THE RAG REPORT

LARGE LANGUAGE MODELS IN LEGAL DUE DILIGENCE

A research paper exploring the use of Large Language Models through a Retrieval Augmented Generation enabled approach to legal due diligence

RON RAINI
Senior Legal Technologist, Research and Development

MIKE KENNEDY
Head of Research and Development, ILT

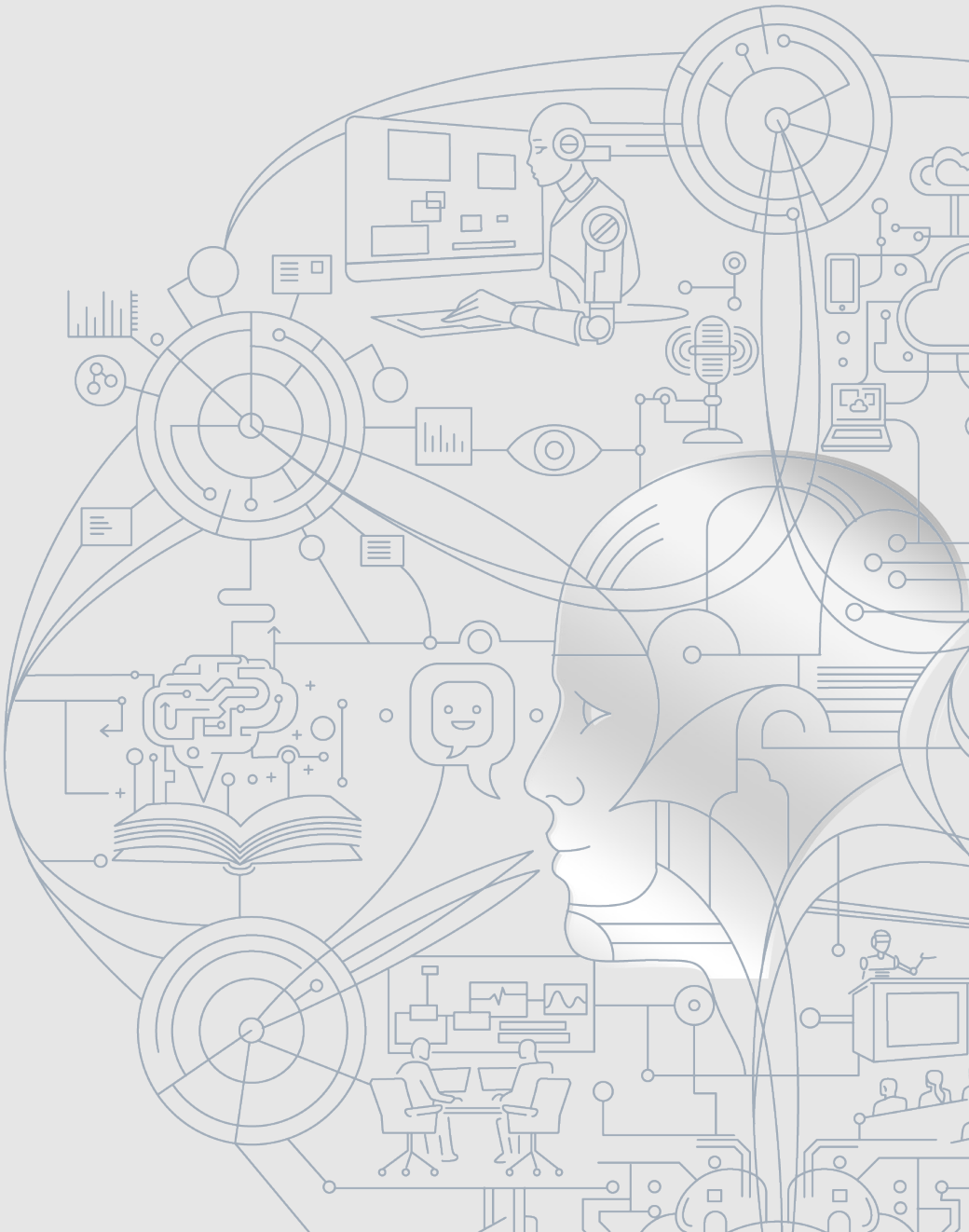
ELLIOT WHITE
Director of Innovation and Legal Technology

KERRY WESTLAND
Partner, Head of the Innovation Group

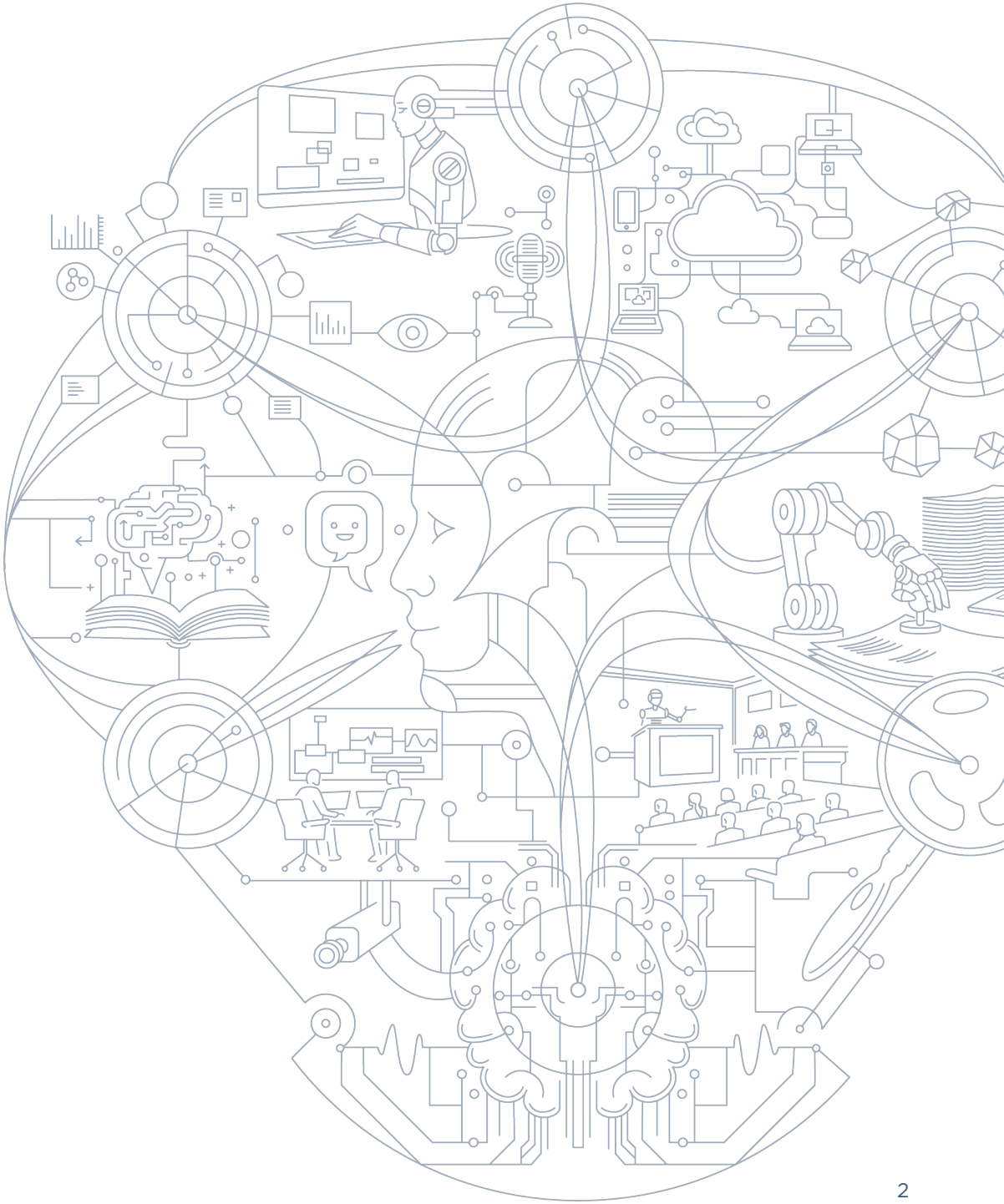
ADDLESHAW GODDARD

MORE IMAGINATION MORE IMPACT

THE LAWYER Awards 2024 LAW FIRM OF THE YEAR



CONTENTS



EXECUTIVE SUMMARY

This research paper, which we believe is the first of its kind from a law firm, is a deep dive into the effectiveness of Large Language Models (LLMs) and their application to legal specific tasks. It has a main focus on the use of LLMs for M&A transaction due diligence. It convincingly demonstrates that LLMs can be used to achieve high accuracy in commercial contract reviews through careful optimisation of a range of components, without fine tuning or training a new model. Through our optimised approach, we can increase the accuracy of LLMs in commercial contract reviews on average, from 74% to 95%.

Addleshaw Goddard (AG) are working to develop a platform that applies LLMs to the extraction of key risks across a document set for the purposes of due diligence. To do this, we wanted to assess alternative approaches beyond partnering with a third party or purchasing off the shelf software.

This paper aims to answer the following questions:

- To what extent can a law firm build and optimise a platform using LLMs for M&A due diligence?
- Is a Retrieval Augmented Generation (RAG) approach suitable for this use case?
- What are the components that have the most optimisation potential within a RAG solution?
- What level of performance or accuracy can be achieved in relation to the extraction of key clauses from M&A transaction documents?

We decided to create a solution utilising a RAG approach, rather than rely on LLMs out of the box. This was partly due to previous experience with this technology as well as a desire to make a long term, cost effective and robust solution.

In order to assess the effectiveness of this approach we amended a series of components across both the retrieval and generation stages and carried out experiments to test the impact of these changes. Our goal was to discover a set of ideal settings to develop a configuration that would optimise our solution for the use case at hand.

We found that a solution utilising RAG can be optimised effectively through a mixture of the selected components, and that this could be done within our own infrastructure at AG by utilising our own team.

Prompt Engineering was a vital tool in refining the generated outputs, showing the continued importance of carefully crafting inputs to LLMs, but this was only possible due to the retrieval element of the approach being optimised to enable the correct information to be fed into the LLM. The components that had the most impact on our results were the amount of context fed into the LLM, the addition of a follow up prompt, the inclusion of keywords which we instructed the LLM to pay extra attention to and in the follow up prompt accusing the LLM of missing information.

We conclude that an RAG approach is the correct approach for this use case, with our solution outperforming a traditional Machine Learning Tool, a Full In-Context solution and a GenAI Contract Review Tool in our testing. The outcomes of this research and related work show that you can make a start with the knowledge you have today. With a mixture of expertise in the technical and legal aspects, and without significant expense, we have been able to develop a sophisticated platform without reliance on largescale data gathering or structuring up front and without training specific models.

FINAL RAG CONFIGURATION

The final configuration for our RAG process consisted of the following components:

01 Using Chunking Strategy 2 (breaking documents down into chunks of 3,500 characters, with an overlap of 700 characters either side).

02 Implementing a customised hybrid retrieval method combining both advanced keywords search and vector search.

03 Creating an optimised Advanced Keywords Query and Vector Search Query for each Provision.

04 Retrieving the Top 10 Chunks and feeding them back to a LLM in the order they appeared in the document.

05 Using GPT4-32K as the LLM for the task.

06 Setting the LLM parameters as temperature 0, maximum tokens to 2,000, and a constant 'seed' value.

07 Drafting a targeted System Prompt that did not unduly increase the context length fed to the LLM.

08 Creating Provision Specific Prompts, improved by our findings in this research, that direct the LLM at specifically what it should be doing.

09 Employing a Follow Up Prompt asking the model to pay special attention to certain aspects and directly accusing it of missing information where necessary.

INTRODUCTION

Since the release of Large Language Models (LLMs), our teams at Addleshaw Goddard (AG) have been exploring how to apply them to legal work. Our journey with Generative AI (GenAI) is thoroughly documented, showcasing our adoption of various GenAI applications. A persistent challenge for us has been identifying a solution that offers flexibility, control, roadmap certainty, accuracy, reliability and affordability – essentially, the age-old question of to buy or build?

We initially focused on the development of our LLM-based internal platform (AGPT) for chat, document review, and other general uses. Following the successful internal rollout of the platform, we turned our attention to applying GenAI to a wider set of use cases. One of these was multi-document review – not only in order to enhance the capabilities within our platform but also to see how GenAI could be specifically applied to legal due diligence.

One of the main challenges of using GenAI technology 'out of the box' is how accurate the models will be when deployed to review large documents and document sets, compared to traditional machine learning tools and manual review.

There are solutions available in the market that aim to solve this challenge. However, we wanted to explore the possibilities ourselves and gain an understanding of how the accuracy of these models could be increased to improve performance, rather than fine-tuning or building a legal-specific LLM.

This paper details our research into how the accuracy of LLMs can be improved using techniques that are already available and how this research underpins a Proof of Concept (PoC) multi-document review platform we are developing for an M&A transaction.

We are sharing our insights to shed some light on how the outputs of LLMs could ultimately reach a level that is acceptable for legal work. We hope this drives the conversation forward in our market and encourages others to share their own findings.

A BRIEF HISTORY: AI ERAS IN LEGAL

Before describing our research in detail, it is worth setting the scene and describing the impact of AI on due diligence over the last decade. We have split the decade into three eras, covering machine learning, GenAI and its wider adoption, and the explosion of foundational models and multi-modal application. Each era has either helped introduce AI to due diligence, advanced it beyond machine learning capability, or fundamentally changed how it is used.

FOUNDATIONAL MACHINE LEARNING (2013 – 2020)

In this period, machine learning and extraction tools entered the legal market and were able to identify and extract specific contract wording from documents. This is the period when it could truly be said that AI was being used on legal work.

We were early adopters of the practical application of these tools for various due diligence exercises. These solutions enabled us to train a model to find specific wording in documents – for example, *force majeure* clauses covering a global pandemic – and then increase the efficiency and speed at which a due diligence exercise was performed. We achieved this using a mixture of technology, processes and people. Typically, this involved training on a subset of documents, validating the output and scoring the models and then running the remaining documents through the system. We exported the results of the AI review to a platform for a team of paralegals or junior associates to review. The results of these reviews were presented using dashboards to highlight areas of risk and a legal report covering recommendations and risk analysis.

After delivering multiple projects, we reached a point where we could save around 50% of the total time spent on a diligence review, reducing the manual review process by approximately 80%. While this represented a substantial improvement on the due diligence process, it still required a high level of human involvement in translating simple extractions of text into usable answers. To take the pandemic example mentioned in the previous paragraph, in this case we would either extract the raw text from the force majeure clause or a Yes / No answer. There was always a risk that the tool would miss a clause or would identify a Yes / No based on an incorrect extraction. This meant that a level of paralegal or junior associate review of the outputs was still required to ensure accuracy, before a more senior lawyer could then also review the output to identify risks and give legal advice. There were limitations with this approach as the time savings per document did not increase when applied at scale. For example, there would be an 80% time saving on the manual review of 100 documents, and the same 80% time saving when reviewing 10,000 documents.

GENERATIVE AI AND WIDER ADOPTION (2020 – 2022)

This period covers the immediate aftermath of the launch of ChatGPT and the other LLMs that quickly followed. This can be defined as a time of simplistic use with the legal industry playing with LLMs, understanding the broader concepts and trying to get to grips with its quirks, failures and potential.

We started to identify the potential for GenAI to be used for document review, particularly as it was able to identify clauses beyond those it was specifically trained on. We spent time exploring the market, identifying providers that were, amongst other use cases, trying to solve large-scale document review using GenAI. At the same time we were also developing AGPT to enhance our knowledge of how LLMs could be used for single document review and the impacts of different prompting techniques.

Our research highlighted the need for a tool to carry out large-scale document review and we consequently invested in a solution. However, we found some limitations in how the outputs were presented and the scale of review that the tool could manage.

At this point, the potential of these tools had become clear. We could now see the step beyond 'Narrow AI' tools, thanks to GenAI's capability to understand natural language requests in the form of the issue a user was looking for, as well as its ability to generate a response in natural language when providing the answer. It allowed us to fail fast when querying documents, as we no longer faced a lead time to train a machine learning model.

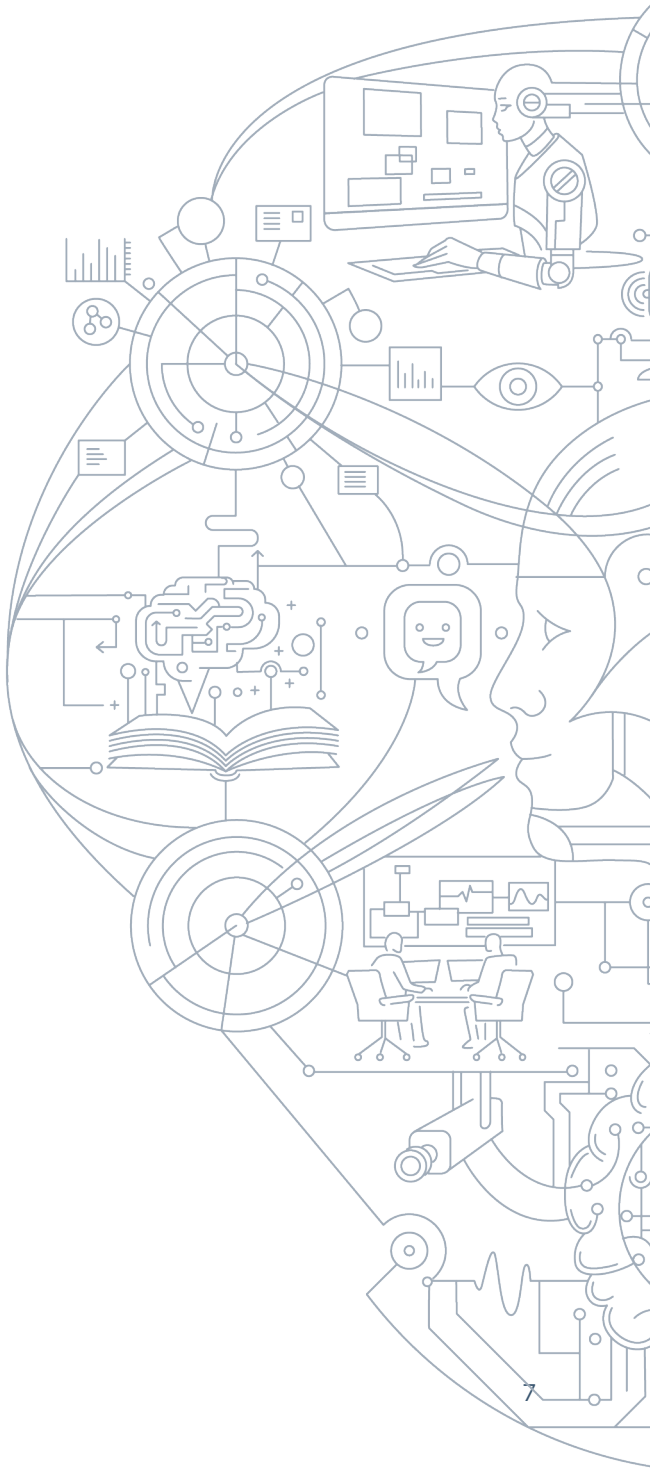
We were also starting to have conversations with clients about whether GenAI could be applied to contract playbooks, as well as the automation of red flag risk reviews of large document sets and other potential use cases, all without the need for manual human intervention.

FOUNDATION MODELS AND MULTI-MODAL (2023 – PRESENT)

At the time of writing (October 2024), solution providers are now using even more powerful LLMs that have been released onto the market, building more sophisticated tools and leveraging more technical aspects to enhance search or data structuring as well as improving System Prompts or task-based prompts. Some are even fine-tuning their own models. The rise of agentic AI is also highlighting the potential of having differing personas or 'personalities' using the same underlying LLMs but tailored to deliver specific tasks.

Through our testing and building we have been able to realise and develop use cases that have been in our 'want to do' list for some time now. Although machine learning and GenAI have created a lot of hype, we have ignored this and focused on assessing the value of this technology and how it can help a law firm deliver legal services. There is a real belief that GenAI is helping to close the expectation gap from lawyers that has existed since 'Narrow AI' tools became prevalent in legal.

Legal firms now have real potential to develop new, advanced workflows that use LLMs alongside other technologies to change how legal work is delivered, building subject matter expertise into an end product that can be used to produce high-quality outputs in a fraction of the time traditionally taken.



FOCUS OF THIS PAPER

This paper details our initial research into how we could carry out multi document reviews using GenAI for an M&A transaction. It forms the basis of how we progressed to a PoC of a M&A Transaction Platform.

