Enhancing Contract Negotiations with LLM-Based Legal Document Comparison

Savinay Narendra, Kaushal Shetty, Adwait Ratnaparkhi
Machine Learning Center of Excellence, JPMorgan Chase & Co.
{savinay.narendra, kaushal.shetty, adwait.ratnaparkhi}@jpmchase.com

Abstract

We present a large language model (LLM) based approach for comparing legal contracts with their corresponding template documents. Legal professionals use commonly observed deviations between templates and contracts to help with contract negotiations, and also to refine the template documents. Our comparison approach, based on the well-studied natural language inference (NLI) task, first splits a template into key concepts and then uses LLMs to decide if the concepts are entailed by the contract document. We also repeat this procedure in the opposite direction — contract clauses are tested for entailment against the template clause to see if they contain additional information. The non-entailed concepts are labelled, organized and filtered by frequency, and placed into a *clause library*, which is used to suggest changes to the template documents. We first show that our LLM-based approach outperforms all previous work on a publicly available dataset designed for NLI in the legal domain. We then apply it to a private real-world legal dataset, achieve an accuracy of 96.46%. Our approach is the first in the literature to produce a natural language comparison between legal contracts and their template documents.

1 Introduction

In the dynamic landscape of contract management, the ability to efficiently negotiate, draft, and manage contracts is paramount for organizations seeking to mitigate risks and streamline operations. This paper explores a comprehensive approach to enhancing contract management processes through the implementation of systematic clause variation analysis, which can be further used to create pre-negotiated Master Service Agreements (MSAs), advanced contract classification and summarization techniques. By leveraging historical contract data and automating key aspects of contract management, organizations can significantly reduce negotiation time frames and improve the consistency and quality of their contractual agreements.

Our work includes several key components aimed at improving contract management through the use of advanced language models:

Demonstrating the Performance of Large Language Models for Natural Language Inference Tasks: We investigate the efficacy of large language models (LLMs) such as Mixtral and GPT-4 in performing Natural Language Inference (NLI) tasks on the contractNLI dataset (Koreeda and Manning, 2021a). This involves not only assessing the models' ability to understand and infer contractual language but also identifying evidence for each NLI task. By demonstrating the superior performance of these models when compared to (Koreeda and Manning, 2021b), we aim to highlight their potential in automating complex contract analysis tasks, thereby enhancing the efficiency and accuracy of contract management processes. The ability of these models to accurately perform NLI tasks is crucial for understanding the nuances and implications of various contract clauses, which in turn supports more informed decision-making during contract negotiations.

Discover Clause Variations: We present the first approach using LLMs to develop clause comparison of contracts agreements with respect to the template agreement as an NLI task. This can be further used to create a comprehensive catalog of approved contract terms based on historical contracts. We explore the application of LLMs in contract management, particularly in reviewing contracts against a template to compare clause variations. To facilitate this, we developed a Retrieval Augmented Generation (RAG) pipeline, which enhances the ability to retrieve relevant clauses and generate appropriate variations. This enables organizations to maintain a high level of consistency and compliance in their contractual agreements, while also speeding up the negotiation process.

Proceedings of the Natural Legal Language Processing Workshop 2024, pages 143–153

We also show how to use LLMs to modify master contracts by incorporating amendments. This involves leveraging the capabilities of advanced language models to automatically generate and integrate amendments into existing contracts. Through these initiatives, our research aims to provide a robust framework for leveraging advanced language models and historical contract data to enhance the efficiency, consistency, and quality of contract management processes. By automating key aspects of contract analysis and negotiation, organizations can achieve significant improvements in operational efficiency and risk mitigation. This paper demonstrates how the integration of LLMs into contract management can transform traditional practices, leading to more streamlined and effective contract lifecycle management.

2 Related Work

Legal contracts are characterized by their intricate logical structures, specialized vocabulary, and the necessity for precise interpretation. The ability to perform document-level Natural Language Inference (NLI) in this context is crucial for various applications, including contract review, compliance checking, and automated legal reasoning. However, existing NLI datasets and models are not well-suited for these tasks, as they are primarily designed for sentence-level inference and lack the context and complexity of full documents.

Reviewing a contract is a time-consuming and complex process that incurs large expenses for companies. To address this gap, (Koreeda and Manning, 2021b)) introduced ContractNLI: A Dataset for Document-level Natural Language Inference for Contracts. This is further discussed in Section 3.1. The task involves using a Span NLI BERT model to classify whether each hypothesis (a sentence) is entailed by, contradicts, or is not mentioned by (neutral to) the contract, and to identify evidence for the decision as spans in the contract. The Span NLI BERT performed significantly better than existing Transformer-based models in terms of NLI. Our task closely parallels their problem statement, as we aim to determine whether each clause in the template agreement is covered (entails or contradicts) or not covered (neutral) in the contract agreement.

