# Metadata Extraction from Legal Contracts Using Large Language Models

Mrs. Sujata Khedkar#1, Tasmiya Sarfaraz Khan#2, Srushti Satish Sambare#3,

Purtee Santosh Mahajan#4, Ketaki Dhananjay Nalawade#5

*#Computer Engineering Department, Vivekanand Education Society's Institute of Technology, Chembur, Mumbai - 400071, India*

[1]sujata.khedkar@ves.ac.in

[2]2021.tasmiya.khan@ves.ac.in

[3]2021.srushti.sambare@ves.ac.in

[4]2021.purtee.mahajan@ves.ac.in

[5]2021.ketaki.nalawade@ves.ac.in

*Abstract*— **Contracts are formal agreements with legal force that govern interactions between people or entities. Different sections and clauses of the contracts are defined regarding obligations, rights, and responsibilities of each party involved. Due to the complexity as well as the volume of legal contracts, extracting key metadata is highly important in ensuring efficiency for legal professionals. This paper addresses the use of Large Language Models to automate the extraction of core metadata from legal contracts, including clauses such as liability, insurance, and jurisdiction, and organizing it into a structured, tabular format for easier analysis. We make use of the OpenAI Azure LLM to produce a full-stack solution, able to parse and structure this metadata into a format that is usable. The study addresses problems related to large prompt sizes and various optimization strategies that could be applied to such systems in order to attain better efficiency and accuracy. In addition, we include real-time FAQ generation from the legal contracts. The goal behind this research therefore is to verify the potential that LLMs may bring to streamline legal workflows while ideally addressing technical and operational challenges found in large-scale metadata extraction tasks.**

*Keywords*— Legal Contracts, LLM, Clauses, FAQ, Metadata.

## I. Introduction

A contract is a legally binding agreement that specifies the terms and conditions of the parties involved and is signed by at least two of them. Since contracts are usually written in text, there is a lot of room for NLP applications in the field of legal papers. However, contract language is repetitive with high inter-sentence similarities and sentence matches, in contrast to most natural language corpora that are typically used in NLP research. [1]

In the legal domain, collecting information from contracts poses three major hurdles for existing methods. The primary difficulty is the scarcity of data needed to train or fine-tune algorithms for high accuracy. Another difficulty is the large size of many contracts, which frequently exceeds the processing capacity of current transformer topologies. Transformer-based models have a limited sequence length that they can accommodate. Contracts that exceed the limit may need to be broken into smaller portions, complicating the analyzing process. The third challenge is that contracts contain a mix of long and short entities, such as full clauses like non-complete and audit rights, and short ones like names and dates. [2]

Metadata extraction is the process of identifying and extracting specific data points from unstructured documents, and is becoming increasingly important for legal professionals. This is particularly true for clauses related to liability, insurance, indemnity, termination, and jurisdiction, which are crucial for understanding the legal implications of contracts. Manually reviewing large, complex contracts for specific clauses can be time-consuming and error-prone. By extracting information about liability, insurance, indemnity, termination, and jurisdiction, etc legal professionals can better assess the potential risks associated with a contract.

To overcome these challenges, in this paper we proposed to extract metadata from an unstructured format to a structured format using LLM. This metadata will be formatted into a table. We proposed four tables: a quarterly report, which includes various clauses and important entities; a termination table, which includes all the termination clauses; an indemnification table, which includes all the indemnification clauses; and a basic table, which includes all important dates from the contracts and party names.

