Enhancing Trust in Legal AI

Optimising Span-Level Retrieval Architectures on LegalBench-RAG

Amal Saad Alshehri, Nelly Bencomo, Amir Atapour-Abarghoue

Durham University, UK

UKAIS 2026 | The University of Sheffield | 9-10 April 2026

Talk Outline

1. The Problem: Why retrieving clauses from contracts is difficult
2. The Benchmark: How LegalBench-RAG measures retrieval quality
3. A Worked Example: A real query, step by step
4. Our Approaches: Dense, keyword-based, and hybrid retrieval
5. Results and Analysis: What we found and what it means

Part 1

The Problem

What This Research is About

We study how well AI systems can find specific clauses inside contracts

The contracts we test include:

privacy policies, non-disclosure agreements, commercial contracts,

and mergers and acquisitions agreements

We compare different retrieval methods to find which is most accurate

Why This Matters

Law firms are adopting AI tools to review contracts

These tools use Retrieval-Augmented Generation (RAG)

RAG works in two steps: first FIND the relevant clause, then ANSWER the question

If the retrieval step fails, the AI gives wrong or hallucinated answers

Our research evaluates how well the retrieval step works for contracts

What is Retrieval-Augmented Generation (RAG)?

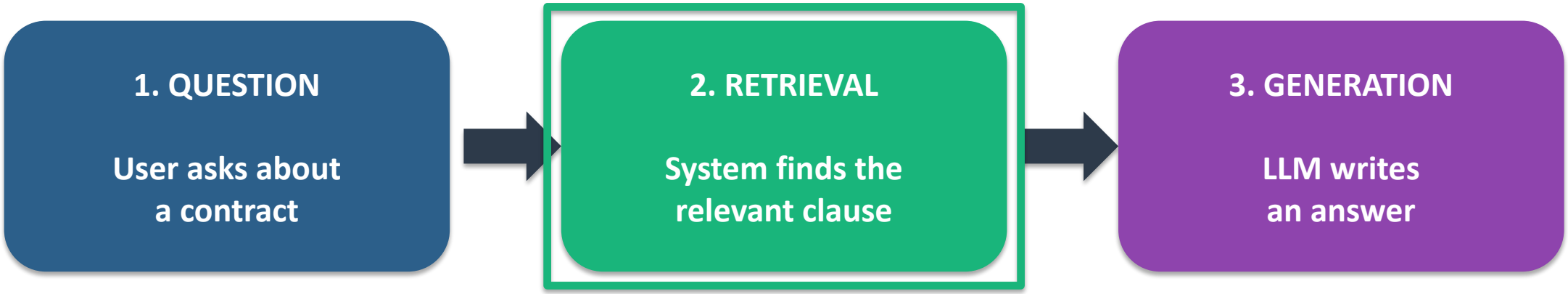
Think of it like a research assistant:

1. You ask a question about a contract
2. The assistant searches the contract to find the relevant clause
3. The assistant reads that clause and writes you an answer

If the assistant brings back the WRONG clause, the answer will be wrong

Our research focuses on step 2: how well can the system find the right clause?

The RAG Pipeline



Our research evaluates this step

If retrieval brings back the wrong clause, the generated answer will also be wrong.