The application of large language models (LLMs) in the context of legal contracts has been extensively explored by (Roegiest et al., 2023). Their problem setup involves legal questions with several answer options, focusing on structured answers rather than generating free text. They employ an embedding-based approach to predict the answer option with the highest similarity to the question text and develop question-specific prompts, eventually landing on a smaller set of reusable prompt templates.

(Lam et al., 2023) present a multi-step method for drafting contract clauses, which includes comparing an input clause to clauses in a trusted repository to yield a set of similar clauses, extracting keyphrase vectors, and clustering these vectors to provide suggestions for modifying the input clause. This method uses the LEDGAR dataset of SEC filings as the trusted repository, offering a robust framework for clause comparison and modification. LegalBench, introduced by (Guha et al., 2024), is a benchmark constructed through a collaborative effort involving legal experts, NLP researchers, and practitioners. LegalBench includes a diverse set of tasks covering various aspects of legal reasoning, from understanding and interpreting legal texts to applying legal principles in specific contexts. This benchmark represents a significant advancement in the intersection of NLP and legal technology, enabling systematic evaluation and comparison of LLMs on legal reasoning tasks and facilitating the development of more sophisticated models tailored to the needs of the legal profession.

Our work differs from the aforementioned studies in several key aspects. While previous research has focused on sentence-level NLI, structured question answering, clause drafting, and benchmarking legal reasoning tasks, our approach is the first to leverage LLMs for the direct comparison of legal contracts with their corresponding template documents. By splitting both templates and contracts into sub-clauses and using LLMs to determine entailment in both directions, we create a comprehensive clause library that aids in refining template documents and assisting in contract negotiations. Our method not only outperforms existing models on a publicly available NLI dataset in the legal domain but also demonstrates high accuracy on a private real-world legal dataset, showcasing its practical applicability and effectiveness.

3 Datasets

3.1 ContractNLI

Before applying large language models (LLMs) to our internal dataset, we wanted to experiment with an external dataset to evaluate their effectiveness and potential. Hence, we utilized the ContractNLI dataset, designed for document-level natural language inference (NLI) specifically tailored to contracts, aiming to automate and support the labor-intensive process of contract review. It is the first dataset to apply NLI to contracts and is the largest annotated corpus of its kind as of September 2021. The dataset includes 607 non-disclosure agreements (NDAs), each annotated with 17 fixed hypotheses, resulting in a substantial corpus for training and evaluating NLI models. The primary tasks involve classifying each hypothesis as Entailment, Contradiction, or NotMentioned, and identifying evidence spans for Entailment and Contradiction labels. For evidence extraction, we need to identify a list of exact spans from the dataset that either contradict or entail the hypothesis, based on the label. This is applicable only when the NLI label is Entailment or Contradiction. The ContractNLI dataset includes evidence as a list of span indices. Each index in the array corresponds to a span where the hypothesis either entails or contradicts the span in the contract.

3.2 Internal Dataset

The internal dataset consists of 25 master contracts, which serve as the primary documents for our analysis. Out of these 25 master contracts, 5 include associated amendments. These amendments reflect changes or additions to the original contract terms, offering a richer context for understanding the evolution of contractual agreements over time. The contracts in the dataset span a significant temporal range, with effective dates ranging from June 2007 to August 2023. This extensive timeframe allows for the examination of contractual language and practices over a period of more than 15 years, providing insights into how contract terms and structures have evolved.

The dataset includes a diverse array of contract types, reflecting the various agreements between JP Morgan and its suppliers. These contract types are:

1. Software and Maintenance Agreement
2. Professional Services Agreement
3. Software License Agreement
4. Application Service Provider Agreement
5. Hardware Agreement

We systematically segmented each clause from the template into distinct key concepts. Subsequently, we employed these segmented concepts within a natural language inference (NLI) framework. In this framework, each key concept from the template was treated as a hypothesis, while the entire contract document was considered the premise. The objective was to predict whether the contract document either contradicts, entails or remains neutral towards the given concept/hypothesis. Additionally, we performed a reverse analysis in which each key concept from the contract clauses were compared against the template document, to identify concepts in the contract that were not covered in the template.