## II. Related work

This paper proposes an unsupervised two-step approach for extracting knowledge from long legal contracts. It overcomes the token limitation in the basic LLMs and the lack of training data. As stated in the abstract, authors propose a query-based summarization model that extracts relevant sentences, thus reducing the text size without losing any core information. The summary is fed into GPT-3.5 to generate accurate metadata extraction. This method does not rely on the necessity of training domain-specific models unlike supervised models but depends on the pre-trained capabilities of LLMs which it uses to extract short and lengthy contract entities without fine-tuning. The performance exhibited was better and brought about state-of-the-art results in zero-resource settings, especially when contrasted with fine-tuned supervised models on domain-specific data. [2] This paper explores the application of Large Language Models (LLMs) in law by designing generative AI to optimize contract management. As already mentioned in the abstract, it uses Retrieval-Augmented Generation (RAG) to improve the analyzing ability and generate legal clauses of LLMs. The system uses semantic embeddings for accurate document retrieval and generation of contract clauses that are contextually relevant. It also employs Azure OpenAI models using alertness of prompting optimization to solve issues on contract complexity; it makes use of retrieval as well as LLMs to reduce the time lawyers take in reading and analyzing a contract, thereby making it possible to draft new contracts fast and accurately. As part of future work, the paper mentions the implementation of Optical Character Recognition (OCR) technology. [3] This will enable the system to handle non-digital legal documents, expanding its application breadth while boosting the depth of the analysis process. It will also enable the system to extract text from scanned papers or PDFs. Optical Character Recognition, or simply OCR, is certainly one of the most important technologies by means of which printed or hand-written text can be scanned and converted to digital versions, saving literally hundreds of hours of time and labor when doing data entry and document management. As highlighted in the paper [4], OCR automates the process of extracting text from various sources, such as images and scanned documents, and converts it into searchable and editable digital content. This technology plays a vital role in industries that handle large volumes of data, enabling organizations to streamline operations, reduce errors, and improve overall productivity.

The paper [7] provides a methodology for automating the reporting of contracts and obligation extraction with LLMs, focusing particularly on filling in the DORA compliance templates and on extracting obligations from the contract. This introduction discusses how the emergence of LLMs like GPT-3 and GPT-4 creates opportunities for new avenues for automation with respect to automating complex tasks like information extraction from legal texts. These models were used in the experiments of this paper as a means of extracting specific data from legal contracts to test their accuracy in compliance and contractual obligation tasks. The system was shown to be highly accurate in filling compliance templates with up to 97.71%, but less so consistent in the extraction of obligations in legal contracts, with up to 70.56% accuracy in some cases. These results showcase the capabilities of LLMs for the purpose of changing the face of legal document management by saving manpower and increasing accuracy. Large language models (LLMs) have found increasing use in the legal sphere, demonstrating how these technologies may be tailored to industry-specific requirements. Prominent models comprise LegalBERT[8], CaseLaw-BERT[9], and FinBERT[15] which have been refined from generic models to tackle assignments including anticipating legal judgments, reviewing contracts, and other domains necessitating an intricate comprehension of legal terminology and concepts. These models demonstrate at least two aspects: the potential for LLMs to increase efficiencies in legal workflows and the role of domain-specific training in reaching almost precise relevance with respect to professional applications

## III. Proposed solution

### A. *System Architecture*

The full-stack solution built around OpenAI's Azure LLM is designed to automate the extraction of key metadata from legal contracts. The system architecture comprises several key components:

#### 1) *Input (Contract Documents):*

Users can upload contract documents in various formats, such as PDF or DOCX. To ensure compatibility and efficient processing, the system extracts the text from these contracts. If the contract is in a non-digital format (e.g., a scanned image), Optical Character Recognition (OCR) is used to extract the text. Tools such as Pytesseract are used to ensure high accuracy when converting image-based contracts into text format.

#### 2) *LLM for Clause Extraction:*

- Once the text is extracted, it is passed through the OpenAI Azure LLM for clause extraction. The LLM is specifically prompted to identify key clauses, such as liability, insurance, and jurisdiction.
- The system uses semantic embeddings and context-aware language models to recognize specific patterns and extract relevant metadata from the contract clauses, which are often complex and filled with legal jargon.
- The LLM processes the contract and returns the extracted metadata in a structured output format.

#### 3) *Structured Output in Tabular Format:*

The extracted metadata is organized into a tabular format for easy review and further analysis. Each row in the table corresponds to a metadata type (e.g., Liability, Insurance, Jurisdiction) with columns containing the extracted information.

Our proposed solution extracts four tables:

Basic Table: This table includes columns such as

Agreement Type, Client, Service Provider and Dates. Dates include:

- Document date: Date when the document was created.
- Effective date: Date when the agreement becomes effective
- End date: Date when the agreement ends.

Termination Table: This table includes:

- Termination clauses: The specific sections or provisions in the contract that outline the conditions under which the agreement can be terminated. This includes reasons for termination such as breach of contract, failure to meet obligations, or termination for convenience. The clauses also specify any notice periods, penalties, or other conditions for ending the contract.
- Clause number
- Who can terminate the agreement: Specifies which party or parties (e.g., Client, Service Provider, or both) have the right to terminate the agreement under the conditions set forth in the termination clause. Some contracts allow only one party to terminate, while others grant termination rights to both parties under certain circumstances.

