLI task as well as on what it required to logically reason more deeply with the aim to interpret contractual intent. This result points to an important border within current transformer capabilities. Transformers learn well using syntax and patterns, but they cannot resolve contradictions or infer meanings common in legal interpretation.

That said, several limitations shaped the scope and depth of this work. First, each model was trained and was evaluated in isolation through using default task definitions, and they did not explore as to whether alternative problem framings might yield much stronger results. Skewed label distributions introduced class imbalance for some datasets especially LEDGAR, so reports of certain micro/macro metrics had limited reliability.Training's scope was minimal because it ensured consistency and feasibility across tasks, so techniques like hyperparameter tuning, cross-validation, or multi-model ensembling saw no exploration within this iteration. Because of time constraints, deeper error analysis and explainability methods were just not incorporated, and this includes SHAP or attention attribution. These methods could then have provided some more actionable perceptions into model behavior.

In general, the work gives a clear as well as reproducible baseline intended for legal NLP experimentation since it brings into one place three datasets that are widely used plus evaluation strategies tailored into a unified framework. This structure makes comparing model behavior for different task types easier. It helps also to identify the locations where current techniques may fall short. The project surfaces each of these contrasts and invites future work toward building more reliable interpretable systems. These systems could incorporate symbolic reasoning, retrieval-based augmentation, or domain-specific prompt engineering. Since legal

documents increase in complexity and volume, these tools can eventually give real support to lawyers, reducing manual work, improving uniformity and opening avenues to broad contract knowledge.

Appendix

All source code and evaluation scripts are made publicly available for transparency and reproducibility. The complete implementation can be accessed at:

`https://github.com/NikStar2/Nikhil_Srinivasan_MRP`

References

[1] D. Hendrycks, C. Burns, A. Chen, and S. Ball, "CUAD: An Expert-Annotated NLP Dataset for Legal Contract Review," *arXiv preprint arXiv:2103.06268*, 2021.

[2] D. Tuggener, P. von Däniken, T. Peetz, and M. Cieliebak, "LEDGAR: A Large-Scale Multi-label Corpus for Text Classification of Legal Provisions in Contracts," in *Proc. LREC*, 2020, pp. 1235–1241.

[3] I. Chalkidis et al., "LexGLUE: A Benchmark Dataset for Legal Language Understanding in English," in *Proc. ACL*, 2022, pp. 4310–4330.

[4] M. Dechtiar, D. M. Katz, and H. Wang, "Software Engineering Meets Legal Texts: LLMs for Auto Detection of Contract Smells," *Patterns*, vol. 6, no. 5, 2025.

[5] S. Moon, S. Chi, and S. Im, "Automated Detection of Contractual Risk Clauses from Construction Specifications using BERT," *Automation in Construction*, vol. 135, p. 104095, 2022.

[6] J. Breton et al., "Leveraging LLMs for Legal Terms Extraction with Limited Annotated Data," *Artificial Intelligence and Law*, 2025.

[7] D. Shu et al., "LawLLM: Law Large Language Model for the US Legal System," *arXiv preprint arXiv:2407.21065*, 2024.

[8] J. Lai, W. Gan, J. Wu, Z. Qi, and P. S. Yu, "Large Language Models in Law: A Survey," *AI Open*, vol. 5, pp. 185–199, 2024.

[9] A. B. Candaş and O. B. Tokdemir, "Automating Coordination Efforts for Reviewing Construction Contracts with Multilabel Text Classification," *Journal of Legal Affairs and Dispute Resolution in Engineering and Construction*, 2022.

[10] S. Wong, C. Zheng, X. Su, and Y. Tang, "Construction Contract Risk Identification Based on Knowledge-Augmented Language Models," *arXiv preprint arXiv:2309.12626*, 2024.

[11] W. Xu et al., "ConReader: Exploring Implicit Relations in Contracts for Contract Clause Extraction," in *Proc. EMNLP*, 2022, pp. 2420–2434.

[12] V. Aggarwal et al., "ClauseRec: A Clause Recommendation Framework for AI-aided Contract Authoring," in *Proc. EMNLP*, 2021, pp. 8741–8754.

[13] P. G. Bizzaro, E. D. Valentina, N. Mana, M. Napolitano, and M. Zancanaro, "Annotation and Classification of Relevant Clauses in Terms-and-Conditions Contracts," in *Proc. LREC*, 2024.

[14] E. W. Kim, Y. J. Shin, K. J. Kim, and S. Kwon, "Development of an Automated Construction Contract Review Framework Using LLM and Domain Knowledge," *Buildings*, vol. 15, no. 6, 2025.

[15] S. Prasad et al., "Evaluating Large Language Models on Contractual Reasoning," *arXiv preprint arXiv:2310.03466*, 2023.

[16] K. Komatsuzaki, M. Koyama, and Y. Matsuo, "ContractNLI: A Dataset for Document-Level Natural Language Inference for Contracts," in *Proc. ACL*, 2021.