4 Motivation

Contract review is a very labor-intensive process and there is a growing need to streamline and automate the process of contract review, which is critical in legal and business environments. Traditional methods of contract analysis are time-consuming, prone to human error, and often require significant expertise. Contract review involves meticulously reading through lengthy and complex documents to identify key clauses, obligations, exceptions, and potential risks. This process demands a deep understanding of legal language and the ability to interpret nuanced terms and conditions, which can vary significantly between contracts. Additionally, the need to cross-reference multiple documents and ensure compliance with relevant laws and regulations further complicates the task. By leveraging advanced natural language processing (NLP) techniques, specifically large language models (LLMs), we aim to enhance the efficiency and accuracy of contract review. Our initial experiments with the ContractNLI dataset provide a valuable opportunity to assess the capabilities of LLMs in handling complex legal language and inference tasks. This research not only contributes to the field of NLP by addressing the unique challenges posed by legal documents but also has practical implications for improving contract management processes in various industries.

5 Experiments on ContractNLI Dataset

In our experiments, we explored the application of large language models to the ContractNLI dataset, focusing on two primary tasks: (1) classifying the relationship between a given contract and a set of hypotheses, and (2) identifying evidence within the contract that supports the classification decision. To guide the models' responses, we employed specific prompts tailored to each task. We tested the performance of both commercial and open-source models, including the GPT-4 model, which is accessed via a commercial API. GPT-4 (OpenAI et al., 2024) is a large-scale, multimodal model that exhibits human-level performance on various professional and academic benchmarks.

Additionally, we fine-tuned the Mixtral 8x7B (Jiang et al., 2024), a Sparse Mixture of Experts (SMoE) language model, which combines multiple expert networks to improve performance while maintaining efficiency. We chose this Mixtral model as it was one of the open-source models available at that time with demonstrated superior performance and reduced inference costs.

To fine-tune the Mixtral model, we employed Low-Rank Adaptation (LoRA) (Hu et al., 2021), which freezes the pre-trained model weights and injects trainable rank decomposition matrices into each layer of the Transformer architecture, greatly reducing the number of trainable parameters for downstream tasks.

We maintained consistent training parameters for both tasks. Specifically, we used the following settings that included a per-device training batch size to 1 and used gradient accumulation to effectively manage memory usage, with accumulation steps also set to 1. Gradient check-pointing was enabled to further conserve memory during training. The total number of training steps was capped at 4000. A small learning rate of 2.5e-5 was selected to ensure stable and gradual fine-tuning of the model. Training was conducted using bf16 precision to optimize computational resources.

Here are the explanations for the two tasks within the ContractNLI dataset.

Natural Language Inference The first task involves classifying the relationship between a contract and a set of hypotheses. Each hypothesis is a single sentence, and the goal is to determine whether the hypothesis is entailed by, contradicts, or is neutral with respect to the contract.

Evidence Extraction The second task focuses on identifying evidence within the contract that supports the classification decision made in the first task.

5.1 Prompts

Here are the prompts that we used for the two tasks described above respectively. We used the same prompts between the two large language models to ensure consistency in inference over the test set. Prompt 1 is used for NLI and prompt 2 is used for evidence identification.

1. Given a document and a hypothesis, determine whether the document entails or contradicts the hypothesis. Answer strictly as "Entailment" or "Contradiction"

2. Given a document and a hypothesis, if the label is 'Entailment' extract evidence verbatim from the document that support the hypothesis. If the label is 'Contradiction', extract evidence verbatim from the document that contradicts the hypothesis \n Evidence:

In the ContractNLI dataset, we did not evaluate whether the hypothesis and the contract are neutral to each other, as our focus was on evidence extraction based on NLI results, applicable only when the NLI label is Entailment or Contradiction.

5.2 Results

	F1(C)	F1(E)	Acc.
GPT-4	0.70	0.91	
	-	-	0.87
Mixtral	0.74	0.93	
	-	-	0.90
Span NLI BERT	0.389	0.839	
	-	-	0.87

Table 1: Comparison of GPT-4(OpenAI et al., 2024), Mixtral 8x7B(Jiang et al., 2024) and Span NLI Bert(Koreeda and Manning, 2021b) on NLI task for ContractNLI test dataset.

In the table above, C refers to Contradiction label while E refers to Entailment Label. The dataset contains a significantly smaller number of instances labeled as Contradiction. We We observe that GPT-4 and Mixtral model achieves a significantly higher F1 score on the Contradiction label compared to the Span NLI BERT model (Koreeda and Manning, 2021b). Additionally, both the LLMs demonstrates superior performance in

calculating the F1 score for the Entailment label. In the ContractNLI dataset, we conducted NLI in just one direction, assessing whether the hypothesis contradicts or entails a given contract, as we aimed to compare how LLMs would outperform the results of (Koreeda and Manning, 2021b)

Model	Mean Average Precision
GPT 4	92.68%
Mixtral	79.8%
Span NLI Bert	92.2%

Table 2: Comparison of performance of GPT-4, Mixtral and Span NLI BERT on evidence identification for ContractNLI test dataset

We observe that GPT-4 model also achieve superior performance on Evidence Identification as compared to the fine-tuned Span NLI Bert model. The mean average precision for Evidence Identification is calculated by averaging the precision across each evidence predicted by the model with respect to the true evidence for that instance at each recall level where a relevant token is retrieved.