Report: This table includes Client name:

- Liability: This term refers to the legal responsibility for damages or breaches under the contract. It can be capped at a certain amount or percentage of the contract's value (e.g., Work Order value). If capped, the liability is limited to the specified value mentioned in the "LIMITATION OF LIABILITY" clause. This clause protects one party by restricting their financial obligations in the case of legal claims.
- Uncapped liability: Certain liabilities may be uncapped, meaning they have no limit on the amount of compensation required. Common uncapped liabilities include claims related to death, personal injury, fraud, or breach of law. If liabilities are capped, this section will list the specific types of liabilities that remain uncapped.
- Warranty: Warranties ensure the quality or performance of products or services. If exceptions are present in the warranty (e.g., "except for certain conditions"), they should be noted. Otherwise, it is marked as "Standard" or "Not available" if the warranty section is absent.
- Indemnity: A legal obligation where one party agrees to compensate the other for losses, damages, or liabilities arising from specified actions or circumstances. If exceptions are present (e.g., indemnity does not apply in certain situations), they should be noted. Otherwise, it is marked as "Standard" or "Not available" if the indemnity section is absent.
- Services/Damage: Refers to the obligations regarding services provided under the contract and any potential damages. If the damages are ongoing, this should be noted. A cap may be specified, which limits the amount of compensation for damages, either as a fixed amount or percentage.
- Jurisdiction/Law: The geographic location (country and city) whose laws govern the contract and where disputes arising under the contract will be resolved.
- Termination for convenience:A clause allowing one party, typically the service provider or client, to terminate the contract without cause. This provides flexibility to exit the agreement at any time, with advance notice as per the contract terms.
- Insurance: Refers to the types and amounts of insurance required to cover potential liabilities or risks in the agreement. This clause outlines the minimum coverage necessary for the parties involved.

The output is exported to csv format, which can be easily integrated into existing contract management systems or used for reporting purposes. The structure of the table includes fields like:

- Clause Type: The specific clause extracted (e.g., Liability).
- Extracted Text: The actual content of the clause from the contract.
- Summary: A brief summary generated by the LLM to give overall content

### *B. Optimizing Prompts Sizes*

Large Language Models (LLMs) are intended to be effectively adapted to specific activities through prompt optimization. An appropriate prompt, whether textual or visual, improves the model's output to better fit the user's task [13]. One of the primary challenges of using Large Language Models (LLMs) like OpenAI's Azure model is handling large input text sizes, as legal contracts can be lengthy and exceed token limitations. To overcome this, the following strategies are employed:

1) Text Chunking:

The contract text is divided into smaller chunks based on logical sections such as clauses, headings, or paragraph breaks. The LLM processes each chunk independently and extracts metadata for each section. This ensures that even large contracts can be processed within the token limit of the LLM without losing context or accuracy.

2) Summarization:

For particularly large sections, a summarization step is added. The system uses the LLM to generate summaries of each section before extracting metadata. This reduces the overall token count while preserving the key information needed for metadata extraction. This step is useful when contracts contain extensive descriptions or legal explanations that may not be directly relevant to the metadata extraction process but are still necessary to retain for legal clarity.

3) Hierarchical Querying

Instead of sending the entire contract at once, the system performs hierarchical querying, where initial queries extract broad metadata (e.g., major sections like "Liability" or "Insurance"). Subsequent, more specific queries are then used to refine the extraction, targeting specific details within each clause. This two-step querying process minimizes the amount of text the model has to process in one go, ensuring efficient and focused metadata extraction.

### *C. Real-time FAQ Generation*

One of the extra features of the system is the ability to generate real-time FAQs from the text of a contract. This is particularly useful for legal professionals who might be needed to provide some quick response to the fundamental questions related to a contract. The process of generating FAQs involves:

1) Natural Language Processing (NLP) for Query Understanding:

When a user asks a question related to the contract (e.g., “What are the liability limits?”), the system uses NLP to interpret the question and match it to the relevant sections of the contract. The OpenAI Azure LLM then searches the contract text to locate the clauses most likely to answer the user’s query.

2) Answer Generation:

Once the relevant text is identified, the LLM generates a natural-language answer to the user's question. This answer includes the direct clause as well as a concise summary or explanation for better comprehension.

For example, if asked about payment obligations, the system extracts the payment terms and provides a summarized version, highlighting key details like deadlines and penalties.