Why Contracts Are Difficult for Retrieval Systems

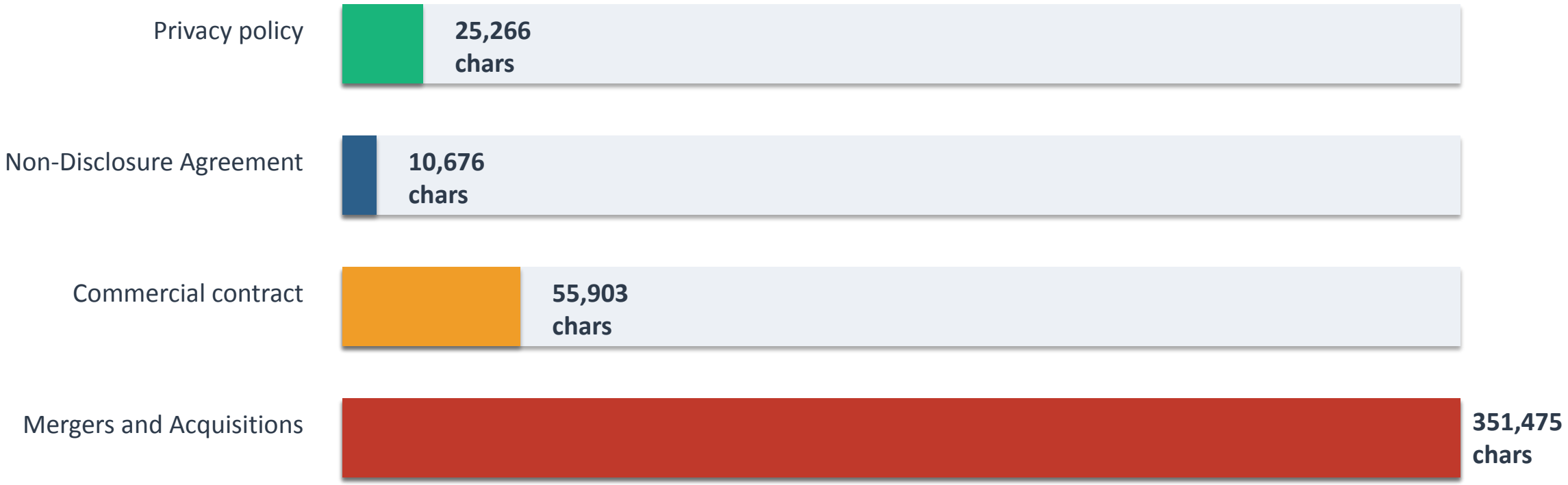
Contracts are very long

(a mergers and acquisitions agreement can be 350,000+ characters)

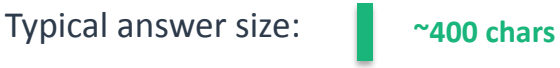
The relevant clause is small (typically 200 to 700 characters)

Contracts use specialised language that differs from everyday English

Contract Sizes (To Scale)



The answer clause is typically only 200-700 characters within these documents



The Vocabulary Gap: How a User Asks a Question

"Does this contract include a non-compete clause?"

"What is the expiration date of this agreement?"

"Is there a limit on how much one party can be sued for?"

These are plain-English questions. The user does not use contract terminology.

The Vocabulary Gap: How the Contract Is Written

A "non-compete clause" is written as:

"Restrictive Covenant" or "Covenant Not to Compete"

An "expiration date" is written as:

"shall remain in effect until the end of the current calendar year

and shall be automatically renewed..."

A "liability limit" is written as:

"Limitation of Liability shall not exceed the aggregate fees..."

The contract uses completely different words. A keyword search for "non-compete" will not match "Restrictive Covenant".

An Important Clarification About Keyword Search

Keyword-based search (known as BM25) fails specifically for

contract clause retrieval because of this vocabulary gap

BM25 performs well in other legal retrieval tasks

(for example, finding relevant case law in the COLIEE competition)

Our finding is specific to retrieving clauses from contracts

Research Questions

RQ1 How well do current retrieval methods find specific clauses inside contracts?

RQ2 Do models trained on legal language outperform general-purpose models for contract retrieval?

Part 2

The Benchmark: LegalBench-RAG

LegalBench-RAG (Pipitone and Houir Alami, 2024)

The first benchmark specifically designed to evaluate contract retrieval

Contains 776 questions about real contracts (194 per dataset)

This is the same mini version used and reported in the original paper

Each question has a ground truth answer marked by exact character positions

For example: file "CopAcc_NDA.txt", characters 7,752 to 8,016

Four Types of Contracts Are Tested

The benchmark covers contracts of increasing complexity:

1. Privacy policies — written for everyday users, simple and easy to read

2. Non-Disclosure Agreements — follow a common template, similar structure across companies

3. Commercial contracts — cover many different topics, often long and detailed

4. Mergers and Acquisitions agreements — use highly technical legal language, very long documents

The Four Datasets in Detail

Contract Type	Documents	Average Length per Document
Privacy policies	7	25,266 characters
Non-Disclosure Agreements	89	10,676 characters
Commercial contracts	461	55,903 characters
Mergers and Acquisitions agreements	**150**	**351,475 characters**

How We Measure Retrieval Quality

Precision

"Of everything the system retrieved,

how much was actually the answer?"

Calculated as:

overlapping characters ÷ total retrieved characters

High precision = mostly relevant text

Recall

"Of the true answer,

how much did the system find?"