6 Proposed solution for Internal Dataset

In our internal problem setting, we are tasked with comparing a negotiated contract against a pre-established template for the contract. These contracts frequently undergo several amendments that add, delete, or modify the original clauses. This scenario closely resembles a Natural Language Inference (NLI) task, wherein we seek to determine whether each concept (hypothesis) in the template clauses is either covered (contradicted or entailed) or not covered (neutral) in relation to the contract agreement. Additionally, since the documents are often available as scanned PDFs, we must explore OCR solutions to accurately convert them into text for further analysis.

One of the main challenges we faced was that many of the documents were images embedded in PDF files, making it difficult to extract and segment the text based on sections. Our initial experiments using Tesseract-OCR were unsuccessful due to errors introduced during OCR and the difficulty of segmenting free-flowing text without clear delimiters. To address this, we used a document image transformer model capable of identifying sections using boundaries and then performing OCR on the bounded boxes. Once the text from each section was extracted, we used GPT-4 model using tailored prompts to extract the correct clauses and compare them with template clauses. This approach allowed us to effectively process and analyze the complex legal language and structure of the contracts, demonstrating the potential of LLMs in automating and enhancing the contract review process.

6.1 PDF Extraction using OCR

The input documents for our tool were PDF documents, and we begin with extracting text from these PDFs. Traditional PDF extraction tools proved inadequate because the PDFs contained text embedded as images. Consequently, we could not rely on regular extraction methods. To address this challenge, we explored two distinct approaches. The first approach involved using Tesseract OCR, while the second approach utilized a Document Image Transformer (DiT) model combined with EasyOCR.

6.1.1 Tesseract OCR

Traditional OCR tools like Tesseract (Smith, 2007) have been widely used for text extraction from various document formats. However, when dealing with PDFs where text is embedded as images, several limitations become apparent including high character error rate, lack of document segmentation and scalability issues.

6.1.2 Document Image Transformer

To address the limitations of traditional OCR tools, we explored the use of a Document Image Transformer (DiT) model(Li et al., 2022). This model serves as the backbone network for a variety of vision-based Document AI tasks, including document image classification, layout analysis, table detection, and text detection for OCR.

Bounding Box Identification: The first step in our approach involved using the DiT model to identify bounding boxes for each section of the document. This segmentation process is crucial for accurately isolating different parts of the document, accommodating the diverse styles and layouts found in image-embedded PDFs. The DiT model's self-supervised pre-training enables it to achieve high accuracy in this task, setting the stage for effective text extraction.

Text Extraction with EasyOCR: Once the sections were identified, we utilized EasyOCR (Baek et al., 2019; Shi et al., 2015), an open-source OCR engine, to extract text from each bounding

box. EasyOCR's robust text recognition capabilities complement the DiT model's segmentation, resulting in a more reliable extraction process. By focusing on smaller, well-defined sections, EasyOCR can achieve higher accuracy compared to processing entire pages at once.

6.2 Large Language Models

Clause Variability Analysis One of the primary tasks in our experiments was to identify the variabilities of specific clauses in the master contract agreements compared to the template master agreements. The clauses analyzed include Limitations of Liability, Insurance, Indemnity, Representations and Warranties, Red Flags, System Modifications, Assignment, Source Code Escrow and Audits.

By comparing these clauses between the master agreements and the template agreements, we aimed to understand the common deviations and variations that occur during contract negotiations and amendments.

6.2.1 Handling Amendments with GPT-4

For contracts that include amendments, we created modified contracts by incorporating all the amendments into the original master agreements. One key observation was that GPT-4 requires very specific context to accurately amend the original master contract agreement. To address this, we employed intelligent chunking of the document using a fine-tuned Document Information Transformer (DiT) model, which helped in breaking down the document into various subsections. The process involved the following steps:

Summarizing Amendments: First, a summary of the amendment document was created to capture all the sections and subsections that needed modification using prompt 1 in Appendix. The amendment was essential to isolate and focus solely on the modified sections of the document. This approach aims to eliminate extraneous information, thereby reducing the potential for errors within the model.

