

9 Major Technical Challenges that Come With Using Generative AI for Contracts

A guide to achieving real business results with Contract AI

A WHITE PAPER BY PEDRAM ABRARI, CTO, PRAMATA

Generative AI is increasingly a part of life and business for a large number of the country's knowledge workers. As such, in-house legal teams are understandably eager to reap the benefits of generative AI, particularly when it comes to dealing with an overwhelming number of contracts and other business-critical documents.

While some organizations have been quick to dive into generative AI with the goal of unlocking its potential to transform the world of legal and contracts, many are facing disappointing results.

Legal teams and the technical resources at their companies are quickly learning it's not as simple as "Contract GPT."

Applying generative AI (GenAI) to contracts presents a unique set of engineering challenges that extend well beyond typical natural language processing tasks. For technical leaders and implementation teams, understanding these challenges is critical to successfully deploying AI for contract analysis at enterprise scale. Success requires mastery of three interconnected disciplines: Context Engineering (how data is structured and retrieved), Prompt Engineering (how instructions are crafted), and Quality Engineering (how accuracy is validated and maintained).

Contracts and their unique challenges

With the advancements of popular large language models (LLMs) over the past several years, legal teams and others who work with contracts might be tempted to load contracts into the LLM of their choice and ask questions about their contents. Confidentiality, data security, and AI training concerns aside, this method may appear effective initially but when used for larger contracts and sets of related contracts, the legal teams are left to fend for themselves. The biggest challenge for legal teams is when the LLM produces suboptimal or factually incorrect responses: a phenomenon known as AI hallucination.

Why do LLMs that seem so capable of producing accurate and useful outputs not do the same with contracts?

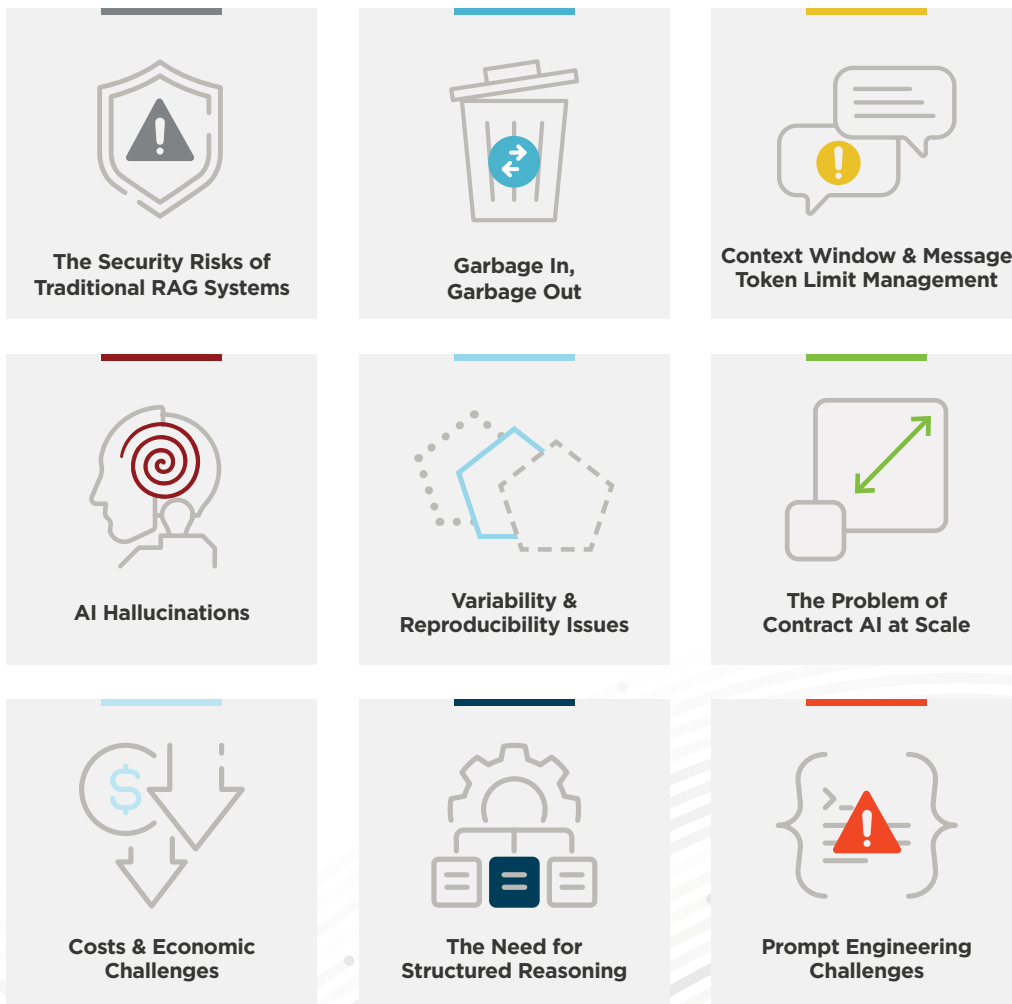
Contracts are not merely text documents. They are highly structured, legally binding instruments with complex internal relationships, specialized terminology, and intricate interdependencies. These distinct characteristics create specific computational constraints that GenAI must address through specialized technical and architectural approaches.

While the theoretical promise of generative AI for contract analysis is compelling, the practical reality has proven far more challenging. Many organizations that have attempted to apply general-purpose AI models to contract analysis have encountered persistent technical limitations that prevent reliable, accurate results at enterprise scale. Most critically, traditional Retrieval-Augmented Generation (RAG) approaches using vector databases create unacceptable security risks by requiring sensitive contract data to be stored in unencrypted form.