3) Answer Generation:

Once the relevant text is identified, the LLM generates a natural-language answer to the user's question. This answer includes the direct clause as well as a concise summary or explanation for better comprehension.

For example, if asked about payment obligations, the system extracts the payment terms and provides a summarized version, highlighting key details like deadlines and penalties.

4) Continuous Learning:

The FAQ system is designed to learn from previous interactions. Over time, it builds a knowledge base of frequently asked questions and their corresponding answers, improving its efficiency and accuracy with repeated use. It also integrates user feedback to refine the generated answers, ensuring that the system evolves based on the needs of its users.

By integrating real-time FAQ generation, the system reduces the time legal professionals spend searching for specific information within contracts, allowing for quicker decision-making and better client support.

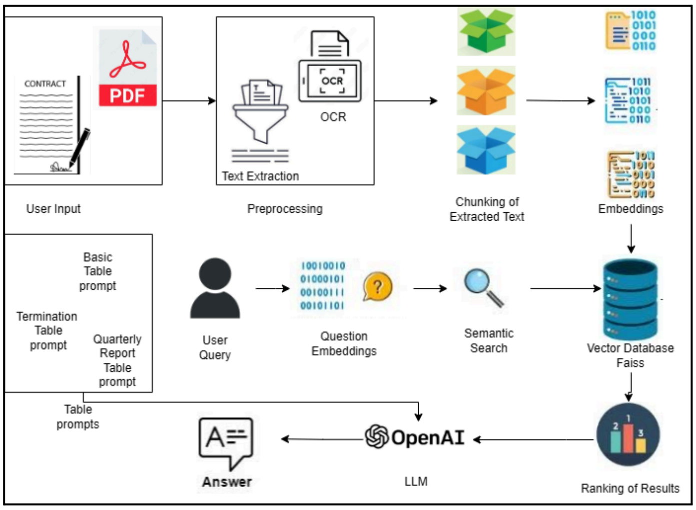



Fig 1. System Architecture

## IV. Experiments and Results

Sample_MSA.pdf

MetaData

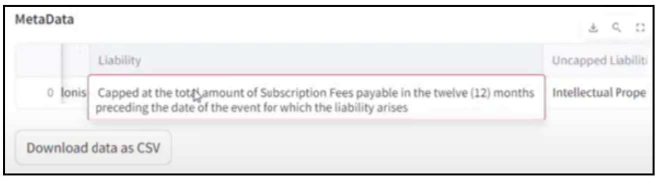
| | Agreement Type | Client | Service Provider | Document Date | Effective Date |
| --- | --- | --- | --- | --- | --- |
| 0 | ---------------------------- | -------- | ------------------ | -------------- | ---------------- |
| 1 | Master Services Agreement | Not specified | Celonis | May 2021 | The earlier of (A) your i |

Fig. 2 Basic Table

MetaData

| | End Date | Clause Number | Title of the Clause | Notice Period | Who can terminate |
| --- | --- | --- | --- | --- | --- |
| 0 | Not specified | 6.2 | Termination for Convenience | 30 days | Either Party |
| 1 | Not specified | 16.2 | Assignment | Not specified | Either Party |
| 2 | Not specified | 16.4 | Governing Law | Not specified | Not specified |
| 3 | Not specified | 16.5 | Amendments | Not specified | Both Parties |
| 4 | Not specified | 16.6 | Entire Agreement | Not specified | Not specified |
| 5 | Not specified | 16.7 | Severability | Not specified | Not specified |
| 6 | Not specified | 16.8 | No Waiver | Not specified | Not specified |
| 7 | Not specified | 16.9 | Export Control and Compliance | Not specified | Not specified |
| 8 | Not specified | 16.10 | Third Party Rights | Not specified | Not specified |

Fig. 3 Termination Table

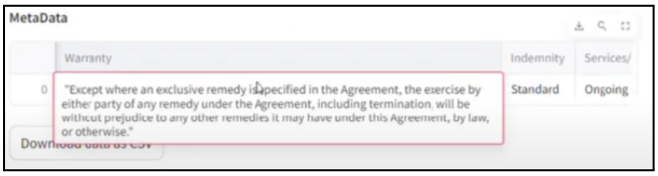


MetaData

| | | Liability | Uncapped Liabiliti |
| --- | --- | --- | --- |
| 0 | lonis | Capped at the total amount of Subscription Fees payable in the twelve (12) months preceding the date of the event for which the liability arises | Intellectual Prope |