Extracting Key Data: Upon extracting the relevant sections and associated text from the amendments in JSON format, the modified master contract, incorporating these amendments, was generated using prompts 2 and 3 in Appendix.

Concept Extraction from Template Clauses: To further analyze the clauses, we divided the template master agreements into multiple concepts or hypotheses using the prompt 4 in Appendix. This step allowed us to break down each template clause into its fundamental concepts, making it easier to compare and analyze against the master agreements.

The term "concept" refers to a specific segment of the original clause, maintaining the integrity and context of the clause. Each clause is divided into multiple concepts. A sample concept/sub-clause generated from the template agreement for the "Red Flags" clause using GPT-4 is shown in table 3.

6.2.2 Retrieval Augmented Generation (RAG) Pipeline

Once the concepts were extracted from each template clause, we implemented a Retrieval Augmented Generation (RAG) pipeline in figure 1 to ask question to the document for each concept in template clause using prompt 5.

For each chunk retrieved in response to the above question, cross-references to other sections were appended to the chunk. This approach ensured that we could accurately determine whether each concept was present in the contract document, providing a comprehensive analysis of clause coverage and variability.

We also did a reverse comparison in which we asked the following question as specified in prompt 6 to find out if there are any additional concepts mentioned in the contract clause not included in the template contract.

These experiments with large language models, particularly GPT-4, demonstrated the importance of providing specific context and intelligent document chunking to accurately amend and analyze contracts. By leveraging advanced NLP techniques and fine-tuned models, we were able to systematically identify clause variabilities, handle amendments, and extract key concepts, thereby enhancing our understanding and management of contractual agreements.

6.3 Prompts

All the prompts used in our work can be found at Appendix A. One of the most challenging aspects of contract review was the incorporation of multiple amendments into the master contract. To address this challenge, we utilized the GPT-4 model to summarize each amendment. The model was prompted to generate output in JSON format, specifying the parent section, the child section, and

"Red Flags" Clause	Sub-Clauses/Concepts
Whenever the Deliverables set forth in ... Supplier having unencrypted ... that contains consumer information, Supplier will have policies and procedures in order to detect ... , practices, or other specific activity that indicates the possible existence of identity theft (“Red Flags”) and will either report the Red Flags to ... prevent or mitigate identity theft.	• Deliverables may include Supplier having unencrypted ... containing consumer information. • Supplier must have policies and procedures to detect, ... identity theft indicators ("Red Flags"). • Supplier is responsible ... to prevent or mitigate identity theft.

Table 3: Red Flags Clause Concept Extraction

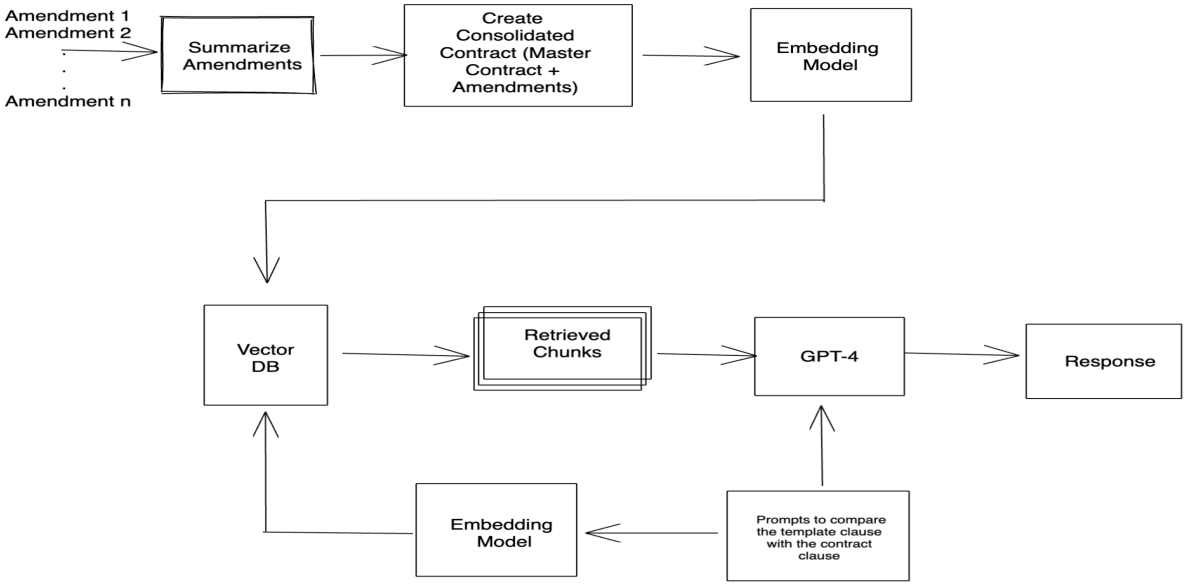

Figure 1: Pipeline to find deviations between Template Clause and Contract Clause

the verbatim text to be added or deleted from the master contract.