The main focus is on how to increase the accuracy of LLMs by optimising the retrieval, extraction and identification of relevant clauses in commercial agreements. The paper sets out the approach and methodology we employed in the testing and evaluation of LLM-powered systems for this specific use case. We discuss the insights and lessons learned with respect to our use case, as well as its applicability for a wider set of legal applications.

We confined our testing to commercial agreements rather than other areas such as Real Estate, Employment, Finance or Tax. This is a relevant place to start, as commercial agreement reviews form a large part of the work required in an M&A due diligence exercise. We expect to be able to apply the same learnings to the other areas mentioned.

OBJECTIVES

Our aim was to take a first step towards evaluating and validating whether LLMs are accurate enough to fulfil our vision for an M&A Transaction Platform.

To do this, our research focused on the technical customisation and optimisation of the components of a Retrieval Augmented Generation (RAG) system, enhancing the system performance for our use case of extracting clauses from documents via the testing of different configuration and parameters.

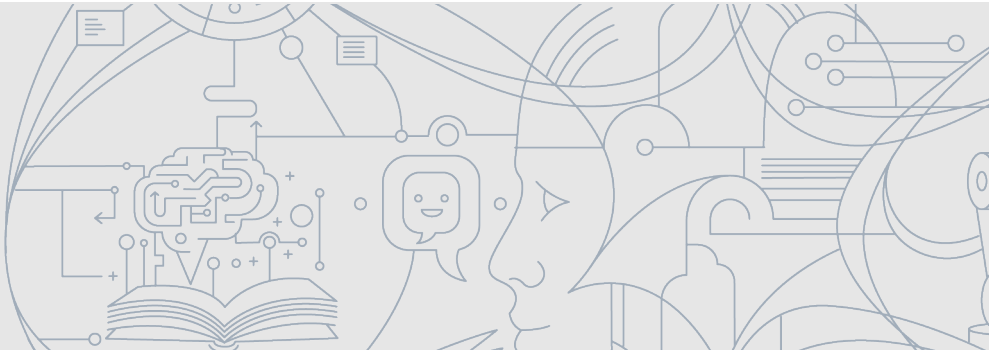
The objective of our research was influenced by our aims for the PoC:

- To build on the success and adoption of AGPT and release more functionality within the platform rather than onboarding alternatives.
- To create a bespoke and configurable platform that would give AG control over broader aspects, such as the LLM used, the system and instruction prompts, the retrieval solution and the custom risk parameters.
- To develop a platform that can help to deliver large scale diligence exercises across a variety of business areas and that is able to do the following:
    - Quickly classify, review and analyse all documents within a deal data room;
    - Apply pre-defined due diligence questions that are relevant to each document type;
    - Generate a comprehensive draft of the due diligence (DD) report, providing detailed answers;
    - Produce a concise risk report, highlighting key issues and potential risks; and
    - Achieve this in a fraction of the time and manpower currently required to generate such reports.
- To advance our learning and knowledge about the application of LLMs to legal work, without relying on outside parties.

SCOPE

The scope of our testing included:

- The specific clauses we were looking to extract and measure;
- The specific task of retrieval and extraction; and
- Our chosen technology workbench.



CLAUSES

We narrowed our focus in our use case to the extraction from commercial contracts of pre-defined contractual provisions and clauses (hereinafter referred to as 'Provisions'). The focus was on those Provisions that are relevant and important in the context of an M&A transaction due diligence exercise and that could have the potential to contain a legal risk or constitute a legal risk by their mere existence or absence in each contract.

The types of commercial contracts we experimented with include, among others: services; supply; licence; development; manufacturing; collaboration; and maintenance.

Although we tested more than 40 Provisions during the research, this paper will focus on our experiments, results and analysis for the Provisions shown in Figure 1.

We selected these Provisions as they were the closest data points to the CUAD[1] dataset we used from the Atticus Project. These are publicly available contracts that have been annotated and can be downloaded alongside specific extraction datasets. This enabled us to run tests on non-confidential documents and freely share the results, in order to evaluate performance and make all results verifiable. All the results shown in this paper are with respect to CUAD agreements only. Further testing and development using AG data is being carried in out alongside this research, with similar results.

We believed that this list of Provisions offered a sufficiently representative sample of clauses that would commonly be focused on during an M&A transaction and contain a sufficient mix of attributes that we know affect the extraction quality. Such attributes include topic and type; usual position in the document; variability or consistency in their formulations; and more.

The intention was that our solution would firstly classify each document and then secondly apply a specific set of questions adapted to the type of the document. However, during the testing, we applied all the above-mentioned Provisions to all commercial contracts.

1 CUAD is a dataset from The Atticus Project that contains publicly available contracts that have been annotated and can be downloaded alongside specific extraction datasets. This dataset is often used by ML solution offerings to test and validate their models. More information can be found here: https://www.atticusprojectai.org/cuad

Assignment	**Audit Rights**	**Cap on Liability**	**Change of Control**	**Effective Date**	**Exclusivity**
Governing Law	**Insurance**	**Licence Grant**	**Minimum Commitment**	**Most Favoured Nation**	**Non-Compete**
Non-Solicit	**Right of First Refusal**	**Source Code Escrow**	**Termination for Convenience**	**Warranty**	

Figure 1. Provisions tested

TASKS

There is a process for due diligence following the extraction of the Provisions where focused questions need to be applied to identify any corresponding risk. In this paper, we have focused on the retrieval and extraction aspects of the process, with risk identification currently ongoing. Each of these aspects will have dedicated prompts and parameters to achieve more accurate responses and results from the LLMs.

TECHNOLOGY WORKBENCH

We used the Microsoft Azure OpenAI service for the research, assessing the broader OpenAI models including GPT-4-Turbo and other OpenAI models. Our choice was influenced by security and infrastructure factors, as well as the deliberate decision that anything we build in the future must be deployable into production within our Microsoft Azure environment. We are working on replicating this research across various other LLM families, including Llama, Claude and Gemini.

The only embedding model we used was OpenAI's 'text-embedding-ada-002'. This is an important retrieval parameter that we expect could boost the performance of our system even further. We will be experimenting with other, more advanced embedding models, as well as other providers and domain-specific (legal) embedding models and looking into training our own domain-specific embedding model.

The specific LLMs that we used in these tests were GPT-4-32K-0613 and GPT-4-Turbo-0125 through the Azure OpenAI service.

METHODOLOGY

There has been much commentary in the market about how and whether Long-Context LLMs would ultimately solve the challenges associated with processing long documents, potentially making the Retrieval Augmented Generation (RAG) approach obsolete. However, as we have seen with the release of new Long-Context LLMs, a larger context window has not resulted in increased performance or improved reasoning quality. If anything, we have seen performance degradation as the context length increases, not to mention increased costs and longer processing times. Consequently, we chose to use a RAG approach for our multi-document extraction use case.

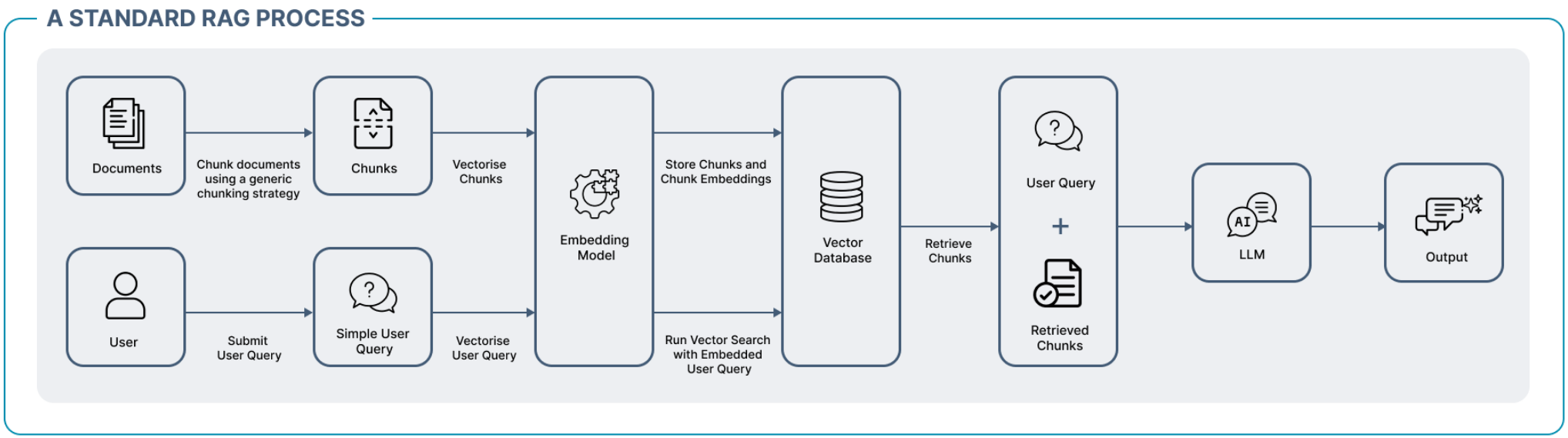


Figure 2. A Non-Optimised Retrieval Approach

To enhance the effectiveness of our RAG approach, our research involved systematically adjusting and evaluating the parameters of several RAG components. This allowed us to determine the most effective settings or combination of settings. To optimise RAG for multi-document extraction and identify the most promising configurations, we started with optimising the Retrieval Components one by one, then moving on to optimising the Generation Components.

Following this optimisation process, we identified a number of RAG configurations that exhibited the best performance. We then tested and evaluated the performance of these RAG configurations in retrieving and extracting the correct Provisions identified above.

Alongside the RAG customisation and optimisation, we also experimented with testing additional methods such as In-Context Learning, traditional machine learning, and a third-party general-purpose legal GenAI-powered tool. This allowed us to run comparisons such as RAG versus In-Context Learning and GenAI versus traditional machine learning, as well as use-case specific GenAI tools versus general purpose GenAI tools, giving us a baseline for the research results.

RETRIEVAL COMPONENTS

The first stage of our research focused on optimising a selection of components related to the retrieval aspect of the RAG approach. Retrieval in this context means building out the ability to find the correct text to be used later in the process. This involves being able to locate the relevant 'chunk' of a document that would contain information relating to a Provision. Our goal was to identify the best performing retrieval configurations to take further in our experiments. We evaluated the performance of the tested retrieval component configurations by using a simple Recall@K metric. This measures the proportion of correctly identified and retrieved relevant chunks (highest ranked), the number of which we define (K) per query.

EMBEDDING MODEL

In the context of RAG, an Embedding Model is used to convert text (as well as other data formats) into vector representations known as embeddings, that capture the semantic meaning of the text which are then stored in a database, known as the Vector Index. This index allows for the efficient retrieval of relevant documents or text segments (i.e. the chunks defined below) based on the similarity of their embeddings to a query's embedding.

CHUNKING STRATEGY

In the context of RAG, the Chunking Strategy is the specific method used to break down large texts or documents into smaller, more manageable pieces known as chunks to store, index, retrieve and process in a more efficient and meaningful way. An effective Chunking Strategy should fit the underlying data and use case to optimise the retrieval.

RETRIEVAL METHOD

This is the technique used to find and fetch relevant information (text chunks in our case) from an index, and includes searching, filtering and ranking the results based on the search query.

VECTOR SEARCH QUERY

This is the text that is converted into a vector representation (using an Embedding Model) and then used in the vector search to find and retrieve the chunks that are most semantically similar to it from a Vector Index based on the corresponding vector representations of the chunks.

ADVANCED KEYWORDS QUERY

This is the set of keywords, search terms and search operators that we designed for each Provision and used in the keywords search retrieval element of our system. The keywords search is not sensitive to the length of the chunks and retrieves every chunk that matches any of the search terms in the Advanced Keywords Query.

NUMBER OF TOP K CHUNKS TO RETRIEVE

This is the parameter that defines how many of the most relevant, top-ranked search results will be retrieved from the index and fed into the LLM to generate a response based on the prompt. In the context of RAG, this parameter reflects a trade-off: the more chunks we retrieve, the better the chances of retrieving all the relevant chunks for a query, but on the other hand the more chunks we retrieve and feed into the LLM, the higher the costs and the lower the quality of the LLM's response.

GENERATION COMPONENTS

The second stage of our research focused on optimising a selection of components related to the generation aspect of the RAG approach. This generation stage covers the ability to generate the relevant text, using the chunks that have been retrieved through the retrieval process, which would reproduce the specific text from the document. In most cases, this would be a reproduction of the actual Provision wording. This output would then enable us to ask detailed questions to identify the relevant risks for our diligence report. Our goal was to identify the best performing generation configurations for our primary task of extracting the target Provisions from the documents. To evaluate the generation configurations, we framed our Provision extraction task as a classification problem and then calculated the corresponding recall, accuracy, precision and F1 scores.

LLM

This is the Large Language Model used as the foundational model for this research. The specific model used in any research project can directly influence the output and therefore the results. Throughout this research, we used two models: OpenAI 'GPT-4-Turbo-0125-Preview' and 'GPT-4-32K-0613'.

LLM PARAMETERS

These are the technical parameters that control and influence the manner and style in which LLMs generate text. In setting these parameters, our objective was to use the values that adjust the behaviour of the model and its impact on accuracy. The Parameters and their corresponding settings that we tested include temperature, maximum tokens and seed value. The temperature is on a scale of 0 to 1, with 0 being restrictive and 1 being creative. The maximum token parameter sets the token limit for the response from the model, which influences the length of the output. Setting a seed value is a way of controlling consistency of outputs, with a constant value forcing more consistency in responses.

ORDER OF CHUNKS

This parameter deals with the order in which the retrieved chunks of each query are processed by the LLM. This could be based on search score and relevance to the search query or by the same order that they appear in the document.

PROMPT ENGINEERING

This refers to the work to create the most effective prompt to elicit the best possible output and means the input or instructions provided to the LLM. The LLM is heavily impacted by the quality of a prompt, whether this is through the clarity of instructions or the input data that is provided at the time of instruction.

SYSTEM PROMPT

This is a predefined message designed to influence the behaviour of the LLM and precedes any subsequent prompts put into the model. It can be used to set personas, deliver additional task-specific information, and provide additional instructions. The System Prompt can also be used to define the LLM's response structure, tone and rules of engagement, amongst other things. We use this message as a way of aligning the output with our objectives for the task at hand.

PROVISION SPECIFIC PROMPT

This is often referred to as a 'User Prompt', as it is usually added by the user. In this context, the Provision Specific Prompt is an additional instruction that is bespoke to the task at hand. In our case, we created these for each Provision. Normally, for a tool like AGPT the Provision Specific Prompt would actually be the User Prompt inserted into the chat window that would then be sent to the LLM along with the System Prompt, along with any additional text provided.

PROMPT SEQUENCING

This is also known as follow-up prompting, and is the method of immediately following up an initial prompt with an additional prompt to instruct the LLM to either correct its output or provide more information. This is a common method used to improve outputs in GenAI conversational tools.

OPTIMISATION PROCESS

To optimise a RAG-based approach to build an M&A Transaction Platform, we first experimented with optimising the Retrieval Components, followed by the Generation Components. Figure 2 shows a standard non-optimised RAG approach, the structure of which can be contrasted against our RAG approach set out in Figure 3.

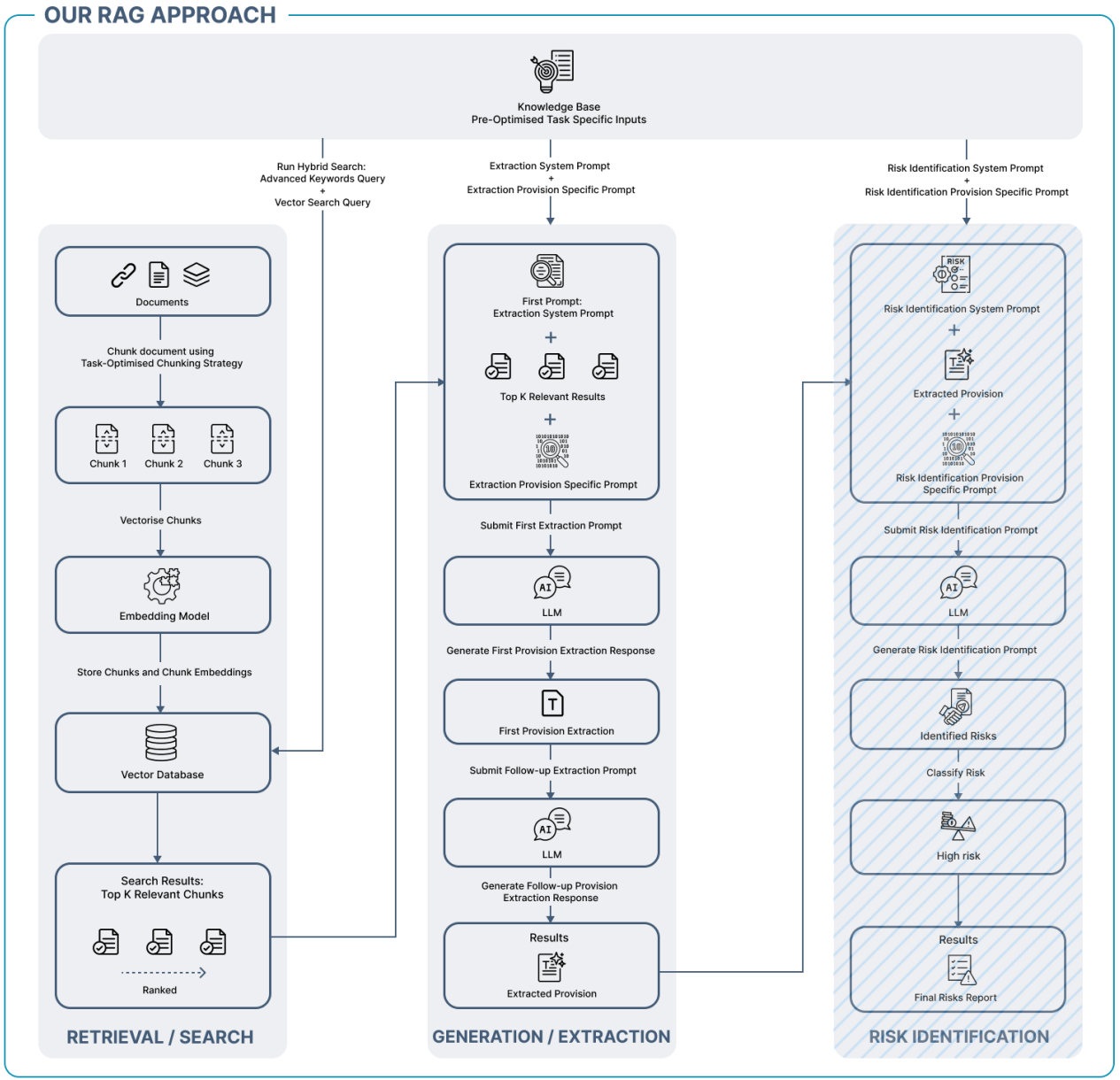


Figure 3. Our RAG Approach

RETRIEVAL APPROACH

This section covers the experiments we carried out in relation to the Retrieval Components, detailing specific configurations, then summarising our finding and results. These experiments focused primarily on ensuring that the correct information is entered into the LLM, rather than on how to achieve the best output – although we found that these two objectives go hand in hand.

EMBEDDING MODEL

Within the scope of this research, we chose OpenAI's 'text-embedding-ada-002' as our constant Embedding Model to understand the full impact of optimising the other components within the experiments. The Embedding Model plays a crucial role in retrieval quality and performance, particularly for domain-specific cases like ours, and we plan to carry out further work in optimising this component in the future. We also aim to explore advanced general and legal domain-specific Embedding Models, including developing our own.

CHUNKING STRATEGY

It should be noted that out of all the components we examined and evaluated in this research, the Chunking Strategy is one of the elements that showed significant potential for improving accuracy.

In our use case, our target Provisions take the form of contractual clauses and subclauses with considerable variation in length. Therefore, our aim was to find a chunking length that was long enough to reduce the chance of splitting a target Provision into two chunks, but that was not so long that it would include too much irrelevant information and reduce the quality of semantic retrieval.

It should be noted that out of all the components we examined and evaluated in this research, the Chunking Strategy is one of the elements that showed significant potential for improving accuracy. Drawing from our research, we aim to explore advanced Chunking Strategies that are dynamic, semantic and document layout-aware to create more meaningful and semantically complete chunks. We are also considering how knowledge graph-based techniques and similar approaches could enhance our chunks and capture relationships between them more effectively.

