Interpretable Long-Form Legal Question Answering with Retrieval-Augmented Large Language Models

Antoine Louis, Gijs van Dijck, Gerasimos Spanakis
Law & Tech Lab, Maastricht University
{a.louis, gijs.vandijck, jerry.spanakis}@maastrichtuniversity.nl

Abstract
Many individuals are likely to face a legal dispute at some point in their lives, but their lack of understanding of how to navigate these complex issues often renders them vulnerable. The advancement of natural language processing opens new avenues for bridging this legal literacy gap through the development of automated legal aid systems. However, existing legal question answering (LQA) approaches often suffer from a narrow scope, being either confined to specific legal domains or limited to brief, uninformative responses. In this work, we propose an end-to-end methodology designed to generate long-form answers to any statutory law questions, utilizing a “retrieve-then-read” pipeline. To support this approach, we introduce and release the Long-form Legal Question Answering (LLeQA) dataset, comprising 1,868 expert-annotated legal questions in the French language, complete with detailed answers rooted in pertinent legal provisions. Our experimental results demonstrate promising performance on automatic evaluation metrics, but a qualitative analysis uncovers areas for refinement. As one of the only comprehensive, expert-annotated long-form LQA dataset, LLeQA has the potential to not only accelerate research towards resolving a significant real-world issue, but also act as a rigorous benchmark for evaluating NLP models in specialized domains. We publicly release our code, data, and models.

Introduction
Legal disputes are an inevitable part of everyday life, with many individuals finding themselves entangled in issues related to marriage, debts, or employment (Farrow et al. 2016; Ponce et al. 2019). However, most people have little to no knowledge about their rights and fundamental legal processes (Balmer et al. 2010). As a result, they either take no action or turn to the internet for advice (Denvir 2016). Unfortunately, the latter often directs users towards commercial websites that prioritize their own marketing efforts over providing thorough, useful legal guidance (Hagan and Li 2020). While invaluable, expert legal assistance is often prohibitively expensive, which results in a considerable number of vulnerable individuals being left unprotected or exploited due to their inability to afford it. This barrier to accessing legal information fosters a significant imbalance within the legal system, impeding the universal right to equal access to justice for all. The global implications of this issue are significant: an estimated 1.5 billion individuals wrestle with unresolved legal challenges, and 4.5 billion may be excluded from the protective measures that the law provides (Garanvano et al. 2019). In light of these circumstances, there is growing consensus that improved access to legal information could dramatically enhance the outcomes of legal disputes for many people (Currie 2009).

The rapid progress in natural language processing and the growing availability of digitized legal data present unprecedented opportunities to bridge the gap between people and the law. For instance, legal text summarization (Bhat-tacharya et al. 2019; Shukla et al. 2022) holds the potential to simplify complex legal documents for the layperson, while legal judgment prediction (Chalkidis et al. 2019; Trautmann et al. 2022) could unveil insightful correlations between an individual’s situation and the probable legal outcome. Similarly, legal question answering (LQA) could offer affordable, expert-like assistance to the masses, thereby empowering marginalized parties when utilized for public welfare. However, existing research on LQA tends to exhibit a constrained scope, often concentrating on specialized legal domains, such as tax law (Holzenberger et al. 2020) or privacy policies (Ravichander et al. 2019), or limiting the responses to uninformative brief answers like yes/no replies (Rabelo et al. 2022) or few-word spans (Duun et al. 2019).

In this paper, we present an end-to-end approach aimed at generating long-form responses to any statutory law questions. Our methodology harnesses the popular “retrieve-then-read” pipeline, which first leverages a retriever over a large evidence corpus to fetch a set of relevant legislative articles, and then employs a reader to peruse these articles and formulate a comprehensive, interpretable answer. Our retriever relies on a lightweight bi-encoder model, which enables fast and effective retrieval. For our reader, we use an instruction-tuned large language model (LLM) that we adapt to our task via two distinct learning strategies: in-context learning, wherein the model learns from instructions and a set of contextually provided examples; and parameter-efficient finetuning, where a small number of extra parameters are optimized on a downstream dataset while the base model’s weights are quantized and remain unchanged.