Calculated as:

overlapping characters ÷ total gold answer characters

High recall = found most of the answer

Why We Measure at the Character Level

Most benchmarks measure at the document level: "Did you find the right document?"

We measure at the character level:

"Did you find the exact right sentence inside a 350,000-character contract?"

This is stricter, but necessary for trustworthy contract analysis

Part 3

A Worked Example from the Benchmark

The Setup for Our Example

Dataset: Non-Disclosure Agreements

Document: An agreement between CopAcc and ToP Mentors

Document length: 14,944 characters (about 30 chunks of 500 characters)

The Question the System Must Answer

From the benchmark:

"Consider the Non-Disclosure Agreement

between CopAcc and ToP Mentors;

Does the document state that Confidential Information

shall only include technical information?"

The system must search through the entire 14,944-character agreement to find the relevant clause.

The Ground Truth Answer (What the System Should Find)

Location: characters 7,752 to 8,016 (264 characters long)

The clause reads:

"Confidential Information" means any Idea disclosed to Mentor,

all data and information, know-how, business concepts,

software, procedures, products, services, development

projects, and programmes contained in such Idea

and/or its description and any conclusions.

Answer to the question: No — Confidential Information includes much more than just technical information.

The Document Is Split into Chunks

The 14,944-character agreement is divided into about 30 chunks

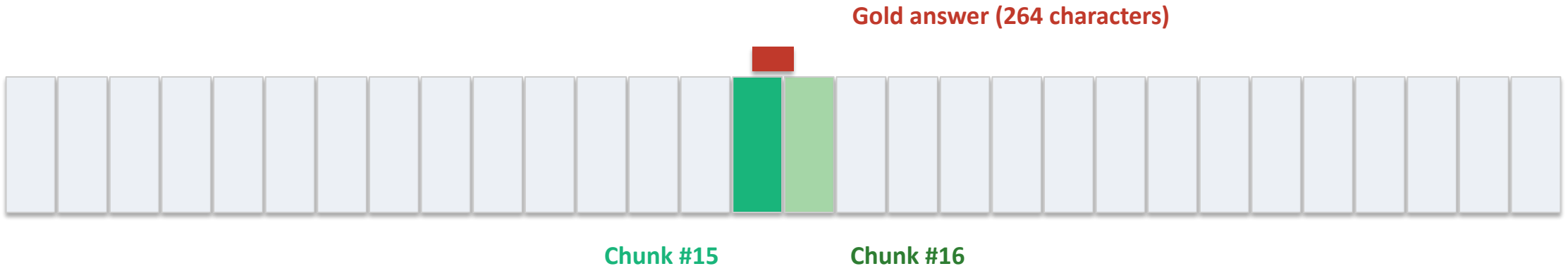
Each chunk is approximately 500 characters

Chunk #15 (characters 7,500 to 8,000) contains:

- Some irrelevant text about fees and taxes
- The beginning of the Confidential Information definition

How the Document Is Chunked

A 14,944-character Non-Disclosure Agreement split into 30 chunks of ~500 characters:



The gold answer starts in Chunk #15 and spills 16 characters into Chunk #16

Actual text inside Chunk #15 (500 characters):

Irrelevant text

...No deductions will be made from the gross fees paid to mentor who shall be solely responsible for ensuring that all and any Government taxes... 4 Definition of Confidential Information

GOLD ANSWER starts here

"Confidential Information" means any Idea disclosed to Mentor, all data and information, know-how, business concepts, software, procedures, products, services, development projects, and programmes contained in such Idea and/or its description and a...

The Gold Answer Spans Two Chunks

The gold answer goes from character 7,752 to 8,016

Chunk #15 ends at character 8,000

So the last 16 characters of the answer fall into the next chunk (Chunk #16)

Even retrieving the perfect chunk cannot capture 100% of the answer

Computing Precision and Recall for This Example

Retrieved: Chunk #15 = 500 characters (positions 7,500 to 8,000)

Gold answer: 264 characters (positions 7,752 to 8,016)

Overlap: 248 characters (positions 7,752 to 8,000)

Precision = 248 / 500 = 49.6% (about half the chunk is useful)

Recall = 248 / 264 = 93.9% (we found 94% of the answer)

The missing 6% of recall (16 characters) is the tail end of the clause that falls into the next chunk.