To find our preferred Chunking Strategy, we carried out experiments aimed at assessing the retrieval performance when applying a vector search retrieval. These experiments are described below. Our aim was to determine which chunk length constituted semantic units that, on average, would be long enough not to split the target Provisions but short enough to preserve sufficient semantic similarity to a given target Provision sample, therefore maximising the retrieval quality. The Vector Search Query used for each Provision was a sample of each Provision that was selected and optimised as described below in the section 'Vector Search Query'. We didn't include a keyword search in this test, as it would have been much less sensitive to chunk length.

For the purposes of this research, we solely considered a fixed-length Chunking Strategy, using a specific number of characters with a specific overlap between chunks. We decided on three Chunking Strategies with a fixed overlap of 20% to capture some text pre-chunk and post-chunk and reduce the risk of losing the context where a chunk was extracted from the middle of a useful piece of text. The details of each of our Chunking Strategies are shown in Figure 4.

Chunking Strategy 1: 2,500 characters long with an overlap of 500 characters either side. Approx. 625 tokens.

Chunking Strategy 2: 3,500 characters long with an overlap of 700 characters either side. Approx. 875 tokens.

Chunking Strategy 3: 4,500 characters long with an overlap of 900 characters either side. Approx. 1125 tokens.

Figure 4. Chunking Strategy configurations

CHUNKING STRATEGY TEST CRITERIA

To test our chosen Chunking Strategies, we selected the 100 longest contracts from the CUAD dataset. This would enable us to put the retrieval approach to the test, as we would only retrieve a small part of each agreement. We chunked the agreements in accordance with our three Chunking Strategies and stored the chunks together with their vector representations in three dedicated Vector Indexes.

We then selected seven Provisions to test across contracts in each Vector Index. We ran a Vector Search Query to test whether the chunk(s) containing the Provision (as labelled by CUAD's annotators) was retrieved in the first chunk (Recall@1) as well as in the top K chunks (Recall@K).

To ensure equal experiment conditions for the three Chunking Strategies, we applied different K values (number of chunks) to equalise the character size. This ensured that no individual strategy had any particular advantage by having a longer character window.

Figure 5 shows how each Chunking Strategy and its particular parameters and number of chunks resulted in a similar character and token count. This allowed us to isolate the chunk configuration and test its effectiveness, rather than simply testing whether retrieving more text would increase the accuracy of results.

Recall@1: Only retrieving one chunk, the top chunk returned from the vector search. Size varies alongside chunk strategy.

Recall@K: Used alongside all strategies, where K is a variable that changes depending on the strategy used.

Variable K Values:

K equal to 7: Used alongside Strategy 1, the retrieval of the top 7 chunks returned from the vector search. Totalling 17,500 characters, approx. 4,375 tokens.

K equal to 5: Used alongside Strategy 2, the retrieval of the top 5 chunks returned from the vector search. Totalling 17,500 characters, approx. 4,375 tokens.

K equal to 4: Used alongside Strategy 3, the retrieval of the top 4 chunks returned from the vector search. Totalling 18,000 characters, approx. 4,500 tokens.

Figure 5. Chunking Strategy Character and Token Counts

CHUNKING STRATEGY TEST RESULTS

Table 1 shows the results of our Chunking Strategy experiments. It is worth mentioning here that, while these numbers may seem low, we applied very strict settings on these experiments, retrieving considerably fewer chunks than we would retrieve in practice, as our objective here was to isolate and stress test the effect of the Chunking Strategy.

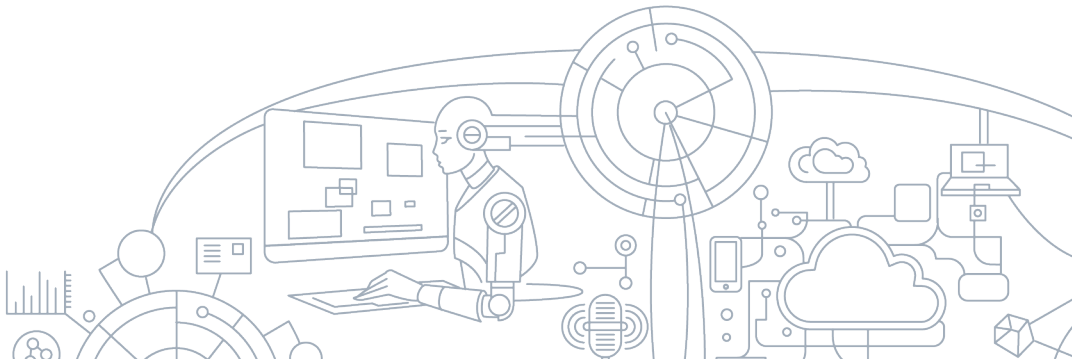
We found that Chunking Strategy 2 was the best performing strategy, achieving the highest Recall@1 and Recall@K in five out of the seven tested Provisions as well as the highest Recall@1 and Recall@K weighted average, with 36.62% and 65.17% respectively. While Chunking Strategy 1's Recall@K weighted average is the second best and not too far behind with 64.58%, its Recall@1 weighted average is less impressive with 27.56%, which is more or less 9% less than Chunking Strategy 2's Recall@1. Finally, although Chunking Strategy 3 achieved a Recall@1 weighted average of 32.44%, outperforming that of Chunking Strategy 1, it got the lowest Recall@K weighted average of 58.81%. This was presumably due to the chunk sizes being too long to preserve enough semantic meaning for a specific Provision.

In light of these results, we adopted Chunking Strategy 2 for the rest of our research, as it exhibited the best fit with the underlying data.

Provision		Chunking Strategy 1		Chunking Strategy 2		Chunking Strategy 3	
	# of Samples	Recall@1	Recall@7	Recall@1	Recall@5	Recall@1	Recall@4
Assignment	225	40.44%	75.11%	49.33%	76.89%	46.67%	68.00%
Change of Control	131	26.72%	64.89%	21.37%	68.70%	27.48%	58.02%
Effective Date	89	38.20%	80.90%	44.94%	71.91%	47.19%	70.79%
Exclusivity	151	9.93%	36.42%	13.25%	33.11%	10.60%	35.76%
Non-Compete	123	27.64%	55.28%	30.08%	57.72%	21.95%	41.46%
ROFR/ROFO/ ROFN	206	21.84%	67.48%	44.66%	68.93%	30.58%	64.56%
Termination For Convenience	80	28.75%	76.25%	50.00%	81.25%	46.25%	76.25%
Weighted Average		27.56%	64.58%	36.62%	65.17%	32.44%	58.81%

Table 1. Chunking Strategy results comparison

As mentioned above, our focus for this research was text chunks. We imagine that other formats such as tables, charts and images would pose different challenges and would require a different Chunking Strategy or OCR to keep the data together.



RETRIEVAL METHOD

As we were dealing with pre-defined Provisions that we wanted to retrieve and extract, we were able to optimise both the Vector Search Query and the Advanced Keywords Query by customising them in advance for each Provision.

This would of course be more challenging to achieve with a conversational AI tool, as the user would need to know what the expected results were going to be. However, our approach gave us an advantage when building solutions for pre-defined use cases. We were able to define specific areas that we wanted to focus on across contracts, and then use the subject matter expertise available within the firm to optimise search parameters and Retrieval Methods.

We implemented a hybrid Retrieval Method that combined both Vector Search Queries and Advanced Keywords Queries. We preferred the Advanced Keywords Query over a simple keywords search, as it was more powerful and provided more flexibility and customisation given the advanced search operators it supported, such as proximity search, term boosting and more.

VECTOR SEARCH QUERY

As we were dealing with pre-defined Provisions, we were able to use a sample of the Provision to optimise the Vector Search Queries, looking for answer-to-answer embedding similarity, rather than the more traditional and less effective question-to-answer embedding similarity commonly used in conversational RAG use cases.

Our objective was to find a candidate sample for each Provision that had good generalisation potential. We looked for candidate samples that had the highest semantic similarity to as many other instances of the same Provision, to use as our Vector Search Query and improve the retrieval.

Based on public and internal data, we compiled clause banks containing hundreds of samples for each Provision. Taking each clause bank in turn, we then created a vector representation of each sample using an Embedding Model (text-embedding-ada-002) and calculated the similarity[2] between each sample and all the other samples of the same Provision. Finally, we selected the samples that were, on average, most similar to the other samples within the same clause bank.

2 Using cosine similarity, which is a metric used to measure the similarity between two vectors in a multi-dimensional space by calculating the cosine of the angle between them. In the context of text analysis and Natural Language Processing (NLP), it's used to quantify the semantic similarity of documents or pieces of text.

Not surprisingly, when we examined those samples that received the highest average similarity scores, it was evident that they tended to be worded in a more standard way, containing terms, phrases and definitions that were more general and commonly associated with the specific type of clause. These samples also tended to be more verbose and included elements beyond just the corresponding main Provision, such as exceptions, exclusions, and sub-conditions.

This can be seen in Figure 6, which is an example of our Vector Search Query of the 'Change of Control' Provision.

An example of the Vector Search Query of 'Change of Control':

Termination Upon Change of Control. Notwithstanding anything to the contrary herein, this Agreement (excluding any then-existing obligations) shall terminate upon (a) the acquisition of the Company by another entity by means of any transaction or series of related transactions to which the Company is party (including, without limitation, any stock acquisition, reorganization, merger or consolidation but excluding any sale of stock for capital raising purposes) other than a transaction or series of transactions in which the holders of the voting securities of the Company outstanding immediately prior to such transaction continue to retain (either by such voting securities remaining outstanding or by such voting securities being converted into voting securities of the surviving entity), as a result of shares in the Company held by such holders prior to such transaction, at least fifty percent (50%) of the total voting power represented by the voting securities of the Corporation or such surviving entity outstanding immediately after such transaction or series of transactions; or (b) a sale, lease or other conveyance of all substantially all of the assets of the Company.

Figure 6. Vector Search Query of the 'Change of Control' Provision

ADVANCED KEYWORDS SEARCH QUERY

Here, our objective was to create an Advanced Keywords Query for each Provision that consisted of a set of meticulously selected keywords, phrases, and terms that were commonly associated with – and specific to – the Provision. This leveraged the fact that contractual clauses often follow certain patterns and use similar terminology, which allowed us to retrieve chunks that included corresponding matches. While the role of a Provision's Vector Search Query is to identify and retrieve chunks that are semantically similar to it, the role of a Provision's Advanced Keywords Query is to complement the semantic search with a more traditional keywords search focused on exact matches.

To design each Provision's Advanced Keywords Query, we used the same clause banks mentioned above and performed text mining and analysis to identify the most common words, terms and phrases for each Provision. We also identified word pairs that commonly appeared in proximity and used all of this information to create our Advanced Keywords Queries.

Our main challenge here was to be as precise as possible, avoiding search terms that were too wide or too general, that were not specific to our target Provision, or that commonly appeared in other Provisions.

Using this method, we created Advanced Keywords Queries containing word pairs found to be most associated with – and specific to – that Provision. The queries also contained advanced search operators such as proximity search (marked with a tilde "~" symbol and followed by the number of words that create the proximity boundary) and term boosting (marked with a caret "^" symbol and followed by the number representing the boost factor). This proved to be highly effective, both in terms of the retrieval performance and in the ease of maintenance and continuous improvement.

To take one example, the term "change control"~5^5 shown in Figure 7 was created to match any occurrence of the words 'change' and 'control' within five words of each other, then retrieve the matching chunk(s), and apply a boost factor of five on the relevancy score of these chunks. This ensured that chunks containing the terms that were most specific to and associated with the target Provision were highly ranked among the retrieved chunks.

On the other hand, terms such as "written notice"~5 and "written consent"~5 were not boosted. This was due to the fact that, although they are very common in Change of Control clauses, they are much more general and commonly appear in many other Provisions as well.

Figure 7 shows examples of the Change of Control keywords.

An example of the Advanced Keywords Query of 'Change of Control':

["change control"~5^5, "control changed"~5^5, "merger consolidation"~10^2, "sale transfer"~10^2, "change ownership"~10^2, "sale substantially"~10^2, "assets substantially"~10^2, "assignment transfer"~10^2, "sale assets"~10^2, "sale merger"~10^2, "transfer interest"~10^2, "business transfer"~10^2, "ownership transfer"~10^2, "transfer assign"~10^2, "management change"~10^2, "written notice"~5, "written consent"~5]

Figure 7. Advanced Keyword Query of the 'Change of Control' Provision

NUMBER OF TOP K CHUNKS TO RETRIEVE

During the research, we chose to experiment with fixed values of Top K Chunks to Retrieve of 10 and 20, rather than using a dynamic value based on the length of the target agreement or using a different value for alternative Provisions. Once we had selected Chunking Strategy 2 (3,500 characters) for the testing, we no longer needed a variable number of chunks to return and so settled on a fixed value for 'K'.

Our objective was to find an 'optimal' Top K value that enabled the best possible retrieval performance as well as the quality of the LLM's generated response. This was due to the fact that most LLMs show a decrease in performance after a certain context length – and this becomes even more pronounced in domain-specific use cases.

RETRIEVAL EXPERIMENTS

The section below compares the results of our optimised retrieval approach with a non-optimised retrieval approach when testing a selection of Provisions from the CUAD contract dataset.

RETRIEVAL TEST CRITERIA

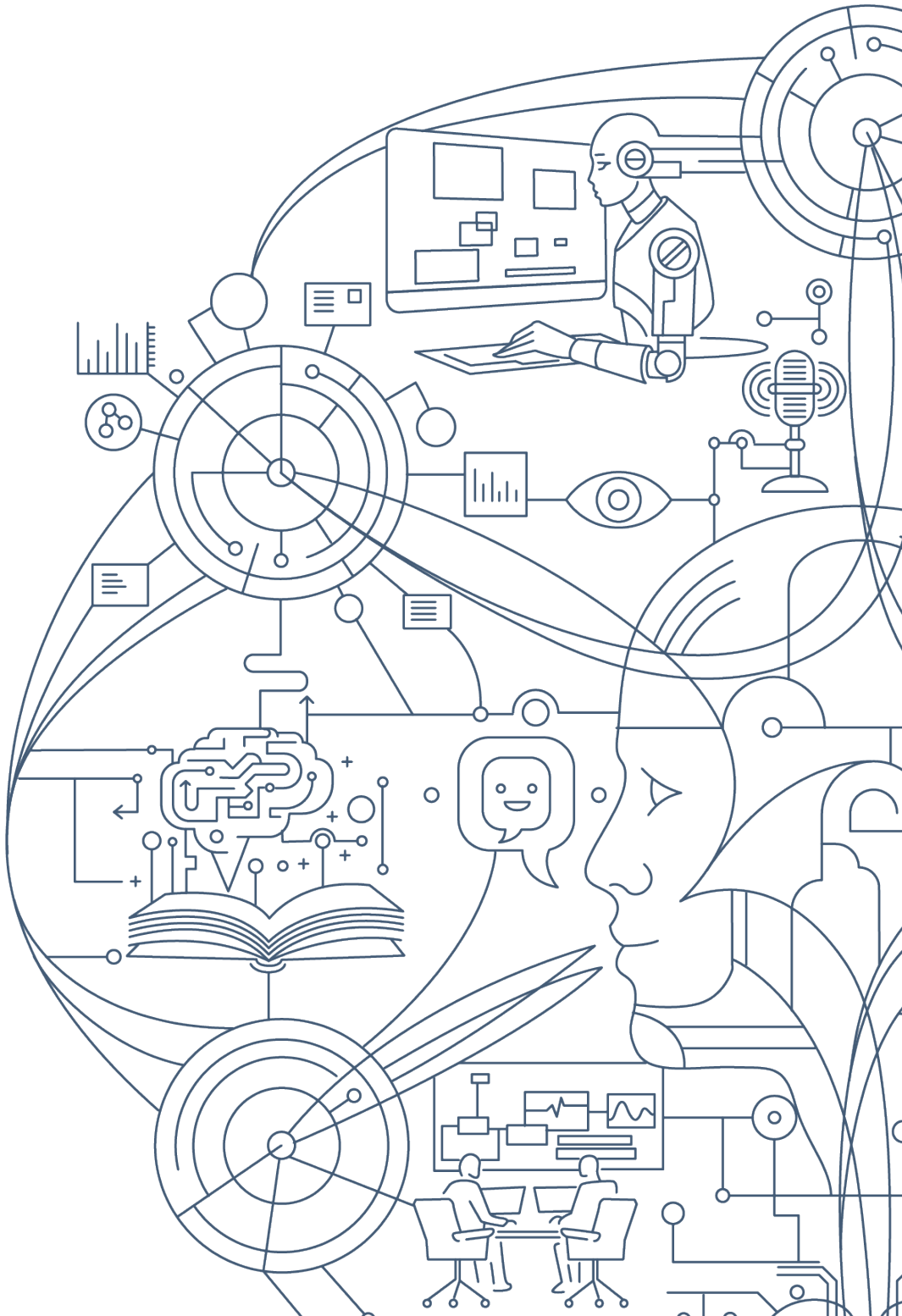
In these experiments, we chunked all of the 510 CUAD contracts in line with our selected Chunking Strategy (Strategy 2), and then stored the chunks together with their vector representations in a dedicated Vector Index. We then tested the retrieval of all instances of a given Provision (as labelled in CUAD) and calculated the Recall@10 and Recall@20 over the entire collection of contracts.

The retrieval results are shown below, covering two approaches to retrieval:

a) A non-optimised approach that takes a simple question-like query for each Provision (such as 'What is the Effective Date of this agreement?') and uses it as a keywords query and a Vector Search Query as part of a hybrid search retrieval.

b) Our optimised retrieval approach that combines advanced keywords search with vector search and utilises pre-optimised Vector Search Query and Advanced Keywords Query for each Provision.

We then tested these approaches across three challenging scenarios which are shown in the tables below along with the results. During the testing, we concentrated on the difference in retrieval performance in the various scenarios.



RETRIEVAL TEST RESULTS

TEST SCENARIO 1

> **...using an optimised retrieval approach increases the effectiveness of identifying the relevant provision by an average of around ~15%.**

This scenario involved running the optimised and non-optimised retrieval approaches on each of the 510 CUAD contracts and retrieving the top 10 chunks, before assessing whether a Provision annotated in the CUAD dataset was present within them. This would then give the Recall@10 score for each approach.

We predicted that this approach would result in a very high accuracy rate as there were a number of contracts within the CUAD dataset that, due to their shorter length, would be fully retrieved within a 10-chunk retrieval strategy. Ultimately, this would mean that both the optimised and non-optimised retrieval approaches would be able to retrieve the relevant Provision within these shorter contracts because the 10 chunks retrieved was the same as running a full context query. The results in Table 2 show that our prediction was correct.

Table 2 shows the accuracy results for this scenario. Despite a number of these agreements achieving 100% accuracy by default, we can still see that using an optimised retrieval approach increases the effectiveness of identifying the relevant provision by an average of around 15%.

Provision	# of Samples	Non-optimised Retrieval	Optimised Retrieval	Difference
Assignment	654	89.14%	98.62%	9.48%
Audit Rights	643	83.05%	98.76%	15.71%
Cap on Liability	672	88.69%	97.47%	8.78%
Change of Control	254	87.80%	96.06%	8.26%
Effective Date	448	94.42%	99.55%	5.13%
Exclusivity	410	51.95%	95.12%	43.17%
Governing Law	464	99.14%	99.78%	0.64%
Insurance	561	92.87%	99.11%	6.24%
Licence Grant	777	63.32%	97.30%	33.98%
Minimum Commitment	424	65.80%	94.10%	28.30%
Non-Compete	260	71.92%	93.08%	21.16%
ROFR/ROFO/ROFN	367	74.93%	91.28%	16.35%
Termination For Convenience	246	94.31%	99.19%	4.88%
Warranty	177	85.31%	98.31%	13.00%
Weighted Average		81.31%	97.28%	15.97%

Table 2. Retrieval Scenario 1 Results: Recall@10 | All CUAD Contracts | Optimised vs Non-optimised Retrieval

TEST SCENARIO 2

...even in a more challenging scenario, an optimised retrieval approach can improve the identification of the correct chunks by around 20%

This scenario involved running the optimised and non-optimised retrieval approaches solely on the CUAD contracts that consisted of 20 or more chunks (approximately over 17,500 tokens or more)[3]. Again, we retrieved the top 10 chunks and then assessed whether a Provision was present within them, which gave the Recall@10 score for each approach. This helped to ensure that the retrieval approach was forced to find and retrieve the chunks from within a large agreement, rather than generating a full retrieval due to the smaller document size. Our prediction was that the optimisation process should show a marked improvement over the non-optimised approach, identifying and retrieving the chunks containing the correct Provisions. The accuracy results shown in Table 3 confirm our prediction.