To support training and evaluating such systems, we collect and release the Long-form Legal Question Answering
(LLeQA) dataset. LLeQA builds upon BSARD (Louis and Spanakis 2022), an information retrieval dataset in French comprising 1,108 legal questions labeled with relevant provisions from a corpus of 22,633 Belgian law articles, and enhance it in two ways. First, we introduce 760 new legal questions (+69%) and 5,308 additional statutory articles (+23%). Second, we supplement the data with new types of annotations, including an exhaustive taxonomy for the question, the jurisdictions concerned, the exact paragraph-level references within the relevant articles, and a comprehensive answer written by seasoned legal professionals. Owing to the rich variety of its annotations, LLeQA serves as a multifaceted resource that extends its utility beyond legal question answering and has the potential to catalyze significant progress in various legal tasks, such as legal inquiry classification, legal topic modeling, and legal information retrieval.

Our experimental results show that retrieval-augmented LLMs exhibit commendable performance on automatic evaluation metrics, measuring alignment with target answers. Yet, a deeper qualitative analysis reveals that these syntactically correct responses, despite seemingly covering the intended topics, frequently harbor inaccuracies and erroneous information. This discrepancy underscores the limitations inherent in relying solely on automatic metrics for assessing such systems, and indicates substantial room for improvement both in terms of modeling and evaluation.

In summary, our main contributions are:

1. A novel dataset for long-form question answering (LFQA) in the legal domain and French language, comprising 1,868 legal questions, meticulously annotated by legal professionals, with detailed answers and references to relevant legal provisions, drawn from a substantial knowledge corpus containing 27,942 statutory articles.

2. A comprehensive evaluation of the retrieve-then-read framework in the context of legal LFQA, while emphasizing interpretability and exploring various learning strategies for the reader.

3. A public release of our code, dataset, and checkpoints to facilitate future research on interpretable legal LFQA.¹

Related Work

Legal question answering. Addressing legal questions has long posed intricate challenges within the legal NLP community, stemming from the inherent complexities of legal texts, including specialized terminology, complex structure, and nuanced temporal and logical connections. To stimulate advancement in this field, an array of datasets and benchmarks has emerged. Duan et al. (2019) craft a judicial reading comprehension dataset in the Chinese language, aimed at fostering the development of systems capable of mining fine-grained elements from judgment documents. Ravichander et al. (2019) present a corpus of questions about privacy policies of mobile applications with the objective of empowering users to comprehend and selectively investigate privacy matters. Holzenberger et al. (2020) introduce a dataset for statutory reasoning in tax law. Zhong et al. (2020) present a multi-choice question answering dataset designed to assess professional legal expertise. Rabelo et al. (2022) hold a competition wherein a task consists in answering “yes” or “no” given a legal bar exam problem related to Japanese civil law. Lastly, both Mansouri and Campos (2023) and Chen et al. (2023a) offer a corpus featuring question-answer pairs in English and Chinese, respectively, sourced from online law-oriented forums.

Knowledge-grounded question answering. Mainstream approaches to tackling knowledge-intensive QA tasks commonly rely on external knowledge sources to enhance the answer prediction, such as collected documents (Voorhees 1999), web-pages (Kwok et al. 2001), or structured knowledge bases (Berant et al. 2013; Yu et al. 2017). These open-book models (Roberts et al. 2020) typically index the knowledge corpus before employing a retrieve-then-read pipeline to predict a response based on multiple supporting documents (Chen et al. 2017). To an extent, this paradigm can be likened to query-based multi-document summarization (Tombros and Sanderson 1998), where the objective lies in providing users with a succinct and precise overview of the top-ranked documents related to their queries. Query-driven summarization may adopt different methodologies, manifesting either in an extractive form, where specific portions of evidence text are selected (Otterbacher et al. 2009; Litvak and Vanetik 2017), or in an abstractive form, where the information is synthesized into new expressions (Nema et al. 2017; Baumel et al. 2018; Ishigaki et al. 2020).

Rationale generation. To gain insights into model predictions (Lei et al. 2016), recent advancements have explored the generation of abstractive textual explanations in areas such as commonsense reasoning (Rajani et al. 2019) and natural language inference (Kumar and Talukdar 2020). Alternatively, Lahoti et al. (2021) proposed the extractive generation of predefined evidence markers instead of decoding raw explanations. Complementing generation, studies have concentrated on extracting rationales from evidence input segments (Bastings et al. 2019; Chalkidis et al. 2021), as well as analyzing saliency maps to underscore key input tokens instrumental to each prediction (Ribeiro et al. 2016).

The LLeQA Dataset

Dataset Construction

In this section, we describe our process to create LLeQA, which involves three main stages. First, we gather and refine annotated legal questions. Then, we build an expansive corpus of supportive statutory articles drawn from Belgian legislation. Finally, we enrich the question annotations by generating paragraph-level references within relevant articles. We elaborate upon each of these steps below.

Collecting question-answer pairs. The data construction process starts with collecting high-quality question-answer pairs on a legal matter. Echoing Louis and Spanakis (2022), we partner with Droits Quotidiens, a Belgian non-profit organization that endeavors to make the law comprehensible and accessible to the most vulnerable. To this end, the organization maintains a rich website featuring thousands of

¹https://github.com/maastrichtlawtech/lleqa
legal questions commonly posed by Belgian citizens. Each question comes with its own individual page, encompassing one or more categorizations, references to relevant legislative statutes, and a detailed answer written in layman’s terms by experienced jurists. With their help, we collect approximately 2,550 legal questions. We then filter out questions that are unsuitable for retrieval-based question answering. Specifically, we discard questions whose references are either missing, too vague (e.g., an entire law or book), or from a statute not collected in our knowledge corpus. Additionally, we group duplicate questions found across different subcategories on the website. This yields a final number of 1,868 question-answer pairs, each with legal references.

Mining supporting information. Next, we build the knowledge corpus of statutory articles used to provide evidence that a system can draw upon when generating an answer. We start by extracting provisions from all publicly available Belgian codes of law via the official government website, forming an exhaustive foundation of 23,759 articles across 35 legal codes, thereby encapsulating a wide range of legal subjects. To enhance the corpus, we then incorporate additional laws, decrees, and ordinances that are frequently cited as supportive references but absent from the initial collection, adding 4,183 articles from 34 legal acts. This brings the final evidence corpus to 27,942 articles. We proceed by assigning a unique identifier to each article and employ regular expressions to match the plain text legal references linked to the questions with their corresponding article in our corpus. Next, we cleanse the articles of recurrent noisy textual elements, such as nested brackets, superscripts, or footnotes that may be present due to revisions or repeals by new legislation. Besides the article’s content, we also collate the complete legislative path leading up to that article, which starts from the statute’s name and progresses through the name of the book, act, chapter, section, and subsection where the article resides. This auxiliary information provides valuable contextual insight on the article’s subject matter. Lastly, we partition the articles into their constituent paragraphs, which serve as the basic units for rationale extraction.

Generating paragraph-level rationales. Only 10.4% of the collected questions come with paragraph-level references, i.e., with mentions to specific article paragraphs as the relevant information to the question. To extend this level of interpretability across all questions in LLeQA, we leverage a large-scale language model to synthetically generate these paragraph-level relevance signals for the remaining samples. This approach serves as an affordable and efficient alternative to hiring costly legal experts for annotation. Utilizing the closed-source gpt-3.5-turbo-0613 model via the OpenAI API, we present the model with all paragraphs from relevant legal articles for each question-answer pair, and instruct it to identify those contributing to the answer. The model responds with a comma-separated list of identifiers corresponding to the deemed relevant paragraphs, or “None” if it discerns no contribution. After parsing the model’s responses, we incorporate these synthetically generated paragraph-level rationales into the dataset.

Finalizing the dataset. We assign the questions with expert-annotated paragraph-level references to the test set and randomly partition the remaining samples into training and development sets, yielding a 10/90/10 split, respectively.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Average # of words Ques.</th>
<th>Evid.</th>
<th>Ans.</th>
<th># Pairs</th>
<th>Answer type</th>
<th>Domain</th>
<th>Source</th>
<th>Lang.</th>
</tr>
</thead>
<tbody>
<tr>
<td>JEC-QA (Zhong et al. 2020)</td>
<td>47</td>
<td>58</td>
<td>15</td>
<td>26,365</td>
<td>Multi-choice</td>
<td>Statutory law</td>
<td>Law exam</td>
<td>zh</td>
</tr>
<tr>
<td>SARA (Holzenberger et al. 2020)</td>
<td>46</td>
<td>489</td>
<td>1</td>
<td>376</td>
<td>Binary, numeric</td>
<td>Tax law</td>
<td>Jurists</td>
<td>en</td>
</tr>
<tr>
<td>PrivacyQA (Ravichander et al. 2019)</td>
<td>8</td>
<td>3,237</td>
<td>140</td>
<td>1,750</td>
<td>Multi-span</td>
<td>Privacy policy</td>
<td>Jurists</td>
<td>en</td>
</tr>
<tr>
<td>CRJC (Duan et al. 2019)</td>
<td>unk.</td>
<td>unk.</td>
<td>unk.</td>
<td>51,333</td>
<td>Binary, span</td>
<td>Case law</td>
<td>Jurists</td>
<td>zh</td>
</tr>
<tr>
<td>FALQU (Mansouri et al. 2023)</td>
<td>144</td>
<td>-</td>
<td>244</td>
<td>9,880</td>
<td>Long-form</td>
<td>Statutory law</td>
<td>Web forum</td>
<td>en</td>
</tr>
<tr>
<td>COLIEE-21 (Rabelo et al. 2022)</td>
<td>41</td>
<td>94</td>
<td>1</td>
<td>887</td>
<td>Binary</td>
<td>Civil law</td>
<td>Law exam</td>
<td>ja, en</td>
</tr>
<tr>
<td>EQUALS (Chen et al. 2023a)</td>
<td>32</td>
<td>252</td>
<td>69</td>
<td>6,914</td>
<td>Long-form</td>
<td>Statutory law</td>
<td>Web forum</td>
<td>zh</td>
</tr>
<tr>
<td>LLeQA (ours)</td>
<td>15</td>
<td>1,857</td>
<td>264</td>
<td>1,868</td>
<td>Long-form</td>
<td>Statutory law</td>
<td>Jurists</td>
<td>fr</td>
</tr>
</tbody>
</table>

Table 1: Comparison of public legal question answering (LQA) datasets. LLeQA has answers an order of magnitude longer and is the only expert-annotated long-form LQA dataset to cover any statutory law subjects.
in Table 2. “Can”, “how”, “what”, and “must” are the most frequently used question words, indicating that people primarily seek information on legal permissions, procedures, definitions, and obligations. These distributions reflect the variety of legal concerns that citizens face, thereby providing valuable guidance for user-centric LQA systems.

**Question evidence.** In LLeQA, approximately 80% of the questions associate with fewer than five articles from the knowledge corpus, with the median number of relevant articles per question being two. These articles have a median length of 84 words, yet 1,515 articles exceed 500 words, with the longest reaching several thousand words. When combining all relevant articles for each question, we find an average evidence length of 1,857 words per question, positioning LLeQA at the upper range among published datasets regarding evidence length. Interestingly, a mere 8% of the articles within our corpus are referenced as relevant to at least one question within the dataset. Moreover, nearly half of these referenced articles originate from five statutes only, implying the critical role a select few laws play in answering the most frequently posed legal inquiries.

**Assessment of annotation quality.** To assess the quality of the synthetically generated paragraph-level rationales, we evaluate the performance of gpt-3.5-turbo-0613 against the ground truth annotations from the test set. Although far from expert quality, we find that the model demonstrates decent annotation performance, achieving a F1 score of 47.5%. By comparison, a naive baseline that randomly selects the relevant paragraphs achieves 15.3% F1, whereas one that always marks the first paragraph as the relevant one scores 27.2% F1. After experimenting with alternative LLMs, we observe that, as of the time of writing, gpt-3.5-turbo-0613 achieves the best overall performance within a limited cost budget. Despite the apparent margin of error, we believe that these imperfect synthetic annotations may still be beneficial for in-context learning purposes as ground truth labels bear less significance in such settings (Min et al. 2022).

**Method**

In this section, we detail the “retrieve-then-read” framework we use for interpretable long-form legal question answering, illustrated in Figure 1. First, a retriever selects a small subset of statutory articles, some of which being relevant to the question. Then, a generator conditions its answer on the subset of articles returned by the retriever. We describe these two components in more detail below.

**Retriever**

The role of our retrieval component is to pinpoint all statutory articles relevant to a question and present them at the forefront of the returned results. More formally, the retriever can be expressed as a function $R : (q, C) \mapsto F$ that takes as input a question $q$ along with a knowledge corpus of legal provisions $C = \{p_1, p_2, \ldots, p_N\}$, and returns a notably smaller filtered set $F \subset C$ of the supposedly relevant provisions, ordered by decreasing relevance.

We employ the widely adopted bi-encoder architecture (Bromley et al. 1993) as the foundation of our retriever, which maps a question $q$ and a legal provision $p$ into dense vector representations and computes a relevance score $s : (q, p) \mapsto \mathbb{R}_+$ between the two by the similarity of their embeddings $h_q, h_p \in \mathbb{R}^d$, i.e.,

$$s(q, p) = \text{sim}(h_q, h_p),$$

where $\text{sim} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a similarity function, such as the dot product or cosine similarity. These embeddings are typically derived from a pooling operation on the output representations of a pretrained autoencoding language model, such as BERT (Devlin et al. 2019), so that

$$h_q = \text{pool}(E(q; \theta_1)), \quad h_p = \text{pool}(E(p; \theta_2)), \quad (2)$$

where the model $E(\cdot; \theta) : \mathcal{V}^n \to \mathbb{R}^{n \times d}$ with weights $\theta$ transforms an input text sequence of $n$ terms from vocabulary $\mathcal{V}$ to $d$-dimensional real-valued word vectors. The pooling function $\text{pool} : \mathbb{R}^{n \times d} \to \mathbb{R}^d$ implements a mean or max operation on the output word embeddings to distill a global representation for the text passage. Beyond the conventional two-tower bi-encoder (Karpuhkin et al. 2020; Yang et al. 2020), which employs two independent encoder models to map queries and articles separately into distinct embedding spaces, the siamese variant (Reimers and Gurevych 2019; Xiong et al. 2021) uses a unique encoder, i.e., $\theta_1 = \theta_2$, to encode the question and article in a shared dense vector space. We use the latter variant in this work.
The training objective for our dense retriever is to learn a high-quality low-dimensional embedding space for questions and legal provisions such that relevant question-provision pairs appear closer to each other than irrelevant ones. Assume our training data \( D = \{ (q_i, p^+_i) \}_{i=1}^{N} \) contains \( N \) instances, each comprising a question \( q_i \), linked to a relevant provision \( p^+_i \). For each question \( q_i \), we sample a set of irrelevant provisions \( P^-_i \), thereby constituting a training set \( T = \{ (q_i, p^+_i, P^-_i) \}_{i=1}^{N} \). Subsequently, we use the instances in \( T \) to contrastively optimize the negative log-likelihood of the relevant provision against the non-relevant ones, i.e.,

\[
\mathcal{L}_0 \left( q_i, p^+_i, P^-_i \right) = - \log \frac{e^{s(q_i,p^+_i)/\tau}}{\sum_{p \in P^-_i \cup \{ p^+_i \}} e^{s(q_i,p)/\tau}},
\]

where \( \tau > 0 \) is a temperature parameter that we set to 0.01.

To select irrelevant provisions, we employ two distinct sampling strategies: random sampling using in-batch negatives (Chen et al. 2017b; Henderson et al. 2017), which considers provisions paired with the other questions within the same mini-batch; and hard negative sampling using BM25 (Robertson et al. 1994), which includes the top provisions returned by BM25 that bear no relevance to the question.

**Generator**

Our generator aims at formulating a comprehensive answer to a short legal question, leaning on corroborative data. Formally, the task can be cast as a conditional text generation problem, where the model requires conditioning its response on a context string that incorporates the question and several supporting statutory articles. We turn to autoregressive large language models (LLMs) based on the Transformer’s decoder block as the backbone architecture for our generator. We then delve into two learning scenarios, specifically in-context learning (Radford et al. 2019) and parameter-efficient finetuning (Lester et al. 2021; Liu et al. 2022).

### In-context learning

To assess the innate performance of our generator without additional training, we start by examining three prevalent in-context learning strategies, namely zero-shot, one-shot, and few-shot learning. Under the zero-shot learning paradigm, the generator receives a natural language instruction and seeks to answer the question directly. The context \( c \) provided to the model is formulated as

\[
c = [d, \{ P^+_i \}, q],
\]

where \( d \) stands for the task description, \( q \) is the test question, and \( P^+_i \) represents the top-\( k \) most pertinent legal provisions to \( q \) as identified by the retriever (\( k \) is set to 5). In one-shot learning, the generator benefits from an additional demonstration to guide its understanding of the task, whereas in a few-shot setting, the model accommodates as many demonstrations as can be included within its context window. Under these scenarios, the context string becomes

\[
c = [d, \{ P^+_j, q, a_j \}_{j=1}^{n}, P^+_i, q],
\]

where \( a_j \) is the gold long-form answer to the training question \( q_j \), and \( n \) denotes the number of demonstrations. We dynamically select demonstrations in the training pool through a similarity-based retrieval method based on the question.

### Parameter-efficient finetuning

Due to the gigantic size of contemporary LLMs, performing full finetuning of all model parameters would be prohibitively expensive. Instead, we employ parameter-efficient finetuning to train our generator, which significantly trims both the training duration and computational cost. Specifically, we apply the QLoR technique (Detmers et al. 2023), which undertakes a preliminary quantization of the pretrained model to 4-bit before freezing all its parameters for training. A small set of learnable low-rank adapter weights (Hu et al. 2022) are then injected into each linear layer of the Transformer and tuned by

\[
\begin{align*}
\text{Top-k Results} & \quad \cdots \\
2 & \quad \cdots \\
3 & \quad \cdots \\
4 & \quad \cdots \\
5 & \quad \cdots \\
\end{align*}
\]
backpropagating gradients through the quantized weights. Formally, given a training sample \((q_i, P_i^+, a_i)\) where \(a_i = (y_1, \cdots, y_T)\) represents the target output sequence, we optimize the adapter’s parameters \(\phi\) using a standard language modeling objective that maximizes the negative log likelihood of generating the target answer \(a_i\) conditioned on the input context, encompassing the source question \(q_i\) and the set of relevant legal provisions \(P_i^+\), i.e.,
\[
\mathcal{L}_\phi (q_i, P_i^+, a_i) = -\log p_\phi (a_i | P_i^+, q_i)
\]
\[
= -\log \prod_{t=1}^{T} p_\phi (y_t | P_i^+, q_i, y_{<t}).
\]

**Context window extension.** LLMs typically come with a predefined context window limit, beyond which perplexity steeply rises due to the weak extrapolation properties of positional encoding. This limitation poses significant challenges for applications requiring the processing of extensive inputs, like ours. Recent efforts have aimed to extend the context window sizes of pretrained LLMs employing rotary position embedding (Su et al. 2021, RoPE), by interpolating positional encoding (Chen et al. 2023b). Guided by promising findings from the open-source community, we perform dynamic NTK-aware scaling,\(^3\) which retains the exact position values within the original context window and progressively downscales the input position indices using a nonlinear interpolation from neural tangent kernel theory (Jacot et al. 2018). Preliminary results suggest this approach substantially mitigates perplexity degradation for sequences exceeding the maximum window size, without necessitating additional finetuning.

**Rationales extraction.** Given the serious implications of flawed legal guidance, ensuring interpretability in generated answers is crucial. This enables users to cross-verify responses through reliable sources while understanding the underlying reasoning, thereby enhancing the trustworthiness of LQA systems. To this end, we impose additional constraints on the model to furnish proper justification for its answers. While prior work on rationale generation has predominantly focused on creating free-form natural language explanations (Latcinnik and Berant 2020; Narang et al. 2020), abstractive models have shown a propensity for fabricating convincing yet misleading justifications, inadvertently supporting inaccurate predictions (Camburu et al. 2020; Wiegreffe et al. 2021). Besides, adapting this strategy for applications involving multiple extensive evidence documents proves challenging. Consequently, we adopt an extractive rationale generation strategy (Lakhotia et al. 2021), prompting the model to generate evidence paragraph markers rather than raw explanations. This technique ensures the production of unaltered rationales that are easily interpretable.

**Experiments**

**Experimental Setup**

**Models.** Our dense retrieval model leverages CamemBERT (Martin et al. 2020), a leading autoencoding model for the French language. To furnish a well-rounded comparative analysis, we incorporate two robust retrieval baselines: BM25 (Robertson et al. 1994), a widely utilized bag-of-words retrieval function; and mE5 (Wang et al. 2022a), currently the top-performing multilingual dense model on the MTEB benchmark (Muennighoff et al. 2023). With regard to our generator, we experiment with several instruction-tuned open-source models derived from LLaMA (Touvron et al. 2023) to harness the benefits of dynamic NTK-aware scaled RoPE. Specifically, we consider four models that are notably high-ranking on the MT-Bench leaderboard (Zheng et al. 2023): Vicuna-1.3 (Chiang et al. 2023), WizardLM-1.0 (Xu et al. 2023a), TULU (Wang et al. 2023), and Gullanco (Dettmers et al. 2023). Due to limited computational resources, we restrict our study to their 7B variant and curtail the extended context window size to 8192 tokens for in-context learning and 4096 tokens for finetuning.

**Implementation.** We start by full finetuning CamemBERT on LLeQA with a batch size of 32 and a maximum sequence length of 384 tokens for 20 epochs (i.e., 5.8k steps), using AdamW (Loshchilov and Hutter 2017) with \(\beta_1 = 0.9\), weight decay of 0.01, and learning rate warm up along the first 60 steps to a maximum value of 2e-5, after which linear decay is applied. We use 16-bit automatic mixed precision to accelerate training, which takes about 1.7 hours. We then optimize our baseline LLMs through 4-bit QLoRA finetuning with an effective batch size of 8 for 10 epochs (i.e., 1.1K steps) using paged AdamW optimizer with default momentum parameters and constant learning rate schedule of 2e-4. We employ NormalFloat4 with double quantization for the base models and add LoRA adapters on all linear layers by setting \(r = 16, \alpha = 32\) while utilizing float16 as computation datatype. Training takes around 7.5 hours to complete. We generate from the LLMs using nucleus sampling (Holtzman et al. 2020) with \(p = 0.95\) and a temperature of 0.1.

**Hardware & Libraries.** Computations are performed on a single 32GB NVIDIA V100 GPU hosted on a server with a dual 20-core Intel Xeon E5-2698 v4 CPU and 512GB of RAM, operating under Ubuntu 16.04.

**Automatic Evaluation**

**Evaluation metrics.** To evaluate our framework’s effectiveness, we assess three core aspects: retrieval performance, generation quality, and rationales accuracy. Firstly, it is essential that the retriever returns as many pertinent provisions as possible within the first top-\(k\) results, given the generator’s limited context window size. This requirement implies a primary interest in recall at small cutoffs. Additionally, we report the mean reciprocal rank, as it offers valuable insights into the position of the first relevant result. Gauging the quality of long-form answers presents more intricate challenges. Automatic metrics, such as ROUGE (Lin 2004), proved to be inadequate due to the intrinsically open-ended nature of long-form responses, which allows for an array of possible formulations that retain semantic similarity (Wang et al. 2022b; Xu et al. 2023b). Besides, these lexical overlap metrics fail to assess essential aspects such as factual correctness and query relevance. Ultimately, a thorough assessment of

---

\(^3\)https://reddit.com/r/LocalLLaMA/comments/14mrgpr/
such systems requires human evaluation, although the latter introduces its own set of challenges (Krishna et al. 2021), in addition to being expensive and noisy. Owing to resource constraints in our study, we opt for an automated evaluation metrics, fully cognizant of its limitations, and earmark human evaluation for future work. In particular, we report METEOR (Banerjee and Lavie 2005), which demonstrated superior correlation with human judgment in the context of long text generation (Sharma et al. 2017; Chen et al. 2022). Lastly, we evaluate the model’s capacity for rationale extraction by measuring the F1 score over the set of predicted paragraph markers as compared to the ground truth markers.

Results. As depicted in Table 4, our dense retriever, fine-tuned on a mere 1.5k in-domain examples, significantly outperforms robust retrieval baselines, underlining the essential role of domain adaptation in enhancing performance. However, the results leave substantial room for improvement; on average, less than half of the relevant articles appear within the top five returned results. This shortfall represents a major bottleneck for our generator, which faces the challenge of answering questions based on partially irrelevant provisions. The impact of this limitation is palpable in the poor performance of rationale extraction, which approaches near-zero effectiveness for most LLMs, though part of this deficiency can be attributed to the models’ tendency to hallucinate. In terms of answer quality, WizardLM and Vicuna display a high degree of overlap with the ground truth responses, indicating accurate engagement with the subject matter. Providing a demonstration appears to improve generation quality as compared to a zero-shot setup, except for Guanaco which shows strong zero-shot results. However, performance does not seem to vary significantly when more demonstrations are provided. Finally, the results suggest that finetuning on our task-specific dataset consistently enhances performance.

Qualitative Analysis
To discern the strengths and shortcomings of our generators, we conduct a detailed manual analysis of 10 randomly selected samples from the test set. We find that TULU exhibits a propensity for producing concise answers, a tendency likely due to its extensive finetuning on instruction datasets averaging a relatively short completion length of 98.7 words, which also explains its low METEOR score as LLeQA answers are markedly longer. We further observe that Guanaco and WizardLM are prone to repetitiveness in their responses, occasionally echoing identical phrases. While presence and frequency penalties could mitigate this issue, they may prove ineffective in instances addressing specialized topics where the same terminology is intrinsically repeated. Regarding response quality, WizardLM and Vicuna stand out, far exceeding Guanaco, which tends to produce nonsensical or linguistically convoluted sentences. In contrast, WizardLM and Vicuna’s outputs are articulated in impeccable French, displaying a persuasive flair that could potentially mislead an unsuspecting reader. Nevertheless, a deeper probe unveils striking hallucinations (Ji et al. 2023). Despite seemingly addressing the question, many facts, dates, sources, and conditions appear to be fabricated, as if the models leveraged the provided context less for factual accuracy and more as a foundation upon which to weave a convincing fictitious answer.

Conclusion
In this work, we introduce LLeQA, an expert-annotated dataset tailored to facilitate the development of models aimed at generating comprehensive answers to legal questions, while supplying interpretable justifications. We experiment with the “retrieve-then-read” pipeline on LLeQA and explore various state-of-the-art large language models as readers, that we adapt to the task using several learning strategies. We find that this framework tends to produce syntactically correct answers pertinent to the question’s subject matter but occasionally fabricate facts. Overall, we believe LLeQA can serve as a robust foundation for advancements in interpretable, long-form legal question answering, thereby contributing to the democratization of legal access.

Limitations and future work. Despite our efforts to make LQA systems more factually grounded in supporting legal provisions, the framework we propose remains vulnerable to hallucinations in both the constructed answers and associated rationales. Additionally, consistent with prior studies (Krishna, Roy, and Iyyer 2021; Xu et al. 2023b), we observe that conventional automatic metrics may not accurately mirror answer quality, leading to potential misinterpretations. These challenges present great avenues for future work.

Ethical considerations. The premature deployment of LQA systems poses a tangible risk to laypersons, who may uncritically rely on the furnished guidance and inadvertently exacerbate their circumstances. To ensure the responsible development of legal aid technologies, we are committed to limiting the use of our dataset strictly to research purposes.

<table>
<thead>
<tr>
<th>Model</th>
<th>Size</th>
<th>R@5</th>
<th>R@10</th>
<th>MRR@10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baselines</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 BM25</td>
<td></td>
<td>17.4</td>
<td>22.8</td>
<td>22.0</td>
</tr>
<tr>
<td>2 mE5_BASE</td>
<td>278M</td>
<td>15.4</td>
<td>21.7</td>
<td>25.8</td>
</tr>
<tr>
<td>3 mE5_LARGE</td>
<td>560M</td>
<td>16.5</td>
<td>26.7</td>
<td>28.3</td>
</tr>
<tr>
<td>4 CamemBERT</td>
<td>111M</td>
<td>48.6</td>
<td>60.6</td>
<td>60.0</td>
</tr>
</tbody>
</table>

Table 4: Retrieval scores of our dense retriever benchmarked against other strong retrieval baselines on LLeQA dev set.

<table>
<thead>
<tr>
<th>Model</th>
<th>0-shot</th>
<th>1-shot</th>
<th>2-shot</th>
<th>Fine-tuned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question Answering (METEOR)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 Vicuna</td>
<td>11.6</td>
<td>16.2</td>
<td>15.3</td>
<td>19.7</td>
</tr>
<tr>
<td>2 WizardLM</td>
<td>12.3</td>
<td>15.5</td>
<td>16.6</td>
<td>20.4</td>
</tr>
<tr>
<td>3 TULU</td>
<td>2.9</td>
<td>4.6</td>
<td>8.5</td>
<td>12.7</td>
</tr>
<tr>
<td>4 Guanaco</td>
<td>11.2</td>
<td>11.2</td>
<td>11.3</td>
<td>20.1</td>
</tr>
</tbody>
</table>

Table 5: Scores of our baseline LLMs on LLeQA test set.
Acknowledgments

This research is partially supported by the Sector Plan Digital Legal Studies of the Dutch Ministry of Education, Culture, and Science. In addition, this research was made possible, in part, using the Data Science Research Infrastructure (DSRI) hosted at Maastricht University.

References