Download data as CSV

Fig. 4 Liability clause of Report Table



MetaData

| | Warranty | Indemnity | Services/ |
| --- | --- | --- | --- |
| 0 | "Except where an exclusive remedy is specified in the Agreement, the exercise by either party of any remedy under the Agreement, including termination, will be without prejudice to any other remedies it may have under this Agreement, by law, or otherwise." | Standard | Ongoing |

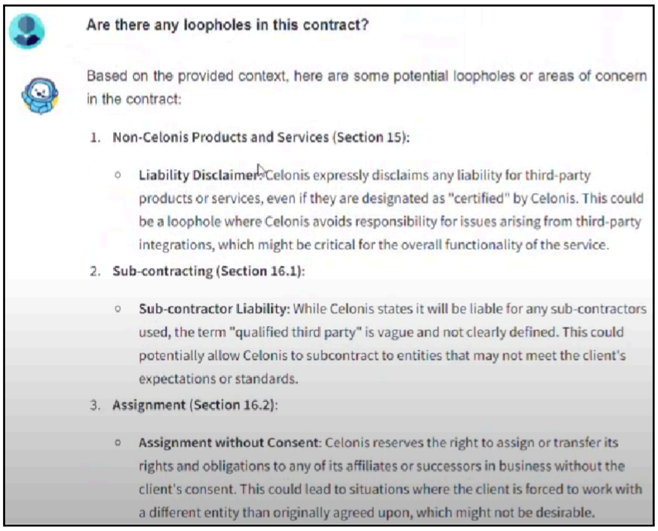
Download data as CSV

Fig. 5 Warranty clause of Report Table

MetaData

| | | Indemnity | Services/Damages | Jurisdiction/Law | Termination for convenience | Insurance |
| --- | --- | --- | --- | --- | --- | --- |
| 0 | se by eitl | Standard | Ongoing | England, London | Not available in the context. | Not provid |

Fig. 6 Report Table

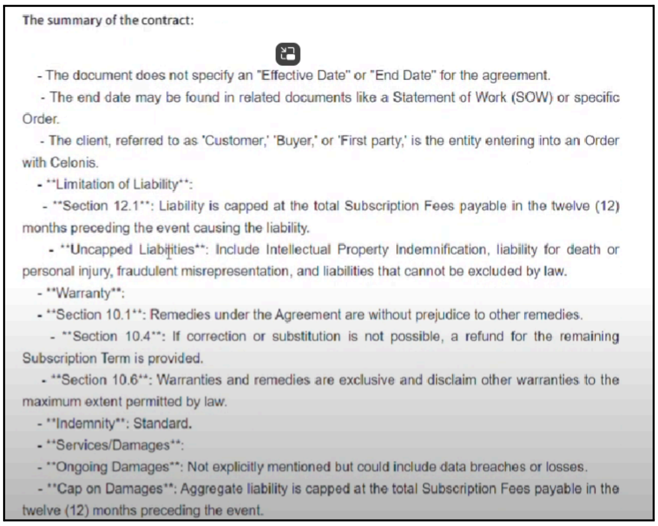


Are there any loopholes in this contract?

Based on the provided context, here are some potential loopholes or areas of concern in the contract:

1. Non-Celonis Products and Services (Section 15):
   - Liability Disclaimer: Celonis expressly disclaims any liability for third-party products or services, even if they are designated as "certified" by Celonis. This could be a loophole where Celonis avoids responsibility for issues arising from third-party integrations, which might be critical for the overall functionality of the service.
2. Sub-contracting (Section 16.1):
   - Sub-contractor Liability: While Celonis states it will be liable for any sub-contractors used, the term "qualified third party" is vague and not clearly defined. This could potentially allow Celonis to subcontract to entities that may not meet the client's expectations or standards.
3. Assignment (Section 16.2):
   - Assignment without Consent: Celonis reserves the right to assign or transfer its rights and obligations to any of its affiliates or successors in business without the client's consent. This could lead to situations where the client is forced to work with a different entity than originally agreed upon, which might not be desirable.

