

REAKASE-8B: Legal Case Retrieval via Knowledge and Reasoning Representations with LLMs

Yanran Tang, Ruihong Qiu, Xue Li, and Zi Huang

The University of Queensland
{yanran.tang, r.qiu, helen.huang}@uq.edu.au, xueli@eeecs.uq.edu.au

Abstract. Legal case retrieval (LCR) is a cornerstone of real-world legal decision making, as it enables practitioners to identify precedents for a given query case. Existing approaches mainly rely on traditional lexical models and pretrained language models to encode the texts of legal cases. Yet there are rich information in the relations among different legal entities as well as the crucial reasoning process that uncovers how legal facts and legal issues can lead to judicial decisions. Such relational reasoning process reflects the distinctive characteristics of each case that can distinguish one from another, mirroring the real-world judicial process. Naturally, incorporating such information into the precise case embedding could further enhance the accuracy of case retrieval. In this paper, a novel REAKASE-8B framework is proposed to leverage extracted legal facts, legal issues, legal relation triplets and legal reasoning for effective legal case retrieval. REAKASE-8B designs an in-context legal case representation learning paradigm with a fine-tuned large language model. Extensive experiments on two benchmark datasets from COLIEE 2022 and COLIEE 2023 demonstrate that our knowledge and reasoning augmented embeddings substantially improve retrieval performance over baseline models, highlighting the potential of integrating legal reasoning into legal case retrieval systems. The code has been released on <https://github.com/yanran-tang/ReaKase-8B>.

Keywords: Legal Case Retrieval · Large Language Models.

1 Introduction

Legal case retrieval (LCR) aims to retrieve precedents from large-scale repositories given a query case. LCR tools become an indispensable tool in modern legal practice for legal professionals such as judges and lawyers. Beyond professional practice, these tools also provide accessible resources for individuals who seek legal guidance but cannot afford costly legal services. Recent research has significantly advanced LCR methods, enabling faster and more accurate retrieval of relevant cases. These approaches are generally classified into two main paradigms: lexical retrieval models and language models (LM). Lexical models, such as BM25 [29], TF-IDF [16], and LMIR [26], focus on computing similarity scores between cases by relying on term frequency statistics. In contrast, LM-based methods [1, 2, 7–9,

[18, 19, 23, 28, 31, 41, 43, 44, 49] leverage pre-trained models to generate semantic representations of cases for similarity comparison. Given the complex nature of legal texts, LM-based methods have further explored similarity computation at different levels of granularity, including sentence-level [47], paragraph-level [31], and document-level [18] strategies.

Although existing methods have made progress by using raw case texts to generate high-dimensional representations for similarity computation, they often overlook two critical aspects in encoding legal cases. **(1) Legal entities relations.** Legal entities are the fundamental objects in a case, such as parties, criminal acts, and evidence. Modelling the relations among these entities provides richer structural and latent information beyond the surface text, enabling more accurate case representations. By integrating entity relations, models can produce embeddings that are semantically richer and structurally faithful to the complexity of legal texts, thereby improving the accuracy in legal case retrieval. **(2) Legal reasoning relations.** Beyond structural elements, legal cases also embody reasoning processes that link facts and issues to judicial decisions. These relations capture how courts interpret evidence, apply statutes, and weigh arguments to reach outcomes. Unlike raw text features, reasoning-based information emphasises the logical pathways that define the uniqueness of each case. Incorporating such reasoning relations into case encoding allows models to better mirror judicial decision-making, producing embeddings that more closely reflect how judges analyse cases. This will improve the distinctiveness of case representations and enhances retrieval accuracy by prioritising cases that share similar reasoning patterns with a query.

To fully exploit both legal entity relations and legal reasoning relations, this paper proposes REAKASE-8B, an LLM-based embedding framework that integrates knowledge and reasoning to generate more informative case representations for the legal case retrieval task. Specifically, a **legal element extraction module** is designed to identify key components of a case, including legal facts, issues, judicial decisions, and relation triplets that capture the connections between facts and issues. Further, a **legal reasoning generation module** is introduced, leveraging an LLM to produce the inferential logic linking facts and issues to judicial decisions. Building on these extracted elements, REAKASE-8B fine-tunes an LLM embedding model with contrastive learning, yielding a reasoning-aware case encoder. Extensive experiments on two benchmark datasets, COLIEE 2022 [13] and COLIEE 2023 [14], show that REAKASE-8B achieves state-of-the-art performance on the LCR task. The main contributions of this paper are as follows:

- **A novel knowledge and reasoning augmented embedding framework.** REAKASE-8B is proposed, which is a novel LLM-based case encoder that jointly integrates legal entity relations and legal reasoning relations, addressing key limitations of prior text only embedding methods.
- **A novel contextualised legal case encoding.** Two modules are designed: (i) a legal element extraction module that captures legal facts, issues, decisions, and their interrelations, and (ii) a legal reasoning generation module that pro-

duces inferential logic linking facts and issues to judicial outcomes. Together, these modules enable knowledge and reasoning-aware case representations.

- **State-of-the-art performance on LCR.** Through contrastive fine-tuning, REAKASE-8B learns embeddings that align closely with legal relational knowledge and judicial reasoning. Experiments on COLIEE 2022 and COLIEE 2023 benchmarks demonstrate consistent and significant improvements over strong baselines, establishing new state-of-the-art results.