» **This whitepaper examines the core challenges of using LLMs and GenAI for contract creation, management and analysis. By understanding these fundamental technical obstacles and how the right technology can address them, tech leaders can make informed decisions about implementing Contract AI within their organizations, rather than letting legal teams and others continue to experiment with contracts in the “wild west” of GenAI.**

Nine fundamental Contract AI obstacles

To successfully use AI to draft, amend, organize, manage, and analyze contracts across an entire organization, Contract AI must address each of these nine fundamental challenges. If these are not addressed, the Contract AI system will hit a critical failure point as it tries to scale beyond basic use-cases, rendering it inaccurate and even useless to legal, sales, procurement, finance, and other teams who rely on enterprise-grade, accessible contract data.





1. The Security Risk of Traditional RAG Systems

Before addressing operational challenges, we must confront a fundamental security risk that can stop Contract AI adoption before it starts. Traditional Retrieval-Augmented Generation (RAG) systems, used by most Contract AI vendors, create an unacceptable security vulnerability for enterprise deployments.

Traditional RAG systems follow this workflow:

1. Decrypt your contract documents
2. Convert text to numerical vectors (embeddings)
3. Store vectors in a database for semantic search
4. Retrieve similar content based on user queries
5. Generate AI responses using retrieved context

The critical vulnerability: Step 2 requires your sensitive contract data to be processed and stored in unencrypted form. To create semantic embeddings, AI systems must decrypt your contracts, process every confidential detail through embedding models, and store the resulting vectors—creating an unacceptable security exposure for contracts containing pricing information, confidential arrangements, and strategic partnerships.

The Enterprise Reality Check

CISO: “Does this system require decrypting our contract data?”

Vendor: “Well, yes, but only to create the embeddings...”

CISO: “Hard pass.”

This conversation happens daily in enterprise security reviews. The fundamental architecture of vector-based RAG creates a non-starter for organizations with mature security policies.

At Pramata, we mitigate this critical security risk using RAG-E technology, which maintains encryption throughout the pipeline. Instead of semantic search through decrypted data, RAG-E uses structured Contract Intelligence and Pramata Prompt Language variables to enable precise, encrypted retrieval. This Context Engineering approach means your contracts stay encrypted, eliminating the vector database vulnerability while meeting enterprise security requirements.

Pramata addresses the “RAG Security Risk” challenge with the following innovations: [TrueDoc OCR](#), [Files Cleanse](#), [Documents Organize](#), [Extract & Enrich](#), [TrueCheck QA](#), [Relationship Object Model](#), [Clause Analysis Models](#), [Templates & Playbooks](#), [Contract Families](#), [Contract Status & Renewals](#), [Pramata Prompt Language](#), [AI Agents \(and Control-Flow\)](#)



2. Garbage In, Garbage Out

The saying “garbage in, garbage out” has never been more relevant than with Contract AI. If your contract storage is messy, cluttered, distributed across multiple systems, full of duplicates and non-contract files, and full of PDFs that can’t be digitally read, there’s no chance of achieving value from Contract AI.

These specific points are usually at the root of a business’s contract chaos, and all undermine even the best GenAI’s performance:

- ▲ **Duplicates, non-contract files and versioning confusion:** When multiple versions of the same contract exist, which one should the AI analyze? What about all the documents a company collects that aren’t contracts at all? When duplicates and non-contract files are loaded into a system, AI may analyze irrelevant documents and bring extraneous or outdated information into its outputs.
- ▲ **Unreadable documents:** Many legacy contracts exist only as scanned images or poorly converted PDFs where text can’t be properly extracted or interpreted. They may also have complex tables that AI struggles to read, reproduce and understand.
- ▲ **Disorganized relationships:** Contracts don’t exist in isolation. They have complex relationships with a specific order of precedence, that includes master agreements, amendments, orders, and related documents. If these commercial relationships aren’t properly established into the right hierarchy, AI can’t provide accurate analysis.
- ▲ **Fragmented storage:** When contracts are scattered across different systems, shared drives, and email inboxes, it would take a significant amount of time and manual effort to collect, organize and provide even the most sophisticated AI what it needs to try and understand the full picture of the commercial relationship.

Poor data quality (i.e., “garbage in”) has a profound impact on the results of any GenAI. If your starting point is a collection of disorganized contracts with duplicates, non-contract files, and missing or unreadable documents, your ContractAI will fail to provide accurate and reliable answers. This is where Quality Engineering becomes essential—establishing clean, validated data as the foundation for AI processing.

Pramata addresses the “garbage in, garbage out” challenge with the following innovations: [TrueDoc OCR](#), [Files Cleanse](#), [Documents Organize](#), [Extract & Enrich](#), [TrueCheck QA](#), [Relationship Object Model](#), [Clause Analysis Models](#), [Templates & Playbooks](#), [Contract Families](#), [Contract Status & Renewals](#)



3. Context Window & Message Token Limit Management

One of the most significant technical constraints affecting GenAI's ability to analyze contracts effectively is the "context window and token limitation". This refers to the finite amount of information an AI model can process in a single instance. This is essentially, the AI's working memory.

To use an easy-to-grasp analogy, think of GenAI's context window as a pitcher of water: It has a finite capacity and will eventually overflow. As you engage with GenAI, it's as though you're adding water to your pitcher every time you or the AI add, or produce, more information.