What Happens When We Retrieve More Chunks (K = 64)

K=1 → Retrieved 500 characters
Found 248 of 264 answer characters
Precision: 49.6% | Recall: 93.9%

K=2 → Retrieved 1,000 characters
Found all 264 answer characters
Precision: 26.4% | Recall: 100%

K=64 → Retrieved 32,000 characters
Found all 264 answer characters
Precision: 0.83% | Recall: 100%

As K increases:

Recall goes UP
(more chunks = more chance to find the full answer)

Precision goes DOWN
(the answer gets buried in irrelevant text)

Part 4

Our Retrieval Approaches

We Tested Four Retrieval Approaches

1. Dense retrieval (finds clauses by meaning, using embeddings)
2. BM25 (finds clauses by keyword overlap, no embeddings)
3. Hybrid (combines BM25 and Dense with weighted scores)
4. Dense + Reranker (retrieve first, then re-score with a second model)

How Dense Retrieval Works

Step 1: Split the contract into 500-character chunks

Step 2: Convert each chunk into a numerical vector (called an embedding)

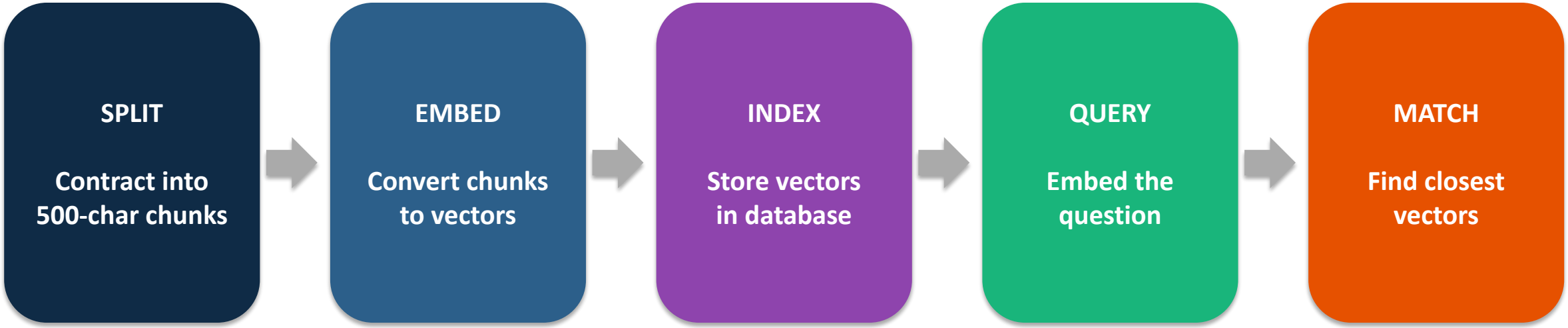
that captures the meaning of the text

Step 3: Store all vectors in a searchable database (we use sqlite-vec)

Step 4: Convert the user's question into a vector using the same model

Step 5: Find the chunks whose vectors are closest to the question's vector

Dense Retrieval Pipeline



Steps 1-3 happen once when indexing. Steps 4-5 happen for every query.

Two Embedding Models Compared

OpenAI text-embedding-3-large

General-purpose model

Trained on broad internet data

Used in the original LegalBench-RAG paper

Understands English well, but not trained specifically on contract language

Voyage voyage-law-2

Fine-tuned for legal text

Optimised for legal documents including contracts, case law, and regulatory filings

Designed to better represent contract-specific terms like 'indemnification' and 'material adverse effect'

Two Chunking Strategies Compared

Naive (Fixed-Size)

Cut every 500 characters exactly

Simple and deterministic

May cut mid-sentence:

"...responsible for ensuring

that all" | CUT | "and any

Government taxes..."