7 Results

The clause variations generated by the GPT-4 model using the pipeline in Figure 1 on the internal dataset were annotated by the annotation team at JP Morgan. The annotators are experienced with handling legal documentation, but may not be able to judge the output at the level of a trained lawyer. The quality of the annotations is deemed sufficient for practical applications. On our internal dataset, the model achieved an accuracy of 96.46%. The accuracy is determined by dividing the total number of correctly identified concepts within each clause by the model, based on their classification as entailed, contradicted, or neutral with respect to the contract document. Refer to Figure 2 for performace of the model on each clause across the dataset.

7.1 Sample Outputs

7.1.1 Comparing Concept in Template Clause with Contract Clause

Here, we show a sample clause variation to determine whether the concept in the template clause "Representations and Warranties" is entailed, contradicted, or neutral with respect to the corresponding clause in the contract. The output from the model offers a natural language explanation of the similarities(entailment) and differences(contradiction) between the template agreement and contract agreement. Please refer to table 4.

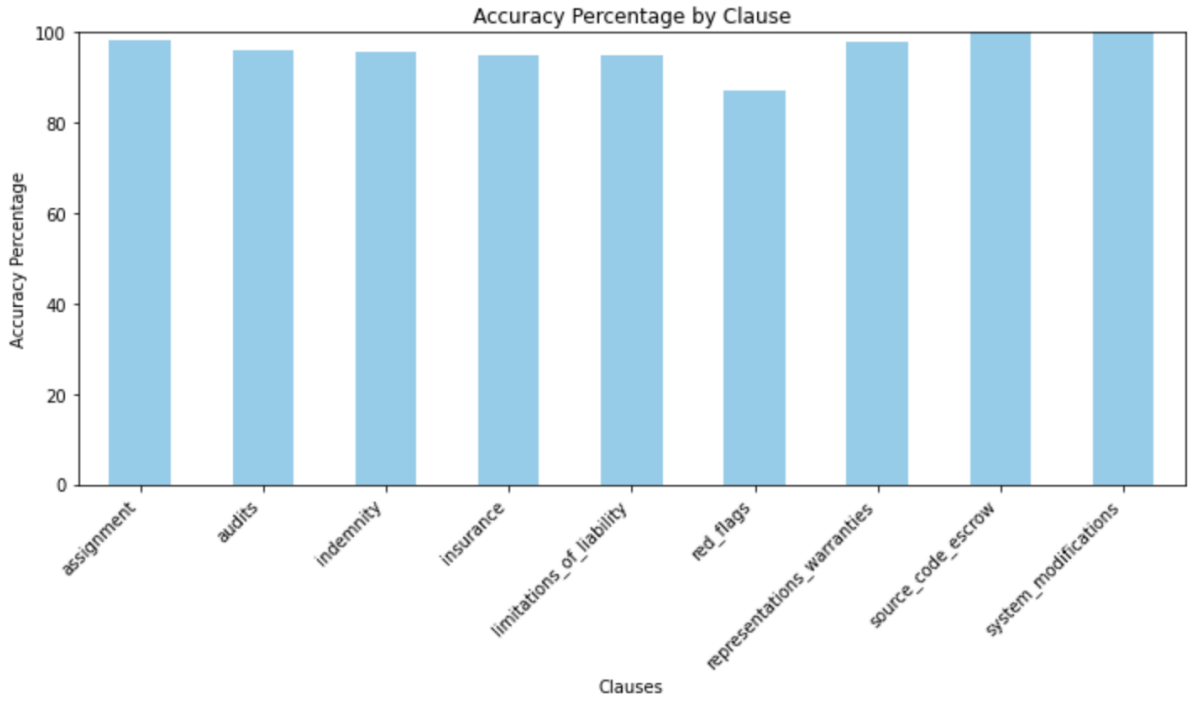

Figure 2: Accuracy Percentage by Clause using GPT-4

The model also generates output that assists a lawyer in identifying any additional sub-clauses present in the contract but absent from the template.