This includes:

- ▲ Your original data, for example, the text of a contract (or multiple contracts)
- ▲ Your instructions, or prompts
- ▲ The AI's thought process
- ▲ The AI's responses
- ▲ Your follow-up questions

The size of your "water pitcher" depends on how sophisticated your LLM is. To give some perspective, current state-of-the-art LLMs have context windows and token limits of up to about 200,000 tokens, which are roughly equivalent to words.

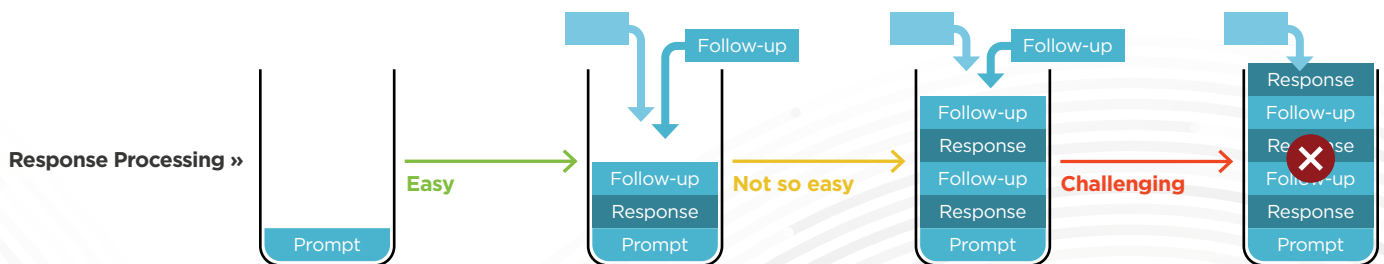
While this might seem substantial, contract-heavy scenarios can quickly exhaust an LLM's capacity. What if a master agreement is dozens of pages, or even a hundred pages long? What about complex contractual relationships that involve multiple, interconnected documents? It's easy to see how quickly you could fill up your AI's context window just by feeding it your source material, forget about asking questions!

What goes wrong when you exhaust a context window?

If you're looking for accurate answers from your Contract AI: *everything*.

When an LLM's context window is loaded beyond a particular threshold, depending on the LLM, anywhere over 70-80 percent of its maximum size, response accuracy begins to degrade resulting in suboptimal answers, and even downright hallucinations.

What makes context window limitations particularly troublesome is that the user has no clue when the LLM is taking shortcuts and producing suboptimal responses. For contracts, where precision is paramount, this behavior creates a dangerous scenario in which the AI may deliver confident-sounding answers that may be fundamentally flawed due to context management issues—and worse yet the user is oblivious to this pitfall! This is where Context Engineering becomes critical—using structured data retrieval and smart variables to minimize context consumption while maximizing relevant information.



Pramata addresses the "Context Window & Token Limits" challenge with the following innovations: Documents Organize, Extract & Enrich, Relationship Object Model, Clause Analysis Models, Templates & Playbooks, Contract Families, Pramata Prompt Language, AI Agents (and Control-Flow), Context Window Overflow Monitoring

“It’s important to understand that generative AI is wired to provide a response to a user query, at almost all costs, even if the response is factually false! Depending on the LLM used, anywhere from 70-80% of context window capacity, is considered the beginning of the danger zone. This is where AI may start to generate suboptimal responses. The closer the context window is to being full, the more the AI takes shortcuts. Of course, once the context window actually overflows then there is no response. But until an actual overflow occurs, the end user is clueless to the fact that the LLM is spewing suboptimal answers or fully hallucinating”

Pedram Abrari, Chief Technology Officer, Pramata



4. AI Hallucinations

“Hallucinations” are instances when LLMs generate information that isn’t grounded in reality. They represent one of the most serious challenges businesses face when they try to apply GenAI to contracts. For general content creation, hallucinations might be merely problematic, but for contract creation, redlines, management, or analysis, they can lead to significant financial and legal risks.

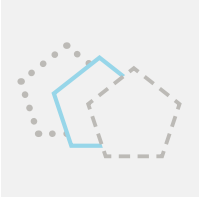
When an AI hallucinates while analyzing a contract, it might confidently state that a termination clause requires 90 days’ notice when the actual requirement is 30 days, or claim a contract contains a price escalation clause that doesn’t exist at all. What makes this particularly dangerous is that hallucinated content often sounds entirely plausible and can be difficult to detect without manually checking against the original documents.

Several factors can trigger hallucinations in contract analysis:

- ▲ **Context window overflows:** When the AI’s working memory fills up, it may begin fabricating information rather than admitting it’s reached its limits.
- ▲ **Knowledge gaps:** If the AI encounters contract concepts or clauses it doesn’t fully understand, it may “fill in the blanks” with plausible but incorrect interpretations.
- ▲ **Pattern completion tendencies:** AI models are designed to recognize and complete patterns, which can lead them to generate clauses that sound like typical contract language but aren’t actually in the document.
- ▲ **Ambiguous contract language:** When contract language is unclear or inconsistent, AI systems may make assumptions about the contract’s intended meaning that aren’t supported by the text.

The most dangerous aspect of hallucinations is that they’re often invisible to the user. Without careful verification and QA, organizations might make critical business decisions based on entirely fabricated contract terms, which is an unacceptable risk. This is where our three-pillar approach becomes essential: Context Engineering provides accurate data retrieval, Prompt Engineering guides precise analysis, and Quality Engineering validates results through our TrueCheck QA—achieving true 99% accuracy rather than claimed accuracy.