RCTS (Recursive Splitter)

Target: 500 characters

Splits at natural breaks:

paragraph endings first,

then sentence endings,

then word boundaries

Preserves the structure of text

How BM25 (Keyword-Based Search) Works

BM25 scores each chunk by how many question words it contains

It does not understand meaning — only word overlap

If the question says "non-compete"

but the contract says "Restrictive Covenant"

BM25 sees zero matching words and ranks this chunk as irrelevant

How Hybrid Retrieval Works

Step 1: Get candidates from both BM25 and Dense retrieval

Step 2: Normalise both score lists to the same 0-to-1 range

Step 3: Combine using weights:

final score = BM25 weight x BM25 score + Dense weight x Dense score

We tested: 20/80, 30/70, 40/60, 50/50, 70/30 (BM25/Dense weights)

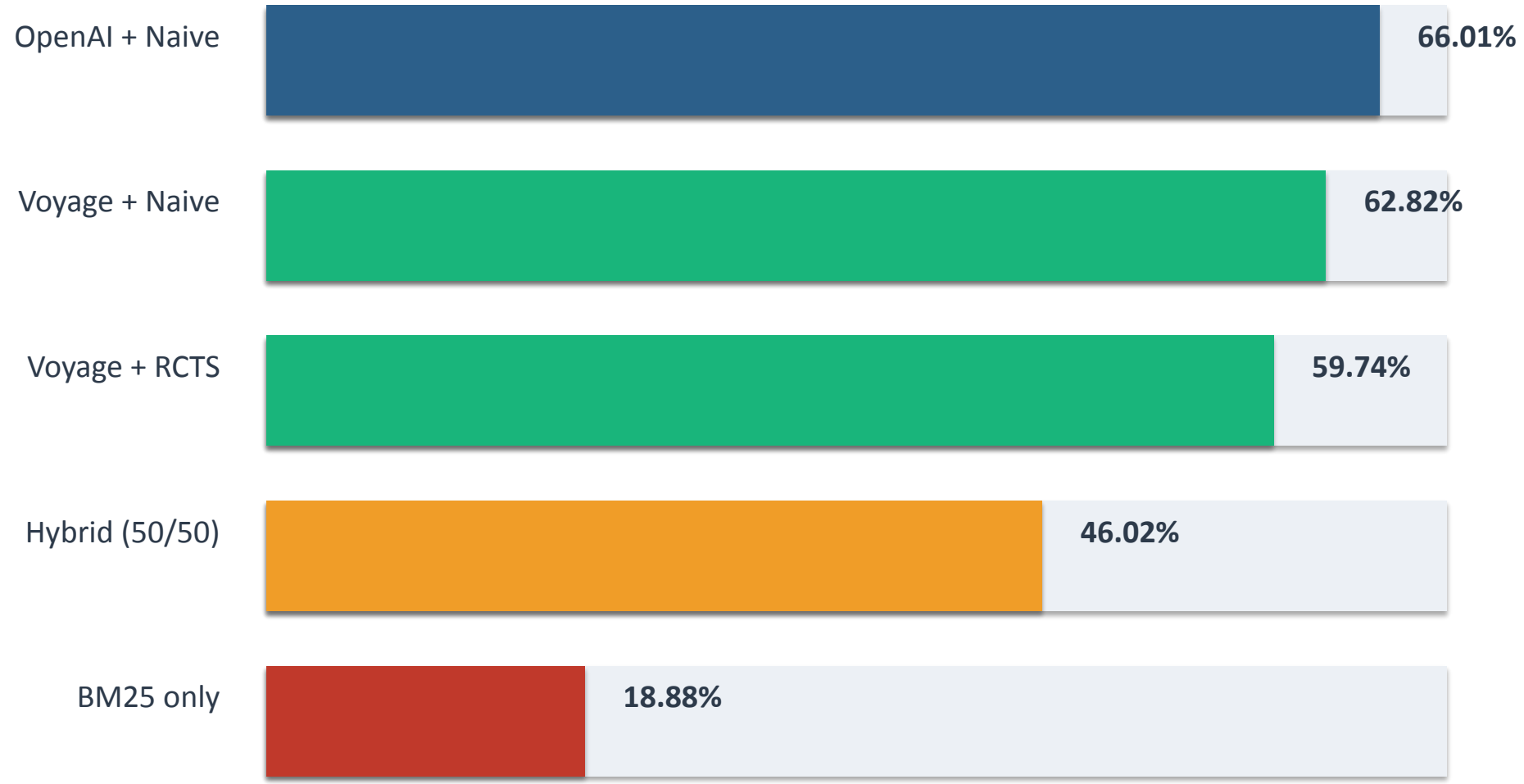
Part 5

Results

Results: All Four Datasets Combined (Equal Weight per Dataset)

Retrieval Method	Precision@1	Recall@64
Voyage voyage-law-2 + RCTS chunking	**11.12%**	**59.74%**
Voyage voyage-law-2 + Naive chunking	8.53%	62.82%
OpenAI + Naive chunking	7.34%	66.01%
OpenAI + RCTS chunking	7.23%	65.86%
Hybrid (50% BM25, 50% Dense)	3.42%	46.02%
BM25 only (keyword search, no embeddings)	0.67%	18.88%

Recall@64 Comparison



Higher is better. Dense retrieval (top 3) far outperforms BM25 keyword search (bottom).