The implication is that, even in a more challenging scenario, an optimised retrieval approach can improve the identification of the correct chunks by around 20%, on average. Considering this task is more difficult than the task set out in scenario 1 above, the increase in improvement from 15% to 20% shows the benefit of this approach when dealing with large documents. The improvement varies depending on the specific Provision retrieved, showing that a use case and even a concept-specific optimisation strategy has a clear benefit.

3 167 documents in CUAD are larger than 20 chunks

Provision	# of Samples	Non-Optimised Retrieval	Optimised Retrieval	Difference
Assignment	338	88.76%	97.34%	8.58%
Audit Rights	472	80.08%	98.31%	18.23%
Cap on Liability	375	80.53%	95.47%	14.94%
Change of Control	173	82.08%	94.22%	12.14%
Effective Date	157	90.45%	98.73%	8.28%
Exclusivity	215	26.98%	90.70%	63.72%
Governing Law	179	97.77%	99.44%	1.67%
Insurance	391	90.54%	98.72%	8.18%
Licence Grant	493	48.88%	95.74%	46.86%
Minimum Commitment	271	60.89%	90.77%	29.88%
Non-Compete	161	59.63%	89.44%	29.81%
ROFR/ROFO/ROFN	265	66.79%	87.92%	21.13%
Termination For Convenience	115	88.70%	98.26%	9.56%
Warranty	101	78.22%	97.03%	18.81%
Weighted Average		73.15%	95.36%	22.21%

Table 3. Retrieval Scenario 2 Results: Recall@10 | CUAD Contracts +20 Chunks | Optimised vs Non-optimised Retrieval

TEST SCENARIO 3

This improvement of around 14% on average may not be as impressive as the other two scenarios, but it is still a clear indication of the effectiveness of our method, even when dealing with large documents.

This scenario involved running the optimised and non-optimised retrieval approaches on only the CUAD contracts that consist of 40 or more chunks (approximately more than 35,000 tokens or more)[4] This time we retrieved the top 20 chunks rather than the top 10, but only across much larger documents. We then assessed whether a Provision was present within those chunks, which gave us the Recall@20 score for each approach. Our initial prediction was that the retrieval would perform worse, but that increasing the chunks retrieved might counteract this, due to there being more chances of retrieving the correct chunk with 20 shots rather than 10.

What was interesting in this scenario is that the non-optimised Retrieval Method achieved a higher performance than in both scenarios 1 and 2. This is largely due to the increase in the chunks being retrieved, which indicated – not surprisingly – that retrieving more chunks increased the likelihood of retrieving the relevant provision. Table 4 shows that there is still an improvement through the use of an optimised retrieval strategy. This improvement of around 14% on average may not be as impressive as the other two scenarios, but it is still a clear indication of the effectiveness of our method, even when dealing with large documents.

4 71 documents in CUAD are larger than 40 chunks.

Provision	# of Samples	Non-Optimised Retrieval	Optimised Retrieval	Difference
Assignment	175	93.71%	99.43%	5.72%
Audit Rights	312	86.54%	99.68%	13.14%
Cap on Liability	165	86.06%	95.15%	9.09%
Change of Control	95	89.47%	98.95%	9.48%
Effective Date	62	91.94%	100%	8.06%
Exclusivity	117	53.85%	97.44%	43.59%
Governing Law	78	98.72%	98.72%	0.00%
Insurance	253	91.70%	99.21%	7.51%
Licence Grant	271	68.27%	98.15%	29.88%
Minimum Commitment	156	78.21%	96.15%	17.94%
Non-Compete	100	79.00%	96.00%	17.00%
ROFR/ROFO/ROFN	166	83.13%	95.18%	12.05%
Termination For Convenience	57	87.72%	100%	12.28%
Warranty	43	90.70%	95.35%	4.65%
Weighted Average		83.07%	97.95%	14.88%

Table 4. Retrieval Scenario 3 Results: Recall@20 | CUAD Contracts +40 Chunks | Optimised vs Non-optimised Retrieval

RETRIEVAL TEST CONCLUSIONS

To conclude, we can clearly see that optimising the retrieval components and defining a method to provide more context and information result in a clear improvement in accuracy of retrieval of between 14 and 22%.

Taking a specific example Provision such as 'exclusivity', we can see that the initial non-optimised retrieval approach to identify exclusivity performs poorly (ranging from approximately 26 to 58%). However, our optimised retrieval approach increased this by at least 43%. The improvement in this specific extraction shows the challenges of basic prompting. Taking a query such as 'Does the agreement contain any exclusivity restrictions or commitments on either party?' and using this as a search parameter may result in similar semantic concepts being returned around many different references to exclusivity, such as exclusive rights to enjoy the agreement, exclusive rights of termination or access to premises and so on. This decreases the likelihood of the actual exclusivity restriction in relation to contracting with other parties being included within the first ten chunks returned. However, once retrieval is optimised, the method becomes much more nuanced and specific, and therefore more likely to return the correct Provision.

At the opposite end of the scale, we can see that Governing Law is the least improved Provision across all three scenarios, as it is the best performer within the non-optimised method. This is due to a lack of variance across those clauses in most agreements, as well as the lack of similar concepts appearing elsewhere across the agreement in contrast to exclusivity. For instance, the Retrieval Method is much more likely to match up a question and answer in relation to 'What is the governing law?' than in other more nuanced Provisions.

The results of the tests for these two Provisions show how optimisation can drastically improve retrieval of certain Provisions, as well as why the effort involved may not be needed for every Provision. It is apparent that adding subject matter expertise into concepts that are more complicated can improve retrieval success, whereas adding more context to relatively simple and more straightforward concepts only has a minor improvement.

To conclude, we can clearly see that optimising the retrieval components and defining a method to provide more context and information result in a clear improvement in accuracy of retrieval of between 14 and 22%. The graph details the percentage improvement per Provision in scenario 1.

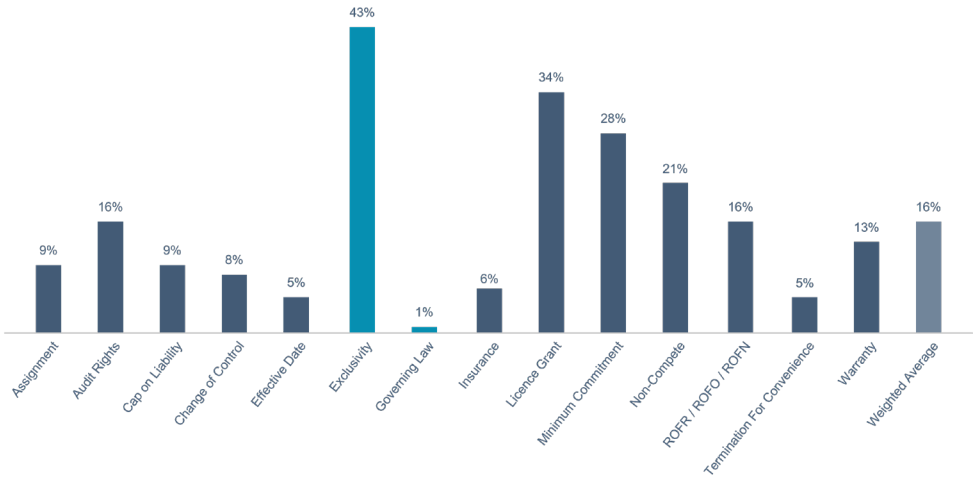


Figure 8. Scenario 1, % Improvement per Provision through an Optimised Retrieval Approach

GENERATION APPROACH

As discussed in the retrieval section above, we split our focus in our testing between Retrieval Components and Generation Components. This section covers the elements in relation to generating output from LLMs, setting out the details of each of these elements and the results. While retrieval is focused on ensuring the correct information is entered into the LLM, generation concentrates on obtaining the best response from the LLM by using that information. This stage is more use-case specific than the retrieval stage, as the methods used may differ if you are focusing on a different use case from a large-scale document review. One of our overall objectives was to produce a high-level risk review across a large document set with short and consistent responses to be able to build red flag risk reports quickly so our generation experiments focused on that use case.

LLMS

For the generation experiments, we utilised the two LLMs set out in the methodology section: 'GPT-4-Turbo-0125-Preview' and 'GPT-4-32K-0613'. This is partly due to the limitations mentioned above around relying on Microsoft Azure and the desire to develop a tool that could be deployed across the firm immediately. This allowed us to test other generation components using a constant through the LLM. However, we are working on building the capability to use different LLMs across the market. We will be running these same experiments with Claude, Gemini and Llama models, with the potential to explore more in time.

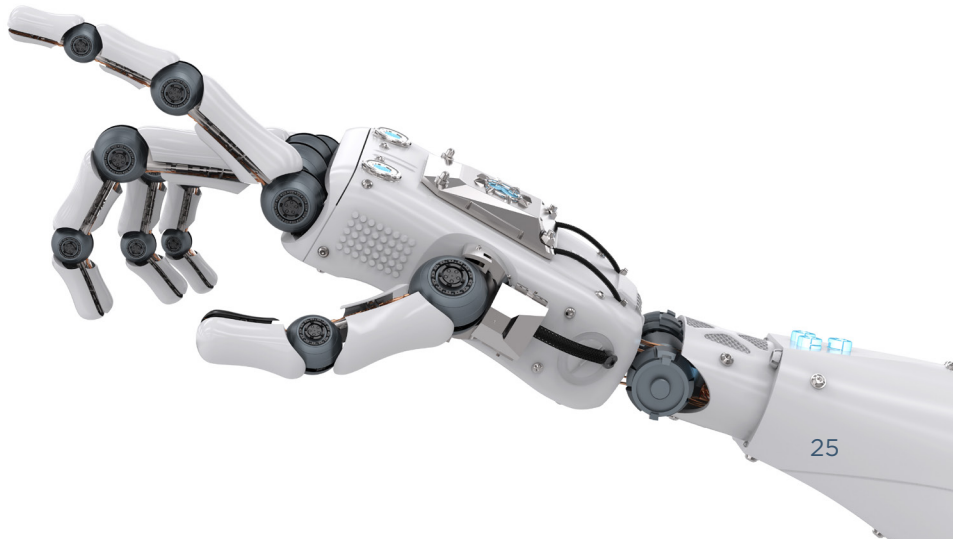
As we progress – and as we are already seeing – the advancement in LLMs will open up even more possibilities. They have the potential to develop to the point where some of the optimisation process can be done through prompting or at the model level. We have seen a marked improvement since moving from GPT-3.5 to GPT-4, for example. While no one can predict whether such advances will be likely in the near future, a steady improvement in performance can clearly be seen. Despite these changes, we still believe that building a strong framework and task-orientated approach will be a future-proof strategy for maximising the benefits of GenAI.

We are aiming to build a platform that is 'LLM agnostic', giving us the ability to select the most effective LLM for a relevant task. The tests outlined in this paper were designed to help us with this objective, demonstrating which models were most suited to particular use cases.

LLM PARAMETERS

For provision extraction, we wanted the model to adhere closely to the text in the document and avoid any unrequired creativity, providing verbatim citations. We also wanted it to be as consistent as possible with its responses, allowing for long extracts if needed. We set the temperature to 0 and the maximum tokens to 2,000, and used a constant 'seed' value.

In the future, we may look to amend these parameters for other generation tasks to build a little more creativity into the model. This will enable us to start testing how effectively we can obtain recommendations while encouraging the model to suggest ideas that may not be present in the base data, or potentially allow users to re-run queries for slightly different responses. For example, in AGPT our parameters are temperature at 0.5 and maximum tokens at 800, which allows a little more flexibility within a conversational tool.



ORDER OF CHUNKS

To determine the order in which the retrieved chunks will be fed into the LLM, we took two main approaches:

a) Sorting the chunks by their relevance score and the search score that was assigned to them in the retrieval process, ranking them from high to low and using a triple hashtag ("###") to separate them – this enabled chunks with higher search scores to appear earlier in the context fed to the LLM (retrieved chunks plus the Provision Specific Prompt).

b) Sorting the chunks by their position in the original document, using a triple hashtag ("###") to separate them – this meant that chunks that appeared earlier in the original document also appeared earlier in the context fed to the LLM (retrieved chunks plus the Provision Specific Prompt).

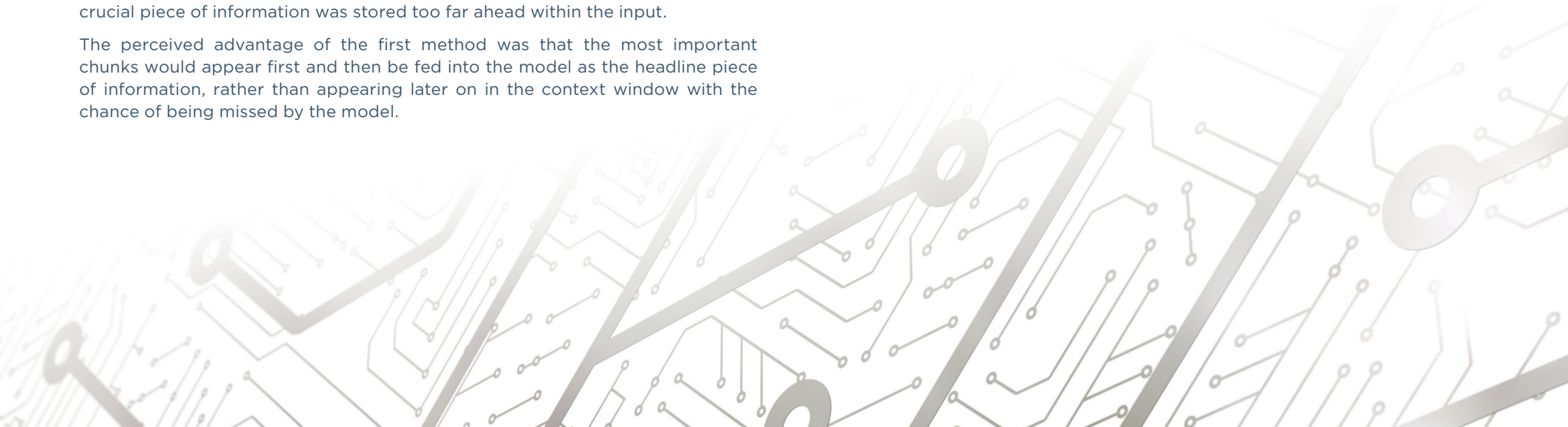
We predicted that trying to replicate the order of the documents through the chunks would enable the LLM to adopt a more common-sense approach. For example, if there was a specific Provision in clause 5 of the agreement, but then a contradicting or ancillary point in clause 12, we could expect the LLM to read this in the order that a human would and apply the previous knowledge from chunk 1 to chunk 2.

Our application of LLMs had already demonstrated the benefits of using a full context approach, and we assumed that this method would bring us as close as possible to maintaining the intended structure of the document, while removing any non-relevant information. However, we were wary that LLMs sometimes miss important information that is further on in their context window, so there was the possibility that not ranking chunks by search score would mean that the most crucial piece of information was stored too far ahead within the input.

The perceived advantage of the first method was that the most important chunks would appear first and then be fed into the model as the headline piece of information, rather than appearing later on in the context window with the chance of being missed by the model.

Following our previous experiences, we decided to adopt the second method, sorting chunks by the position in the original document. There were five main reasons for our decision:

- This method preserved the original continuity and lines of reasoning as well as the hierarchical structure of the original document, which is crucial when it comes to agreements;
- It enabled us to bring in defined terms and insert them at the beginning of the input, even where those defined terms would have low search scores – this is important in future iterations when we look at relating chunks and bring them in for extra context;
- It allowed us to merge consecutive chunks to reduce the chance of splitting Provisions or having fragmented segments of Provisions that would be more difficult for the LLM to identify and extract;
- It allowed us to remove any overlaps when merging consecutive chunks, which reduced the number of tokens to be fed into the LLM; and
- It allowed us to feed the LLM with a limited amount of mostly relevant text, thus avoiding crossing the theoretical threshold at which the LLM tends to prioritise or overlook information based on its position in the context window.



PROMPT ENGINEERING

We have been working on Prompt Engineering ever since we began testing GenAI solutions. It has been an important part of our training, both internally with our colleagues and externally with our clients. We believed that optimising the prompt wording would give us a significant chance of improving the quality of the output. Consequently, we spent some time experimenting with and amending the language, creating a number of methods to test.

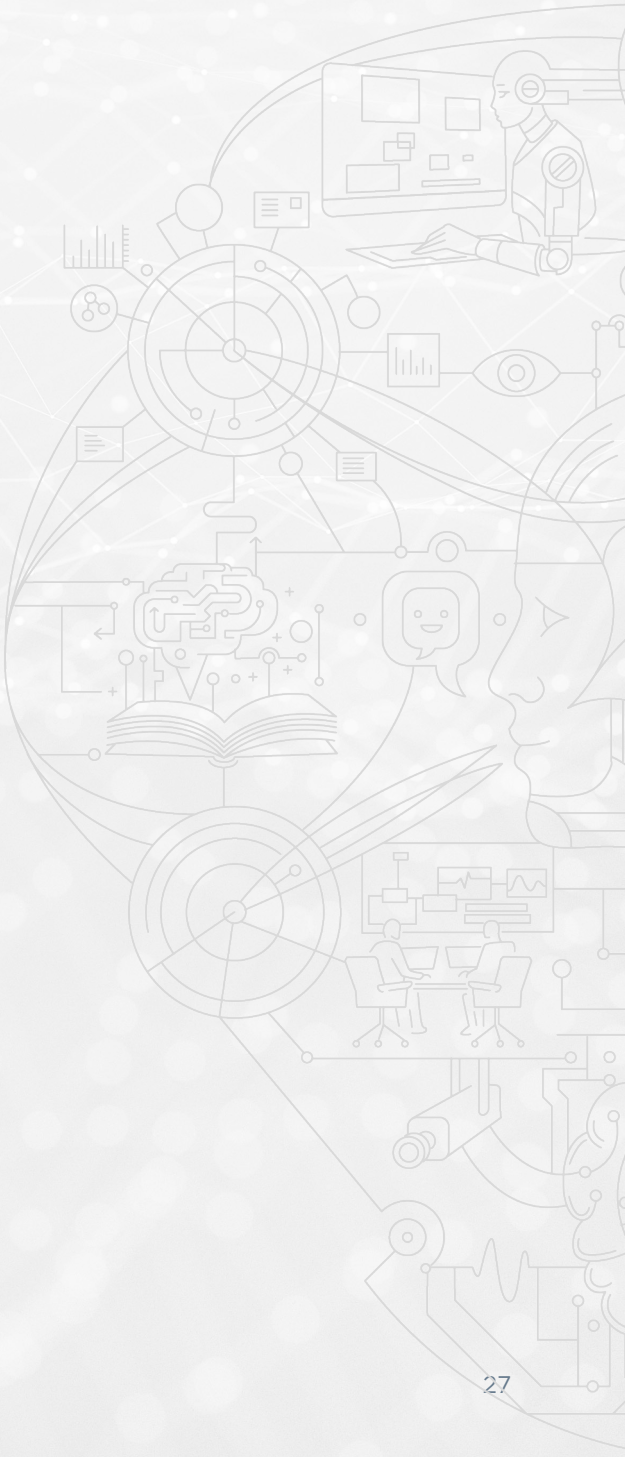
Current solutions in the market build Prompt Engineering steps into their backend workflows, taking this process away from the user. However, we believe that understanding how to get the best out of these models is important for anyone interacting with LLMs, even if that interaction is less involved in drafting prompts. Better prompting skills lead to an improvement in the ability to spot errors, temper expectations and use AI for what it is good for rather than attempting to do the impossible. For solutions that carry out standard tasks and workflows, building prompts into the backend system works well, but with more sophisticated use cases we believe the user should have an element of control – whether that is through the ability to enter more detailed prompts or by having more visibility of the instructions in the system and the facts available.

The main areas of Prompt Engineering we looked at were:

- The contents of the System Prompt, including whether a detailed persona is needed;
- The contents of the Provision Specific Prompt;
- The level of legal specific context needed;
- The effect of emotional / emotive language on LLMs;
- The effect of follow-up and sequential prompting; and
- The location and wording of task specific and general instructions.

Our detailed results are outlined below. Overall, we found that LLMs' output can be improved through more intentional and detailed prompting but that there are some specific pitfalls to be aware of. Although we already knew this, it is useful to have it confirmed in our results. It is also worth noting that most people can achieve a similar enhanced result, as it involves tailoring inputs controlled by the user, without any technical building.

In our opinion, Prompt Engineering will be an important skill in the future, but it will vary in importance depending on the level and type of user. To build an M&A Transaction Platform, we will need to create sophisticated prompts in order to get the best responses from LLMs – however, this will not necessarily require lawyers themselves to draft prompts. Our existing AGPT platform contains a mixture of pre-prepared prompts as well as a robust System Prompt, and also enables the user to interact in any way they like through the chat interface. LLMs may never reach a level where prompting doesn't matter, but advances in knowledge across solutions providers and internal innovation teams will mean that the end user experience can be supported much more.



SYSTEM PROMPT

Using a more elaborate and task-bespoke message does significantly improve the quality and depth of the model's responses. However, we found that being too granular with task instructions has a detrimental effect on model performance

We experimented with several System Prompts. Our System Prompt within AGPT is different from the one we used throughout this testing and that will eventually sit behind our bespoke M&A Transaction Platform. This is because users will interact with a chat solution differently from how they would interact with a solution intended to run large-scale reviews in a consistent manner. From our work with AGPT, we knew that building personas or setting more detailed instructions allows you to achieve better outputs and so we predicted that developing a task-specific System Prompt for diligence would increase the quality of responses.

The base structure for our System Prompt is made up of four key elements:

- A persona;
- The overarching instruction and task that the model is expected to carry out;
- The anticipated input and context of the text to be received; and
- The expected structure of a response and guidelines on how to achieve this.

Using a more elaborate and task-bespoke message does significantly improve the quality and depth of the model's responses. However, we found that being too granular with task instructions has a detrimental effect on model performance. Providing too much context or including instructions on how the LLM needs to answer the question, where to find the information or what wording to look for did not improve the output. It seemed that after a certain length, the LLM tended to 'forget' some instructions, degrading the quality and consistency of its responses. Taking into account that after the System Prompt we then inserted the retrieved document chunks and the Provision Specific Prompt, it appears that putting too much content and instructions into the System Prompt could be a hindrance. Following these findings, we decided to have a relatively simple System Prompt and only include the most high-level persona and task description instructions, moving some of the more specific instructions and directions into a Provision Specific Prompt, which is outlined below.

A SIMPLE EXAMPLE OF A SYSTEM PROMPT COULD BE:

"You are a UK based legal expert that specialises in identifying legal risk within commercial contracts. You will be given the text from a commercial contract followed by a specific question about the contents of that text. Your task is to answer the question in relation to the provided text and highlight any legal risk identified. Your answer must be in two parts, first part must be a short answer to the question asked, with the second part explaining your reasoning and providing any references available to the given text."

A MORE COMPLEX AND EFFECTIVE EXAMPLE, AND THE ACTUAL SYSTEM PROMPT WE USED IN OUR TESTING, IS BELOW:

"You are a UK Lawyer specialising in Corporate and Commercial law. Your expertise is in performing legal due diligence in the context of M&A and investment transactions, which means reviewing and analysing different types of agreements and providing answers to due diligence-related questions about the content of such agreements. You will be provided with one or more text excerpts taken from a single agreement as well as a specific question that will follow such text excerpts. Each of the text excerpts as well as the question will be delimited by triple hashtags ('###'). Your task is to review and analyse each one of the provided text excerpts in light of the question that follows them and then provide a precise and accurate answer to the question based on the information in the text excerpts. Your answer must be based only on information appearing in the text excerpts, and you must avoid making any assumptions or providing speculative information. Do not use markdown. Only use plain text."

Figure 9. Examples of a simple and then more complex System Prompt

PROVISION SPECIFIC PROMPT

The Provision Specific Prompts are the prompts we used in our testing to instruct the LLM to extract the relevant Provision. When drafting our prompts, we concentrated on making them modular, systematic, easy to adjust and repeatable for each Provision. They also had to contain reproducible concepts that could be used across deal types. To achieve this, our prompts consisted of three main parts: a Provision Definition, the Main Request; and the General Instructions.

PROVISION DEFINITION

The Provision Definition we used went further than simply providing a very strict dictionary legal definition of the Provision. It provided the LLM with more context about the Provision Specific Prompts task, guiding it towards the relevant information and steering it away from irrelevant information. For the time being, LLMs are not legally trained, and while there is certainly some knowledge within these models about legal concepts, this knowledge is not yet at the level that would be needed to ensure accuracy when dealing with a complex legal concept. There is progress being made in the market on Legal LLMs (L-LLMs), but our testing does not incorporate any models of that type.

The definition we provided in relation to each Provision was intended to give the model a foundational knowledge of that concept, while also highlighting and clarifying any subtleties relating to the Provision. We believed that this would help the model avoid common mistakes that we had already seen through our use of LLM-based tools. As we carried out our research, we were able to spot these common mistakes and directly address them by adding context into the Provision Specific Prompt. Figure 10 shows some examples of the tweaks we made to Provision Definitions.

Our findings showed that adding this Provision Definition did improve the extraction of certain Provisions, and in particular it reduced the numbers of false positives. This is potentially due to the fact that the additional context helped narrow the scope of what the model would deem relevant to the query. The example in Figure 11 is the full Provision Definition for Change of Control that we used in our testing.

Example: Effective Date

After conducting an error analysis of the Effective Date extractions and realising that most of the false positives are cases where the LLM returned the Agreement Date instead, we added a clarification regarding the difference between Effective Date and Agreement Date, which led to a significant performance improvement.

Example: Exclusivity

After seeing that the LLMs tend to miss Exclusivity provisions that are worded as an obligation to avoid from engaging in certain activities as opposed to a right to exclusive performance, we were able to reduce the number of false negatives made by adding a corresponding clarification to the Exclusivity definition.

Figure 10. Examples of Improvements to Provision Definitions

Example of the definition used for 'Change of Control':

"In the context of Contract Law, Change of Control provisions are contractual provisions that specify the rights, obligations, and consequences that arise when there is a significant alteration in the ownership or management structure of one of the contracting parties. These clauses typically define what constitutes a change of control event, which may include mergers, acquisitions, asset sales, or shifts in voting power. Change of Control provisions can grant certain rights to the non-changing party, such as the ability to terminate the agreement, renegotiate terms, or receive compensation. They may also impose obligations on the party undergoing the change, such as providing notice or seeking approval from the other party."

Figure 11. Example of the Provision Definition used for Change of Control

THE MAIN REQUEST

This part of the Provision Specific Prompt is essentially the task we are asking the model to do. This is in addition to the task-based instructions given in the System Prompt as it focuses on the Provision in question and provides a clear task for that specific Provision. In our case, this task was to extract the specific Provisions which would then be used later when we analysed them in greater detail to identify any specific risks that they may contain, and ascertain whether the mere existence or absence of certain Provisions constituted a risk.

Placing the Main Request after the Provision Definition allowed us to give the model the concept's context and description – while also guiding it towards the right parameter space – before presenting the Main Request to enable the model to carry out the specific task. The language used here was not overly complex as it related to a simple information extraction request. However, experience has shown that it is vital to ensure that the model has a way of responding when the task cannot be carried out. In our case, we gave the model the chance to respond with 'not found', adding to our other components and guarding against hallucinations.

An example of the main request for 'Change of Control':

"Does this agreement contain any Change of Control provisions? If so, please extract all such provisions verbatim and in their entirety. In case you are unable to identify any Change of Control provisions, please respond with 'Not found'."

Figure 12. Example of the Main Request used for Change of Control

GENERAL INSTRUCTIONS

Alongside the Provision Definition and Main Request, we provided additional instructions within the Provision Specific Prompt. As mentioned above, our findings have shown that including these instructions within the Provision Specific Prompt was more effective than including them in the System Prompt. This was because they tended to be 'forgotten' when included in the System Prompt due to the amount of information being provided. We kept these instructions consistent throughout our experiments, but we can see how this could be used in the future to effectively extract provisions that differ in form and style.

Through our use and understanding of AGPT, we had already compiled a list of tactical instructions to obtain the best results from LLMs. We wanted to ensure that these learnings were reflected in our M&A Transaction Platform – or in any other large-scale application of LLMs to documents – so we decided to include these within the prompts for Provision extraction. Whilst these instructions are quite generic, they do prove effective in guiding the LLMs and enhancing performance.

Examples of additional instructions given:

- Make sure to review and analyse each provided excerpt thoroughly. Relevant information for answering the question may be found within different parts of a given excerpt, as well as across multiple excerpts.
- You should assume that every provision that contains any of the following terms is relevant and should be thoroughly examined: 'change of control', 'change in control', 'merger', 'consolidation', 'sale', 'transfer', 'ownership', 'ownership change', 'assignment', 'assets', 'management', 'substantially', 'interest', 'business', 'assigned', 'assign', and 'delegate'.
- Make sure to take your time and think step-by-step before providing your answer.
- Where applicable, make sure to start each part of your answer with a reference to the specific clause, subclause, section or subsection from which the information is taken.
- Do not provide any introductions, just respond with the relevant provisions or with 'Not found'.

Figure 13. Example of the General Instructions in a Provision Specific Prompt

The first two instructions were intended to deal with the main challenges of the LLMs – recalling and then synthesising disperse text instances located in different positions along the context window to answer one question.

The first instruction is more generic and simply guides the model to consider the possibility that relevant information can be found in multiple places along the context window. This was quite effective in making the LLM to 'have another look' before returning the first or most obvious bit of information.

The second instruction is more Provision-specific and guides the LLMs to pay extra attention to Provisions that contain certain terms that are associated with the target Provision. Adding this to our prompts did not require much effort, as the terms used are taken from the same list of advanced keywords that we generated for the retrieval stage. The instruction enhanced the overall performance, resulting in a recall improvement of up to 16% on some Provisions. We implemented this addition in five out of the nine Provisions.

Interestingly, our results seem to indicate that, a level of emotive prompting can lead to a slight increase in performance. We had already noticed this with our use of AGPT. Inserting phrases such as 'This is of utmost importance', 'My job relies on the accuracy of your output' or 'Try your best!' resulted in longer and more comprehensive answers. We have not explored the effectiveness of these emotive instructions throughout our research as we have not isolated this wording to run specific tests, but we have brought an aspect of it into our strategy for generally improving prompts. There is ongoing research in relation to this 'emotional blackmail' (at the risk of personifying machines), but there does not appear to be a clear answer as to why it has such an effect at the moment.

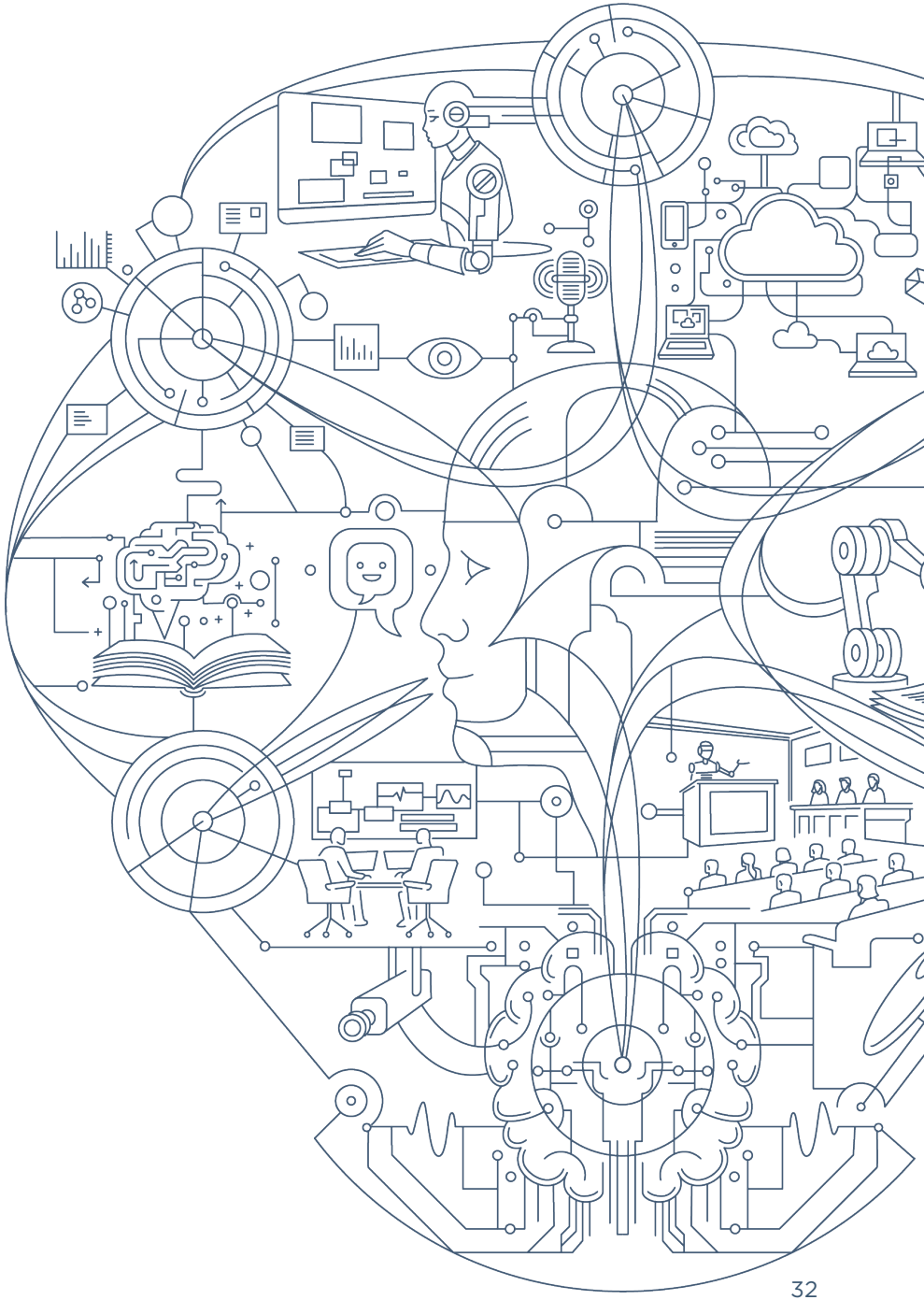
PROMPT SEQUENCING (FOLLOW UP PROMPTS)

The Follow Up Prompt variation that improved performance the most was when we directly accused the LLM of missing relevant information.

We wanted to test the difference between building in an automatic follow-up prompt rather than simply relying on the initial response. We hypothesised that having an in-built follow up may encourage the LLM to go back over answers and provide further information that it initially missed or even bring down the number of mistakes. However, we were also concerned that forcing the model to recheck something that may be correct might result in an increase in hallucinations, as it could assume it had missed something and then bring back false information.

To test this second prompt, we followed the process outlined above, sending the System Prompt, the relevant document chunks and the Provision Specific Prompt to the LLM. Regardless of the output, we then immediately input a follow-up prompt which went back to the LLM alongside all of the above as well as the first response from the model.

We experimented with different variations of the content for the Follow Up Prompt. Initially we thought that a simple request using neutral language, asking the LLM to double check its response would be enough to have an effect of reducing errors and extracting relevant information that was initially missed. However, this didn't seem to result in any noticeable improvement, as the LLM tended to just repeat its previous response. The Follow Up Prompt variation that improved performance the most was when we directly accused the LLM of missing relevant information. This seemed to stimulate the LLM to validate its first response, reviewing the retrieved chunks fed into it with greater care, and providing additional information when needed.



GENERATION EXPERIMENTS

The section below outlines the results we obtained from testing a number of generation configurations in large-scale document review. Due to the iterative nature of the RAG process, our generation experiments naturally built on the findings from our retrieval experiments.

To ensure the most accurate generation tests, we used the optimised approaches we tested for retrieval as we are ultimately building an end-to-end system for the task of Provision extraction.

GENERATION TEST CRITERIA

For the generation tests, we used a subset of CUAD agreements. This consisted of agreements between 40 and 60 chunks in length, which ensured that we did not run a full context review using our RAG approach by the inclusion of small documents. This also meant that we could stress test our retrieval approach in a more realistic scenario.

OPTIMISED CONFIGURATION

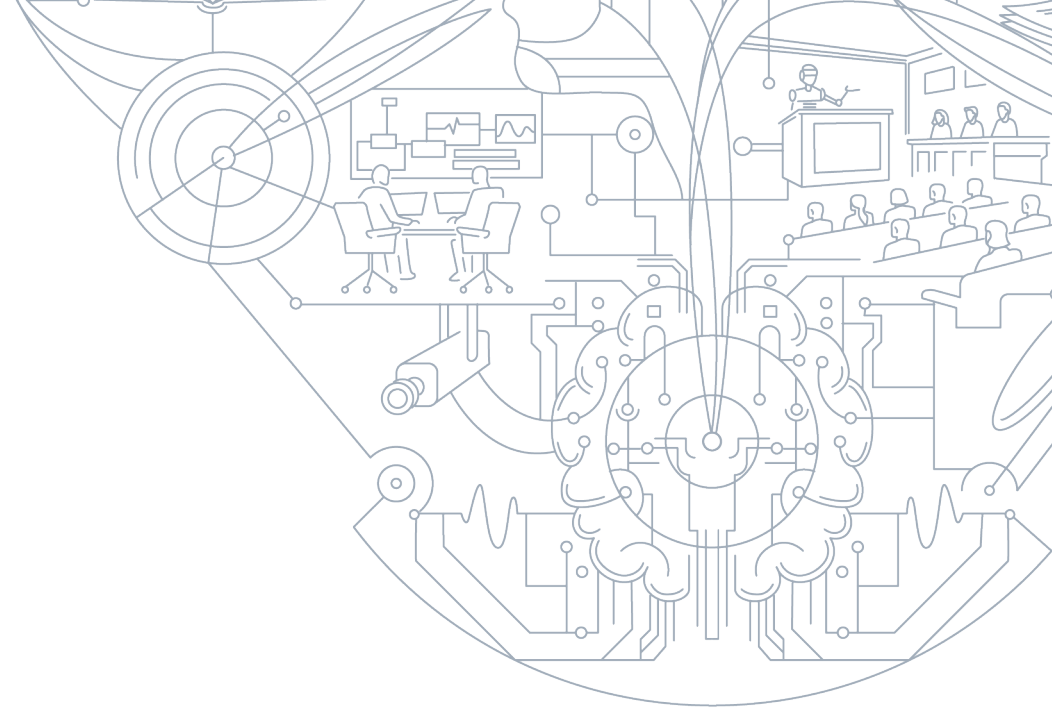
To test the proposed end-to-end process, we applied the optimised retrieval and generation components outlined above.

For retrieval, our approach incorporated the following components:

- Our chosen Chunking Strategy 2;
- A customised hybrid Retrieval Method combining both advanced keywords search and vector search; and
- Each Provision's optimised Advanced Keywords Query and Vector Search Query.

For generation, we included the following fixed components:

- Our System Prompt;
- LLM Parameters;
- Order of Chunks; and
- The structure of the Provision Specific Prompts.



Beyond these components, we tested four additional generation components:

- The inclusion of a Follow Up Prompt as opposed to just a First Prompt without any follow up. This allowed us to obtain detailed results on the effects of a Follow Up Prompt.
- The document context length component – either using our RAG approach, retrieving 10 or 20 chunks (denoted RAG10 and RAG20, respectively), or a Full In-Context approach which involved feeding through the entire document. This enabled us to examine what effect the length of the context had on the quality of the LLM's responses in order to form a baseline comparing RAG and Full In-Context approaches.
- The specific LLM used – this varied between GPT-4-Turbo-0125-Preview and GPT-4-32K-0613.
- An improved Provision Specific Prompt that included an extra instruction to pay attention to Provisions containing specific keywords that were associated with the target Provision.

OPTIMISED CONFIGURATION

These alternative components resulted in 10 configuration variations that we tested on our Provision extraction task. These focused on nine Provisions which are shown in Figure 14.

To complete the picture, as well as these 10 configurations, we also conducted the same tests across two third-party tools: a traditional Machine Learning Extraction Tool and a GenAI Contract Review Tool.

Assignment	**Audit Rights**	**Cap on Liability**	**Change of Control**	**Effective Date**
Exclusivity	**Governing Law**	**Licence Grant**	**Termination for Convenience**	

Figure 14. Provisions used.

As mentioned above, since the task at hand was Provision extraction, we were able to frame it as a classification problem and evaluate the performance of the different configuration variations and third-party tools by comparing their outputs and responses to the corresponding CUAD labels, then calculating their recall, accuracy, precision and F1 scores. When we take our next steps in this research, looking at Risk Identification, we will move towards more complex and subjective evaluation metrics and dedicated risks datasets that we will develop.

While Provision extraction is not in itself a generative task, we found it still served as an excellent foundational study for understanding LLMs' capabilities in legal domains. It tests an LLM's ability to parse and comprehend complex legal language, including specialised terminology and unique sentence structures common in legal documents, across various types of legal documents and practice areas. It requires the LLM to accurately identify and delineate specific legal concepts within a broader context, demonstrating its capacity for fine-grained understanding of legal texts. This task also often involves subtle legal reasoning, as the LLM must discern which parts of a document constitute a distinct Provision or directly relate to a given target Provision or concept. Finally, performance on Provision extraction can indicate an LLM's potential for more complex legal tasks, such as Risk Identification and more. These add up to the advantage of LLMs being 'Zero-Shot Learners' that can be nudged to perform better with simple Prompt Engineering – rather than being trained from scratch with new training data, which is the case with traditional machine learning models. This attribute is valuable not only in pure generative tasks, but also in our Provision extraction tasks, as demonstrated above in the Prompt Engineering section.

SUCCESS CRITERIA

In the evaluation of the configurations and third-party tools, we defined specific criteria tailored to our Provision extraction task and where this task was undertaken in the full process of our LLM-powered multi-document review tool.

We defined six categories of possible responses:

1. A response containing an extraction of the target Provision.
2. A response containing an extraction of Provisions that are not the target Provision but are conceptually related to it (e.g., an Assignment Provision when the target Provision was Change of Control).
3. A response containing an extraction of Provisions that are not the target Provision and are not related to it (e.g., a Governing Law Provision when the target Provision was Effective Date).
4. A response stating 'Not found'.
5. A response containing Hallucinations (e.g., an invented Provision that does not exist in the target agreement).
6. A response containing Misleading information (e.g., an introductory phrase that directly contradicts the Provision's extraction that follows it).

In light of the above, we classified the responses as follows:

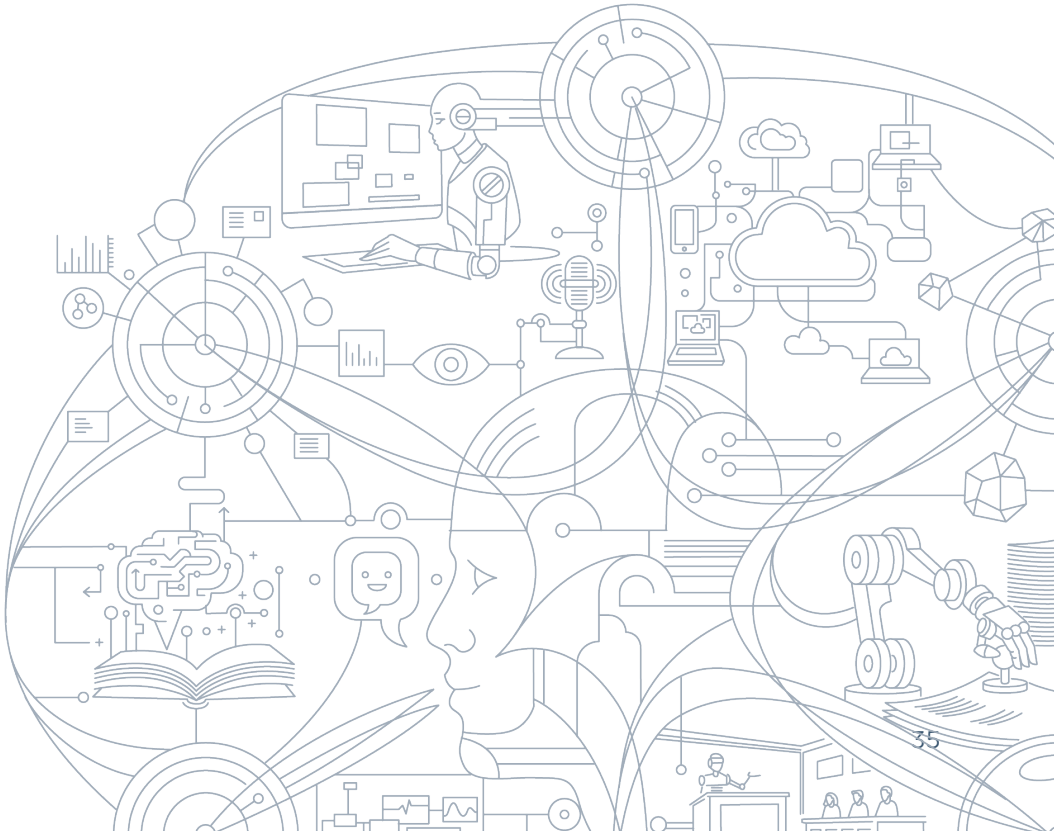
1. A True Positive was defined as any case where the target agreement contained an instance of the target Provision and the response returned at least 75% of the target Provision's instance (as annotated in CUAD), as long as it did not include Hallucinations or Misleading Information.
2. A False Negative was defined as any case where the target agreement contained an instance of the target Provision and the response returned less than 75% of the target Provision's instance or 'Not found'.
3. A True Negative was defined as any case where the target agreement did not contain an instance of the target Provision and the response returned 'Not found' and/or extractions of Provisions that are conceptually related to the target Provisions.
4. A False Positive was defined as any case where the response contained either Hallucinations or Misleading Information or extractions of Provisions that are not conceptually related to the target Provision where the target agreement did not contain an instance of the target Provision.

The purpose of including conceptually similar Provisions within our True Positive classification was to not penalise a solution for extracting information that still has some relevance to the target Provision but may be considered over inclusive. This approach stemmed from the fact that our M&A Transaction Platform will have an additional stage of information processing following the extraction stage, which is the risk identification stage. Our objective in the extraction stage is to perform further and more targeted filtering of irrelevant information beyond what was done in the retrieval stage, in order to reach the risk identification stage with as much targeted and relevant information as possible, allowing the model to focus on the more complex task of risk identification.

We evaluated all the configurations and third-party tools tested using the criteria described above.

When testing the Machine Learning Extraction Tool, we only used its out-of-the-box models that were pre-trained by the vendor.

When testing the GenAI Contract Review Tool, we used its specific 'Contract Review' feature together with our Provision Specific Prompts.



GENERATION TEST RESULTS

This section covers the results of our generation experiments, following the same systematic and phased order in which the experiments were conducted. The aim of the research was to isolate the individual generation components and ascertain their impact on the overall performance on our Provision extraction task.

We started with the context length component, comparing Full In-Context, RAG 20 (20 chunks) and RAG 10 (10 chunks) configurations. We then progressed to the LLM component, examining two different LLMs, specifically GPT4-Turbo and GPT4-32K. For our specific process, we concluded with the Improved Provision Specific Prompt, focusing on the impact of using an Improved Prompt.

Although we defined the inclusion of a Follow Up Prompt as a varying generation component, we did not test it in isolation but applied it across all configurations, as its application provided a better indication of the potential of each configuration. Having said that, a section below outlines the effect of a Follow Up Prompt. Finally, this section details the results of the two third-party tools we tested: a traditional Machine Learning Extraction Tool and a GenAI Contract Review Tool.

Appendix 2 contains our full generation results in relation to all configurations and tools tested, including a detailed breakdown for each individual Provision.

DOCUMENT CONTEXT LENGTH TEST SCENARIO

These results demonstrate the advantage of RAG configurations [...] This is driven by our optimised retrieval approach that allows us to accurately and precisely feed the LLM with the minimum amount of context needed to accurately answer the question.

The first stage of our generation experiments focused on examining what impact the document context length had on the overall performance of our Provision extraction task. This involved experimenting with three different scenarios, altering the context length from the longest to shortest – 'Full In-Context', 'RAG 20', and 'RAG 10' – while keeping all the other generation components fixed.

Figure 15 shows a comparison between 'Full In-Context', 'RAG 20', and 'RAG 10' configurations. The results show the average F1 scores calculated across all Provisions, for both the First Prompt and Follow Up Prompt scenarios. Figure 16 shows a similar comparison but displays the average F1 scores, excluding Effective Date and Governing Law, which are the two Provisions we consider the easiest to extract as all configurations extract them with a perfect or nearly perfect F1 score. These figures show a general decrease in performance as the length of the context fed into the LLM increases. This effect is most prominent in the transition from a Full In-Context to a RAG 20 configuration in the First Prompt scenario. There is a 6% increase in the F1 score when considering the average across all Provisions (Figure 15) and an 8% increase when considering the average across all Provisions excluding Effective Date and Governing Law (Figure 16). This effect is also evident when examining the Follow Up Prompt scenarios in both Figure 15 and Figure 16, with an increase in F1 score of 2 to 3% in the transition from a Full In-Context to a RAG 20 configuration, as well as an additional increase in F1 score of 2% in the transition from a RAG 20 to a RAG 10 configuration.

These results demonstrate the advantage of RAG configurations – and particularly RAG 10 – in comparison with a Full In-Context configuration on our Provision extraction task. This is driven by our optimised retrieval approach that allows us to accurately and precisely feed the LLM with the minimum amount of context needed to accurately answer the question. We believe this is even more significant when dealing with the type of complex domain-specific tasks that we are used to at AG.

More broadly, these findings suggest that when designing and optimising an LLM-powered system, whether with a Full In-Context or a RAG configuration, special attention should be paid to the effect of the context length on the overall performance. This needs to balance a range of factors including the LLM's capabilities, the specific domain, the task and use case at hand and the quality of the retrieval process.

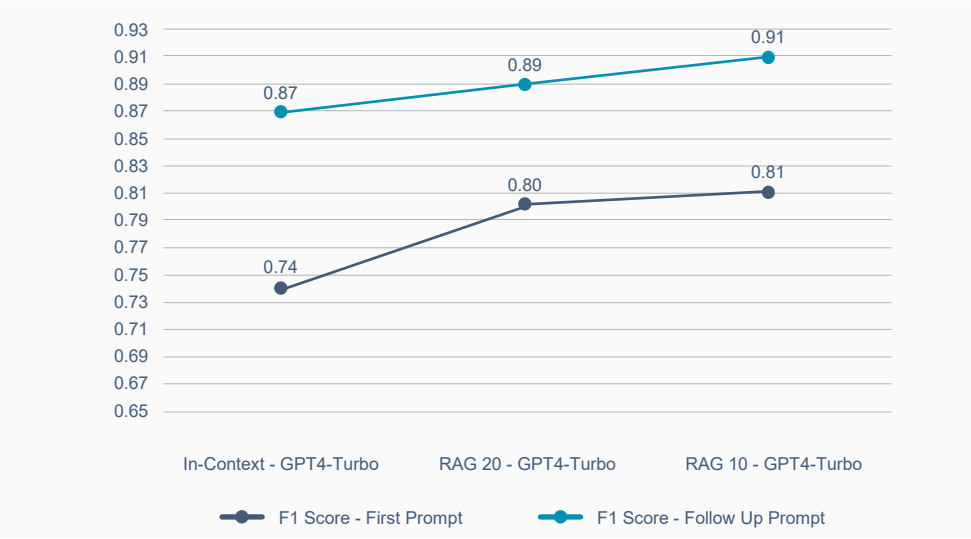


Figure 15. First Prompt and Follow Up Prompt Average F1 Score by Configuration across all Provisions.

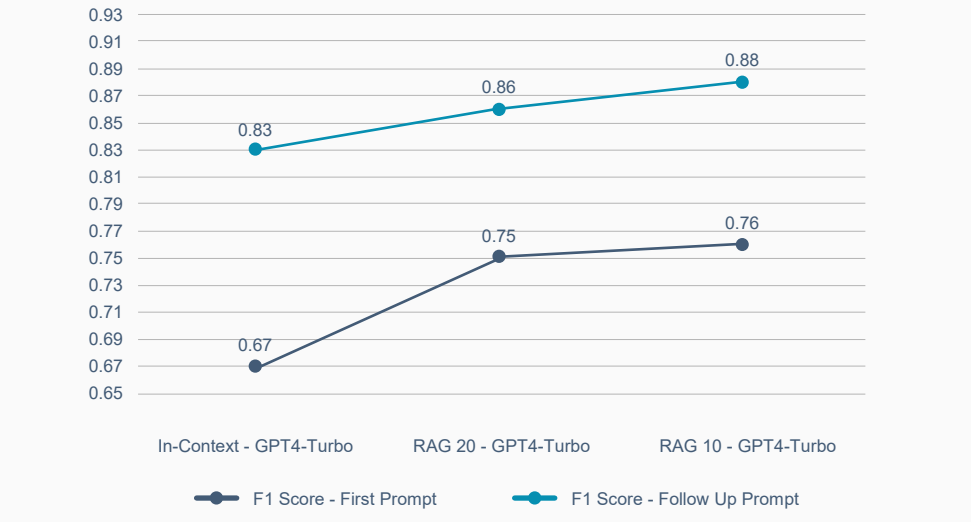


Figure 16. First Prompt and Follow Up Prompt Average F1 Score by Configuration across all Provisions (Excluding Effective Date & Governing Law).

LLM TEST SCENARIO

These results demonstrate the superiority of GPT4-32K over GPT4-Turbo in our Provision extraction task, with a stronger performance both on average as well as in most of the test scenarios.

Based on the results of our document context length tests above and in light of its relative advantage, we decided to focus on the 'RAG 10 – GPT4-Turbo' configuration and conduct an additional set of experiments to compare it with the 'RAG 10 – GPT4-32K' configuration. This meant we could isolate and assess the impact of the LLM used on our Provision extraction task.

Figure 17 compares the 'RAG 10 - GPT4-Turbo' and 'RAG 10 - GPT4-32K' configurations, displaying the average F1 scores for both the First Prompt and Follow Up prompt scenarios, calculated across all Provisions. Figure 18 shows a similar comparison but displays the average F1 scores, excluding Effective Date and Governing Law. These figures show how GPT4-32K performs better than GPT4-Turbo on our Provision extraction task, with an improvement of 4% in the First Prompt and 1% in the Follow Up Prompt when considering the average F1 scores across all Provisions (Figure 17). The results also show an improvement of 7% in the First Prompt and 2% in the Follow Up Prompt when considering the average F1 scores, excluding Effective Date and Governing Law (Figure 18).

Furthermore, when comparing pairs of configurations across each Provision and for both First Prompt and Follow Up Prompt scenarios, we can see that out of 18 pairs, GPT4-32K achieved a higher F1 score in 11 cases, the same F1 score in four cases, and a lower F1 score in only three cases. Appendix 2 has more details of paired comparisons, such as 'RAG 10 – GPT4-Turbo – First Prompt (Assignment)' versus 'RAG 10 – GPT4-32K – First Prompt (Assignment)', 'RAG 10 – GPT4-Turbo – Follow Up (Assignment)' versus 'RAG 10 – GPT4-32K – Follow Up (Assignment)', and so on.

These results demonstrate the superiority of GPT4-32K over GPT4-Turbo in our Provision extraction task, with a stronger performance both on average as well as in most of the test scenarios. This was particularly the case in the scenario of First Prompt, without the moderating effect of the Follow Up Prompt.

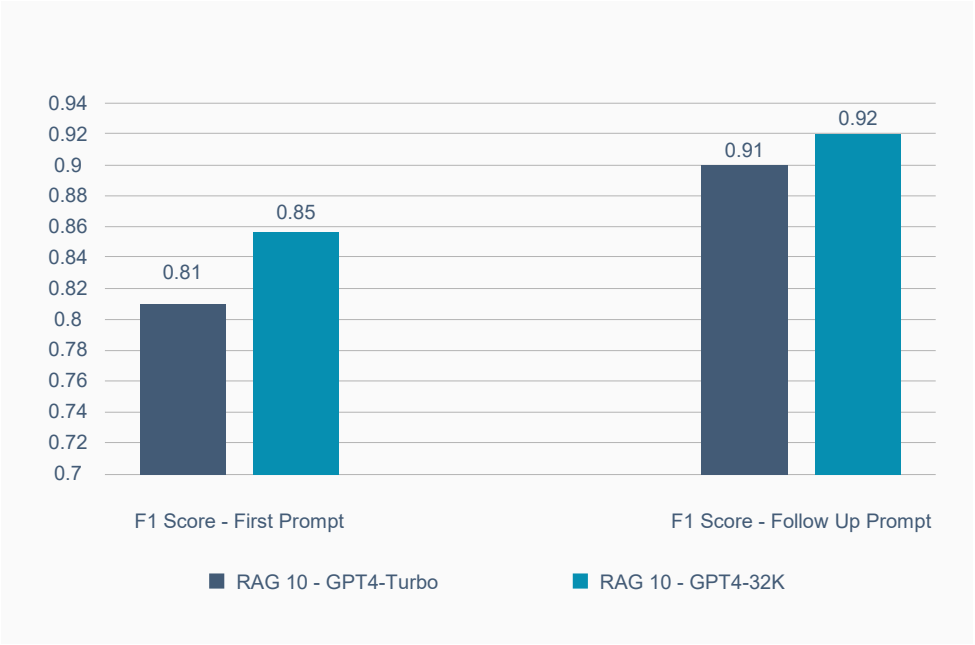


Figure 17. RAG 10 GPT4-Turbo vs RAG 10 GPT4-32K First Prompt and Follow Up Prompt Average F1 Score across all Provisions.

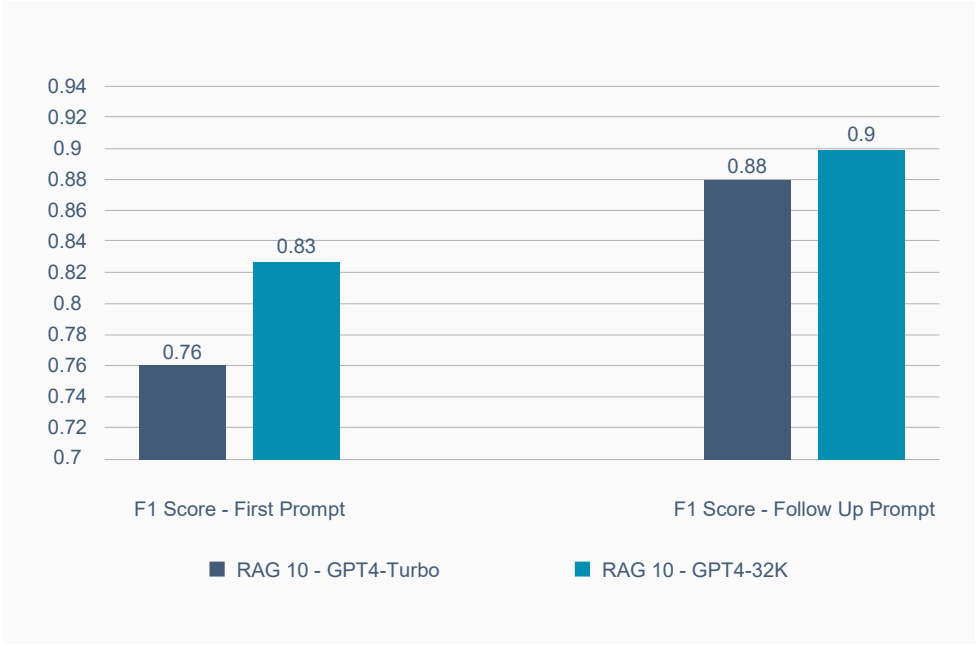


Figure 18. RAG 10 - GPT4-Turbo vs RAG 10 - GPT4-32K First Prompt and Follow Up Prompt Average F1 Score across all Provisions (Excluding Effective Date & Governing Law).

IMPROVED PROMPT TEST SCENARIO

Using an Improved Provision Specific Prompt led to an increase in F1 score in three out of the five tested Provisions...

Following our LLM tests and the performance improvement we achieved with the 'RAG 10 - GPT4-32K' configuration (compared to 'RAG 10 - GPT4-Turbo'), we wanted to test whether we could see additional improvements by using the same configuration but with a better Provision Specific Prompt.

As we had already achieved high F1 scores on some of the Provisions, we decided to focus on the five Provisions where we had not achieved a recall score of at least 90% with 'RAG 10 – GPT4-32K' and the initial Provision Specific Prompt configuration.

We carried out an error analysis of the instances when the LLM failed to extract across the five Provisions (i.e., the false negatives), looking for common patterns and potential causes for such extraction failure. This showed that many of these failures were relatively straightforward instances that the LLM overlooked and failed to recall, highlighting the fact that LLMs cannot reliably attend to their entire context even when it is much shorter than their context window length.

To tackle this issue, we adjusted the Provision Specific Prompts of the five Provisions to include an additional instruction to pay attention to sections containing specific keywords that are associated with the target Provision. Our hypothesis was that this would guide the LLM to better review 'suspicious' sections that might contain the information relevant to the question. We could easily and systematically implement this across different Provisions by using the keywords and terms from each Provision's Advanced Keywords Query, which we had already created as part of the retrieval components optimisation process. This also allowed us to avoid overfitting, as we did not gather keywords from the missed samples, but only from each Provision's Advanced Keywords Query.

Figure 19 compares 'RAG 10 - GPT4-32k' with the initial Provision Specific Prompt and then with the Improved Provision Specific Prompt configurations, displaying the F1 scores by Provision for the Follow Up Prompt scenario. Using an Improved Provision Specific Prompt led to an increase in F1 score in three out of the five tested Provisions, with Cap on Liability and Exclusivity each achieving a 10% increase and Audit Rights achieving a 4% increase. We didn't see a decrease in either Change of Control or Licence Grant, which were the two provisions that received the highest scores with the initial Provision Specific Prompt. This adds further weight to the argument that this method offers robust potential.

These findings suggest that refined prompts can be effective in improving the performance of LLMs, and even mitigate inherent issues such as their difficulty in reliably recalling information from different parts of a document, especially when dealing with long documents. Despite the increase in context windows of some LLMs, we do see a reduction in performance as text length increases, even when this is well below the current context windows.

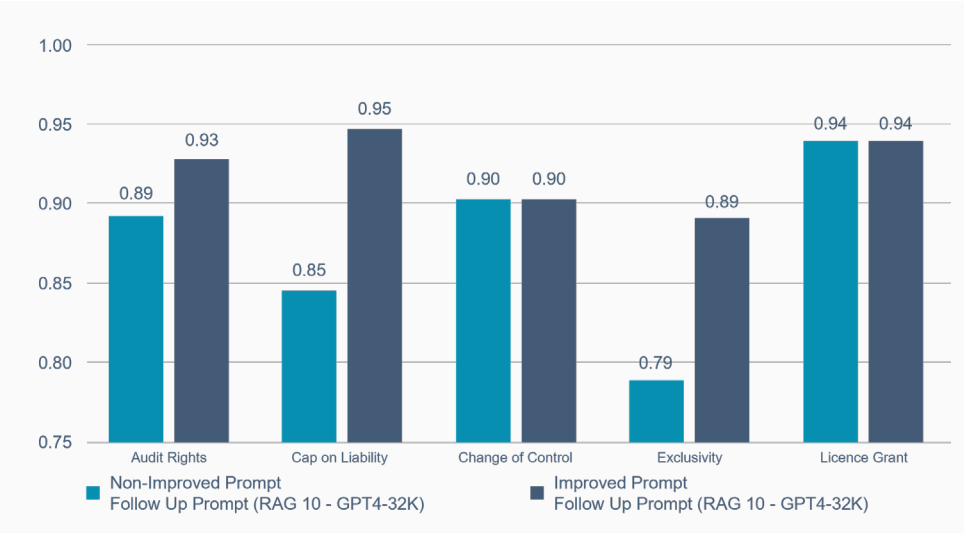


Figure 19. Improved Prompt F1 Score Follow Up Prompt by Provision.

OVERALL CONFIGURATION RESULTS

...the use of a Follow Up Prompt gives us an average improvement of 9.2% across all configurations.

Following the description of the reasoning behind our experimental process, we now present the overall generation experiment results, highlighting the best and worst performing configurations and tools, as well as some of our principal findings.

Figure 20 shows a comparison between all the configurations and third-party tools we tested, displaying their average F1 scores, calculated across all Provisions for both the First Prompt and Follow Up Prompt scenarios. Firstly, it can be seen that the 'RAG 10 - GPT4-32K - Improved Prompt' configuration is our best performer, with an average F1 score of 92% for the First Prompt scenario and 95% for the Follow Up Prompt scenario.

Figure 20 also provides a further demonstration that the use of a Follow Up Prompt gives us an average improvement of 9.2% across all configurations. Appendix 2 goes into greater detail, showing that using a Follow Up Prompt led to an F1 score increase across all tested configurations and Provisions (80+ in total), except for a few Governing Law and Effective Date configurations in which the First Prompt had already achieved a perfect or nearly perfect score. It is also worth mentioning that, in certain cases, the Follow Up Prompt led to an F1 score increase of up to 30% to 35%. 'In-Context', 'RAG 20', and 'RAG 10' configurations on Cap on Liability in Appendix 2 exemplify this.

On a more Provision-specific level, Figure 21 shows a comparison between all tested Provisions, displaying each Provision's average F1 score, calculated across all tested configurations and third-party tools. This standpoint allows us to identify the Provisions that are easier to extract, such as Governing Law, Effective Date and Assignment, with F1 scores of 0.99, 0.97, and 0.91 respectively, as well as those that are more difficult, such as Audit Rights, Exclusivity and Cap on Liability, with F1 scores of 0.79, 0.76, and 0.67 respectively.

Finally, Figure 20 provides us with an important comparative perspective of our configuration, focusing on the two third-party tools we tested: the Machine Learning Extraction Tool and the GenAI Contract Review Tool.

The Machine Learning Extraction Tool lagged behind most of the LLM-based configurations we tested, with an average F1 score of 0.86. Analysis of the performance and output of the Machine Learning Extraction Tool showed that it generally struggled with the extraction of Provisions that were relatively long, were less standard in their language and structure, used more unique or agreement-specific language and terms, or that were split across different pages. In contrast, the LLM-based configurations – especially the best-performing ones – exhibited greater conceptual understanding and flexibility, allowing them to more effectively extract even non-standard Provisions.

Compared to all other tools and configurations, the GenAI Contract Review Tool was the worst performer, with an average F1 score of 0.72. The tool generally struggled to follow instructions, insisted on rephrasing Provisions rather than extracting them verbatim as requested, provided very partial citations, tended to return very short responses and provided inconsistent responses in general. This behaviour is probably the result of system-level constraints, such as its System Prompt and LLM Parameters, which we could not adjust. This illustrates the limitations of using general non-customisable tools and strengthens the case for investing time and resource into building bespoke solutions for specific use cases.

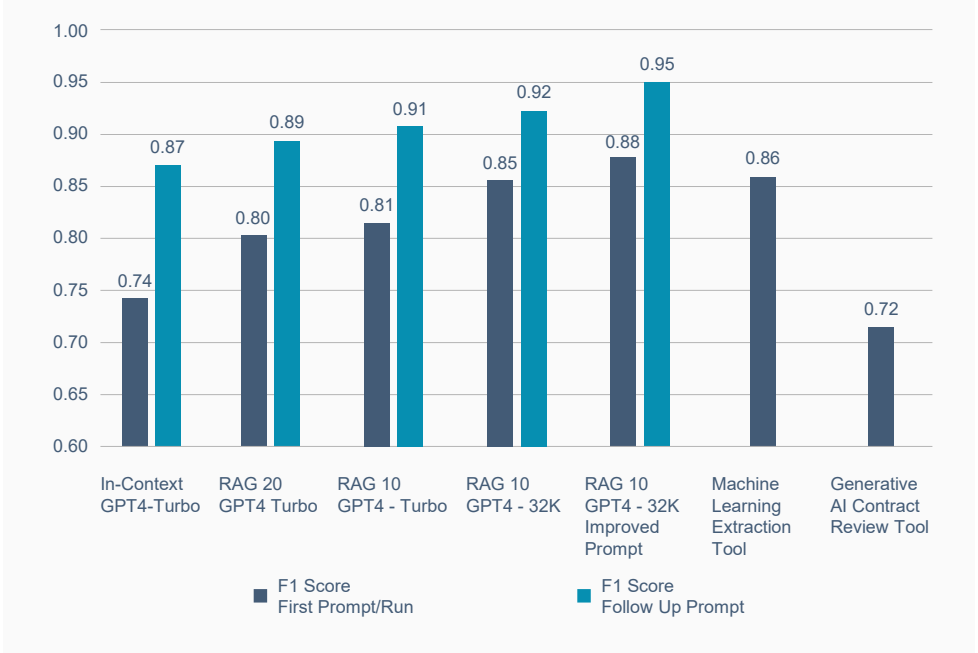


Figure 20. Average F1 Score by Configuration/Tool Across all Provisions.

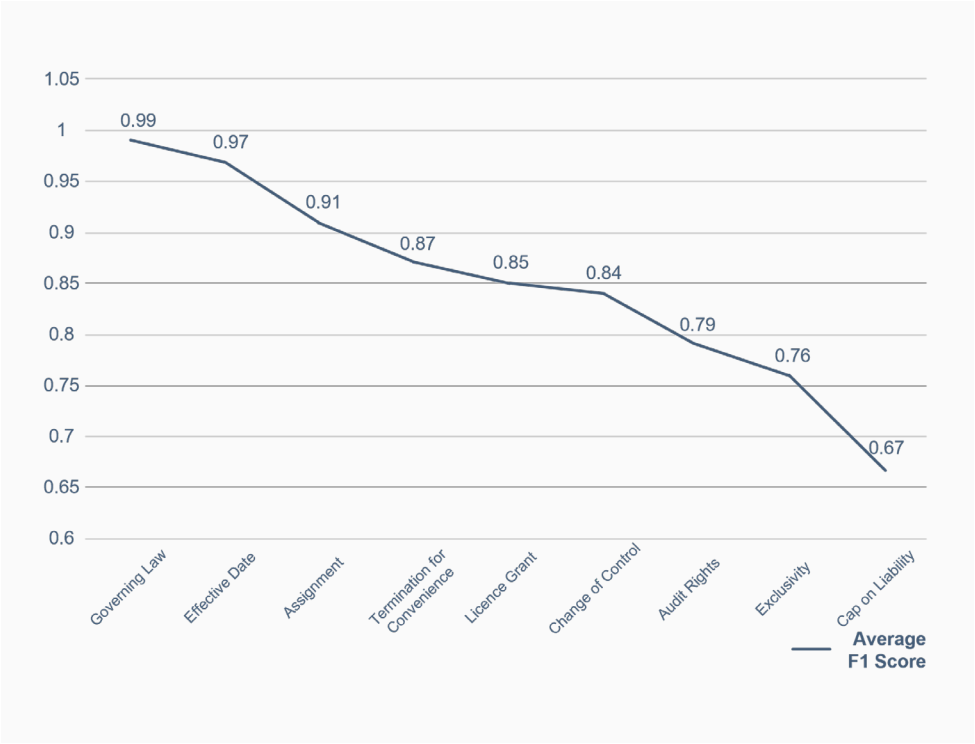


Figure 21. Average F1 Score by Provision across all Configurations and Tools.

GENERATION TESTS CONCLUSIONS

Drawing conclusions from these generation experiments, optimising a range of generation components and building on an optimised retrieval approach allows us to substantially improve the performance of LLMs for our task, as shown in our findings above. This performance improvement varies across different concepts, with some easier to extract than others due to the reasons we have covered.

We can see how Provisions characterised by more consistency and low variability in their formulations – such as Governing Law and Effective Date – are easier to extract. On the other hand, Provisions such as Exclusivity and Cap on Liability are characterised by greater variability and formulations and are more closely tied to the specific commercial content of the agreement. This seems to make them more challenging to extract. We see this across all tools and all configurations.

Beyond formulation, we have identified a range of additional factors that affect the ease of extraction, including the following:

- The location of the Provision in the agreement, as Provisions with less variance in their usual location, such as Effective Date – which almost always appears in the beginning of the agreement – are easier to extract.
- The length of the Provision – as Provisions that tend to be longer are more likely to only be partially extracted.
- Whether the Provision is a standalone Provision or part of a broader one.

These insights are not merely theoretical, but also operative as they help us to better define and characterise our target Provisions in a way that would facilitate their quality extraction.

We have seen that optimising specifically for certain concepts improves the performance in relation to those concepts, and we can reach levels of accuracy by doing this ourselves much more than by simply using a GenAI Contract Review Tool (0.72 vs 0.95 F1 Scores). We have also discovered that through this optimisation process we can get better results than by using traditional Machine Learning Extraction Tools, although these solutions are closer in performance (0.86 vs 0.95 F1 Scores). General prompt improvements also increase the accuracy of the output, which is promising as it means that we can use tools where we have less control over the retrieval components and apply our learning to improve the outputs solely through Prompt Engineering.

The fact that customising a tool for a particular use case improves performance may not be a surprise; however, it strengthens our belief that investing time and resource into building solutions ourselves is a good approach. This is a similar argument to the one we put forward in 2014 when we began training our own machine learning models within a third-party tool, as we could feed AG documents and content in and therefore train models to be more bespoke to us, rather than using the general models provided by the vendor.

CONCLUSION

Throughout this paper, we have discussed our journey as we set out to develop a platform for using LLMs in a real-life practical scenario. We wanted this scenario to go beyond what we had built with AGPT and be more than a simple wrapper for an LLM. We pursued this project ourselves in order to find a solution that balances flexibility, control, reliability and cost effectiveness. The development of our PoC and the findings along the way is a significant milestone on this journey.

Our research and experimentation have given us a valuable insight into the use of LLMs in legal-specific work and how we can pre-define their configurations for very specific use cases. We have shown that it is possible to optimise LLMs using a range of components to increase performance – in some cases by quite a margin. Our findings have highlighted the importance of Prompt Engineering, the use of follow-up prompts and the careful process of optimising retrieval components in increasing LLM performance. Our testing has shown that, through optimised retrieval techniques and improved prompting approaches, we can increase the accuracy of LLMs in commercial contract reviews from 74% to 95%, on average.

We have shown that, out of all the components we evaluated, the Chunking Strategy is one of the elements that demonstrated the most significant potential for improving accuracy. Combining a good Chunking Strategy with other retrieval components resulted in a clear accuracy improvement of between 14% and 22%. We discovered that giving LLMs a more detailed and bespoke message does improve the quality of its responses, but being too granular did have a detrimental effect. Humanising the information given to the models also showed some performance improvement – asking a model to pay extra attention and accusing it of missing relevant information both led to higher accuracy – with improvements of up to 16% in our experiments. We demonstrated that GPT4-32K performed better than GPT4-Turbo in the specific use case of extraction recall and generation. Overall, our results show that a RAG approach has major advantages over a Full In-Context configuration as you can accurately feed the LLM with the minimum amount of context needed.

Following our testing, we found the best performing configuration to be as follows:

1. Using Chunking Strategy 2.
2. Implementing a customised hybrid Retrieval Method combining both advanced keywords search and vector search.
3. Creating an optimised Advanced Keywords Query and Vector Search Query for each Provision.
4. Retrieving the Top 10 Chunks and feeding them back to an LLM in the order they appeared in the document.
5. Using GPT4-32K as the LLM for the task.
6. Setting the LLM Parameters as temperature 0, maximum tokens to 2,000, and a constant ‘seed’ value.
7. Drafting a targeted System Prompt that did not unduly increase the context length fed to the LLM.
8. Creating Provision Specific Prompts, improved by our findings in this research, that direct the LLM towards what it should be doing.
9. Employing a Follow Up Prompt asking the model to pay special attention to certain aspects and directly accusing it of missing information where necessary.

We aimed to provide concrete examples and give some context to the rhetoric in the market in relation to the use of LLMs for legal work. While we still have a long way to go before we can create some of the solutions we want, we are already seeing a lot of value from GenAI in the work we do every day. We would welcome any comments and feedback following this paper and hope that sharing this approach drives a wider discussion across law firms, legal service providers, in-house teams, legal tech solution providers and academics.

KEY TAKEAWAYS

Chunking Strategy. The best performing Chunking Strategy was the use of chunks 3,500 characters long, with an overlap of 700 characters either side, set out in more detail as Chunking Strategy 2 in the retrieval section above. This enabled us to break documents down into sensibly sized text excerpts, with the overlap allowing us to maintain clause structure and context for each specific chunk. We found bringing back 10 chunks was the sweet spot for retrieval with this Chunking Strategy, but believe that a more intelligent strategy in the future will lead to the retrieval of a more flexible range of chunks.

Retrieval Optimisation. We achieved substantial improvements by optimising the Retrieval Methods used throughout our PoC. We did this by giving more context to the LLM, adding related text to a Vector Search Query while also running an Advanced Keyword Query across the chunks which had specific weighting for the keywords that had most impact in retrieving a particular concept. We had expected this due to our work so far with LLMs, but it was useful to see the numbers and the scale of the improvement from our experiments. We also found that these optimisation methods have differing impacts across different concepts, with simple and unique facts such as Governing Law not benefiting as much from additional context as complex Provisions such as Exclusivity.

Prompt Engineering. The process of using well-crafted and intentional prompts to instruct LLMs adds an advantage and improves performance. This is reflected both in the System Prompt used to set the persona and task focus for the model and the Provision Specific Prompt that is either added for a use case or input by a user. This will still be an important skill in the future, but it may be that the end user only needs a surface level awareness of this due to solution developers applying best practice behind the scenes. We found that adding urgency or importance to the prompts did slightly increase performance, and this emotive style of prompting is something that we will continue to investigate. Although giving extra context and information to the LLMs works well, there is a limit; the complexity of the information can start to reduce the effectiveness of the wider prompt.

Follow Up Prompting. Using a follow up prompt improved the responses in all the scenarios we tested. This is especially true when we increased the pressure or accused the model of being incorrect. This is closely linked to the use of emotive prompting and the performance improvement when using stronger or more emotive language. Being able to systemise Follow Up Prompting and take it away from the user will be crucial, as we want to avoid users having to constantly follow up on prompts. Instead, we want to build this into the system to always push a second request to the LLM. There is also the opportunity to ask the model several times and take the best response, in comparison to some clear scoring metrics.

Hallucinations. These didn't pose a problem throughout our testing, potentially due to the strict nature of our experiments and the task we were carrying out. All of the optimisation steps we took seemed to reduce elements of hallucination. Perhaps the most impactful step we took was giving the model a clear option where it couldn't find information, as well as being very specific about the expected input and therefore the task itself. Fundamentally, we know that there is a near-zero chance of removing hallucination risk from the use of LLMs completely. However, our intentionality in the use case, the nature of our task being extraction focused, the optimisation steps we took and the Prompt Engineering we carried out all resulted in a much more accurate response. We have seen that without at least some level of Prompt Engineering and improvement, there is a real risk of hallucinations and general poor performance with LLMs, especially where you have no visibility or control over the retrieval components or System Prompts in a third-party tool.

Traditional Machine Learning. ML extraction is still effective at finding and extracting clauses; however, a well optimised retrieval approach using LLMs is close to or on a par with this performance. There is an added advantage to using GenAI as it is possible to add new Provisions on the go, rather than labelling examples to run a supervised machine learning process, with the only overhead being the drafting of specific prompts. We can get to an answer quicker using GenAI, but this is only a clear advantage for bespoke extractions as most of the solutions in the market have a large list of pre-trained concepts. An additional benefit with GenAI is the ability to get to the next stage of querying the extractions to identify risks – this is the next focus of our research and our work to date has shown this to be effective. Using LLMs, we can create specific risk query prompts that get us to an answer, rather than just flagging the language a human would need to check.

Subject Matter Expertise. Expert knowledge is a valuable commodity when dealing with GenAI, at least for now as we are working with general models. The importance of prompting and optimising search queries as shown in this paper means that subject matter expertise in the area where GenAI is being deployed is crucial. We can call on a wealth of knowledge across our lawyers and within our Innovation team at AG, which has meant that we have been able to optimise our approach and guide an LLM towards more accurate, useful and AG-specific outputs. Consequently, we have been able to achieve satisfactory results by adding our domain knowledge into the inputs for the model, rather than focusing on fine tuning. We don't believe that fine tuning a model would help to create the solution we are working towards, but this is partly due to the technical hurdles, costs and time investment needed. A more domain-specific Legal-LLM may enter the market in the next few years, and it will be interesting to test this with this research in mind.

NEXT STEPS

This paper covers the start of our journey and is intended to share some of our findings we have discovered along the way. We will be taking forward a number of aspects, both to enhance our internal tools and to increase our understanding around GenAI. This is a space that is moving very quickly, and there is a certain amount of effort needed to just keep up with new innovations.

Our key focuses for the coming year will be to continue our technical learning, incorporating other LLMs into our testing stack and potentially into production, building our diligence offering and rolling it out across the firm, and to keep delivering on our specific use cases, whether through AGPT, our M&A Transaction Platform or our range of third-party tools. From a technical perspective, there is a need to enhance our use of Azure, and we are in the process of setting up a sandbox with Google Vertex, which also involves upgrading our Embedding Models. We are investigating the potential of fine-tuning and would love to test the output of this compared to the methods we have used throughout this paper.

An ongoing challenge is the more intelligent chunking of documents and an improved search. We have had these challenges for a long time – particularly in relation to working on clause banks and playbooks – and we are having some very useful conversations about how to solve them. We are looking at approaches such as Hypothetical Document Embeddings, graph databases and natural language chunk relationships. In the future, we hope to deploy some more sophisticated Chunking Strategies, or potentially combine some extraction machine learning tools to help build out a better Retrieval Method.

As always, we are excited to work with others across the legal industry and continue to work closely with our vendors, academic partners, clients and other law firms through consortiums and working groups. We will continue to share our journey and are always open for conversations and questions.

APPENDIX 1: WORKED EXAMPLE

This worked example shows the process for identifying a termination for change of control risk and applying the approaches set out in this paper.

VECTOR SEARCH QUERY

Termination Upon Change of Control. Notwithstanding anything to the contrary herein, this Agreement (excluding any then-existing obligations) shall terminate upon (a) the acquisition of the Company by another entity by means of any transaction or series of related transactions to which the Company is party (including, without limitation, any stock acquisition, reorganisation, merger or consolidation but excluding any sale of stock for capital raising purposes) other than a transaction or series of transactions in which the holders of the voting securities of the Company outstanding immediately prior to such transaction continue to retain (either by such voting securities remaining outstanding or by such voting securities being converted into voting securities of the surviving entity), as a result of shares in the Company held by such holders prior to such transaction, at least fifty percent (50%) of the total voting power represented by the voting securities of the Corporation or such surviving entity outstanding immediately after such transaction or series of transactions; or (b) a sale, lease or other conveyance of all substantially all of the assets of the Company.

ADVANCED KEYWORDS

["change control"~5^5, "control changed"~5^5, "merger consolidation"~10^2, "sale transfer"~10^2, "change ownership"~10^2, "sale substantially"~10^2, "assets substantially"~10^2, "assignment transfer"~10^2, "sale assets"~10^2, "sale merger"~10^2, "transfer interest"~10^2, "business transfer"~10^2, "ownership transfer"~10^2, "transfer assign"~10^2, "management change"~10^2, "written notice"~5, "written consent"~5].

SYSTEM PROMPT

You are a UK Lawyer specialising in Corporate and Commercial law. Your expertise is in performing legal due diligence in the context of M&A and investment transactions, which means reviewing and analysing different types of agreements and providing answers to due diligence-related questions about the content of such agreements. You will be provided with one or more text excerpts taken from a single agreement as well as a specific question that will follow such text excerpts. Each of the text excerpts as well as the question will be delimited by triple hashtags ('###'). Your task is to review and analyse each one of the provided text excerpts in light of the question that follows them and then provide a precise and accurate answer to the question based on the information in the text excerpts. Your answer must be based only on information appearing in the text excerpts, and you must avoid making any assumptions or providing speculative information. Do not use markdown. Only use plain text.

TEXT CHUNKS

###CHUNK 1###

###CHUNK 2###

###CHUNK 3###

PROVISION SPECIFIC PROMPT

In the context of Contract Law, Change of Control provisions are contractual provisions that specify the rights, obligations, and consequences that arise when there is a significant alteration in the ownership or management structure of one of the contracting parties. These clauses typically define what constitutes a change of control event, which may include mergers, acquisitions, asset sales, or shifts in voting power. Change of Control provisions can grant certain rights to the non-changing party, such as the ability to terminate the agreement, renegotiate terms, or receive compensation. They may also impose obligations on the party undergoing the change, such as providing notice or seeking approval from the other party.

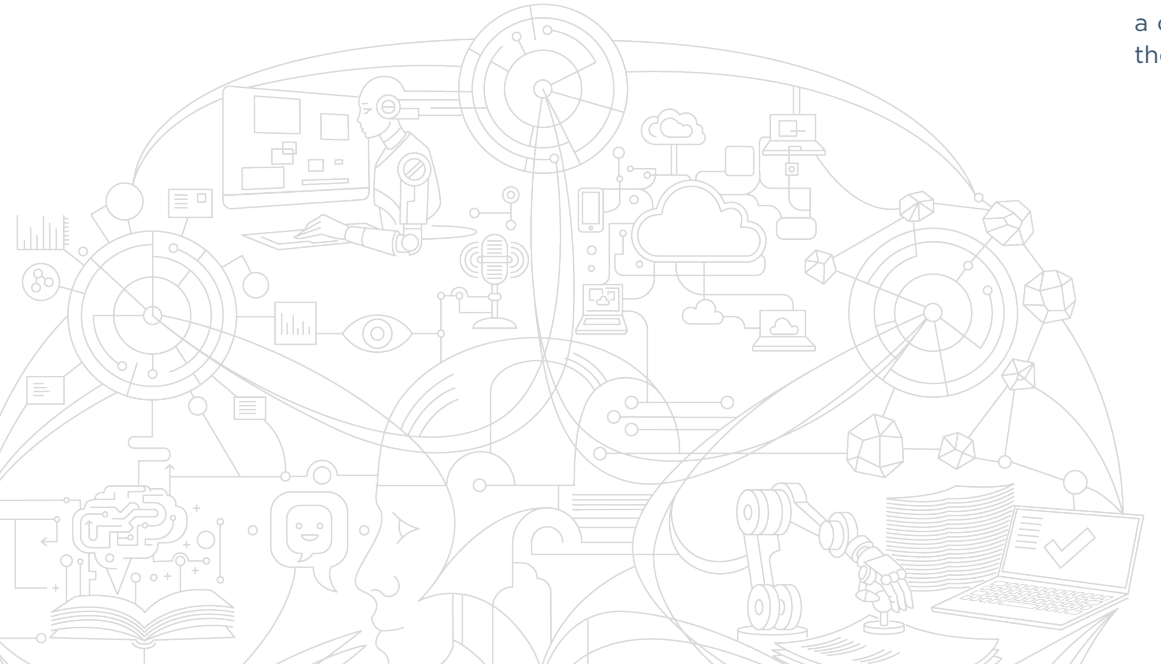
Does this agreement contain any Change of Control provisions? If so, please extract all such provisions verbatim and in their entirety. In case you are unable to identify any Change of Control provisions, please respond with 'Not found'.

GENERAL ADDITIONAL INSTRUCTIONS

Make sure to review and analyse each provided excerpt thoroughly. Relevant information for answering the question may be found within different parts of a given excerpt, as well as across multiple excerpts. You should assume that every provision that contains any of the following terms is relevant and should be thoroughly examined: 'change of control', 'change in control', 'merger', 'consolidation', 'sale', 'transfer', 'ownership', 'ownership change', 'assignment', 'assets', 'management', 'substantially', 'interest', 'business', 'assigned', 'assign', and 'delegate'. Make sure to take your time and think step-by-step before providing your answer. Where applicable, make sure to start each part of your answer with a reference to the specific clause, subclause, section or subsection from which the information is taken. Do not provide any introductions, just respond with the relevant provisions or with 'Not found'.

OUTPUT

Clause 2.4 of the agreement contains Change of Control provisions: "Either party may terminate this Agreement at any time by giving not less than four weeks' notice in writing to the other party if the other party undergoes a change of control, or if it is realistically anticipated that it will undergo a change of control within two months. For the purposes of this clause 2.4, 'control' shall mean beneficial ownership of more than 50% of the issued share capital of a company or the legal power to direct or cause the direction of management of the company."



RISK IDENTIFICATION PROMPT

Is there an ability for either party to terminate this agreement upon a change of control or ownership of the other party?

Respond in the following format:

If the answer is yes, output 'Yes' and the name of the relevant party that has the right to terminate, for example, 'Yes - [Party Name]', or 'Yes - Both parties'. If the answer is No, simply output 'No' and no other text. Then provide the reasoning behind your answer: 'Reasoning: [the specific reason behind your answer].' Then provide the text you have used to come to this answer, do not change this text in any way just reproduce it exactly: 'Source Text: [text reproduced from agreement].' If the answer is No to the question above, just insert 'Text Not Found'. Do not start your answer with any introductory text. Follow the schema set out above.

RISK OUTPUT

Yes - Both parties

Reasoning: The agreement explicitly states that either party may terminate the agreement if the other party undergoes a change of control, with a detailed definition of what constitutes a change of control.

Source Text: "Either party may terminate this Agreement at any time by giving not less than four weeks' notice in writing to the other party if the other party undergoes a change of control, or if it is realistically anticipated that it will undergo a change of control within two months. For the purposes of this clause 2.4, 'control' shall mean beneficial ownership of more than 50% of the issued share capital of a company or the legal power to direct or cause the direction of management of the company."

APPENDIX 2: GENERATION EXPERIMENT RESULTS

GENERATION EXPERIMENTS OUTPUT TABLE

Configuration/Tool	# of Samples	Accuracy	Precision	Recall	F1 Score
Average					
In-Context - GPT4-Turbo - First Prompt	-	0.69	1.00	0.63	0.74
In-Context - GPT4-Turbo - Follow Up Prompt	-	0.82	1.00	0.79	0.87
RAG 20 - GPT4-Turbo - First Prompt	-	0.74	1.00	0.70	0.80
RAG 20 - GPT4-Turbo - Follow Up Prompt	-	0.84	0.99	0.83	0.89
RAG 10 - GPT4-Turbo - First Prompt	-	0.75	1.00	0.71	0.81
RAG 10 - GPT4-Turbo - Follow Up Prompt	-	0.86	0.99	0.85	0.91
RAG 10 - GPT4-32K - First Prompt	-	0.79	1.00	0.76	0.85
RAG 10 - GPT4-32K - Follow Up Prompt	-	0.88	1.00	0.86	0.92
RAG 10 - GPT4-32K - Improved Prompt - First Prompt	-	0.82	1.00	0.79	0.88
RAG 10 - GPT4-32K - Improved Prompt - Follow Up Prompt	**-**	**0.92**	**1.00**	**0.90**	**0.95**
Machine Learning Extraction Tool	-	0.79	1.00	0.76	0.86
GenAI Contract Review Tool	-	0.64	0.97	0.60	0.72
Assignment					
In-Context - GPT4-Turbo - First Prompt	47	0.77	1.00	0.73	0.84
In-Context - GPT4-Turbo - Follow Up Prompt	47	0.87	1.00	0.85	0.92
RAG 20 - GPT4-Turbo - First Prompt	47	0.85	1.00	0.83	0.90
RAG 20 - GPT4-Turbo - Follow Up Prompt	**47**	**0.91**	**1.00**	**0.90**	**0.95**
RAG 10 - GPT4-Turbo - First Prompt	47	0.87	1.00	0.85	0.92
RAG 10 - GPT4-Turbo - Follow Up Prompt	**47**	**0.91**	**1.00**	**0.90**	**0.95**
RAG 10 - GPT4-32K - First Prompt	47	0.89	1.00	0.88	0.93
RAG 10 - GPT4-32K - Follow Up Prompt	**47**	**0.91**	**1.00**	**0.90**	**0.95**
Machine Learning Extraction Tool	**47**	**0.91**	**1.00**	**0.90**	**0.95**
GenAI Contract Review Tool	47	0.68	1.00	0.63	0.77
Audit Rights					
In-Context - GPT4-Turbo - First Prompt	74	0.49	1.00	0.43	0.60
In-Context - GPT4-Turbo - Follow Up Prompt	74	0.69	1.00	0.66	0.79
RAG 20 - GPT4-Turbo - First Prompt	74	0.57	1.00	0.52	0.69
RAG 20 - GPT4-Turbo - Follow Up Prompt	74	0.69	1.00	0.66	0.79
RAG 10 - GPT4-Turbo - First Prompt	74	0.61	1.00	0.57	0.72
RAG 10 - GPT4-Turbo - Follow Up Prompt	74	0.76	1.00	0.73	0.84
RAG 10 - GPT4-32K - First Prompt	74	0.68	1.00	0.64	0.78
RAG 10 - GPT4-32K - Follow Up Prompt	74	0.82	1.00	0.81	0.89
RAG 10 - GPT4-32K - Improved Prompt - First Prompt	74	0.84	1.00	0.82	0.90

Configuration/Tool	# of Samples	Accuracy	Precision	Recall	F1 Score
RAG 10 - GPT4-32K - Improved Prompt - Follow Up Prompt	**74**	**0.88**	**1.00**	**0.87**	**0.93**
Machine Learning Extraction Tool	74	0.77	1.00	0.75	0.85
GenAI Contract Review Tool	74	0.54	1.00	0.49	0.66
Cap on Liability					
In-Context - GPT4-Turbo - First Prompt	37	0.32	1.00	0.17	0.29
In-Context - GPT4-Turbo - Follow Up Prompt	37	0.57	1.00	0.47	0.64
RAG 20 - GPT4-Turbo - First Prompt	37	0.43	1.00	0.30	0.46
RAG 20 - GPT4-Turbo - Follow Up Prompt	37	0.70	1.00	0.63	0.78
RAG 10 - GPT4-Turbo - First Prompt	37	0.43	1.00	0.30	0.46
RAG 10 - GPT4-Turbo - Follow Up Prompt	37	0.70	1.00	0.63	0.78
RAG 10 - GPT4-32K - First Prompt	37	0.65	1.00	0.57	0.72
RAG 10 - GPT4-32K - Follow Up Prompt	37	0.78	1.00	0.73	0.85
RAG 10 - GPT4-32K - Improved Prompt - First Prompt	37	0.78	1.00	0.73	0.85
RAG 10 - GPT4-32K - Improved Prompt - Follow Up Prompt	**37**	**0.92**	**1.00**	**0.90**	**0.95**
Machine Learning Extraction Tool	37	0.68	1.00	0.60	0.75
GenAI Contract Review Tool	37	0.46	1.00	0.33	0.50
Change of Control					
In-Context - GPT4-Turbo - First Prompt	46	0.74	1.00	0.70	0.82
In-Context - GPT4-Turbo - Follow Up Prompt	**46**	**0.85**	**1.00**	**0.83**	**0.90**
RAG 20 - GPT4-Turbo - First Prompt	46	0.80	1.00	0.78	0.87
RAG 20 - GPT4-Turbo - Follow Up Prompt	**46**	**0.85**	**1.00**	**0.83**	**0.90**
RAG 10 - GPT4-Turbo - First Prompt	46	0.72	1.00	0.68	0.81
RAG 10 - GPT4-Turbo - Follow Up Prompt	**46**	**0.85**	**1.00**	**0.83**	**0.90**
RAG 10 - GPT4-32K - First Prompt	46	0.70	1.00	0.65	0.79
RAG 10 - GPT4-32K - Follow Up Prompt	**46**	**0.85**	**1.00**	**0.83**	**0.90**
RAG 10 - GPT4-32K - Improved Prompt - First Prompt	46	0.70	1.00	0.65	0.79
RAG 10 - GPT4-32K - Improved Prompt - Follow Up Prompt	**46**	**0.85**	**1.00**	**0.83**	**0.90**
Machine Learning Extraction Tool	46	0.78	1.00	0.75	0.86
GenAI Contract Review Tool	46	0.50	0.87	0.50	0.63
Effective Date					
In-Context - GPT4-Turbo - First Prompt	**25**	**1.00**	**1.00**	**1.00**	**1.00**
In-Context - GPT4-Turbo - Follow Up Prompt	**25**	**1.00**	**1.00**	**1.00**	**1.00**
RAG 20 - GPT4-Turbo - First Prompt	25	0.96	1.00	0.95	0.97
RAG 20 - GPT4-Turbo - Follow Up Prompt	**25**	**1.00**	**1.00**	**1.00**	**1.00**
RAG 10 - GPT4-Turbo - First Prompt	25	0.96	1.00	0.95	0.97
RAG 10 - GPT4-Turbo - Follow Up Prompt	25	0.96	0.95	1.00	0.97
RAG 10 - GPT4-32K - First Prompt	25	0.88	1.00	0.85	0.92
RAG 10 - GPT4-32K - Follow Up Prompt	**25**	**1.00**	**1.00**	**1.00**	**1.00**
Machine Learning Extraction Tool	25	0.84	1.00	0.80	0.89
GenAI Contract Review Tool	25	0.92	1.00	0.9	0.95
Exclusivity					
In-Context - GPT4-Turbo - First Prompt	52	0.62	1.00	0.57	0.72
In-Context - GPT4-Turbo - Follow Up Prompt	52	0.69	1.00	0.65	0.79

Configuration/Tool	# of Samples	Accuracy	Precision	Recall	F1 Score
RAG 20 - GPT4-Turbo - First Prompt	52	0.58	1.00	0.52	0.69
RAG 20 - GPT4-Turbo - Follow Up Prompt	52	0.71	1.00	0.67	0.81
RAG 10 - GPT4-Turbo - First Prompt	52	0.65	1.00	0.61	0.76
RAG 10 - GPT4-Turbo - Follow Up Prompt	52	0.79	1.00	0.76	0.86
RAG 10 - GPT4-32K - First Prompt	52	0.67	1.00	0.63	0.77
RAG 10 - GPT4-32K - Follow Up Prompt	52	0.69	1.00	0.65	0.79
RAG 10 - GPT4-32K - Improved Prompt - First Prompt	52	0.67	1.00	0.63	0.77
RAG 10 - GPT4-32K - Improved Prompt - Follow Up Prompt	**52**	**0.83**	**1.00**	**0.80**	**0.89**
Machine Learning Extraction Tool	52	0.75	1.00	0.72	0.84
GenAI Contract Review Tool	52	0.31	0.92	0.24	0.38
Governing Law					
In-Context - GPT4-Turbo - First Prompt	**24**	**1.00**	**1.00**	**1.00**	**1.00**
In-Context - GPT4-Turbo - Follow Up Prompt	**24**	**1.00**	**1.00**	**1.00**	**1.00**
RAG 20 - GPT4-Turbo - First Prompt	**24**	**1.00**	**1.00**	**1.00**	**1.00**
RAG 20 - GPT4-Turbo - Follow Up Prompt	**24**	**1.00**	**1.00**	**1.00**	**1.00**
RAG 10 - GPT4-Turbo - First Prompt	**24**	**1.00**	**1.00**	**1.00**	**1.00**
RAG 10 - GPT4-Turbo - Follow Up Prompt	**24**	**1.00**	**1.00**	**1.00**	**1.00**
RAG 10 - GPT4-32K - First Prompt	**24**	**1.00**	**1.00**	**1.00**	**1.00**
RAG 10 - GPT4-32K - Follow Up Prompt	**24**	**1.00**	**1.00**	**1.00**	**1.00**
Machine Learning Extraction Tool	24	0.92	1.00	0.92	0.96
GenAI Contract Review Tool	24	0.88	1.00	0.88	0.93
Licence Grant					
In-Context - GPT4-Turbo - First Prompt	69	0.64	1.00	0.60	0.75
In-Context - GPT4-Turbo - Follow Up Prompt	69	0.77	1.00	0.74	0.85
RAG 20 - GPT4-Turbo - First Prompt	69	0.72	1.00	0.69	0.82
RAG 20 - GPT4-Turbo - Follow Up Prompt	69	0.81	1.00	0.79	0.88
RAG 10 - GPT4-Turbo - First Prompt	69	0.72	1.00	0.69	0.82
RAG 10 - GPT4-Turbo - Follow Up Prompt	69	0.83	1.00	0.81	0.89
RAG 10 - GPT4-32K - First Prompt	69	0.78	1.00	0.76	0.86
RAG 10 - GPT4-32K - Follow Up Prompt	**69**	**0.90**	**1.00**	**0.89**	**0.94**
RAG 10 - GPT4-32K - Improved Prompt - First Prompt	69	0.75	1.00	0.73	0.84
RAG 10 - GPT4-32K - Improved Prompt - Follow Up Prompt	**69**	**0.90**	**1.00**	**0.89**	**0.94**
Machine Learning Extraction Tool	69	0.74	1.00	0.71	0.83
GenAI Contract Review Tool	69	0.65	1.00	0.61	0.76
Termination For Convenience					
In-Context - GPT4-Turbo - First Prompt	34	0.62	1.00	0.48	0.65
In-Context - GPT4-Turbo - Follow Up Prompt	34	0.91	1.00	0.88	0.94
RAG 20 - GPT4-Turbo - First Prompt	34	0.76	1.00	0.68	0.81
RAG 20 - GPT4-Turbo - Follow Up Prompt	34	0.91	0.92	0.96	0.94
RAG 10 - GPT4-Turbo - First Prompt	34	0.82	1.00	0.76	0.86
RAG 10 - GPT4-Turbo - Follow Up Prompt	34	0.94	0.96	0.96	0.96
RAG 10 - GPT4-32K - First Prompt	34	0.88	1.00	0.84	0.91
RAG 10 - GPT4-32K - Follow Up Prompt	**34**	**0.97**	**1.00**	**0.96**	**0.98**
Machine Learning Extraction Tool	34	0.76	1.00	0.68	0.81
GenAI Contract Review Tool	34	0.79	0.91	0.8	0.85

MORE IMAGINATION **MORE IMPACT**

addleshawgoddard.com