Fig. 6 Chatbot Q&A



The summary of the contract:

- The document does not specify an "Effective Date" or "End Date" for the agreement.
- The end date may be found in related documents like a Statement of Work (SOW) or specific Order.
- The client, referred to as 'Customer,' 'Buyer,' or 'First party,' is the entity entering into an Order with Celonis.
- **Limitation of Liability**:
  - **Section 12.1**: Liability is capped at the total Subscription Fees payable in the twelve (12) months preceding the event causing the liability.
  - **Uncapped Liabilities**: Include Intellectual Property Indemnification, liability for death or personal injury, fraudulent misrepresentation, and liabilities that cannot be excluded by law.
- **Warranty**:
- **Section 10.1**: Remedies under the Agreement are without prejudice to other remedies.
  - **Section 10.4**: If correction or substitution is not possible, a refund for the remaining Subscription Term is provided.
  - **Section 10.6**: Warranties and remedies are exclusive and disclaim other warranties to the maximum extent permitted by law.
- **Indemnity**: Standard.
- **Services/Damages**:
- **Ongoing Damages**: Not explicitly mentioned but could include data breaches or losses.
- **Cap on Damages**: Aggregate liability is capped at the total Subscription Fees payable in the twelve (12) months preceding the event.

Fig. 7 Summary of Quarterly Report

## V. Limitations

### A. Scalability Issues with Large Contracts

- The proposed system relies on Large Language Models (LLMs) such as OpenAI's Azure LLM, which have inherent token limitations. Legal contracts often exceed the maximum context length that these models can process in a single query, requiring text chunking and hierarchical querying. This can lead to loss of contextual understanding between related clauses spread across different sections of a contract.
- Processing multiple large contracts simultaneously could significantly increase computational load, leading to higher latency and processing costs when deployed in a real-world legal setting with a high volume of contracts.

### B. Handling Diverse Legal Documents

- Contracts vary significantly in structure, language, jurisdiction, and legal terminology. While the proposed system performs well with common legal clauses (e.g., liability, insurance, jurisdiction), it may struggle with domain-specific contracts such as those in healthcare, real estate, or mergers and acquisitions, where unique clauses exist.
- The system primarily focuses on English-language contracts, and while future work mentions multilingual support, legal translation and jurisdictional variations in phrasing and clause structuring remain a major challenge.

## VI. Future Work

To address the current limitations and further improve the system, several potential enhancements are proposed:

*A. Overcoming Token Limitations:*

There are many strategies that can be pursued to overcome token limitations:

- Text Chunking with Overlap: Implement smarter chunking mechanisms with overlapping content to divide the text into chunks, which may help preserve continuity in clauses. This can ensure that even clauses spanning over multiple sections are captured appropriately.
- Hierarchical summarization: In case one has a very huge contract, that input can be summarized in a hierarchical procedure wherein a summary is made first, and then the summary will be processed toward metadata extraction. It will reduce the amount of the input text while conserving the most important information for the end user. We aim to prototype and test hierarchical summarization within the next quarter, focusing initially on reducing text size while maintaining critical information.
- Memory-Augmented Models: Using memory-augmented LLMs or longer context windows, for instance in GPT-4, which extends token capability, the model can process huge blocks of text while having contextual information from previous sections.

*B. Multi-lingual Contract Support:*

Most contracts are multilingual. International business has a significant share of contracts in multiple languages. It would be very precious to expand the system to accommodate such multilingual contracts. There would be scope for using multilingual models like mBERT or XLM-R. More importantly, the model could be fine-tuned for legal documents in different languages- French, Spanish, German, or Arabic. In the next six months, we plan to introduce support for European languages (French, Spanish, German), followed by Arabic and Asian languages in the next year.This would enable the system to operate with contracts spread across various legal jurisdictions and less dependent on translation services.

## VII. Conclusion

This work demonstrates the capacity of LLMs, such as the OpenAI Azure LLM, to automate the extraction of essential metadata from legal contracts. The system quickly identifies and extracts all the pertinent clauses concerning liability, insurance, jurisdiction, and terms of payment, then structures them in tabular form for easy analysis and review. We have proposed four tables which provides with effective analysis of legal contracts. In addition, real-time generation of FAQs based on the content of a contract has more functionality integrated into the system: legal professionals can now query particular details of a contract and receive an accurate response instantly. It has reduced manual efforts in analyzing contracts, with LLMs instead hastening workflows, minimizing human error, and improving the efficiency associated with legal processes. The result is the ability of LLM-based systems in changing legal professionals' interfaces with complex contracts, especially in metastuffs like data extraction and compliance checking.