Finding 1: Legal-Specialised Embeddings Improve Precision

11.1% vs 6.4%

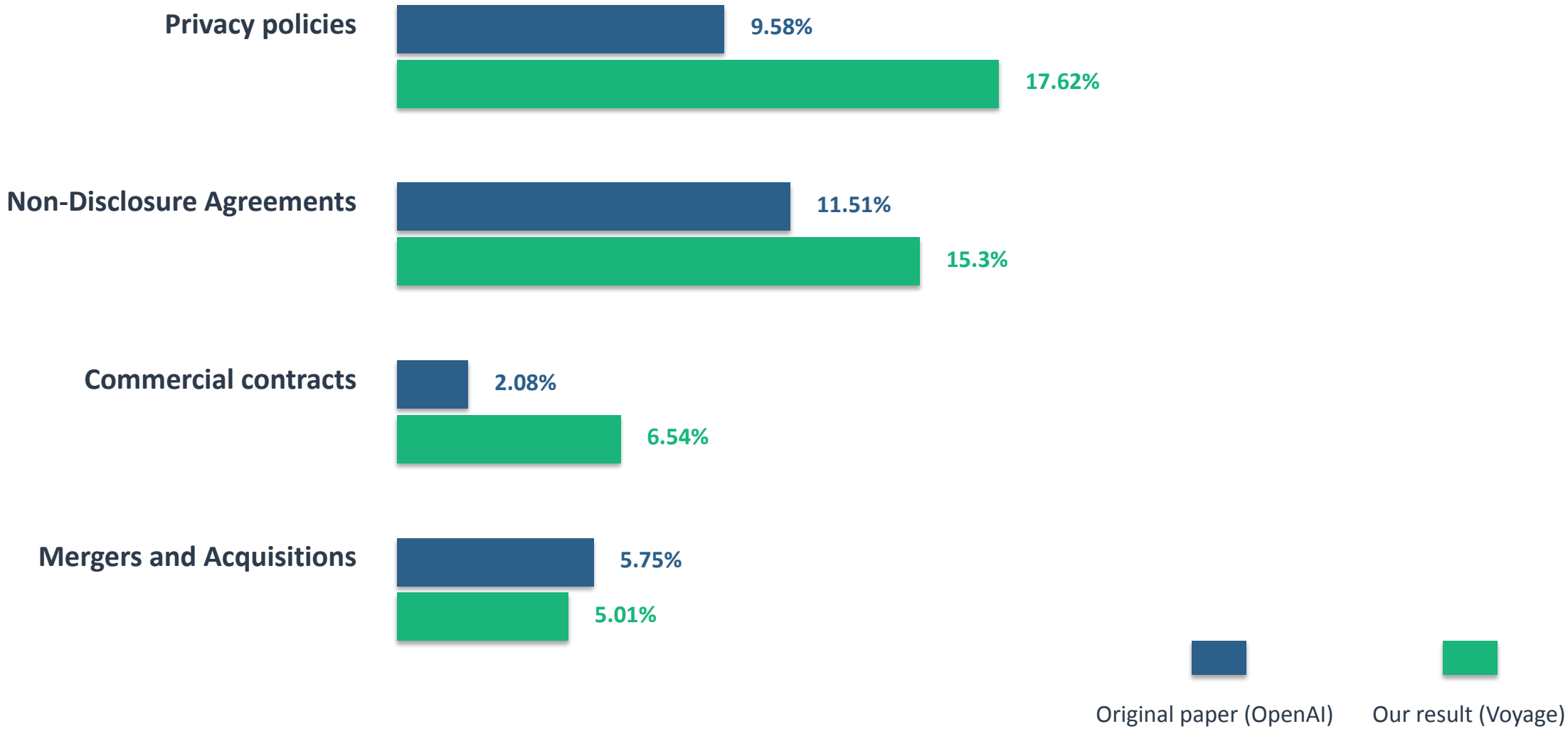
Our best result (Voyage voyage-law-2) vs the original paper's best (OpenAI)

