WIRATUNGA, N., ABEYRATNE, R., JAYAWARDENA, L., MARTIN, K., MASSIE, S., NKISI-ORJI, I., WEERASINGHE, R., LIRET, A. and FLEISCH, B. 2024. CBR-RAG: case-based reasoning for retrieval augmented generation in LLMs for legal question answering. In Recio-Garcia, J.A., Orozco-del-Castillo, M.G. and Bridge, D (eds.) *Case-based reasoning research and development: proceedings of the 32nd International conference of case-based reasoning research and development 2024 (ICCBR 2024),* 1-4 July 2024, Merida, Mexico. Lecture notes in computer science, 14775. Cham: Springer [online], pages 445-460. Available from: https://doi.org/10.1007/978-3-031-63646-2_29

CBR-RAG: case-based reasoning for retrieval augmented generation in LLMs for legal question answering.

WIRATUNGA, N., ABEYRATNE, R., JAYAWARDENA, L., MARTIN, K., MASSIE, S., NKISI-ORJI, I., WEERASINGHE, R., LIRET, A. and FLEISCH, B.

2024

CBR-RAG: Case-Based Reasoning for Retrieval Augmented Generation in LLMs for Legal Question Answering *

Nirmalie Wiratunga[1], Ramitha Abeyratne[1], Lasal Jayawardena[1,2], Kyle Martin[1], Stewart Massie[1], Ikechukwu Nkisi-Orji[1], Ruvan Weerasinghe[2], Anne Liret[3], and Bruno Fleisch[3]

[1] Robert Gordon University, Aberdeen, UK
{n.wiratunga, r.abeyratne, l.jayawardena, k.martin3, s.massie, i.nkisi-orji}@rgu.ac.uk,
[2] Informatics Institute of Technology, Sri Lanka
ruvan.w@iit.lk
[3] BT France
{anne.liret, bruno.fleisch}@bt.fr

Abstract. Retrieval-Augmented Generation (RAG) enhances Large Language Model (LLM) output by providing prior knowledge as context to input. This is beneficial for knowledge-intensive and expert reliant tasks, including legal question-answering, which require evidence to validate generated text outputs. We highlight that Case-Based Reasoning (CBR) presents key opportunities to structure retrieval as part of the RAG process in an LLM. We introduce CBR-RAG, where CBR cycle's initial retrieval stage, its indexing vocabulary, and similarity knowledge containers are used to enhance LLM queries with contextually relevant cases. This integration augments the original LLM query, providing a richer prompt. We present an evaluation of CBR-RAG, and examine different representations (i.e. general and domain-specific embeddings) and methods of comparison (i.e. inter, intra and hybrid similarity) on the task of legal question-answering. Our results indicate that the context provided by CBR's case reuse enforces similarity between relevant components of the questions and the evidence base leading to significant improvements in the quality of generated answers.

Keywords: CBR, RAG, LLMs, Text Embedding, Indexing, Retrieval

1 Introduction

Retrieval-Augmented Generation (RAG) enhances the performance of large language models (LLMs) on knowledge-intensive NLP tasks by combining the strengths of pre-trained parametric (language models with learned parameters) and non-parametric (external knowledge resources e.g., Wikipedia) memories [15]. This

* This research is funded by SFC International Science Partnerships Fund.

hybrid approach not only sets new benchmarks in open-domain question answering by generating more accurate, specific, and factually correct responses but also addresses critical challenges in the field such as the difficulty in updating stored knowledge and providing provenance for generated outputs. For example, in the context of legal question-answering, RAG-based systems can retrieve publicly available legislation documents from open knowledge bases to provide context for user queries. However, such a system would require output that could be validated, and moderation of content generated by LLMs has been highlighted as a critical concern [11].

Case-Based Reasoning (CBR) presents an excellent opportunity here, as previous solutions form the basis of a knowledge-base which can be evidenced in best practice or regulations [17]. CBR can enhance the retrieval process in RAG models by organising the non-parametric memory in a way that cases (knowledge entries or past experiences) are more effectively matched to queries. Previous work on ensemble CBR-neural systems has also highlighted the benefits of CBR integration, demonstrating improvements in factual correctness over purely neural methods [22]. Accordingly, in this work we present three contributions.

- Firstly, we formalise the role of the CBR methodology to form context for RAG systems.
- Secondly, we provide an empirical comparison of different retrieval methods for RAG, examining different representations (i.e. general and domain-specific embeddings) and methods of comparison (i.e. inter, intra and hybrid similarity).
- Finally, we present these contributions in the context of the legal domain, and provide results for a generative legal question-answering (QA) application [1]. Our results highlight the opportunities of CBR-RAG systems for knowledge-reliant generative tasks.

This paper is structured as follows. In Section 2 we describe influential work targeting the legal domain from CBR and LLM literature. In Section 3, we formalise CBR-RAG and its application to legal QA, while in Section 4 we detail the different methods for creating and comparing embeddings to perform case retrieval. In Section 5 we describe the different encoders used to create embeddings. Finally, in Section 6 we discuss the methodology and results of our evaluation, followed by conclusions in Section 7.

2 Related Work

CBR boasts a long-standing history in the legal domain, with initial efforts concentrating on extracting features to index law cases effectively. These features, referred to as 'factors', are pivotal in systems like HYPO [3], which employ fact-oriented representations applied to trade secret law and extended to legal tutoring with the CATO [1] system. More generally with Textual CBR, a

[1] Reproducible code is available: https://github.com/rgu-iit-bt/cbr-for-legal-rag

key focus has been on extracting features for case comparison, using methods that range from decision trees, as exemplified by the SMILE system in creating indexing vocabularies for legal case retrieval [5], to association rules for case representation [24]. Much of this has now been advanced by the adoption of neural transformations of input text through transformer-style embeddings with LLMs.

The application of LLMs presents an interesting approach in addressing challenges within the legal domain. LLMs use language understanding capabilities to interact with users, enabling extraction of key elements from legal documents, and present information in an understandable manner to enhance decision-making processes. For this reason, GPT-based [14,18] and BERT-based [7] transformer models are popular for LegalAI. However their black-box nature and tendency to hallucinate and lack of factual faithfulness present significant challenges for deployment [13]. Retrieval-Augmented Generation (RAG) systems address this by presenting the LLM with factual data to generate responses [15,16], employing a variety of sophisticated fact identifying mechanisms [2,19]. However currently such retrieval methods in RAG do not make use of CBR's potential for varying matching strategies across different segments of the content being matched. Here we are reminded of research in CBR involving the integration with IR systems, specifically where CBR has been effectively used to retrieve 'most on-point' cases to guide the search and browse of vast IR collections [17]. Our work draws inspiration from these integrative approaches, applying the principles of CBR to improve contextual understanding within LLMs.

LLMs are typically pre-trained on general text and subsequently fine-tuned on legal texts to learn domain-specific representations. Obtaining sufficiently large data sets for LLM training poses a significant challenge. For example, pre-trained LLMs may be fine-tuned using the LEDGAR dataset [20] for legal text classification downstream tasks or on extensive corpora like the Harvard Law case corpus[1] using masked-language modelling or next-sentence prediction techniques. Thereafter tested on other legal downstream tasks, such as those found in the CaseHOLD dataset [8], which include tasks related to Overruling, Terms of Service, and CaseHOLD itself. We observe that there remains a notable scarcity of LLMs specifically applied to legal question answering tasks. This is likely because existing legal QA datasets are mostly small, manually curated datasets (for example, the rule-qa task in the LegalBench collection [10] is formed from 50 question-answer pairs). However the recent release of the Open Australian Legal Question-Answering (ALQA) dataset[2], comprising over 2,100 question-answer-snippet triplets (synthesised by GPT-4 from the Open Australian Legal Corpus), presents an opportunity for LLMs to expand to legal QA.

3 CBR-RAG: Using CBR to form context for LLMs

In CBR-RAG, we integrate the initial retrieve-only stage of the CBR cycle with its indexing vocabulary and similarity knowledge containers to enable the re-

[1] https://case.law/
[2] https://huggingface.co/datasets/umarbutler/open-australian-legal-qa

trieval of cases that serve as context for querying an LLM. Consequently, the original LLM query is augmented with content retrieved via CBR, creating a contextually enriched prompt for the LLM.

Figures 1 illustrates a high-level architecture of a generative model for Question-Answering systems, highlighting the integration of CBR within it. Here we denote the generative LLM model as, G, and the prompt, P, used to generate the response as a tuple, $p = (\mathcal{Q}, \mathcal{C})$, where $\mathcal{Q}$ is the question reflecting the user's query, and, $\mathcal{C}$, is the context text with relevant details to guide the response generation. The response generated by the model for the query, $\mathcal{Q}$, is denoted as the answer, A.

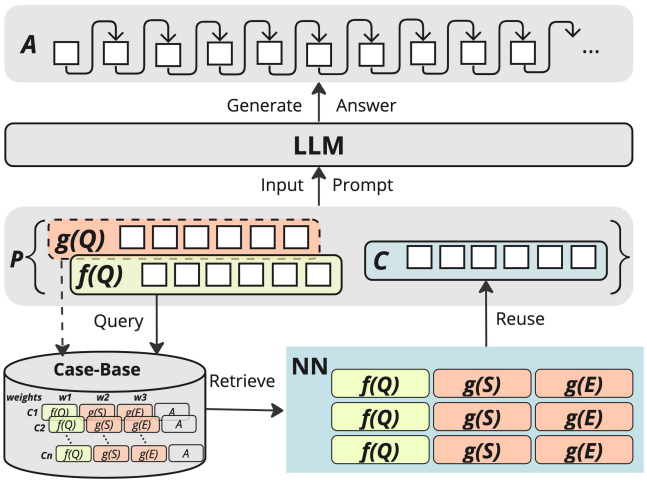

Fig. 1: CBR-RAG

3.1 Casebase

We used the Australian Open Legal QA (ALQA) [6] dataset to create the case-base. The dataset is formed of 2,124 LLM-generated question-answer pairs from the Australian Open Legal Corpus dataset. Each QA pair is corroborated by a supporting textual snippet from a legal document within the corpus for factual validation. An example case about the 'interpretation of reasonable grounds in searches without a warrant' appears in Table 1. Here the support text provides the context in which the question should be answered. The bold text further highlights examples of named entities that might usefully be captured separately for case comparison purposes.

Figure 2 provides a frequency distributions of the legal acts identified in the casebase with the most frequently referenced legal acts in the dataset listed in

Table 1: Examples Legal Q&A case

Component	Description
Case Name	Smith v The State of New South Wales (2015)
Question	How did the case of Smith v The State of New South Wales (2015) clarify the interpretation of 'reasonable grounds' for conducting a search without a warrant?
Support	In [**Case Name: Smith v The State of NSW (2015)**], the plaintiff [**Action: was searched without a warrant**] [**Location: near a known drug trafficking area**] based on the plaintiff's nervous demeanor and presence in the area, but [**Outcome: no drugs were found**]. The legality of the search was contested, focusing on [**Legal Concept: whether 'reasonable grounds' existed**].
Answer	The case ruled 'reasonable grounds' require clear, specific facts of criminal activity, not just location or behavior.

Index	**Act** (**Freq**)
1	Federal Court Rules (9)
2	Civil Aviation Regulations 1998 (8)
3	Land & Env. Court Act 1979 (8)
4	Corporations Act 2001 (Cth) (7)
5	Migration Act 1958 (Cth) (6)
6	Customs Act 1901, Tariff Act 1995 (6)
7	Industrial Relations Act 1996 (6)
8	Environmental Planning and Assessment Act 1979 (5)
9	Trees (Disputes Between Neighbours) Act 2006 (5)
10	Migration Act 1958 (5)

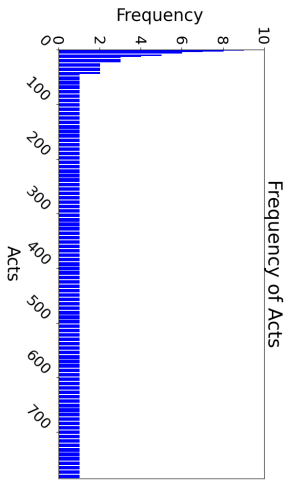

Fig. 2: Ten most frequent legal acts in the casebase are listed on the left, and the legal act frequency distribution appears on the right.

the table (extracted using the prompt in Table 3). The most frequently mentioned act is the 'Federal Court Rules' out of 785 unique legal acts. Within the dataset, 1,183 (57%) cases were found to have no reference to legal acts, while only 44 acts appeared in more than 1 case. Accordingly, relying solely on legal acts for indexing would not be suitable for this casebase. Instead, it presents an ideal opportunity for experimentation with neural embeddings. These embeddings form the indexing vocabulary, with weighted similarity contributing to the similarity knowledge.

4 Representation and Similarity

We formalise our retrieval and representation methods. Let's denote the casebase as a collection of cases $C = \{c_1, c_2, \ldots, c_n\}$ containing question and answer legal cases. Each case c, in the context of RAG is formalised as a tuple,

$$c = < Q, S, E, A >$$

where Q represents a **question**, A represents the **answer**, S represents the **support** for a given answer from an evidence base, and E represents a set of **entities** extracted from S. This case representation underpins RAG in its use of S. Typical CBR for question answering would be composed mainly of problem-solution components (i.e. question and answer respectively together with any lessons learnt), structuring cases as problem-solution-support enhances answer generation for the LLM by providing factually accurate context. Furthermore, while only the most relevant component of a document is extracted as the support text, the link to the full document is available in the original ALQA corpus.

4.1 Representation

Initially, in textual form, each part is represented by neural embeddings. For diverse retrieval scenarios, we use a dual embedding form for Q to enable matching it with not only other questions but where necessary matching it to the supporting text or the entities as follows:

- **Intra Embeddings**, $f(\cdot)$, optimised for attribute matching. These embeddings facilitate attribute-to-attribute comparisons where local similarities can be computed between the same types of attributes (e.g., questions with questions).
- **Inter Embeddings**, $g(\cdot)$, designed for information retrieval (IR) scenarios. These embeddings allow for matching that is not restricted to like attributes, enabling inter-attribute similarity assessment. This approach is particularly useful in situations where a question may be relevant for comparison to the support text or the entities.

A representation using intra-embedding is useful for tasks like semantic textual similarity that focus on finding sentences with closely related meanings, even if phrased differently. For example, a sentence like 'The judge dismissed the case due to lack of evidence.' would find a semantically similar counterpart in 'The court threw out the lawsuit because there was insufficient proof' Conversely, an inter-embedding representation is suited for tasks aimed at searching for documents relevant to a query. For example, a legal query on 'copyright infringement in digital media' may yield cases with outcomes such as rulings on unauthorised content distribution or streaming without permission. These highlight the distinction between representations needed for ascertaining semantic similarity with intra-embeddings ($f(.)$), and finding relevant content with inter-embeddings ($g(.)$).

The dual-embedding case representation, accommodating both forms of retrieval tasks, is given by:

$$c = < f(Q), g(Q), g(S), g(E), A >$$

Similarly, the prompt can be expressed using the inter embedding as follows[1]:

$$p = < f(\mathcal{Q}), g(\mathcal{Q}), \mathcal{C} >$$

4.2 Case Retrieval

We use case retrieval to augment the context part of a prompt, p, given its query, $\mathcal{Q}$. Accordingly, there are three comparison strategies for case retrieval: intra-, inter-, and hybrid-embedding based retrieval.

Intra-embedding retrieval involves matching on the basis of embeddings obtained from function $f(.)$. Here $f(\mathcal{Q})$ which represents the embeddings from the prompt's query is matched to query parts of the cases in the casebase. The best matching case identified from intra-embedding retrieval is defined by:

$$\beta_k = \underset{c_i \in C,\, k}{\text{top-k}}\ \text{Sim}(f(\mathcal{Q}), f(Q_i)) \tag{1}$$

Here, top-k refers to the selection of indices corresponding to the k highest-scoring cases as determined by the similarity measure. β represents the indices of these retrieved cases, while Sim is a similarity metric (e.g., cosine similarity) that measures the similarity between the intra-embedding of the prompt's question and the intra-embeddings of the question parts of each case in the casebase.

Inter-embedding retrieval uses $g(\mathcal{Q})$ from the prompt to search the casebase, focusing on identifying relevant cases akin to an information retrieval style search, where attribute-to-attribute matching is not strictly followed. The best matching case is identified as follows:

$$\beta_k = \underset{c_i \in C,\, k}{\text{top-k}}\ \text{Sim}(g(\mathcal{Q}), g(X_i)) \tag{2}$$

where X_i can be either S_i (the Snippet part) or E_i (the Entity part) of the case.

Hybrid embedding retrieval is an alternative matching that involves a combination of both intra and inter-embeddings of the prompt's $\mathcal{Q}$ representations being used to match cases in a hybrid weighted retrieval approach:

$$\begin{aligned}\beta_k = \underset{c_i \in C,\, k}{\text{top-k}}\ (&w_1 \cdot \text{Sim}(f(\mathcal{Q}), f(Q_i)) + w_2 \cdot \text{Sim}(g(\mathcal{Q}), g(S_i)) + \\ &w_3 \cdot \text{Sim}(g(\mathcal{Q}), g(E_i)))\end{aligned} \tag{3}$$

[1] Note: In our notation, we employ a calligraphic font for the prompt components $(f(\mathcal{Q}), g(\mathcal{Q}), \mathcal{C})$ to distinguish them from those of cases. Despite this stylistic difference, it is important to understand that both prompts and cases utilise similar embedding representations.

This approach utilises both representation forms of the prompt's query $f(\mathcal{Q})$ and $g(\mathcal{Q})$ for retrieval. Given the top k cases $\{c_{\beta_j}\}_{j=1}^{k}$, we can extract the context for the prompt based on the retrieval option ρ as follows:

$$\text{Context, } \mathcal{C}(\rho) = \begin{cases} \{S_{\beta_j}\}_{j=1}^{k} & \text{if } \rho = 1, \text{ “support-text-only”} \quad (4a) \\ \{Q_{\beta_j}, S_{\beta_j}, E_{\beta_j}, A_{\beta_j}\}_{j=1}^{k} & \text{if } \rho = 2, \text{ “full-case”} \quad (4b) \end{cases}$$

5 Embedding models

In this work, we explore embeddings generated by BERT, AnglEBERT, and LegalBERT; the latter being pre-trained on diverse English legal documents, with the rest all being general-purpose embeddings. We next provide an overview of these models, as illustrated in Figure 3, and discuss how they can be used to generate the f and g forms of representations.

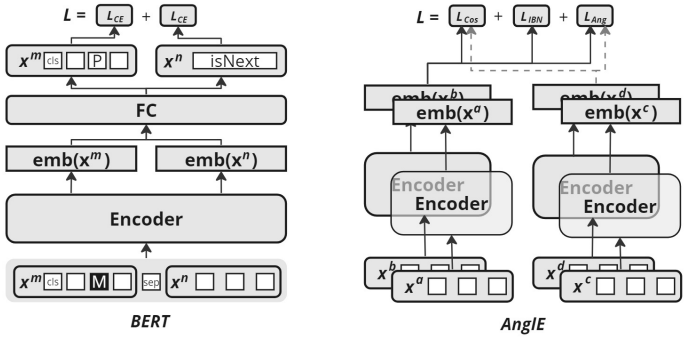

Fig. 3: Architecture and training process for BERT and AnglEBERT. Note that LegalBERT has the same architecture as BERT, but is pre-trained on legal text.

5.1 BERT

The Bidirectional Encoder Representations from Transformers (BERT) model [9] is a language model for learning generalisable text embeddings. The model is formed of an encoder block (taken from the transformer architecture in [23]), followed by a fully-connected layer. The bidirectional nature of BERT is derived from its pre-training technique, which conditions on both the left and right contexts of input sentences simultaneously. The model uses a self-supervised learning strategy that combines masked-language modeling (MLM) with next sentence prediction (NSP) to acquire contextually rich word embeddings. During training, a pair of sentences is fed into the model, and a random subset of words is replaced with the [MASK] token, establishing a sequence-to-point task. The

objective here is to predict the masked words by using the context provided by both previous and subsequent words. Furthermore, a [CLS] token is inserted at the beginning of the input sequence to accumulate contextual information from the entire sequence, facilitating tasks like sentence relationship classification. Pre-training with both MLM and NSP, can be seen as typical text prediction and is controlled by the standard cross-entropy objective function. BERT is pre-trained on a combination of large general purpose text datasets totalling 3.3B words and has demonstrated strong performance across many domains.

5.2 LegalBERT Trained on General Legal Data

Domain-specific knowledge is known to be beneficial for legal tasks [7,21]. For example, the word 'case' may refer to a variety of containers (brief case, suit case, display case, etc), but also a court case. While semantic relations with the latter are likely to be impactful for legal question answering, the greater frequency of the former in general corpora will result in embeddings more weighted towards that context. The LegalBERT family of models [7] enhance the BERT model by further pre-training on an additional 12 GB of diverse English legal documents from a mix of UK, EU, and US legislation and case law. The goal is to embed domain knowledge into produced embeddings by learning from a relevant document set.

5.3 BERT with AnglE Embeddings

The AnglE embeddings [16] adopts a contrastive learning strategy (similar to Siamese networks [4]) to learn embeddings through matching both positive and negative text pairs, where a positive pair is considered to be similar (above some threshold usually). The process begins with BERT embeddings, input to AnglE for optimisation. Its novelty stems from one of the three loss functions, that are designed to overcome the vanishing gradient issue encountered in cosine similarity-based comparisons, particularly at the extremes of similarity (or dissimilarity). This is achieved by comparing embeddings based on angle and magnitude within complex spaces, effectively bypassing cosine similarity's saturation problem. Once trained the AnglE embedding model can be used to generate text embeddings using a final pooling layer [16].

One of the difficulties when using AnglEBERT for a specific domain is that one must generate a supervised training set (unlike with BERT which can be trained in an unsupervised manner using the MLM and NSP self-supervision methods). This is because the AnglE method adopts a contrastive learning strategy, where the supervised dataset must include paired instances for training. This can be prohibitive in contexts where domain expertise is required for labelling.

5.4 Dual-embedding Case Representation with AnglE

For the purposes of this work, we leverage dual-embeddings introduced in [16]. In terms of the inter-embedding retrieval, a specific embedding prompt cue $Cue(\mathcal{Q})$

is used to contextualise the relevance of the query embeddings towards matching with attributes other than that of questions, as follows:

$$Cue(Q) = \text{“Represent this sentence for searching relevant passages: ”}\{Q\}$$

The idea here is that the cue is used to influence the embedding generation towards inter-retrieval oriented embeddings as follows:

$$g(Cue(\mathcal{Q}))$$

For intra-embedding retrieval the prompt text is empty, i.e. input the text without specifying an additional prompt cue. This means the embedding function f processes the query $\mathcal{Q}$ directly, as $f(\mathcal{Q})$. Table 2 provides an example of question and support text version that are used as input to each of BERT, AnglEBERT and LegalBERT with relevant prompts to enable the generation of the alternative embeddings that we intend using during case matching with CBR.

Table 2: Comparison of an example question with and without the *Cue* text (in blue) to create inter and intra embeddings.

Embedding	**Question**
intra	f(“What were the court’s findings regarding the financial liabilities of Allco Finance Group Ltd to Blairgowrie Trading Ltd?”)
inter	g(“Represent this sentence for searching relevant passages:” + “What were the court’s findings regarding the financial liabilities of Allco Finance Group Ltd to Blairgowrie Trading Ltd?”)

6 Evaluation

The aim of our evaluation is two-fold: 1) to understand the impact of inter and intra embeddings on weighted retrieval, and 2) to understand the generative quality of RAG systems when coupled with a case-based retrieval-only system. To analyse weighted retrieval we compared several representation combinations of embedding models and similarity weights as follows:

- compare three alternative forms of text embeddings using the encoders presented in Section 5 - BERT, LegalBERT and AnglEBERT;
- compare four alternative weighting schemes to assess utility of question, support, and entity components within case representations. These include:
  - Question only, represented by the weights [1,0,0] (see Equation 1),
  - Support only, with weights [0,1,0],
  - Entities only, using weights [0,0,1] (see Equation 2),
  - A hybrid approach, combining these components with weights [0.25, 0.40, 0.35] (refer to Equation 3).

Accordingly BERT[0,1,0] would denote using a Support only version of retrieval with the BERT embedding; and using the same naming convention, AnglE-BERT[0.25,0.4,0.35] indicates a hybrid dual embedding method where AnglE-BERT embeddings are used with the specified weights for case retrieval. Our baseline comparator is an LLM with no case retrieval i.e. No-RAG.

We selected Mistral [12] for answer generation at test time, due to its open-source availability, allowing us to use a model distinct from OpenAI's GPT-4, which was employed for formulating the Q&A casebase. This approach effectively simulates consulting an alternative expert in place of a human specialist.

6.1 Legal QA Dataset Analysis

The ALQA dataset, introduced in Section 3 is a synthetic Q&A dataset generated from real legal documents in the Australian Open Legal Corpus. Here sentences extracted from documents, each coupled with a prompt, are used to generate corresponding questions and answers from OpenAI's GPT-4 model. We performed a multi-stage analysis to ensure that this dataset was appropriate for our retrieval and follow-on answer generation tasks.

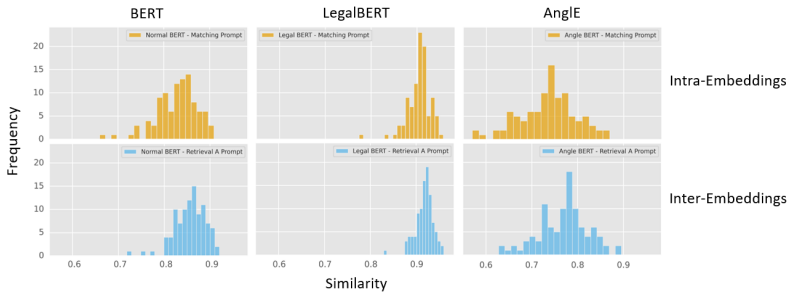

Fig. 4: Cosine similarity distribution for intra- and inter-embeddings.

Dataset Validation involved a randomly sampled manual analysis of the questions and answers performed by the research team to ensure the dataset contained no LLM-based anomalies (i.e. hallucination, factually incorrect statements, etc). We then converted the question text into intra- and inter-embeddings using BERT, LegalBERT and AnglEBERT and examined the similarity distribution by calculating the cosine similarity between each instance and its nearest neighbour (shown in Figure 4). We observe that the similarity distributions are mostly Gaussian, with the most frequent values between 0.7 and 0.8 cosine similarity for both embedding types produced by BERT and AnglEBERT. Embeddings learned by LegalBERT seem to be more densely clustered, as indicated by the higher similarity. This could suggest a reduced ability to discriminate based on similarity compared to BERT and AnglEBERT. We believe the results of this analysis were very promising, as they suggest the embeddings avoid the

traditional issues associated with feature engineering approaches (such as sparse similarity distributions).

Case-Base consisted of the ALQA dataset where each case consists of the full Q&A content as discussed in Section 3.1. While originally containing 2,124 question-support-answer triplets, 40 were removed due to offensive content, and therefore our case-base contains 2,084 cases. The case representation was also expanded to include entities to form the complete tuple as discussed in Section 4.

Table 3: Prompts used in this research.

Scenarios & Generator	**Prompt**
Extract legal acts to pair cases for test set creation. **Gpt-3.5-turbo-0125**	Extract the legal act(s) in this text. Print 'None' if nothing is found. { TEXT }
Extract entities for case representation. **Gpt-3.5-turbo-0125**	Extract named entities and unique identifiers as a single text (separated with white-space) line from this " + TEXT. Print '' if nothing is found. { TEXT }
Generate Qs from text pairs for test set creation. **Mistral-7B**	Produce a question and answer where the answer requires detailed access to both Text 1 and Text 2. Don't refer texts in the question or answer. { Text1: TEXT1 — Text2: TEXT }
Generate answers from snippets for retrieval analysis. **Mistral-7B**	Answer QUESTION by using the following contexts: { TEXT1 — TEXT2 }
Generating answers from cases for retrieval analysis. **Mistral-7B**	Answer QUESTION as a simple string (with no structure) by using the following question, citation, and answer tuples as context: { Question: Q1, Citation: C1, Answer: A1 — Question: Q2, Citation: C2, Answer: A2 }

Test Set Creation focused on creating a discrete test set of questions that reference applicable knowledge in the case-base, without directly mapping to a single case in the case-base. To guide the generation of unique questions for test purposes, we first analysed the case-base in terms of unique legal acts mentioned in all cases. We then selected case pairs based on the common acts and, using Mistral-7B [12] generated 35 new question-answer pairs, each answerable using the combined information from both cases in a pair, as detailed in the prompt presented in Table 3. The rationale is that by pairing cases with common legal acts, we can encourage Mistral to create novel test Q&A pairs. These pairs are unique in that they necessitate synthesising information from both cases in the pair to form a coherent question and corresponding answer, ensuring that the resulting pairs differ from the questions and answers of the individual cases in the case-base, thereby creating a set of new test cases that are reasonably disjoint.

All outputs were manually reviewed to ensure they were suitable test cases[1]. After pruning cases that lacked suitable answers or contained answers merely linked to the case pair by conjunctions, we were left with a total of 32 cases.

6.2 Retrieval Analysis

We first evaluated the quality of case retrieval, by exploiting the fact that each of our 32 test cases had originated from a pair of parent cases. Accordingly, for each test case, we treated the pair of parent cases from the training set as relevant, and all other cases as irrelevant (allowing calculation of ranked precision and recall). We then performed a similarity-based retrieval using the k-Nearest Neighbors algorithm, exploring a range of k values consisting of prime numbers between 1 and 37 We calculated results using F1-score for retrieval@k and visualised using a heat-map (see Figure 5). Here the best performing algorithm was Hybrid AnglEBERT with [0.25, 0.40, 0.35] weights and $k = 3$. Accordingly, k was selected as 3 to be used in the subsequent generative experiments with $k = 1$ as a comparative baseline.

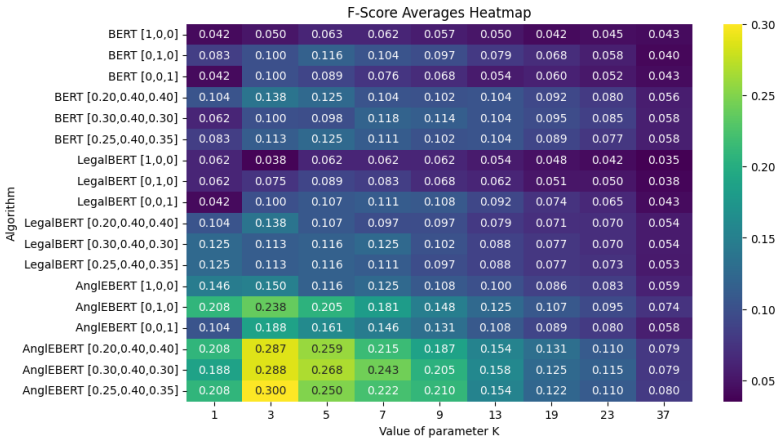

Fig. 5: F1 score for Retrieval@k

6.3 Generation Results

We evaluated the quality of generated output using six algorithms run with $k = 1$ and $k = 3$, along with the baseline for No-RAG. The results are presented in Table 4.

Next, we evaluated the quality of generated output. Six algorithms were run with $k = 1$ and $k = 3$, along with the baseline for No-RAG.

[1] Test dataset available at open-australian-legal-qa-test

Table 4: Cosine scores for hybrid algorithms

		No Context	*Support*	*Full Case*
$k = 0$	No-RAG	0.8967		
$k = 1$	Hybrid BERT	-	0.8986	**0.9068**
	Hybrid LegalBERT	-	0.9020	0.9043
	Hybrid AnglEBERT	-	0.9121	0.9074
$k = 3$	Hybrid BERT		0.9007	0.8998
	Hybrid LegalBERT	-	0.9034	**0.9045**
	Hybrid AnglEBERT	-	0.9092	***0.9141**

The Mistral-7B-open model was utilised as the LLM to generate the answers given a question from a test case with the RAG context (formed using the CBR-RAG retrieval approaches). The generated content was then converted to an embedding using Mistral for cosine-based comparison with the expected answer from the test case (i.e. reference text) for comparison, with the expectation that higher the similarity the better the CBR-RAG setup.

Relatively high cosine scores were observed with the No-RAG baseline due to the generator having parametric memory in most of the legal questions asked by default. The best semantic similarity was noted with the Hybrid AnglEBERT variant when 3 nearest neighbours were fed into the generator in the form of full cases forming context for RAG. It provided answers on average with 1.94% increase in performance. All hybrid variants performed better than the No-RAG baseline. We also observed that including the full case in the prompt provided better results compared to including only the Support text in most hybrid algorithms. Overall, Hybrid AnglEBERT outperforms the BERT and LegalBERT variants with higher semantic similarity observed when $k = 3$.

We performed a series of ANOVA tests to evaluate whether results were significant. Following this we carried out paired tests between the best-performing methods from each of the BERT and LegalBERT groups (in bold), as well as the baseline No-RAG. Here we found that hybrid AnglEBERT k=3, significantly outperforms (asterik) both 'No-RAG' and hybrid LegalBERT k=3, at the 95% confidence level, as shown by a one-tailed T-test. Against hybrid BERT k=1, AnglEBERT k=3 shows significant improvement at the 90% confidence level.

7 Conclusions

In this paper we have presented CBR-RAG, improving LLM output by augmenting input with supporting information from a case-base of previous examples. We have performed an empirical evaluation of different retrieval methods with knowledge representation and comparison using BERT, LegalBERT, and AnglEBERT embeddings. Responses generated by CBR-RAG outperforms those of baseline models in similarity to ground truth. Our findings confirm that using a

case-retrieval approach in RAG systems lead to clear performance benefits, but selecting an appropriate embedding for case representation is key. A qualitative analysis with a domain expert would be an ideal next step to validate these results. The fact that AnglEBERT had the best performance suggests that its contrastive approach to optimising for embeddings (based on similarity comparisons) remains more important than the standard self-supervised masked training strategies used by LegalBERT, even if trained on general legal data. We note that none of the embedding methods, including LegalBERT (trained on broader legal collections), were fine-tuned to the ALQA-specific legal corpus. We are keen to explore the impact of fine-tuning using contrastive self-supervision methods and determine the necessary data supervision burdens for this process, which could pose disadvantages in certain domains.

In future work, we aim to explore further opportunities within representation and specifically alternative methods for text embeddings. Moreover, given the hybrid embeddings' success, which combines multiple representations for fine-grained similarity comparison, we are keen to expand CBR-RAG with more retrieval capabilities. Finally, we found that combining multiple neighbours while maintaining a coherent prompt is challenging, so we plan to explore case aggregation strategies in the future.

References

1. Aleven, V., Ashley, K.D.: Teaching case-based argumentation through a model and examples: Empirical evaluation of an intelligent learning environment. In: Artificial intelligence in education. vol. 39, pp. 87–94. Citeseer (1997)
2. Asai, A., Wu, Z., Wang, Y., Sil, A., Hajishirzi, H.: Self-rag: Learning to retrieve, generate, and critique through self-reflection. arXiv preprint arXiv:2310.11511 (2023)
3. Ashley, K.D.: Reasoning with cases and hypotheticals in hypo. International journal of man-machine studies 34(6), 753–796 (1991)
4. Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., Shah, R.: Signature verification using a" siamese" time delay neural network. Advances in neural information processing systems 6 (1993)
5. Brüninghaus, S., Ashley, K.D.: The role of information extraction for textual cbr. In: International Conference on Case-Based Reasoning. pp. 74–89. Springer (2001)
6. Butler, U.: Open australian legal corpus (2024), https://huggingface.co/datasets/umarbutler/open-australian-legal-corpus
7. Chalkidis, I., Fergadiotis, M., Malakasiotis, P., Aletras, N., Androutsopoulos, I.: Legal-bert: The muppets straight out of law school. arXiv preprint arXiv:2010.02559 (2020)
8. Chalkidis, I., Jana, A., Hartung, D., Bommarito, M., Androutsopoulos, I., Katz, D.M., Aletras, N.: Lexglue: A benchmark dataset for legal language understanding in english (2022)
9. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)

10. Guha, N., Nyarko, J., Ho, D.E., Ré, C., Chilton, A., Narayana, A., Chohlas-Wood, A., Peters, A., Waldon, B., Rockmore, D.N., Zambrano, D., Talisman, D., Hoque, E., Surani, F., Fagan, F., Sarfaty, G., Dickinson, G.M., Porat, H., Hegland, J., Wu, J., Nudell, J., Niklaus, J., Nay, J., Choi, J.H., Tobia, K., Hagan, M., Ma, M., Livermore, M., Rasumov-Rahe, N., Holzenberger, N., Kolt, N., Henderson, P., Rehaag, S., Goel, S., Gao, S., Williams, S., Gandhi, S., Zur, T., Iyer, V., Li, Z.: Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models (2023)
11. Hacker, P., Engel, A., Mauer, M.: Regulating chatgpt and other large generative ai models. In: Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency. pp. 1112–1123 (2023)
12. Jiang, A.Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D.S., Casas, D.d.l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., et al.: Mistral 7b. arXiv preprint arXiv:2310.06825 (2023)
13. Lai, J., Gan, W., Wu, J., Qi, Z., Yu, P.S.: LLMs in law: A survey (2023)
14. Lee, J.S.: Lexgpt 0.1: pre-trained gpt-j models with pile of law (2023)
15. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T., et al.: Retrieval-augmented generation for knowledge-intensive nlp tasks. Advances in Neural Information Processing Systems 33, 9459–9474 (2020)
16. Li, X., Li, J.: Angle-optimized text embeddings (2023)
17. Rissland, E.L., Daniels, J.J.: A hybrid cbr-ir approach to legal information retrieval. In: Proceedings of the 5th international conference on Artificial intelligence and law. pp. 52–61 (1995)
18. Tang, C., Liu, Z., Ma, C., Wu, Z., Li, Y., Liu, W., Zhu, D., Li, Q., Li, X., Liu, T., Fan, L.: Policygpt: Automated analysis of privacy policies with large language models (2023)
19. Thulke, D., Daheim, N., Dugast, C., Ney, H.: Efficient retrieval augmented generation from unstructured knowledge for task-oriented dialog. arXiv preprint arXiv:2102.04643 (2021)
20. Tuggener, D., von Däniken, P., Peetz, T., Cieliebak, M.: LEDGAR: A large-scale multi-label corpus for text classification of legal provisions in contracts. In: Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S. (eds.) Proceedings of the Twelfth Language Resources and Evaluation Conference. pp. 1235–1241. European Language Resources Association, Marseille, France (May 2020), https://aclanthology.org/2020.lrec-1.155
21. Tuggener, D., von Däniken, P., Peetz, T., Cieliebak, M.: LEDGAR: A large-scale multi-label corpus for text classification of legal provisions in contracts. In: Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S. (eds.) Proceedings of the Twelfth Language Resources and Evaluation Conference. pp. 1235–1241. European Language Resources Association, Marseille, France (May 2020), https://aclanthology.org/2020.lrec-1.155
22. Upadhyay, A., Massie, S.: A case-based approach for content planning in data-to-text generation. In: International Conference on Case-Based Reasoning. pp. 380–394. Springer (2022)
23. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems 30 (2017)

24. Wiratunga, N., Koychev, I., Massie, S.: Feature selection and generalisation for retrieval of textual cases. In: European Conference on Case-Based Reasoning. pp. 806–820. Springer (2004)