2 Related Work

Legal Case Retrieval. In legal case retrieval, there are two main branches of methods to capture the semantic similarity between legal cases. (1) Statistical models mainly rely on using the term frequency and the inverse document frequency to measure the semantic similarity between cases. For example, TF-IDF [16], BM25 [29], and LMIR [26] are typical methods in using these measurements. (2) In the language modelling era, most methods have been using advanced language models, such as BERT [11], RoBERTa [21] and MonoT5 [25], as legal case encoder to generate embeddings for retrieval [1, 2, 4–9, 18–20, 23, 28, 31–33, 36–41, 43, 44, 46, 47, 49, 51]. For example, BERT-PLI divides a extremely long legal case by paragraphs to obtain paragraph-level embeddings and measures the case relevance by embedding interactions between cases [31]. SAILER develops a fact encoder to decode reasoning and decision based on BERT and encode the case with the truncated case text [18]. More recently, LLMs have been applied to introduced into embedding-based legal case retrieval by feeding the case into LLMs and use the last-layer hidden state as the case embedding, such as in LawLLM [32]. Different from these methods, our proposed REAKASE-8B identifies the key components in a case and generate the legal reasoning and the knowledge relation triplets to support effective case retrieval with LLMs.

Embedding Models Based on Large Language Models. Given the powerful context comprehension ability of LLMs, recent methods have been trying to utilise LLMs to obtain meaningful document embeddings for retrieval. MMTEB benchmark serves as a testbed for different retrieval tasks, including a collection of legal question answering tasks [12]. Among various methods, voyage-law-2 is the best performing models yet it is a closed-source one¹. E5-Mistral-7B-Instruct [42] is one of the high performing open-sourced LLMs in legal task. Recently, Qwen3-Embedding-8B [50] achieves state-of-the-art embedding ability among various tasks in MMTEB. Note that the legal question answering tasks in MMTEB is not the main focus of this paper.

Legal Large Language Models. There are recent efforts in developing legal specific LLMs for various legal tasks [10, 32, 48]. For example, general purpose legal LLMs are developed using legal domain data with supervised fine-tuning as in LawLLMs [32, 48], SaulLM [10]. There are also frameworks directly evaluating the

¹ <https://blog.voyageai.com/2024/04/15/domain-specific-embeddings-and-retrieval-legal-edition-voyage-law-2/>

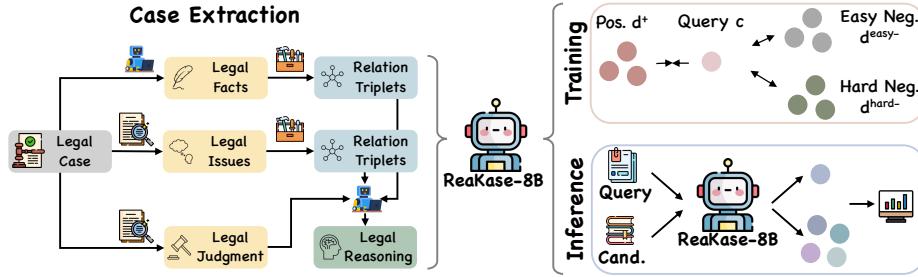


Fig. 1: REAKASE-8B designs key information extraction for knowledge triplets and legal reasoning to obtain contextualised case representations. The training is based on contrastive learning with positive (Pos.), easy negative (Neg.) and hard negative samples given a query case. During inference, REAKASE-8B embeddings for the query case and candidate (Cand.) cases will be used for retrieval.

capability of LLM embeddings in legal scenarios by integrating LLM embeddings into various existing retrieval frameworks [27, 34, 35, 52]. With the prevalence in retrieval-augmented generation in LLMs, legal case retrieval has been integrated as part of the retrieval process for trustworthy legal generation [22, 30]. The proposed REAKASE-8B does not aim to build another LLM that generates token-level response to user input. But rather, the main focus of this paper is to develop a powerful legal embedding model for legal case retrieval.

3 Method

This section presents REAKASE-8B in detail with the diagram in Figure 1.

Task Definition. In the legal case retrieval task, the goal is to identify precedents relevant to a given query case q from a collection of candidate cases. Formally, let $\mathcal{D} = \{d_1, d_2, \dots, d_n\}$ denote a repository of n candidate cases. The objective is to retrieve a subset of relevant cases, defined as $\mathcal{D}^* = \{d_i \mid d_i \in \mathcal{D}, \text{relevant}(d_i, q)\}$, where $\text{relevant}(d_i, q)$ indicates that case d_i is legally relevant to the query case q .

3.1 Legal Element Extraction

To capture the key legal elements that characterise a case, legal facts, issues, and judgements are extracted by leveraging legal knowledge from well-structured legal case documents.

Legal Facts. Legal facts capture the essential elements of a case, describing the “who, when, what, where, and why”. For example, in legal cases from COLIEE, these details are typically located in the Background section $c_{(Bg)}$, which often spans thousands of words and includes redundant or repetitive content. Such noise hinders the generation of effective case representations. To mitigate this,

we employ GPT-5 with the prompt in below to generate concise and accurate summaries of case facts c_{Fact} with the Background:

Prompt Template for Legal Fact Extraction
Summarize in 50 words: $\{c_{(Bg)}\}$

Legal Issues. In the legal domain, an issue refers to “a critical feature that focuses on the dispute points between the parties in the case”². In the COLIEE datasets, issues typically appear in the Analysis section, where judges articulate the contested points and provide supporting explanations. In this process, facts, issues, or judgements from precedents are often cited to justify the final decision. To anonymise these references, the datasets replace cited case names with placeholders such as `FRAGMENT_SUPPRESSED`. These placeholders indicates that there is a reference of precedent case in this sentence based on the legal logic presented in the case text. Following this design, all sentences containing such placeholders are regarded as the legal issues c_{Issue} of the case c , as they explicitly capture both the disputes under consideration and the precedents invoked in judicial reasoning. The extraction legal issues are then defined as:

$$c_{\text{Issue}} = \{ s \in c_{\text{Ana}} \mid [\text{PH}] \in s \}, \quad (1)$$

where c_{Ana} denotes the set of sentences in the Analysis section of case c , and s is a sentence. PH is the placeholder and it can be special tokens, such as `FRAGMENT_SUPPRESSED`.

Legal Judgements. In legal cases, the judgement represents the court’s final decision, reflecting both the resolution of the dispute and the authoritative determination of the legal issues. Within the COLIEE datasets, judgements are typically located in the concluding section of case documents, often signposted by headings such as “Judgement” or “Order.” This consistent formatting provides a reliable basis for automatically extracting the judgement text. However, in some cases, editor names or attribution notes appear after the true ending of the judgement, which are not part of the judicial decision. To maintain accuracy and relevance, such irrelevant content is removed during preprocessing. The resulting judgement text for each case c is denoted as c_{Jud} :

$$c_{\text{Jud}} = \{ s \in c_{\text{Con}} \mid s \text{ follows headings “Judgement” or “Order”} \}, \quad (2)$$

where c_{Con} denotes the set of sentences in the concluding section of the case c .

3.2 Legal Relation Triplets

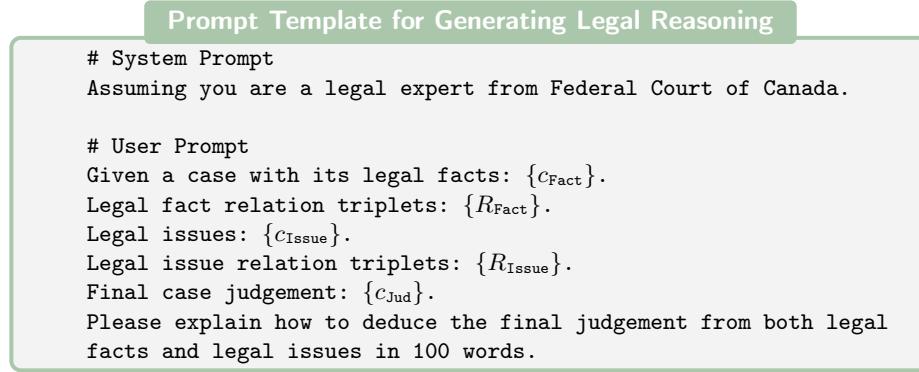
To capture structural information in legal texts, we extract relational triplets from the previously identified legal facts and issues. This extraction leverages open-source named entity recognition (NER) and relation extraction tools, which

² <https://www.uscourts.gov/glossary>

identify entities in the text and the semantic relations between them. For instance, in the COLIEE datasets derived from the Federal Court of Canada [13,14], the sentence “The claimant filed an appeal” yields the triplet (*claimant*, *filed*, *an appeal*). Formally, a case is represented as a set of relational triplets $\mathcal{R} = (h, r, t)_{i=1:n}$, where h and t denote the head and tail entities, r denotes the relation, and n is the total number of extracted triplets. Specifically, we denote the triplets extracted from legal facts and legal issues as $\mathcal{R}_{\text{Fact}}$ and $\mathcal{R}_{\text{Issue}}$, respectively. Modelling the relations among diverse legal entities captures richer structural and latent information than raw text alone, providing complementary semantic and structural signals to enhance case representations for downstream tasks.

3.3 Legal Reasoning Generation

Legal reasoning refers to the logical process through which a final judgement is derived from the underlying legal facts and issues of a case. To generate reasoning that reflects the judicial process of judges, the previously extracted legal facts, legal issues, and legal judgements are utilised as inputs for reasoning generation. A structured prompt template is designed for this purpose:



Using this prompt, the legal reasoning for each case is represented as:

$$c_{\text{Reason}} = \text{LLM}\left(c_{\text{Fact}}; R_{\text{Fact}}; c_{\text{Issue}}; R_{\text{Issue}}; c_{\text{Jud}}\right), \quad (3)$$

where c_{Reason} denotes the generated reasoning from legal facts and issues towards the judgement of a case. “;” denotes the concatenation. GPT-5 is employed to produce concise explanations that outline the inferential steps linking legal facts and issues to the final judgement. This generated reasoning serves as a natural language interpretation of judicial decision-making, bridging the gap between raw case descriptions and outcomes. Incorporating such reasoning enhances interpretability and supports the legal case retrieval.

3.4 REAKASE-8B Framework

REAKASE-8B is built upon a base and general embedding model, Qwen3-Embedding-8B [50], which does not have natural language generation ability.

The model first employs a contextualised encoding module to reconstruct each case into a structured textual input that integrates facts, issues, judgements, relation triplets, and reasoning. Building on these enriched representations, a contrastive learning objective is then utilised to fine-tune REAKASE-8B, ensuring the embeddings of relevant cases converge while irrelevant ones diverge. This fine-tuning process allows REAKASE-8B to effectively capture fine-grained semantic similarities and achieve strong generalisation across diverse legal scenarios, yielding robust and accurate precedent retrieval.

Contextualised Case Encoding. During training, each case is reformulated as c_{Cont} by concatenating its extracted legal facts, legal issues, as well as their knowledge-enhanced triplets, and the generated legal reasoning:

Prompt Template for Contextualised Case Encoding

```
# System Prompt
The following contains key components of a legal case.

# User Prompt
Legal facts: { $c_{\text{Fact}}$ }.
Legal fact relation triplets: { $\mathcal{R}_{\text{Fact}}$ }.
Legal issues: { $c_{\text{Issue}}$ }.
Legal issue relation triplets: { $\mathcal{R}_{\text{Issue}}$ }.
Legal reasoning: { $c_{\text{Reason}}$ }.
```

The judgement itself is not included in this prompt template is because that the judgement itself, a fairly short decisive sentence, does not provide too much information in helping the legal case retrieval. This is also observed in empirical study that with the judgement, the retrieval performance is not improved.

Feeding the reformulated case c_{Cont} into the LLM, the case is encoded into a dense vector representation by REAKASE-8B as:

$$\mathbf{x}_c = \text{REAKASE-8B}(c_{\text{Cont}}), \quad (4)$$

where $\mathbf{x}_c \in \mathbb{R}^d$ denotes the d -dimensional semantic embedding of the case, capturing its integrated factual, legal, and reasoning elements for downstream retrieval and comparison tasks.

Contrastive Learning Objective. The contrastive objective encourages embeddings of the query case and positive cases to be pulled closer together while pushing embeddings of negative cases apart, thereby enabling the model to learn discriminative and semantically enriched case representations. To further enhance training effectiveness, hard negative samples are introduced by retrieving cases with high BM25 relevance scores that are not labelled as ground-truth matches, ensuring the model is challenged with difficult but informative distinctions:

$$\ell = -\log \frac{e^{(s(\mathbf{x}_q, \mathbf{x}_{d+}))/\tau}}{e^{(s(\mathbf{x}_q, \mathbf{x}_{d+}))/\tau} + \sum_{i=1}^n e^{(s(\mathbf{x}_q, \mathbf{x}_{d_i^{easy}}))/\tau} + \sum_{j=1}^m e^{(s(\mathbf{x}_q, \mathbf{x}_{d_j^{hard}}))/\tau}}, \quad (5)$$

In Equation 5, q denotes the query case. The positive sample d^+ is derived from the ground-truth labels. Negative samples are composed of two types: easy negatives d^{easy-} , which are randomly drawn from the candidate pool and include in-batch samples; and hard negatives d^{hard-} , which are retrieved using high BM25 similarity scores. The variables n and m denote the number of easy and hard negative samples, respectively. The similarity function s , implemented using either the dot product or cosine similarity, measures the similarities between case representations. Finally, the temperature coefficient τ scales the similarity scores, controlling the sharpness of the softmax distribution during contrastive learning.

4 Experiments

The following research questions (RQs) are studied in this section:

- **RQ1:** How does REAKASE-8B perform compared with existing state-of-the-art legal case retrieval models?
- **RQ2:** What is the impact of the extracted legal entities and the generated legal reasoning in legal case retrieval?
- **RQ3:** How does different prompt-based contextualisation impact legal case representations?
- **RQ4:** When does the generated legal reasoning help with effective legal case retrieval?

Datasets. To evaluate the effectiveness of the proposed REAKASE-8B model, two widely adopted benchmark datasets are utilised: COLIEE2022 [13] and COLIEE2023 [14]. Both datasets are released as part of the Competition on Legal Information Extraction and Entailment (COLIEE) and contain legal cases from the Federal Court of Canada with statistics in Table 1. These two datasets constitute the most widely used English benchmarks for legal case retrieval. Beyond English, REAKASE-8B can be readily extended to multilingual legal systems by integrating domain-specific information extraction pipelines and multilingual language models.

Baselines. Three categories of baselines are included: (1) **Statistical model:** BM25 [29], using term frequency and inverse document frequency. (2) **Legal-specific language model (LM):** LEGAL-BERT [8], fine-tuning BERT with legal data; MonoT5 [25], being trained on broad retrieval tasks; SAILER [18], leveraging legal case structure to fine-tune BERT; PromptCase [36], reformulating the legal facts and issues in BERT. (3) **Top-performing open-sourced large language models (LLM):** E5-Mistral-7B-Instruct (Mistral-7B) [42], Qwen3-Embedding-8B (Qwen3-8B) [50] and Inf-Retriever-V1 (Inf-7B) [45] are three representative LLM embedding models on MMTEB legal tasks.

Table 1: Statistics of datasets.

Datasets	COLIEE2022		COLIEE2023	
	train	test	train	test
# Query	898	300	959	319
# Candidates	4415	1563	4400	1335
# Avg. relevant cases	4.68	4.21	4.68	2.69
Avg. length (# token)	6724	6785	6532	5566
Largest length (# token)	127934	85136	127934	61965

Metrics. We evaluate ReaKase using seven standard metrics widely adopted in information retrieval and legal case retrieval, with a focus on the top-5 retrieved cases. Following prior LCR studies [18, 24, 36], the metrics include Precision, Micro-F1, Macro-F1, Mean Reciprocal Rank (MRR@K), Mean Average Precision (MAP@K), and Normalized Discounted Cumulative Gain (NDCG@K). In all cases, higher values indicate better performance.

Implementation. For dataset preprocessing, all French text is removed from both datasets. Relation and entity extraction are conducted using open-source tools, including spaCy³, Stanford OpenIE [3], and LexNLP⁴. Qwen3-Embedding-8B [50] is adopted as the base model. For training, the batch size is set to 2. We use Adam [17] as the optimiser, with the learning rate selected from 1e-5, 5e-6, 1e-6, 5e-7, 1e-8 and weight decay from 1e-5, 1e-4, 1e-3. For each query, one positive sample, one randomly selected easy negative sample, and one hard negative sample are included. In-batch samples from other queries are additionally treated as easy negatives. The model input length is limited to 2048 tokens. To enable efficient fine-tuning, we employ LoRA [15] with rank $r = 8$, scaling factor $\alpha = 32$, and dropout rate 0.1 for regularising the adaptation matrices. All training experiments are run for 1,000 steps on 4 AMD Mi300x GPUs.

4.1 Overall Performance (RQ1)

We evaluate REAKASE-8B against multiple baselines on COLIEE2022 and COLIEE2023 (Table 2). REAKASE-8B consistently outperforms lexical methods, LM-based models, and LLM-based embeddings across nearly all metrics, demonstrating the effectiveness of legal relation entities and contrastive learning for legal case retrieval. Among traditional LMs, BM25 and PromptCase achieve moderate results, while recent LLM-based retrievers (Mistral-7B, Qwen3-8B, Inf-7B) enhance retrieval performance, attaining improved scores across all metrics. Notably, **REAKASE-8B generalises strongly across both datasets, outperforming all baselines** in all seven metrics, highlighting its robustness and competitive advantage in legal case retrieval.

Table 2: Results on COLIEE2022 and COLIEE2023 (%; avg. of five runs). Boldface indicates the best method (paired t-test, $p \leq 0.05$, Bonferroni corrected).

Methods	COLIEE2022										COLIEE2023									
	P@5 R@5 Mi-F1 Ma-F1 MRR@5 MAP NDCG@5					P@5 R@5 Mi-F1 Ma-F1 MRR@5 MAP NDCG@5					P@5 R@5 Mi-F1 Ma-F1 MRR@5 MAP NDCG@5					P@5 R@5 Mi-F1 Ma-F1 MRR@5 MAP NDCG@5				
BM25	17.9	21.2	19.4	21.4	23.6	25.4	33.6	16.5	30.6	21.4	22.2	23.1	20.4	23.7						
LM	LEGAL-BERT	4.47	5.30	4.85	5.38	7.42	7.47	10.9	4.64	8.61	6.03	11.4	11.3	13.6						
	MonoT5	0.71	0.65	0.60	0.79	1.39	1.41	1.73	0.38	0.70	0.49	0.47	1.17	1.33	0.61					
	SAILER	16.6	15.2	14.0	16.8	17.2	18.5	25.1	12.8	23.7	16.6	17.0	25.9	25.3	29.3					
	PromptCase	17.1	20.3	18.5	20.5	35.1	33.9	38.7	16.0	29.7	20.8	21.5	32.7	32.0	36.2					
LLM	Mistral-7B	21.4	25.4	23.2	25.7	26.8	28.5	38.0	16.0	29.7	20.8	21.5	21.9	22.9	30.8					
	Qwen3-8B	21.6	25.7	23.5	26.0	26.7	29.0	37.8	18.4	34.1	23.9	25.2	25.8	27.5	36.4					
	Inf-7B	21.1	25.1	22.9	25.5	26.6	28.7	37.9	19.0	35.3	24.7	26.0	26.5	28.0	37.6					
	ReaKase-8B	24.7	29.5	27.0	29.5	52.7	50.6	55.6	22.1	41.0	28.7	29.8	48.1	46.9	52.5					

³ <https://spacy.io/>

⁴ <https://github.com/LexPredict/lexpredict-lexnlp>

4.2 Ablation Study (RQ2)

To assess the contribution of legal entities and legal reasoning in REAKASE-8B, we perform an ablation study by selectively removing each component (Table 3). Removing both components leads to a substantial performance drop on COLIEE2022 and COLIEE2023, highlighting their critical roles in legal case retrieval. Incorporating legal entities alone yields modest gains over the baseline, whereas introducing legal reasoning alone provides a larger improvement, particularly in ranking metrics such as MRR@5 and NDCG@5. Combining both components achieves the best overall performance across most metrics, with Mi-F1 reaching 27.0 and 28.7, and NDCG@5 reaching 55.6 and 52.5 on COLIEE2022 and COLIEE2023, respectively. These results suggest that while legal reasoning contributes more substantially, the **integration of both the legal relation knowledge and the reasoning produces a complementary effect, leading to the most robust and consistent retrieval performance.**

Table 3: Ablation study. Entity and Reason denote legal entities and legal reasoning, separately. (%)

Entity	Reason	COLIEE2022							COLIEE2023						
		P@5	R@5	Mi-F1	Ma-F1	MRR@5	MAP	NDCG@5	P@5	R@5	Mi-F1	Ma-F1	MRR@5	MAP	NDCG@5
✗	✗	24.6	29.2	26.7	29.0	50.7	48.5	53.8	20.7	38.5	27.0	28.0	44.9	43.8	48.6
✓	✗	22.3	26.4	24.2	26.5	48.8	46.5	51.4	21.7	40.3	28.2	29.0	46.8	45.1	50.7
✗	✓	24.5	29.1	26.6	29.1	51.6	49.1	54.5	22.0	40.9	28.6	29.3	47.1	46.2	52.0
✓	✓	24.7	29.5	27.0	29.5	52.7	50.6	55.6	22.1	41.0	28.7	29.8	48.1	46.9	52.5

4.3 Effectiveness of Reasoning Context (RQ3)

To examine the impact of reasoning-context prompt templates on legal case retrieval, we evaluate three alternative user prompts within the contextualised legal case encoder on COLIEE2022 and COLIEE2023. This experiment investigates how prompt phrasing influences retrieval performance. In addition to the default template introduced in Section 3.4, two alternative prompts are illustrated in the below prompt figure, with corresponding results reported in Table 4.

Table 4: Effectiveness of different user prompts. Prompts are shown below.

Templates	COLIEE2022							COLIEE2023						
	P@5	R@5	Mi-F1	Ma-F1	MRR@5	MAP	NDCG@5	P@5	R@5	Mi-F1	Ma-F1	MRR@5	MAP	NDCG@5
Prompt 1	25.0	29.8	27.2	29.9	50.3	48.3	54.5	21.6	40.0	28.0	29.2	48.8	47.5	52.8
Prompt 2	24.5	29.0	26.6	29.2	49.7	48.7	54.2	22.0	40.8	28.6	29.8	47.8	46.7	52.5
Default	24.7	29.5	27.0	29.5	52.7	50.6	55.6	22.1	41.0	28.7	29.8	48.1	46.9	52.5

Across both datasets, **all prompts demonstrate an effective contextualisation for case encoding.** On COLIEE2022, Prompt 1 achieves the strongest performance on classification metrics such as Mi-F1 and Ma-F1, while the default prompt performs best on ranking metrics including MRR@5, MAP, and NDCG@5. Prompt 2 consistently falls between the two. On COLIEE2023, performance differences narrow considerably, with most metrics differing by less than 0.5 points across prompts. Nevertheless, a similar trend emerges: Prompt

1 provides stable, balanced performance, while the default prompt achieves the best results on most metrics across both datasets. These findings suggest that concise label-style prompts (Default) sharpen ranking effectiveness and improve recall, whereas explicit instructional prompts (Prompt 1) deliver more consistent and generalizable gains across datasets.

Prompt Template for Contextualised Case Encoding (Continued)

```
# System Prompt
The following contains key components of a legal case.

# User Prompt 1
Provide the key factual background: {cFact}.
Provide the legal fact relation triplets: {RFact}.
Provide the key legal disputes: {cIssue}.
Provide the legal issue relation triplets: {RIssue}.
Provide the legal reasoning between legal facts and legal
issues: {cReason}.

# Or User Prompt 2
List the important legal facts as: {cFact}.
List the important fact relations among events and parties
as: {RFact}.
List the important legal issues as: {cIssue}.
List the important issue relations among events and parties
as: {RIssue}.
List the important legal reasoning as: {cReason}.
```

4.4 Case Study

To illustrate the important role of reasoning in REAKASE-8B, we compare a query case (090305) with two candidate precedents: a positive match (010294) and a negative match (005442), where 010294 is correctly retrieved only when reasoning is used, and 005442 is wrongfully retrieved without reasoning but correctly excluded when reasoning is incorporated.

As shown in the red text in below examples of generated legal reasoning, for query case 090305, which turns on whether coercion amounts to duress under Section 34(1)(f) of Canada's Immigration and Refugee Protection Act (IRPA), a reasoning-aware model correctly retrieves case 010294. Both cases apply the same legal test from Ryan, whether the coercion meets the threshold of imminent peril, even though the outcomes differ. By contrast, a non-reasoning-based model retrieves 090305 with case 005442, which also cites Section 34(1) but focuses on procedural fairness and evidentiary flaws rather than duress. This shows that **reasoning alignment captures substantive relevance while filtering out statute-only matches**. This is because the statute-only matches are easily confused by the token level or wording level similarity that would otherwise lead to false positives.

Examples of Generated Legal Reasoning.

Generated Reasoning of Query Case (090305)

The Officer used the correct legal tests: ‘membership’ under s.34(1)(f) is broad; the complicity test (voluntary, significant, knowing contribution) applies to s.98/35(1)(a), not to membership; the SNM’s legitimacy is irrelevant. Duress requires imminent physical peril (Ryan). Mr. Mohamed showed general coercion, not imminent threats; thus his payments were not legally ‘under duress.’ On reasonableness, evidence supported that he was an SNM member. Therefore, inadmissibility was reasonably found; judicial review is dismissed. No certified question: no unsettled, dispositive legal issue.

Generated Reasoning of Positive Match (010294)

Applicant served in Iranian intelligence; officer found s.34(1) IRPA inadmissibility. Applicant claimed coercion/duress. Was the officer’s decision reasonable, including treatment of duress? Membership finding under s.34(1)(f) was reasonably made. But the officer unreasonably failed to fully assess the duress claim (a material defense affecting culpable “engaging/membership”). Because the decision was unreasonable on duress, it was quashed and remitted to a different officer. Application granted.

Generated Reasoning of Negative Match (005442)

Apply reasonableness to the s.34(1) IRPA finding and no deference on procedural fairness. ‘Membership’ requires more than suspicion: an institutional link/knowing participation, assessed contextually, considering exculpatory facts; low-level, compelled tasks (pamphlets/soap in high school) alone are insufficient. The officer ignored relevant membership criteria and contrary evidence, relied on credibility concerns as proxy, and withheld influential CBSA/CTS advocacy memoranda, breaching fairness. Thus the decision was unreasonable and procedurally unfair, so it must be set aside and remitted to a different officer.

5 Conclusion

This paper addresses two main challenges in legal case retrieval: limited legal reasoning and weak representation of structured legal relations. We introduce REAKASE-8B, which integrates legal element extraction and reasoning generation modules through contextualized prompting to enhance case semantics. REAKASE-8B achieves SOTA performance on both datasets, demonstrating its robustness and effectiveness for reasoning-oriented legal retrieval.

6 Acknowledgements

This work is supported by Australian Research Council CE200100025, DP230101196, DP230101753 and DE250100919.

References

1. Abolghasemi, A., Verberne, S., Azzopardi, L.: Improving bert-based query-by-document retrieval with multi-task optimization. In: ECIR (2022)
2. Althammer, S., Askari, A., Verberne, S., Hanbury, A.: Dossier@coliee 2021: Leveraging dense retrieval and summarization-based re-ranking for case law retrieval. CoRR [abs/2108.03937](#) (2021)
3. Angeli, G., Premkumar, M.J.J., Manning, C.D.: Leveraging linguistic structure for open domain information extraction. In: ACL (2015)
4. Askari, A., Abolghasemi, A., Pasi, G., Kraaij, W., Verberne, S.: Injecting the BM25 score as text improves bert-based re-rankers. In: ECIR (2023)
5. Askari, A., Peikos, G., Pasi, G., Verberne, S.: Leibi@coliee 2022: Aggregating tuned lexical models with a cluster-driven bert-based model for case law retrieval. CoRR [abs/2205.13351](#) (2022)
6. Askari, A., Verberne, S.: Combining lexical and neural retrieval with longformer-based summarization for effective case law retrieval. In: DESIRES. CEUR (2021)
7. Askari, A., Verberne, S., Abolghasemi, A., Kraaij, W., Pasi, G.: Retrieval for extremely long queries and documents with RPRS: a highly efficient and effective transformer-based re-ranker. CoRR [abs/2303.01200](#) (2023)
8. Chalkidis, I., Fergadiotis, M., Malakasiotis, P., Aletras, N., Androutsopoulos, I.: LEGAL-BERT: the muppets straight out of law school. CoRR [abs/2010.02559](#) (2020)
9. Chalkidis, I., Kampas, D.: Deep learning in law: early adaptation and legal word embeddings trained on large corpora. Artif. Intell. Law **27**(2), 171–198 (2019)
10. Colombo, P., Pires, T.P., Boudiaf, M., Culver, D., Melo, R., Corro, C., Martins, A.F.T., Esposito, F., Raposo, V.L., Morgado, S., Desa, M.: Saullm-7b: A pioneering large language model for law. CoRR [abs/2403.03883](#) (2024)
11. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT (2019)
12. Enevoldsen, K., Chung, I., Kerboua, I., Kardos, M., Mathur, A., Stap, D., Gala, J., Siblini, W., Krzemiński, D., Winata, G.I., Sturua, S., Utpala, S., Ciancone, M., Schaeffer, M., Sequeira, G., Misra, D., Dhakal, S., Rystrøm, J., Solomatin, R., Ömer Çağatan, Kundu, A., Bernstorff, M., Xiao, S., Sukhlecha, A., Pahwa, B., Poświata, R., GV, K.K., Ashraf, S., Auras, D., Plüster, B., Harries, J.P., Magne, L., Mohr, I., Hendriksen, M., Zhu, D., Gisserot-Boukhlef, H., Aarsen, T., Kostkan, J., Wojtasik, K., Lee, T., Šuppa, M., Zhang, C., Rocca, R., Hamdy, M., Michail, A., Yang, J., Faysse, M., Vatolin, A., Thakur, N., Dey, M., Vasani, D., Chitale, P., Tedeschi, S., Tai, N., Snegirev, A., Günther, M., Xia, M., Shi, W., Lù, X.H., Clive, J., Krishnakumar, G., Maksimova, A., Wehrli, S., Tikhonova, M., Panchal, H., Abramov, A., Ostendorff, M., Liu, Z., Clematide, S., Miranda, L.J., Fenogenova, A., Song, G., Safi, R.B., Li, W.D., Borghini, A., Cassano, F., Su, H., Lin, J., Yen, H., Hansen, L., Hooker, S., Xiao, C., Adlakha, V., Weller, O., Reddy, S., Muennighoff, N.: Mmteb: Massive multilingual text embedding benchmark. CoRR [abs/2502.13595](#) (2025)
13. Goebel, R., Kano, Y., Kim, M.Y., Loro, M.N., Minh, N.L., Rabelo, J., Rossi, J., Satoh, K., Savelka, J., Shao, Y., Shimazu, A., Tojo, S., Tran, V., Valvoda, J., Westermann, H., Yamada, H., Yoshioka, M., Wehnert, S.: Competition on legal information extraction/entailment (COLIEE) (2022)
14. Goebel, R., Kano, Y., Kim, M.Y., Loro, M.N., Minh, N.L., Rabelo, J., Rossi, J., Satoh, K., Savelka, J., Shao, Y., Shimazu, A., Tojo, S., Tran, V., Valvoda, J.,

- Westermann, H., Yamada, H., Yoshioka, M., Wehnert, S.: Competition on legal information extraction/entailment (COLIEE) (2023)
15. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Chen, W.: Lora: Low-rank adaptation of large language models. CoRR **abs/2106.09685** (2021)
 16. Jones, K.S.: A statistical interpretation of term specificity and its application in retrieval. *J. Documentation* **60**(5), 493–502 (2004)
 17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
 18. Li, H., Ai, Q., Chen, J., Dong, Q., Wu, Y., Liu, Y., Chen, C., Tian, Q.: SAILER: structure-aware pre-trained language model for legal case retrieval. CoRR **abs/2304.11370** (2023)
 19. Liu, B., Hu, Y., Wu, Y., Liu, Y., Zhang, F., Li, C., Zhang, M., Ma, S., Shen, W.: Investigating conversational agent action in legal case retrieval. In: ECIR (2023)
 20. Liu, B., Wu, Y., Zhang, F., Liu, Y., Wang, Z., Li, C., Zhang, M., Ma, S.: Query generation and buffer mechanism: Towards a better conversational agent for legal case retrieval. *Inf. Process. Manag.* (2022)
 21. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized BERT pretraining approach. CoRR **abs/1907.11692** (2019)
 22. Luo, K., Huang, Q., Jiang, C., Feng, Y.: Automating legal interpretation with llms: Retrieval, generation, and evaluation. In: ACL (2025)
 23. Ma, Y., Ai, Q., Wu, Y., Shao, Y., Liu, Y., Zhang, M., Ma, S.: Incorporating retrieval information into the truncation of ranking lists for better legal search. In: SIGIR (2022)
 24. Ma, Y., Shao, Y., Wu, Y., Liu, Y., Zhang, R., Zhang, M., Ma, S.: Lecard: A legal case retrieval dataset for chinese law system. In: SIGIR (2021)
 25. Nogueira, R., Jiang, Z., Pradeep, R., Lin, J.: Document ranking with a pretrained sequence-to-sequence model. In: EMNLP (2020)
 26. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. SIGIR (2017)
 27. Premasiri, D., Ranasinghe, T., Mitkov, R.: Llm-based embedders for prior case retrieval. CoRR **abs/2507.18455** (2025)
 28. Rabelo, J., Kim, M., Goebel, R.: Semantic-based classification of relevant case law. In: JURISIN (2022)
 29. Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In: SIGIR (1994)
 30. Santosh, T.Y.S.S., Jia, C., Goroncy, P., Grabmair, M.: Relexed: Retrieval-enhanced legal summarization with exemplar diversity. In: NAACL (2025)
 31. Shao, Y., Mao, J., Liu, Y., Ma, W., Satoh, K., Zhang, M., Ma, S.: BERT-PLI: modeling paragraph-level interactions for legal case retrieval. In: IJCAI (2020)
 32. Shu, D., Zhao, H., Liu, X., Demeter, D., Du, M., Zhang, Y.: Lawllm: Law large language model for the US legal system. In: CIKM (2024)
 33. Sun, Z., Xu, J., Zhang, X., Dong, Z., Wen, J.: Law article-enhanced legal case matching: a model-agnostic causal learning approach. CoRR **abs/2210.11012** (2022)
 34. Sun, Z., Zhang, K., Yu, W., Wang, H., Xu, J.: Logic rules as explanations for legal case retrieval. In: LREC/COLING (2024)
 35. Tang, Y., Qiu, R., Huang, Z.: Uqlegalai@coliee2025: Advancing legal case retrieval with large language models and graph neural networks. CoRR **abs/2505.20743** (2025)

36. Tang, Y., Qiu, R., Li, X.: Prompt-based effective input reformulation for legal case retrieval. In: Databases Theory and Applications - 34th Australasian Database Conference ADC. vol. 14386, pp. 87–100. Springer (2023)
37. Tang, Y., Qiu, R., Liu, Y., Li, X., Huang, Z.: Casegnn++: Graph contrastive learning for legal case retrieval with graph augmentation. CoRR **abs/2405.11791** (2024)
38. Tang, Y., Qiu, R., Liu, Y., Li, X., Huang, Z.: Casegnn: Graph neural networks for legal case retrieval with text-attributed graphs. In: ECIR 2024. pp. 80–95. Springer (2024)
39. Tang, Y., Qiu, R., Yin, H., Li, X., Huang, Z.: Caselink: Inductive graph learning for legal case retrieval. In: SIGIR. pp. 2199–2209. ACM (2024)
40. Tran, V.D., Nguyen, M.L., Satoh, K.: Building legal case retrieval systems with lexical matching and summarization using A pre-trained phrase scoring model. In: ICAIL (2019)
41. Vuong, T., Nguyen, H., Nguyen, T., Nguyen, H., Nguyen, T., Nguyen, H.: NOWJ at COLIEE 2023 - multi-task and ensemble approaches in legal information processing. CoRR **abs/2306.04903** (2023)
42. Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., Majumder, R., Wei, F.: Text embeddings by weakly-supervised contrastive pre-training. CoRR **abs/2212.03533** (2022)
43. Wang, Z.: Legal element-oriented modeling with multi-view contrastive learning for legal case retrieval. In: IJCNN (2022)
44. Xiao, C., Hu, X., Liu, Z., Tu, C., Sun, M.: Lawformer: A pre-trained language model for chinese legal long documents. AI Open **2**, 79–84 (2021)
45. Yang, J., Jiahe Wan, Y.Y., Chu, W., Xu, Y., Qi, Y.: inf-retriever-v1 (revision 5f469d7) (2025). <https://doi.org/10.57967/hf/4262>, <https://huggingface.co/infly/infly-retriever-v1>
46. Yao, F., Xiao, C., Wang, X., Liu, Z., Hou, L., Tu, C., Li, J., Liu, Y., Shen, W., Sun, M.: LEVEN: A large-scale chinese legal event detection dataset. In: ACL (2022)
47. Yu, W., Sun, Z., Xu, J., Dong, Z., Chen, X., Xu, H., Wen, J.: Explainable legal case matching via inverse optimal transport-based rationale extraction. In: SIGIR (2022)
48. Yue, S., Liu, S., Zhou, Y., Shen, C., Wang, S., Xiao, Y., Li, B., Song, Y., Shen, X., Chen, W., Huang, X., Wei, Z.: Lawllm: Intelligent legal system with legal reasoning and verifiable retrieval. In: DASFAA (2024)
49. Zhang, H., Dou, Z., Zhu, Y., Wen, J.R.: Contrastive learning for legal judgment prediction. ACM Trans. Inf. Syst. **41**(4), 25 (2023)
50. Zhang, Y., Li, M., Long, D., Zhang, X., Lin, H., Yang, B., Xie, P., Yang, A., Liu, D., Lin, J., Huang, F., Zhou, J.: Qwen3 embedding: Advancing text embedding and reranking through foundation models. CoRR **abs/2506.05176** (2025)
51. Zhong, H., Wang, Y., Tu, C., Zhang, T., Liu, Z., Sun, M.: Iteratively questioning and answering for interpretable legal judgment prediction. In: AAAI (2020)
52. Zhou, Y., Huang, H., Wu, Z.: Boosting legal case retrieval by query content selection with large language models. In: SIGIR-AP (2023)