Precision@1 improved by 73% by using embeddings trained on contract language

Precision@1 Comparison Per Contract Type (RCTS Chunking, No Reranker)

Contract Type	Original Paper (OpenAI)	Our Result (Voyage)
Privacy policies	9.58%	17.62%
Non-Disclosure Agreements	11.51%	15.30%
Commercial contracts	2.08%	6.54%
Mergers and Acquisitions	5.75%	5.01%

Precision@1: Original Paper vs Our Result



Finding 2: Keyword Search (BM25) Fails on Contract Retrieval

0.67%

BM25 Precision@1 (see last row of the main results table)

The vocabulary gap between user questions (plain English)

and contract language makes keyword matching nearly useless

Finding 3: Adding BM25 to Dense Retrieval Reduces Performance

Dense-only Recall@64: 66.01% (see main results table, row 3)

Hybrid Recall@64: 46.02% (see main results table, row 5)

Adding BM25 reduces recall by 20 percentage points

Why? Because BM25 scores are noisy for contracts (vocabulary gap)

and dilute the quality of the dense embedding signal

In general web search, hybrid retrieval often helps. For contract retrieval, it does not, because the keyword signal is unreliable.

Finding 4: Retrieval Difficulty Varies by Contract Type

Contract Type	Best Recall@64	Avg Document Length
Privacy policies	85%	25,266 characters
Non-Disclosure Agreements	67%	10,676 characters
Commercial contracts	72%	55,903 characters
Mergers and Acquisitions	**27%**	**351,475 characters**

Why Mergers and Acquisitions Agreements Are the Hardest

Each document averages 351,475 characters

- That means approximately 700 chunks of 500 characters each

The retrieval system must find the right 1 or 2 chunks out of 700

- More chunks means more candidates to sort through, more noise

The language is highly specialised:

- "Material Adverse Effect", "Bring-Down Condition", "Intervening Event"

Finding 5: Embedding Model Matters More Than Chunking Strategy

Naive vs RCTS chunking: 0.1% to 3% difference in recall

OpenAI vs Voyage voyage-law-2: up to 73% difference in precision

The choice of embedding model has a much larger impact

than how you split the contract into chunks

Best Result from the Paper vs Best Result from Our Experiments

	Paper's Best (Pipitone et al.)	Our Best
Embedding model	OpenAI (general-purpose)	Voyage voyage-law-2 (legal)
Chunking	RCTS, 500 characters	RCTS, 500 characters
Reranker	Cohere (reduced performance)	None used
Precision@1	**6.41%**	**11.12% (+73%)**
Recall@64	62.22%	59.74% (-4%)

Part 6

Implications and Future Work

Practical Recommendations for Contract Retrieval

1. Use embedding models trained on legal language

   (in our experiments, this improved precision by 73%)

2. Do not rely on keyword search alone for contract clause retrieval

   (BM25 scored 0.67% precision due to the vocabulary gap)

3. Test hybrid retrieval carefully before deploying it

   (adding BM25 reduced our recall from 66% to 46%)

Limitations

We tested one chunk size (500 characters)

other sizes may yield different results

Evaluated on 776 queries (the same mini benchmark reported in the original paper)

The full benchmark contains 6,858 queries and remains to be evaluated

Future Work

1. Legal-aware chunking:

   Instead of splitting contracts every 500 characters,

   split them at clause boundaries (definitions, obligations, termination)

2. Fine-tune a reranker (a second-stage scoring model)

   specifically on contract data to improve re-scoring accuracy

3. Query reformulation:

Before searching, use a language model to rephrase the user's question from plain English into contract language

4. Evaluate on the full LegalBench-RAG benchmark (6,858 queries)

Key Takeaways

1. Contract clause retrieval is hard — best precision is only 11%
2. Embedding models trained on legal language give 73% better precision
3. Keyword search fails on contracts due to the vocabulary gap
4. Longer, more specialised contracts are harder for all methods
5. Retrieval precision is the foundation of trustworthy legal AI

Thank You

Questions and Discussion

Amal Saad Alshehri | Durham University, UK