## References


[1] Dan Simonson, Daniel Broderick, and Jonathan Herr. 2019. The extent of repetition in contract language. In Proceedings of the Natural Legal Language Processing Workshop 2019, pages 21–30, Minneapolis, Minnesota. Association for Computational Linguistics.

[2] Zin, May Myo, et al. "Information Extraction from Lengthy Legal Contracts: Leveraging Query-Based Summarization and GPT-3.5." *Legal Knowledge and Information Systems*. IOS Press, 2023. 177-186.

[3] Mongoli, Alessio. *The Use of LLMs in the Legal Field: Optimizing Contract Management with Generative Artificial Intelligence.* Diss. Politecnico di Torino, 2024.

[4] A. D et al., "Image Text Detection and Documentation Using OCR," *2024 International Conference on Smart Systems for Electrical, Electronics, Communication and Computer Engineering (ICSSEECC)*, Coimbatore, India, 2024, pp.410-414,doi:10.1109/ICSSEECC61126.2024.10649443.

[5] Sisodia, Prakhar, and Syed Wajahat Abbas Rizvi. "Optical character recognition development using Python." *Journal of Informatics Electrical and Electronics Engineering (JIEEE)* 4.3 (2023): 1-13.

[6] Mukherjee, Sumita, et al. "OCR Using Python and Its Application." *Journal of Advanced Zoology* 44 (2023).

[7] Geerligs, Cornelis. *Information Extraction from Contracts Using Large Language Models*. Diss. 2024.

[8] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion An droutsopoulos. "LEGAL-BERT: The muppets straight out of law school". In: arXiv preprint arXiv:2010.02559 (2020). DOI: https://doi.org/10.48550/arXiv.2010.02559.

[9] Lucia Zheng, Neel Guha, Brandon R. Anderson, Peter Henderson, and Daniel E. Ho. When DoesPretraining Help? Assessing Self-Supervised Learning for Law and the CaseHOLDDataset. 2021. DOI: https://doi.org/10.48550/arXiv.2104.08671.arXiv:2104.08671[cs.CL]

[10] McCamus, John. *The Law of Contracts, 3/e*. University of Toronto Press, 2020.

[11] Fosbrook, Deborah, and Adrian C. Laing. *The AZ of contract clauses*. Bloomsbury Publishing, 2022.

[12] Wang, Brydon T. "Prompts and large language models: A new tool for drafting, reviewing and interpreting contracts?." *Law, Technology and Humans* 6.2 (2024): 88-106.

[13] Sabbatella, Antonio, et al. "Prompt Optimization in Large Language Models." *Mathematics* 12.6 (2024): 929.

[14] Frieda Josi, Christian Wartena, and Ulrich Heid. 2019. Detecting Paraphrases of Standard Clause Titles in Insurance Contracts. In RELATIONS - Workshop on meaning relations between phrases and sentences, Gothenburg, Sweden. Association for Computational Linguistics.

[15] Dogu Araci. “Finbert: Financial sentiment analysis with pre-trained language models”. In: arXiv preprint arXiv:1908.10063 (2019). DOI: https://doi.org/10.48550/arXiv.1908. 10063.

[16] TomBrownetal.“Language Models are few-shot learners”. In: Advances in neural information processing systems 33 (2020), pages 1877–1901. DOI: 10.48550/arXiv.2005.14165.

[17] Lewis, Patrick, et al. "Retrieval-augmented generation for knowledge-intensive nlp tasks." *Advances in Neural Information Processing Systems* 33 (2020): 9459-9474.

[18] Bulles, John, Hennie Bouwmeester, and Anouschka Ausems. "A best practice for the analysis of legal documents." *On the Move to Meaningful Internet Systems: OTM 2019 Workshops: Confederated International Workshops: EI2N, FBM, ICSP, Meta4eS and SIAnA 2019, Rhodes, Greece, October 21–25, 2019, Revised Selected Papers*. Springer International Publishing, 2020.

[19] Wei-Lin Chiang et al. “Chatbot Arena: An Open Platform for Evaluating LLMs by Human Preference”. In: arXiv preprint arXiv:2403.04132 (2024). DOI: https://doi.org/10. 48550/arXiv.2403.04132.

[20] Chiang, Wei-Lin, et al. "Chatbot arena: An open platform for evaluating llms by human preference." *arXiv preprint arXiv:2403.04132* (2024)