8 Discussion

Our experiments and results on the ContractNLI dataset reveal that both the GPT-4 and Mixtral models outperform the SpanNLI BERT model introduced by Koreeda and Manning (2021) on Natural Language Inference (NLI) tasks. For NLI, GPT-4 achieved an F1 score of 0.91 on the Entailment label, compared to 0.834 by the SpanNLI BERT model. Additionally, GPT-4 attained an F1 score of 0.70 on the Contradiction label, significantly higher than the 0.357 achieved by the SpanNLI BERT model. For evidence identification, GPT-4 achieved slightly superior performance than the Span NLI Bert model and demonstrated a mean average precision of 92.68%. These results demonstrate the potential of leveraging state-of-the-art language models for enhancing the accuracy and reliability of NLI tasks and evidence identification in legal contexts.

The results on the internal dataset demonstrate the efficacy of Large Language Models (GPT-4) in generating outputs that are highly accurate when applied to our internal dataset. The model achieved an impressive accuracy of 96.46%, indicating a high level of precision in its outputs and can be used as a very effective tool by legal professionals in comparing contracts and simplifying the contract negotiation process. The insights provided by the model can help in informed decision-making, further improving the quality of contract review. Additionally, the analysis of clause variations enabled the creation of a clause library, focusing on the most frequent additions and deletions of subclauses/concepts. Due to the highly confidential nature of the data, we are unable to disclose the specific modified clauses.

The integration of the GPT-4 model into the contract negotiation process has the potential to significantly improve outcomes and streamline operations for legal professionals. However, one potential risk in our study is that, although the annotators are experienced in handling legal documentation, their evaluations may lack the nuanced judgment and expertise of a trained lawyer. This could potentially impact the accuracy and reliability of the annotations, particularly in complex legal scenarios.

9 Conclusion

In this study, we explore the efficacy of our novel approach on the Contract-NLI dataset, a publicly available dataset designed for natural language inference and evidence identification within legal

Concept from template	The deliverables will not contain any malware, .. malicious programs and will not store any data on computers, systems, or network.
GPT-4 output from comparing template concept to contract	"The document does not explicitly state that the deliverables will not contain any malware, .. will not store any data on computers, systems, or network. However, it does mention that the supplier will comply with certain security and risk management policies, and that the supplier is responsible for assessing and remediating security vulnerabilities."
Sources from the Contract	• "5.10 Application Security." • "5.5 Critical Vulnerabilities."

Table 4: Sample Clause Variation Comparing a Concept in the Template Clause "Representations and Warranties" with the Contract

contracts. Our methodology demonstrates superior performance compared to all previously established techniques for NLI task and Evidence Identification on the Contract-NLI dataset.

We introduce the first approach that leverages large language models (LLMs) to generate natural language comparisons between legal contracts and their corresponding templates, conceptualized similarly to a natural language inference (NLI) problem on the internal dataset, where we have achieved high accuracy. Additionally, we illustrate the capability of LLMs to perform comparative analysis against both the source text and the text of citations cross-referenced elsewhere in the document.

Our approach involves framing the comparisons as an NLI problem, thereby enabling a more structured and interpretable analysis. The results indicate that our approach not only outperforms existing methods on the Contract-NLI dataset but also provides a robust framework for the natural language comparison of legal documents. The implications of these findings suggest significant advancements in the automation of legal document analysis and the potential for broader applications in the legal domain.

References

Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. 2019. Character region awareness for text detection.

Neel Guha, Julian Nyarko, Daniel Ho, Christopher Ré, Adam Chilton, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, et al. 2024. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Advances in Neural Information Processing Systems*, 36.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts.

Yuta Koreeda and Christopher Manning. 2021a. ContractNLI: A dataset for document-level natural language inference for contracts. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1907–1919, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yuta Koreeda and Christopher D Manning. 2021b. Contractnli: A dataset for document-level natural language inference for contracts. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1907–1919.

Kwok-Yan Lam, Victor CW Cheng, and Zee Kin Yeong. 2023. Applying large language models for enhancing contract drafting. In *Proceedings of the Third International Workshop on Artificial Intelligence and Intelligent Assistance for Legal Professionals in the Digital Workspace (LegalAIIA 2023)*.

Junlong Li, Yiheng Xu, Tengchao Lv, Lei Cui, Cha Zhang, and Furu Wei. 2022. Dit: Self-supervised pre-training for document image transformer.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello . . . , and Barret Zoph. 2024. Gpt-4 technical report.

Adam Roegiest, Radha Chitta, Jonathan Donnelly, Maya Lash, Alexandra Vtyurina, and Francois Longtin. 2023. Questions about contracts: Prompt templates for structured answer generation. In *Proceedings of the Natural Legal Language Processing Workshop 2023*, pages 62–72.

Baoguang Shi, Xiang Bai, and Cong Yao. 2015. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition.

