Report on Challenges

Identification of Challenges and Solutions

Enhancing User Retention in Accorda

27th of April, 2025

Prepared by

Ivan Lubyako, Francisco de Jesus, and Sara de Almeida

Prepared for

ELTEMATE

TABLE OF CHALLENGES

1. Legal Challenges

1.1. General Legal Challenges

1.1.1. Legal Language

Unsurprisingly, we found that contracts are often written in highly complex, archaic, and specialized legal terms. In several cases – especially involving nuanced legal concepts – the AI returned inaccurate definitions.

To solve this, we carefully engineered our prompts and built agentic flows to handle tasks like category extraction, clause evaluation, and semantic comparison. We also created custom prompt templates and structured *JSON schemas*, embedding precise legal descriptions for each concept. This ensured consistent and interpretable outputs.

1.2. Specific Legal Challenges

1.2.1. Comparison Tool

1.2.1.1. Handling Semantic Variations

Contracts often undergo semantic variations or paraphrasing, which do not necessarily alter their underlying meaning. Similarly, structural changes – such as the reordering of clauses or their appearance in different sections – may not affect the contract's substantive content.

Initially, we considered applying segmentation algorithms to divide contracts into standardized sections. However, we ultimately decided against using traditional rule-based or machine learning-based segmentation methods. Instead, we focused on smart prompt design: instructing the AI to recognize and extract clauses based on linguistic patterns, headings, numbering, and context. This gave the system flexibility to handle different contract formats without rigid parsing rules.

1.2.2. Analysis Tool

1.2.2.1. Differences Between Legal Systems

One of our first challenges was the private and flexible nature of contract law itself. Differences between civil law and common law traditions became very clear. For example, in civil law, basic principles are implied even if not written into the contract. In common law, however, these principles usually must be explicitly stated.

To address this, our analysis tool lets users set the jurisdiction based on the contract's origin, adapting the legal checks accordingly.

1.2.2.2. Identifying Contract-Specific Requirements and National Differences

Another challenge arose: certain contract types have specific legal requirements.

Our first solution was to make the system detect the type of contract and adjust the analysis for its specific characteristics, shown in the Specific Analysis section.

1.2.2.3. Handling National Differences

Furthermore, we also needed to account for country-specific requirements – certain types of contracts have additional obligations under different national laws.

To manage this, we added a system to alert users when specific local rules might apply, helping them verify compliance with governing law, jurisdiction, and contract type.

1.2.2.4. Defining Scope

Finally, when it came to additional obligations like pre-contractual negotiations or verbal agreements, we had to set a clear boundary:

- Accorda focuses on analyzing the content of written agreements only.
- Some pre-contractual aspects or parties information are therefore outside its scope – intentionally.

2. Technical Challenges

2.1. General Technical Challenges

2.1.1. Model Selection and Architecture Flexibility

Choosing the right AI model was one of the major technical challenges during development.

After evaluating several options:

- We initially leaned towards **Gemini**, given its strong performance with long context windows — a key factor for processing lengthy legal contracts.
- **Flash 1.5** offered impressive speed but was too non-deterministic for precise clause-by-clause comparisons.
- **Flash 2.5** produced more detailed and consistent outputs, although with slower performance.

To fine-tune results, we also configured a low temperature setting, which helped produce more controlled and reliable responses.

To accommodate different user needs, the final system allows **configurable model selection** – users can choose the AI model that best fits their preferences.

Additionally, the application's architecture was designed to be **extensible**, making it easy to integrate alternative models such as **Qwen, Claude, or OpenAI models** with minimal modification.

2.1.2. Deployment and Hosting

DevOps and deployment proved to be one of the most difficult aspects of the project.

At first, we explored cloud services that supported **Docker Compose** deployments, but many were either too restrictive or unsuitable for the project's specific needs.

Ultimately, we decided to **host the application on a private VPS**. Although we had limited experience with server administration at the start, we successfully managed the deployment – overcoming challenges like **CORS issues** during backend/frontend communication setup. In the end, the **private VPS** approach proved to be the most **reliable and cost-effective** hosting solution.

2.2. Specific Technical Challenges

2.2.1. Document Comparison and Visualization

Another major technical challenge was **displaying differences between two documents** in a way that was clear and user-friendly – especially for non-technical users.

To handle this:

- We built a **dedicated Python service** that compares clauses and generates **structured diff data**.
- On the frontend, we integrated **diff2html**, a library designed to visualize text differences.

However, because diff2html’s default output is highly technical and designed for programming code, we **heavily customized** its styling and behavior.

The goal was to:

- Simplify the visual presentation;
- Remove programming-specific indicators;
- Make the document comparison clean, intuitive, and easy to read for legal and business professionals.

These modifications made the comparison tool much better suited to real-world legal review workflows.

Concluding Remarks

Building Accorda involved navigating a wide range of legal and technical challenges.

On the legal side, we tackled the inherent complexity of legal language, the nuances between civil law and common law systems, and the need to account for national-specific requirements – all while ensuring clarity, flexibility, and precision in analysis, without losing the “lawyer's intuition”.

On the technical side, we faced critical decisions in model selection, architectural design for extensibility, deployment strategy, and user experience optimization, particularly in presenting document comparisons in a clear and accessible manner.

Addressing these challenges required careful planning, iterative problem-solving, and a focus on practical solutions that prioritize reliability, adaptability, and user trust.

While Accorda does not aim to replace professional legal analysis, it empowers users – especially legal professionals – to compare and analyze contracts faster, with greater accuracy, and with greater confidence.