Pramata addresses the “AI Hallucinations” challenge with the following innovations: [TrueDoc OCR](#), [Files Cleanse](#), [Documents Organize](#), [Extract & Enrich](#), [TrueCheck QA](#), [Relationship Object Model](#), [Clause Analysis Models](#), [Templates & Playbooks](#), [Contract Families](#), [Contract Status & Renewals](#), [Pramata Prompt Language](#), [AI Agents \(and Control-Flow\)](#), [Context Window Overflow Monitoring](#), [Few-Shot Prompting Support](#), [Thought Process Support](#)



5. Variability & Reproducibility Issues

Unlike traditional software systems that produce consistent, deterministic results, generative AI models exhibit significant variability in their outputs. This characteristic poses particular challenges for contract analysis, where consistency and reliability are essential.

In the context of contracts, this variability creates several serious problems:

- ▲ **Inconsistent interpretations:** The same contract clause analyzed multiple times might yield different interpretations, making it difficult to establish a definitive understanding of rights and obligations.
- ▲ **Unpredictable outputs:** Report formats, extraction results, and summaries may vary with each attempt.
- ▲ **Difficult quality assurance:** Traditional software testing relies on predictable inputs and outputs, but LLM variability makes it challenging to validate that the AI is consistently producing correct analyses.
- ▲ **Risk of contradictory guidance:** When the same contract is analyzed in separate sessions, contradictory conclusions might emerge, creating confusion about contractual rights and responsibilities.

While some level of variation is expected (after all, there are multiple ways to phrase the same answer), the lack of consistent responses from GenAI creates significant technical challenges when organizations rely on it for contract analysis and insights. The lack of consistency and reproducibility means one session might correctly identify all relevant clauses while another might miss critical clauses, without any clear indication of which result is more reliable. Prompt Engineering with few-shot examples and standardized templates, combined with Quality Engineering validation, dramatically reduces this variability.

“Some LLMs are good at certain tasks, while others are good at entirely different tasks. You need to have multiple tools in your AI toolkit, and know which one to use for which types of tasks. Pramata handles a lot of this technical complexity, so the users can focus on their business problems.”

Pedram Abrari, Chief Technology Officer, Pramata

Pramata addresses the “Variability and Reproducibility” challenge with the following innovations: [TrueDoc OCR](#), [Extract & Enrich](#), [Templates & Playbooks](#), [Contract Families](#), [Multi-LLM Support](#), [Pramata Prompt Language](#), [AI Agents \(and Control-Flow\)](#), [Context Window Overflow Monitoring](#), [Few-Shot Prompting Support](#), [Thought Process Support](#)



6. The Problem of Contract AI at Scale

Getting GenAI to successfully analyze a single contract is challenging enough, but scaling that capability across an enterprise's entire contract portfolio introduces an entirely different level of complexity. For most organizations, having AI analyze just one or a handful of contracts isn't valuable. Rather, they need insights across thousands of agreements to derive meaningful business intelligence.

Enterprise-scale contract analysis presents several significant technical hurdles:

- ▲ **Manual processing limitations:** Running contracts one-by-one through standard AI interfaces is prohibitively time-consuming and resource-intensive for organizations with more than even a handful of contracts.
- ▲ **Consistency across volume:** Maintaining consistent analysis quality across thousands of contracts, including those on third-party paper, requires sophisticated approaches beyond what's possible with ad-hoc processing.
- ▲ **Cross-contract analysis:** Enterprise value often comes from analyzing patterns across contracts, not just within them. This requires the GenAI to access and analyze contracts in aggregate—a capability you can't get from LLMs alone, or from LLMs not trained on contracts.

These scalability challenges directly impact how valuable the entire concept of Contract AI is to a business. If a system can impressively analyze a single contract but can't scale to process thousands, its practical utility for enterprise deployments is severely limited. Organizations might find themselves with impressive demos but no path to implementing an enterprise Contract AI solution that actually works. Scalability requires all three engineering pillars working in concert: Context Engineering for efficient data access, Prompt Engineering for optimized processing, and Quality Engineering for consistent accuracy at scale.

Pramata addresses the "Scalability" challenge with the following innovations: [Extract & Enrich](#), [Multi-LLM Support](#), [Pramata Prompt Language](#), [AI Agents \(and Control-Flow\)](#), [Scalable Agent Processing & Reporting](#)



7. Costs & Economic Challenges

Implementing GenAI for enterprise-wide contract management and analysis introduces significant cost considerations that technical teams must carefully evaluate and manage. At the end of the day, businesses have to decide if the cost of the Contract AI system is worth it to solve the business problems they're looking at GenAI to do.

Several specific factors drive the economics of Contract AI implementations:

- ▲ **Model complexity trade-offs:** More capable models with larger context windows and superior reasoning capabilities cost significantly more for every token they process, creating tension between performance and cost.
- ▲ **Processing volume considerations:** When dealing with thousands of contracts, even small per-token costs multiply rapidly. Enterprise-scale processing can quickly become uneconomical without careful optimization.
- ▲ **Reprocessing requirements:** When context windows overflow or hallucinations occur, documents often need to be reprocessed, which means costs can multiply.
- ▲ **Continuous monitoring expenses:** Maintaining accuracy over time requires ongoing processing to keep contract information current, creating recurring cost commitments.
- ▲ **Economic scalability barriers:** For many organizations, the most significant barrier to Contract AI adoption isn't technical feasibility but economic viability at scale.

Simply dumping contracts into popular commercial LLMs without optimization would be cost prohibitive for anything beyond limited, experimental use. Context Engineering reduces costs by minimizing token consumption through targeted retrieval, while Quality Engineering prevents expensive reprocessing by ensuring accuracy the first time.

Pramata addresses the "Cost & Economic" challenge with the following innovations: [Extract & Enrich](#), [Multi-LLM Support](#), [Pramata Prompt Language](#), [AI Agents \(and Control-Flow\)](#), [Scalable Agent Processing & Reporting](#)



8. The Need for Structured Reasoning

Contract analysis often requires complex, multi-step reasoning to reach accurate conclusions.

For example, analyzing a complex price escalation clause might require everything from identifying the base price terms, finding the escalation trigger conditions and determining the calculation method to checking for caps or exceptions and verifying if any amendments have modified these terms.

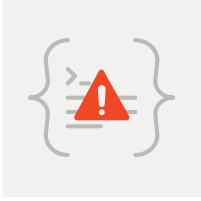
This type of multi-step process is challenging for GenAI, which has traditionally struggled with the requirement to provide a clear logical progression.

Despite how difficult it may be for GenAI to achieve, Contract AI must be able to perform structured reasoning because:

- ▲ Contract terms often reference and modify each other, requiring analysis that considers multiple clauses in relation to one another.
- ▲ Understanding how master agreements, amendments, and addenda interact requires multi-step reasoning about precedence and modifications.
- ▲ Many contract clauses contain elaborate if/then conditions that must be evaluated in sequence.
- ▲ Determining current contractual status often requires reasoning about effective dates, amendment history, and renewal periods.
- ▲ Accurate analysis frequently demands cross-referencing between different sections, exhibits, and related documents.

Early LLMs struggled significantly with this kind of structured reasoning. When faced with complex contract analysis requiring multiple logical steps, they would often take shortcuts, conflate steps, or jump to conclusions without proper analytical progression. Structured reasoning requires sophisticated Prompt Engineering with chain-of-thought patterns, supported by Context Engineering that maintains document relationships and Quality Engineering that validates logical consistency.

Pramata addresses the “Structured reasoning” challenge with the following innovations: [TrueDoc OCR](#), [Extract & Enrich](#), [Templates & Playbooks](#), [Contract Families](#), [Contract Status & Renewals](#), [Multi-LLM Support](#), [Pramata Prompt Language](#), [AI Agents \(and Control-Flow\)](#), [Few-Shot Prompting Support](#), [Thought Process Support](#)



9. Prompt Engineering Challenges

One of the most significant yet often underestimated technical challenges of implementing GenAI for contract analysis is effective prompt engineering. As the newly created role of “prompt engineer” indicates, drafting effective prompts for any number of GenAI tasks isn’t simply about asking clear questions or providing detailed instructions.

Prompt engineering is a complex technical discipline that requires specialized expertise for any field, and all the more so for Contract AI. Without proper prompt engineering, even the most advanced AI models will produce inconsistent, inaccurate, or unusable results when analyzing contracts.

Several specific technical hurdles make prompt engineering for Contract AI particularly challenging:

- ▲ Contracts contain specialized legal language that requires extremely precise instructions to interpret correctly. Slight variations in prompt phrasing can yield dramatically different interpretations of the same contract clauses.
- ▲ Getting the AI to consistently return results in the desired format requires complex prompting techniques that most business users don’t know.
- ▲ Creating effective prompts often requires multiple rounds of trial and error, which means significant development overhead for business-critical contract analysis workflows.
- ▲ Different AI models respond differently to the same prompts, requiring model-specific optimization that further complicates enterprise deployments.
- ▲ Detailed instructions improve accuracy but consume valuable context window space, creating tension between providing detail and maximizing available processing capacity.

For organizations attempting to implement Contract AI, these prompt engineering challenges create a significant technical burden. Teams must either develop or resource deep expertise in prompt engineering or risk inconsistent and unreliable contract analysis. Pramata’s Prompt Engineering expertise, built on analyzing millions of contracts over two decades, is embedded in our Pramata Prompt Language—eliminating the need for organizations to reinvent the wheel.

Pramata addresses the “Prompt Engineering” challenge with the following innovations: [TrueDoc OCR](#), [Extract & Enrich](#), [Templates & Playbooks](#), [Contract Families](#), [Contract Status & Renewals](#), [Multi-LLM Support](#), [Pramata Prompt Language](#), [AI Agents \(and Control-Flow\)](#), [Few-Shot Prompting Support](#), [Thought Process Support](#)

Beyond “Contract GPT”: Why effective Contract AI requires a purpose-built solution

As this whitepaper has demonstrated, applying generative AI to contracts isn't simply a matter of feeding documents into commercially available LLMs and asking questions. The nine fundamental challenges we've explored—from security risks and garbage data to context window limitations, hallucinations and prompt engineering complexities—create a perfect storm of technical obstacles that can derail even the most promising Contract AI initiatives.

What makes contracts particularly challenging for generic AI approaches is their unique combination of characteristics:

- ▲ They contain specialized legal language with precise meanings
- ▲ They form complex document families with interdependent relationships
- ▲ They require flawless accuracy where even small errors can have significant consequences
- ▲ They exist in massive volumes that must be analyzed consistently at enterprise scale
- ▲ They contain sensitive information that must remain encrypted (addressed by RAG-E)

Organizations attempting to implement Contract AI with generic LLMs or simple AI tools inevitably encounter the limitations we've detailed in this guide. These technical barriers aren't merely inconveniences. They represent fundamental constraints that prevent reliable, accurate contract analysis at enterprise scale. Success requires:

Security-first architecture *(RAG-E) that keeps data encrypted*

Context Engineering *for structured data management*

Prompt Engineering *for consistent, accurate analysis*

Quality Engineering *for validated, trustworthy results*

How Pramata's Contract AI innovations help solve these challenges

For each of these challenges, and (more significantly) all of them combined, the solution is to implement Enterprise Grade Contract AI that was designed specifically to solve these business problems.

Pramata's purpose-built Contract AI platform was designed from the ground up to address these nine fundamental challenges through its integrated three-pillar approach:

- ▲ **Context Engineering:** Structured data models and encrypted retrieval (RAG-E) that eliminate security risks while optimizing performance
- ▲ **Prompt Engineering:** Twenty years of contract expertise embedded in reusable, tested patterns
- ▲ **Quality Engineering:** Human-curated answer keys and AI validation agents that achieve true 99% accuracy through our AI TrueCheck methodology

These three pillars work together through our proprietary **Contract AI Knowledge Engine, Contract AI Repository, and AI Design Studio**. We'll dive deeper into each one and how they address the specific needs of businesses trying to manage and efficiently gain intelligence and insights from their contracts.

		RAG Security Risks	Garbage-In; Garbage-Out	Context Window Management	Single Message Token Limit	Hallucinations	Variability &	Scalability	Costs & Economics	Structured Reasoning	Prompt Engineering Challenges
Knowledge Engine	TrueDoc OCR [°]	●	●			●	●			●	●
	Files Cleanse [*]	●	●			●					
	Documents Organize [*]	●	●	●	●	●					
	Extract & Enrich [*]	●	●	●	●	●	●	●	●	●	●
	TrueCheck QA [*]	●	●			●					
AI Repository	Relationship Object Model [*]	●	●	●	●	●					
	Clause Analysis Models [*]	●	●	●	●	●					
	Templates & Playbooks [*]	●	●	●	●	●	●			●	●
	Contract Families [*]	●	●	●	●	●	●				
	Contract Status & Renewals [*]	●	●			●					●
AI Design Studio	Multi-LLM Support [*]						●	●	●	●	●
	Pramata Prompt Language [*]	●		●	●	●	●	●	●	●	●
	AI Agents (and Control-Flow) [*]	●		●	●	●	●	●	●	●	●
	Context Window Overflow Monitoring [*]			●	●	●	●				●
	Few-Shot Prompting Support [*]					●	●		●	●	●
	Thought Process Support [*]					●	●		●	●	●
	Scalable Agent Processing & Reporting [°]							●			

[°]To Be Filed ^{*}Patent Pending

Contract AI Knowledge Engine

Pramata's Contract AI Knowledge Engine is the foundation of everything else the platform can do. Through each of the following components, organizations can get from their current state—no matter how disorganized their contracts are—to a place in which all contracts are ready to power Contract AI intelligence and insights. This represents our Quality Engineering pillar in action by establishing clean, validated data as the foundation.

TrueDoc OCR

Addresses the following AI challenges:

- ✓ *Garbage-in, garbage-out*
- ✓ *Structured Reasoning*
- ✓ *Hallucinations*
- ✓ *Prompt Engineering*
- ✓ *Variability & Reproducibility*

Traditional optical character recognition (OCR) technology is sufficient for converting text into a digital format that can be searched and analyzed. It is not, however, capable of taking complex and non-text-based information and converting it into a format that GenAI can use. If presented with a table, for example, traditional OCR will flatten the contents into plain text, effectively removing its meaning. This is where Pramata's TrueDoc OCR stands out and makes a critical difference for working with contracts within an AI setting.

Unlike traditional OCR, Pramata's TrueDoc OCR technology precisely captures complex contract elements including tables, multi-page clauses, and intricate layouts that regular OCR often misses or can't process. TrueDoc provides a near-lossless conversion of contracts to text, retaining as much of the original context as possible to ensure our Contract AI is working off of all the information in the contract.

File Cleanse

Addresses the following AI challenges:

- ✓ *Garbage-in, garbage-out*
- ✓ *Hallucinations*

If you have duplicate contracts, missing contracts, or contracts organized within the wrong account or document family, you shouldn't even attempt to use GenAI to solve higher-level problems.

This is why cleansed contract files is a mandatory first step. This process includes:

- ▲ Removing duplicates and non-contract files
- ▲ Converting all documents to searchable PDFs through optical character recognition (OCR)
- ▲ Merging and splitting files when needed
- ▲ Rotating pages and ensuring proper orientation
- ▲ Detecting missing signature pages
- ▲ Identifying out-of-scope documents

Pramata's patent-pending File Cleanse technology is uniquely capable of doing all of this and more, accurately, and without manual effort and heavy lifting.

Document Organize

Addresses the following AI challenges:

- ✓ Garbage-in, garbage-out
- ✓ Single Message Token Limit
- ✓ Context Window Management
- ✓ Hallucinations

Contracts exist within complex and interconnected relationships including master agreements, amendments, orders, and many others. This is why organized contract families are a prerequisite to performing and real analysis with Contract AI. This process includes:

- ▲ Creating a clear document hierarchy based on order of precedence
- ▲ Tracking document history and changes
- ▲ Establishing rules-based relationships between master agreements and amendments or other related documents

This level of organization is vital to ensure you receive outputs based on all relevant documents. Pramata's patent-pending Document Organize technology is uniquely capable of creating the highly organized contract environment necessary to achieve your Contract AI goals.

Extract & Enrich

Addresses the following AI challenges:

- ✓ Garbage-in, garbage-out
- ✓ Scalability
- ✓ Context Window Management
- ✓ Costs & Economics
- ✓ Single Message Token Limit
- ✓ Structured Reasoning
- ✓ Hallucinations
- ✓ Prompt Engineering
- ✓ Variability & Reproducibility

Pramata's Extract & Enrich capability pre-tags key contract clauses, dramatically improving processing efficiency when running analyses at scale. Which means that when analyzing thousands of contracts, the system doesn't need to re-discover the same types of clauses repeatedly. This enables much faster processing across large contract portfolios that general AI technology can't do. This exemplifies Context Engineering—structuring data for optimal AI consumption.

TrueCheck QA

Addresses the following AI challenges:

- ✓ Garbage-in, garbage-out
- ✓ Hallucinations

TrueCheck QA represents the three-legged stool of Quality Engineering: human-curated answer keys, precision-engineered extraction prompts, and AI validation agents that work together to achieve true 99% accuracy. The human expertise in TrueCheck QA is strategically applied upfront:

- ▲ **Answer Key Curation:** Subject matter experts create comprehensive answer keys from representative contract sets, establishing the ground truth for training validation agents
- ▲ **Extraction Prompt Engineering:** Specialists craft and refine prompts that accurately extract specific contract terms across diverse language variations
- ▲ **Validation Prompt Engineering:** Experts develop validation prompts that enable AI agents to verify extraction accuracy with high precision

This approach transforms human expertise into scalable AI validation. Once the answer keys are curated and prompts are engineered, AI validation agents can verify thousands of contract extractions automatically. Unlike vendors who claim 90% accuracy (leaving you to review 100% because you don't know which 10% is wrong), TrueCheck QA tells you exactly which results to trust and which need review.

This patent-pending TrueCheck QA technology is uniquely capable of providing the highest possible level of accuracy on which your Contract AI can perform to its maximum potential. By embedding human expertise into the foundation through answer keys and prompt engineering, we achieve enterprise-grade accuracy that scales without requiring human review of every result.

Contract AI Repository

Pramata's Contract AI Repository is the next layer of our platform that builds on what's come out of the Contract AI Knowledge Engine. That is, cleansed and organized files that have been de-duplicated, digitized using TrueDoc OCR, and ensured high levels of accuracy by TrueCheck QA. Now your contracts and their data are ready to power your Contract AI Repository, which allows you to unlock your contract data for business intelligence using the following features and functions. This demonstrates Context Engineering at work by maintaining structured relationships while keeping data encrypted.

Relationship Object Model

Addresses the following AI challenges:

- ✓ Garbage-in, garbage-out
- ✓ Single Message Token Limit
- ✓ Context Window Management
- ✓ Hallucinations

Rather than relying solely on free-form text generation, Pramata implements structured data architectures that constrain AI outputs into predefined formats and schemas. This structured approach ensures that regardless of how the AI phrases its analysis, the underlying data points remain consistent and can be reliably processed by downstream systems.

This patent-pending Relationship Object Model is uniquely capable of providing structured outputs that make Contract AI useful far beyond typical legal team use cases, thanks to standardized data outputs. This structured approach is what enables RAG-E to work without vector databases.

Templates & Playbooks

Addresses the following AI challenges:

- ✓ Garbage-in, garbage-out
- ✓ Variability & Reproducibility
- ✓ Context Window Management
- ✓ Structured Reasoning
- ✓ Single Message Token Limit
- ✓ Prompt Engineering
- ✓ Hallucinations

Contract Templates and Playbooks help our Contract AI standardize contract risk assessment criteria and guidelines, providing consistent benchmarks for AI to evaluate contract positions against organizational policies and risk thresholds.

Our patent-pending Templates & Playbooks technology takes your capabilities far beyond mere "redlining" to assess how compliant any contract is with your organization's rules and best practices—even suggesting areas and ways to amend for greater consistency.

Clause Analysis Models

Addresses the following AI challenges:

- ✓ Garbage-in, garbage-out
- ✓ Single Message Token Limit
- ✓ Context Window Management
- ✓ Hallucinations

Clause Analysis Models are pre-tagged clauses that allow for much more efficient querying without needing full-text access to hundreds, thousands, or millions of contracts. Pramata's patent-pending Clause Analysis Model technology allows natural language contract searching and analysis, and the ability for the AI to recognize the meaning and context of clauses even when the exact wording varies.

Contract Families

Addresses the following AI challenges:

- ✓ Garbage-in, garbage-out
- ✓ Hallucinations
- ✓ Context Window Management
- ✓ Variability & Reproducibility
- ✓ Single Message Token Limit

Pramata's patent-pending Contract Families technology is how Pramata maintains contract relationship precedence.

This is essential for enabling our Contract AI to determine effective terms across multiple related contracts, while minimizing hallucinations.

Contract Status & Renewals

Addresses the following AI challenges:

- ✓ Garbage-in, garbage-out
- ✓ Prompt Engineering
- ✓ Hallucinations

With patent-pending Contract Status & Renewals technology, Pramata delivers levels of status and renewal management accuracy and visibility that meet the needs of even the most stringent organizations.

Renewal Management: Pramata automatically tracks and derives the current state and future contract renewal dates based on a variety of contract term parameters, keeping them up-to-date even as new contracts are signed.

Contract Status: Because of our Cleanse and Organize process, our Contract AI platform understands contract relationships and activation status when analyzing contractual obligations.

AI Design Studio

Sometimes “out of the box” isn’t enough. Only Pramata’s AI Design Studio puts the power in your hands, all with no technical or engineering expertise required. Building on your Contract AI Knowledge Engine and Contract AI Repository, you can use the AI Design Studio to harness the full power of GenAI on your organization’s entire body of contracts. These features and functions set Pramata’s AI Design Studio apart from other GenAI tools, allowing allow Pramata users full control over how they use GenAI while also solving common Contract AI challenges. This is where our Prompt Engineering expertise shines—making sophisticated AI accessible without requiring technical knowledge.

Multi-LLM Support

Addresses the following AI challenges:

- ✓ Variability & Reproducibility
- ✓ Scalability
- ✓ Costs & Economics
- ✓ Structured Reasoning
- ✓ Prompt Engineering

Pramata’s patent-pending Multi-LLM support technology allows the platform to leverage various LLMs based on the specific task requirements. This optimizes AI performance and accuracy, for example utilizing fast-performing models for simple queries to deep reasoning models for complex analysis. This flexibility saves costs and technical resources compared to forcing one LLM model to perform all of your contract management and analysis functions.

AI Agents (and Control-Flow)

Addresses the following AI challenges:

- ✓ Context Window Management
- ✓ Single Message Token Limit
- ✓ Hallucinations
- ✓ Variability & Reproducibility
- ✓ Scalability
- ✓ Costs & Economics
- ✓ Structured Reasoning
- ✓ Prompt Engineering

Rather than forcing the AI to scan entire collections of contracts to locate relevant clauses, which consumes precious context window space, Pramata’s targeted approach means the AI can focus its processing power on analysis rather than discovery.

Pramata’s patent-pending AI Agents and Control-Flow technology implements specialized orchestrator agents that coordinate other agents’ activities. While specialized AI agents can reliably do the same task over and over, an orchestrator agent directs the flow and progression of more complex questions from one specialized agent to another.

Pramata Prompt Language

Addresses the following AI challenges:

- ✓ Context Window Management
- ✓ Single Message Token Limit
- ✓ Hallucinations
- ✓ Variability & Reproducibility
- ✓ Scalability
- ✓ Costs & Economics
- ✓ Structured Reasoning
- ✓ Prompt Engineering

Our patent-pending Pramata Prompt Language uses expert-designed prompts that target the precise, most relevant parts of your contracts instead of requiring the AI to cull through every word of every contract. Using specialized variables like `{{master_agreement}}` and `{{price_escalator}}`, the system can extract specific clauses or terms, dramatically reducing the amount of context needed for analysis, while ensuring superior accuracy even without exact wording matches, thanks to the aforementioned extracted terms functionality.

Our Contract AI foundation models have been tuned on millions of contracts over the past two decades, resulting in a deep understanding of legal verbiage and context. This, combined with our technical expertise, allows Pramata’s Contract AI to do what teams like legal, sales, finance, and procurement need, without anyone reinventing the prompt-engineering wheel.

Context Window Overflow Monitoring

Addresses the following AI challenges:

- ✓ Context Window Management
- ✓ Variability & Reproducibility
- ✓ Single Message Token Limit
- ✓ Prompt Engineering
- ✓ Hallucinations

Pramata actively monitors context utilization during processing, alerting users when the AI is approaching thresholds that might impact reliability. This transparency prevents the silent performance degradation that occurs when using standard LLMs.

When the system detects that a particular analysis might exceed safe context limits, it warns users that results may be inaccurate and suggests alternative approaches, like breaking the analysis into smaller components. This patent-pending Context Window Overflow Monitoring technology prevents many of the problems other LLMs face when tasked with working with large numbers of complex documents like contracts.

Thought Process Support

Addresses the following AI challenges:

- ✓ Hallucinations
- ✓ Structured Reasoning
- ✓ Variability & Reproducibility
- ✓ Prompt Engineering

Pramata requires its Contract AI to document its thought process step-by-step when analyzing contracts during the reasoning stage. This patent-pending Thought Process Support technology ensures a separate and thorough evaluation of multiple factors before ever reaching a final, concise and accurate output for users. Which significantly reduces hallucination rates, similar to how asking someone to “show their work” helps humans catch their own errors.

Few-Shot Prompting Support

Addresses the following AI challenges:

- ✓ Hallucinations
- ✓ Structured Reasoning
- ✓ Variability & Reproducibility
- ✓ Prompt Engineering

Pramata leverages a patent-pending Few-Shot Prompting Support technology. This utilizes a technique that provides the AI with a small number of clear examples of expected inputs and outputs. These examples serve as templates to guide the AI towards consistent interpretations and output formats and significantly reduce variability across outputs.

Scalable Agent Processing & Reporting

Addresses the following AI challenges:

- ✓ Scalability
- ✓ Costs & Economics

Pramata's agent-powered reporting capability transforms individual AI analyses into standardized, tabular outputs that maintain consistency across the enterprise. This approach ensures that even if the underlying AI processing varies, the final reports delivered to users maintain a consistent structure and format, making them suitable for business-critical decisions.

The result? Enterprise Grade Contract AI that actually works. Pramata delivers reliable, accurate insights across your entire contract portfolio without requiring your team to have any specialized AI expertise.

GenAI + Contracts comes with challenges, but they're not insurmountable

As we've laid out in this whitepaper, generative AI has many complex technical issues that make the systems available publicly today unsuitable for complex contract management, analysis, creation, and insights.

Fortunately, each of GenAI's pitfalls when it comes to dealing with contracts are solvable by using a solution with Enterprise Grade Contract AI that actually works: Pramata.

Pramata's three-pillar approach—**Context Engineering, Prompt Engineering, and Quality Engineering**—combined with our RAG-E security architecture, delivers what "Contract GPT" cannot: enterprise-grade accuracy, security, and scale.

READY TO EXPLORE TRUE CONTRACT AI?

Contact Pramata today to see how our purpose-built Contract AI solution can transform how your organization manages, analyzes, and derives value from your contracts.

[REQUEST A DEMO](#)