R. Smith. 2007. An overview of the tesseract ocr engine. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, volume 2, pages 629–633.

A Appendix A: Prompts

1. You are a US attorney that reviews the amendments made to a master agreement and modifies the master agreement based on that.

   MASTER AGREEMENT:

   {master_agreement}

   Edit the master agreement with the changes in the following amendment compared to master agreement. Only edit the master agreement. Follow the instructions in the amendment below to modify the master agreement. Add the amendments text to the relevant sections verbatim. If the amendment instructs to add the text, add it to the relevant section in the master agreement at the appropriate position. Figure out where the amendment should be made and then add it at the relevant position.

   AMENDMENT:

   {amendment}

   OUTPUT:

   {{amended_master_agreement}}

   Strictly follow the instructions below to produce the output:

   If the amendment is not at all related to the text in the master agreement, only output the master agreement as it is.

   (a) Only output the modified master agreement.

   (b) Do not make up facts.

   (c) Do not add the prompt text to the final output.

   (d) Do not add reason to the final output on how the output was generated.

2. You are a US attorney that works on extracting the amendments from the document below that need to be amended in the master agreement.

   Extract the exact section number where the modification has to take place in the original document, the text that needs to be replaced and the modified text verbatim in a RFC8259 compliant JSON format. Sections are identified with numbers. Include the section header in `parent_section_no` and `child_section_no`. Do not include any explanation or comment.

   AMENDMENT DOCUMENT:

   `{amendment}`

   The output should be strictly in the format as below without any comments. The output is RFC8259 compliant JSON. Follow the below format strictly. Do not add any comment to the answer. Only return the JSON.

   ```
   [[parent_section_no: , parent_title: , child_section_no:, child_title:, amendment_text: , parent_section_no: , parent_title: ,child_section_no:, child_title:, amendment_text: ]]
   ```

   The `parent_section_no` is the parent section number that needs to be modified in the master agreement. The `parent_title` is the title of the parent section number that needs to be modified in the master agreement. The `child_section_no` is the child section number that needs to be modified in the master agreement. The `child_title` is the title of the child section number that needs to be modified in the master agreement.

   HERE IS AN EXAMPLE OF HOW THE FINAL JSON OUTPUT SHOULD LOOK LIKE:

   AMENDMENT DOCUMENT:

   Section 2, Indemnity is hereby amended as follows:

   The first paragraph of Section 2.2, Indirect Damages, is hereby deleted and replaced with

the following: «amendment_text»

OUTPUT:

parent_section_no: «2», parent_title: «Indemnity» , child_section_no: «.2», child_title: «Indirect Damages» , amendment_text: «amendment_text»" parent_section_no: «3» parent_title:«Communications» , child_section_no: «(g)», child_title:«Publicity» , amendment_text: «amendment_text»" parent_section_no: «», parent_title: «Pricing Schedule Exhibit» , child_section_no: «», child_title: «» , amendment_text: «amendment_text»"

INSTRUCTIONS WHILE CREATING THE OUTPUT:

- In cases, when there are section numbers specified, extract the section header and add it to parent_section_no.
- Do not add the list item numbers in the document as parent_section_no.
- Create a RFC8259 compliant JSON.
- Check for double quotes (") in amendment_text key and replace them with single quotes.

3. Given the document below, the section number and the title, determine whether this is the right section where the chunk should be added. Return True if this is the document where the chunk should be added, else return False.

   Information:

   Parent Section Number: {parent_section_number}
   Child Section Number: {child_section_number}
   Parent title: {parent_title}
   Child title: {child_title}
   Document Chunk: {chunk}

4. You are a US attorney that helps your clients extract key and broad concepts from the clauses.

   Only extract key and broad points from the template clause below each separated by a new line. Each bulleted point mentioned is a single concept. Include all key points within each bulleted point.

   Template Clause: {template_clause}

5. Is the following concept covered within the document? ALWAYS return a "SOURCES" part in your answer. Don't try to make up an answer.

   CONCEPT: {question} {section_text} ———————— FINAL ANSWER:
   SOURCES:

6. Based on the following key points below from the template, answer the following question. ALWAYS return a "SOURCES" part in your answer.

   If the answer is "Yes" and there is additional information in the contract document not included in the template, include the "SUB CLAUSE" from the contract which is included else include "NA" in "SUB CLAUSE".
   QUESTION: What additional information is in the contract clause {key} that is not included in the template concepts below?
   ALL CONCEPTS: {all_template_concepts}
   ————————

   FINAL ANSWER:
   SOURCES:
   SUB CLAUSE: