JustifIT-QA: Evidence-Grounded Legal Question Answering in Italian

Anonymous ACL submission

Abstract

Grounded legal question answering requires retrieving the relevant statutory provisions, generating legally sound answers, and supporting them with explicit evidence. We introduce *JustifIT-QA*, the first Italian statutory QA dataset with expert-written answers and fine-grained evidence spans, and present *DARv2*, a dense retriever that improves coverage of relevant legal texts. Building on this retrieval layer, we study a judge-based feedback loop in which an LLM critiques and refines generated answers, and we evaluate LLMs as judges for legal answer correctness. Results show that feedback improves expert-rated answer quality when relevant context is retrieved, while evidence extraction benefits from stronger models but remains challenging. We release the dataset, code, and retriever to support future research on grounded and explainable legal QA in Italian.

1 Introduction

Legal question answering requires replies that are not only fluent but also normatively grounded, since even minor inaccuracies may lead to incorrect interpretations of obligations, rights, or sanctions. This is particularly challenging for LLM-based systems: recent work shows that they may produce legally unsupported statements and fabricated authorities, and that long-form generation can remain only partially grounded even when conditioned on retrieved sources (Dahl et al., 2024; Stolfo, 2024). At the same time, grounded legal QA remains under-resourced outside English, and correctness is difficult to assess at scale in open-ended settings (Guha et al., 2023). Recent work on LLM-based judging further suggests both the promise of automatic evaluation and the need to carefully analyze its reliability and biases (Gu et al., 2025; Chen et al., 2024).

This paper addresses these challenges in the Italian statutory domain. First, to the best of our knowledge, we introduce the first Italian dataset with expert-written answers and fine-grained evidence extracted from legislative text, enabling evaluation beyond article-level retrieval. Second, we present a dense retriever designed specifically for legal contexts, which improves the coverage of relevant provisions. Third, we study an iterative feedback loop in which generated answers are refined through evidence-grounded critiques, improving expert-rated correctness using the retrieved context. Finally, we provide a feasibility study of LLM-based judging for legal answers, analyzing agreement with experts and error patterns. To support future research, we will make publicly available the dataset, retrieval model[1], and the code[2].

2 Related Work

Legal NLP has recently seen the release of several benchmarks for retrieval and question answering, most of which focus on identifying relevant provisions and supporting downstream legal decision making. Statutory retrieval datasets such as BSARD (Louis and Spanakis, 2022) provide expert-annotated mappings between natural-language questions and applicable legal articles, while broader benchmark suites such as LexGLUE (Chalkidis et al., 2022) cover a range of legal tasks but are less focused on grounded QA. JuriFindIT introduces statutory article retrieval for the Italian domain and explicitly encodes cross-article references, enabling graph-aware modeling of inter-provision dependencies in retrieval (Noce et al., 2026). More recent resources move toward answer generation grounded in legal sources. LLeQA targets long-form legal question answering with retrieval over legislative material and answer generation conditioned on retrieved passages (Louis et al., 2024). Complementary work in other legal genres emphasizes pinpointing legally salient

[1] justifit/DARv2 - Huggingface [2] JustifIT-QA repository (code & dataset)

spans, as in contract understanding datasets such as CUAD (Hendrycks et al., 2021). Beyond English, datasets such as JEC-QA for judicial exam questions in Chinese highlight retrieval and reasoning over legal materials under different legal systems and question styles (Zhong et al., 2020). Despite this progress, non-English statutory QA remains comparatively under-resourced, and span-level evidence annotations paired with expert-written answers are still uncommon across jurisdictions and languages. A second relevant line of work concerns retrieval-augmented generation, where generation is explicitly conditioned on retrieved documents to reduce reliance on parametric memory (Lewis et al., 2020). In parallel, a rapidly expanding body of research studies the use of large language models as evaluators of open-ended outputs. LLM-as-a-judge methods investigate how rubric-based prompting can approximate human preference judgments and support scalable evaluation (Zheng et al., 2023; Liu et al., 2023). Iterative self-refinement methods further explore how natural-language critique can be used to revise and improve model outputs across multiple passes (Madaan et al., 2023; Shinn et al., 2023). Finally, evidence identification and rationale extraction have been widely studied to make predictions inspectable and to ground outputs in supporting text, notably in fact verification and rationale benchmarks such as FEVER and ERASER (Thorne et al., 2018; DeYoung et al., 2020). These directions collectively motivate increasing attention to grounded generation, structured evaluation, and evidence-linked explanations in high-stakes domains such as law.

3 JustifIT-QA: Legal Q&A with Evidences

We introduce *JustifIT-QA*, an Italian statutory QA dataset built on top of JuriFindIT[3]. The dataset is designed to address two main limitations of that benchmark: the absence of expert-written answers and the lack of fine-grained evidence annotations. For each question, JustifIT-QA provides (i) an expert drafted answer and (ii) explicit annotations of the relevant text spans within the labelled articles drawn from the JuriFindIT corpus. This yields a stand-alone resource for grounded legal question answering, and evidence-based interpretability.

The extension is built on the expert-authored portion of the original dataset and does not introduce new questions. Instead, it increases the annotation granularity of the existing 895 queries by adding answers and span-level evidence over previously labelled relevant articles, preserving continuity with the original benchmark while enabling supervision beyond retrieval.

3.1 Expert-Written Answers

For each expert-authored question, we collected a reference answer written by a legal annotation team covering the four domains represented in the dataset: civil law, criminal law, anti-money laundering and counter-terrorism regulations, and privacy. Using a team with matching domain expertise helps maintain consistency across question interpretation, article selection, and answer drafting. The answers were written from scratch by human experts, without generation or post-editing from model outputs. Each answer is designed to resolve the question directly, identify the applicable rule, and mention the relevant conditions or consequences when required. We report some examples in Appendix A.4. Given the high cost of independent expert judgments, we report an auxiliary LLM-based agreement analysis in Appendix A.1. This analysis is not intended to replace human inter-annotator agreement, but to provide an consistency check, using two independent LLM evaluators.

3.2 Fine-Grained Evidence Annotation

In addition to writing answers, annotators marked the specific portions of the relevant articles that support them. Unlike the original dataset, which provides article-level supervision, JustifIT-QA introduces span-level evidence capturing the normative statements underlying the expert answer. Evidence was annotated at paragraph, sub-paragraph, clause, or shorter span level when necessary. Annotators were instructed to mark only the text directly needed to justify the answer, rather than entire articles. To reduce anchoring effects, annotation followed a two-step workflow: experts first drafted the answer, and only afterwards identified the minimal supporting excerpts in the relevant articles. This backward procedure encourages evidence selection based on justification rather than topical overlap. The resulting annotations define an explicit many-to-many mapping between questions, answers, and statutory text: a question may require multiple articles, and each article may contribute multiple disjoint excerpts. We report some samples from the dataset in Table 3.

[3] jurifindit/JuriFindIT

Area	mean	std	median	max
AML/CTF	24.85	52.83	5.5	360
Civil Code	1.07	0.31	1.0	4
Criminal Law	1.69	1.04	1.0	7
Privacy	32.88	33.37	22.0	141

Table 1: Number of highlighted evidence sections per question by normative area.

Area	mean	std	median	max
AML/CTF	3462.64	7248.08	760.0	41891
Civil Code	49.79	49.11	39.0	489
Criminal Law	156.01	277.23	76.5	2582
Privacy	4341.94	4461.66	2540.0	20748

Table 2: Total words in highlighted evidence sections per question by normative area.

4 Dataset Statistics

We report descriptive statistics for the two new annotation layers in JustifIT-QA, i.e. expert-written answers and relevant evidence sections, stratified by normative area. For answers, we compute word-count statistics and extract linguistic profiling features with Profiling-UD (Brunato et al., 2020) (Appendix A.2). For evidence, we report both the number of annotated sections per question and their total length in words.

Evidences. Tables 1 and 2 jointly describe the annotated evidence in terms of fragmentation and length. The two measures show a consistent domain-level pattern. *Civil Code* and *Criminal Law* mostly correspond to a compact-evidence setting, where answers can usually be justified through a small number of localized excerpts. *Criminal Law* is slightly more variable than *Civil Code*, but remains relatively bounded in most cases. By contrast, *AML/CTF* and *Privacy* exhibit substantially more fragmented and longer evidence, with heavy-tailed distributions. This suggests that questions in these areas often require aggregating several statutory passages and connecting dispersed normative constraints. The distinction between compact and long-context evidence settings motivates reporting downstream results by normative area.

Answers. The linguistic indicators reported in Appendix A.2 show domain-specific variation in answer formulation. *Privacy* answers are the most syntactically articulated, with longer sentences, greater dependency depth, and a higher proportion of subordinate structures. *Criminal Law* shows the strongest nominal profile, reflecting a more definition- and qualification-oriented style. *AML/CTF* and *Civil Code* exhibit broadly similar lexical and syntactic patterns. These trends indicate that the answer layer captures meaningful variation across normative areas, complementing the stronger cross-domain differences observed in the evidence statistics.

5 Answer Refinement and Evidence Generation

Legal question answering requires outputs that are not only fluent but *normatively grounded*: even minor inaccuracies may trigger incorrect interpretations of obligations, rights, or sanctions. This makes hallucinations and unsupported details particularly harmful in professional legal settings, where LLM-generated content has been shown to produce legally inconsistent statements and fabricated references (Dahl et al., 2024; Huang et al., 2025). To address this risk, we adopt a pipeline (Figure 1) that ties generation to statutory sources and produces reviewable support. Given a question, we (i) retrieve a small set of relevant provisions, (ii) generate an answer conditioned on the retrieved context via retrieval-augmented generation (Lewis et al., 2020), (iii) iteratively refine the answer through evidence-based verification, and (iv) extract verbatim supporting excerpts from the same provisions, in line with long-form legal QA settings that combine retrieval with grounded answer generation (Louis et al., 2024). Together, these components aim to reduce unsupported statements while improving interpretability by linking the final answer to explicit textual evidence.

5.1 Article Retrieval

Given a legal question in natural language, the goal of statutory article retrieval is to return a small set of legislative provisions that are useful for answering the query. We formalize retrieval as a function

$$R : (q, C; k) \to F \tag{1}$$

that takes a query q and a corpus $C = \{a_1, \ldots, a_N\}$ of N articles, and outputs a ranked subset $F \subset C$ of size $k \ll N$. Let $s_\theta(q, a) \in \mathbb{R}$ be a parametric scoring function, and let $\mathrm{rank}_q(a)$ denote the position of article a when sorting the corpus by decreasing score. The retrieved list is thus defined as:

$$\begin{aligned} F = R(q, C; k) &= \{\, a \in C \mid \mathrm{rank}_q(a) \le k \,\} \\ &= \operatorname*{TopK}_{a \in C} s_\theta(q, a). \end{aligned} \tag{2}$$

question	relevant_spans	answer
Mario takes out a life insurance policy on Lucia without having the authority to do so. Can Lucia ratify the policy after learning of it only after a claim has been made?	{21095: [0, 537]}	Yes, Lucia may ratify the contract even after the loss has occurred. Until the insurer has received notice of ratification or rejection, Mario remains personally liable for...
What are the main duties of the Data Protection Officer (DPO) under the GDPR?	{10662:[58-1223, 1301-2702, 3404-4676, . . .]}	The DPO must advise the controller/processor, monitor GDPR compliance...
In a company with four members, three of them decide to expel one member for serious breaches of duty. What are the rights of the expelled member, and when does the expulsion take effect?	{'21565': [41, 362]}	The expulsion is decided by a majority of the members, excluding the member to be expelled from the count. The expulsion takes effect 30 days after notification to...

Table 3: Examples from JustifIT-QA (translated into English). For each question, *relevant_spans* is a dictionary whose key is the identifier of a legal text within JuriFindIT corpus and whose value is a list of evidence spans, each represented by the start and end character offsets of the corresponding excerpt.

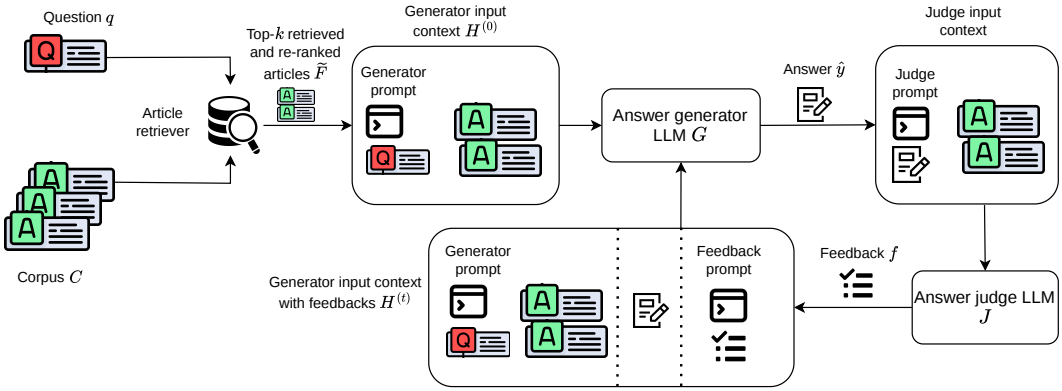

Figure 1: Given a question, an article retriever returns the top-k relevant provisions. A generator produces an answer conditioned on the retrieved articles. A judge model verifies the answer against the same articles and, when needed, produces feedback used to refine the next generation step. Prompt templates are shown in Appendix G.

We adopt a two-stage retrieval setup (Guo et al., 2022; Tao et al., 2023). In the first stage, we use a dense bi-encoder architecture (Reimers and Gurevych, 2019; Karpukhin et al., 2020) built on Transformer encoders (Vaswani et al., 2017), which maps queries and articles into a shared d-dimensional embedding space. A single encoder $\mathrm{Enc}_\theta(\cdot)$ is applied to both inputs, producing representations $\mathbf{h}_q = \mathrm{Enc}_\theta(q)$ and $\mathbf{h}_a = \mathrm{Enc}_\theta(a)$ with $\mathbf{h}_q, \mathbf{h}_a \in \mathbb{R}^d$. We compute each representation using the *last token* embedding. Relevance is estimated via cosine similarity, $s_\theta(q,a) = \cos(\mathbf{h}_q, \mathbf{h}_a) = \frac{\mathbf{h}_q \cdot \mathbf{h}_a}{\|\mathbf{h}_q\| \, \|\mathbf{h}_a\|}$, and all $a \in C$ are ranked according to $s_\theta(q,a)$.

In the second stage, a re-ranker refines the initial ranking by scoring each query–article pair jointly rather than comparing independently encoded representations. Given the candidate set returned by the bi-encoder, the re-ranker takes (q,a) as paired input and outputs a refined relevance score $\tilde{s}_\phi(q,a) \in \mathbb{R}$, which is used to reorder the candidates: $\widetilde{F} = \mathrm{RankK}_{a \in F} \; \tilde{s}_\phi(q,a)$, where $\tilde{s}_\phi(q,a)$ is the probability assigned by the re-ranker to the token *yes* for the paired input (q,a), i.e.,

$$\tilde{s}_\phi(q,a) = \frac{\exp(z_{\text{yes}}(q,a))}{\exp(z_{\text{yes}}(q,a)) + \exp(z_{\text{no}}(q,a))}, \qquad (3)$$

with $z_{\text{yes}}(q,a)$ and $z_{\text{no}}(q,a)$ denoting the final logits associated with the tokens *yes* and *no*, respectively. For convenience, in the rest of this work we use the function R to indicate the retrieval and re-ranking pipeline.

Learning. We train the bi-encoder with a contrastive objective over a dataset $\mathcal{D} = \{(q_i, a_i^p)\}_{i=1}^N$ of question–positive-article pairs. Negatives are sampled via in-batch negatives (Karpukhin et al., 2020): for a batch of size B, the negative set for q_k is $\mathcal{N}(q_k) = \{a_j \mid j \neq k, \; j \in [1, B]\}$. The loss is:

$$\mathcal{L}(q_i, a_i^p, \mathcal{N}(q_i)) = -\log \frac{\exp(s_\theta(q_i, a_i^p)/\tau)}{\sum_{a \in \{a_i^p\} \cup \mathcal{N}(q_i)} \exp(s_\theta(q_i, a)/\tau)}, \qquad (4)$$

where τ is a temperature parameter.

5.2 Judge-Based Feedback Loop

To reduce hallucinations and enforce consistency with the retrieved statutory context, we introduce a feedback loop driven by a *judge* language model. LLM-based judges have recently emerged as a practical alternative to costly human evaluation, with evidence of useful correlation with expert preferences in controlled settings (Zheng et al., 2023; Liu et al., 2023). We formalize the judge as

$$J : (q, \widetilde{F}, \hat{y}) \rightarrow (\ell, f), \tag{5}$$

where ℓ is a compliance label and f is a feedback message. The judge evaluates $\hat{y}$ *only* against the retrieved articles $\widetilde{F}$, treating them as the authoritative source and thus directly targeting unsupported content, a central failure mode of neural generation (Maynez et al., 2020; Ji et al., 2023). Its decision follows expert-defined criteria covering support in $\widetilde{F}$, absence of contradictions, preservation of normative force and quantitative details, preservation of scope (e.g., subjects, conditions, exceptions, jurisdiction), and faithful integration without arbitrary additions; the prompt with the criteria is reported in Appendix G.2.

The label ℓ follows a three-level scale designed with a legal team: NON-COMPLIANT, PARTIALLY-COMPLIANT, and FULLY-COMPLIANT. This granularity is important in legal QA, where answers often include both core legal claims and ancillary details (e.g., minor conditions, procedural details, contextual remarks) enriching content of the answers. A binary label would conflate errors affecting the main conclusion with inaccuracies confined to secondary information. We therefore use NON-COMPLIANT for errors on core legal content, PARTIALLY-COMPLIANT for answers whose main claim is correct but include unsupported peripheral details, and FULLY-COMPLIANT for answers fully consistent with $\widetilde{F}$.

When $\ell \neq$ FULLY-COMPLIANT, the judge returns a feedback f identifying unsupported or contradicted statements in $\hat{y}$. Following iterative self-refinement approaches (Madaan et al., 2023), we implement revision in conversational form: the generator receives f as a new user turn while preserving the dialogue history. Let $\mathcal{H}^{(t)}$ denote the conversation state at iteration t; refinement is:

$$\hat{y}^{(t+1)} = G\big(\mathcal{H}^{(t)} \oplus f^{(t)}\big), \tag{6}$$

where $\oplus$ appends the new feedback to the history. The process stops when the judge returns FULLY-COMPLIANT or when a maximum number of iterations is reached. Algorithm 1 summarizes the procedure. The proposed loop implements evidence-grounded self-correction: the judge flags claims unsupported by $\widetilde{F}$, and the generator revises its answer while the statutory context remains fixed.

Algorithm 1 Conversational judge-based feedback loop for answer refinement.

Require: question q, corpus C, retrieval pipeline R, generator G, judge J, top-k, max steps T

$\widetilde{F} \leftarrow R(q, C; k)$
$\mathcal{H}^{(0)} \leftarrow \langle (q, \widetilde{F}) \rangle$
$\hat{y}^{(0)} \leftarrow G(\mathcal{H}^{(0)})$
$\mathcal{H}^{(0)} \leftarrow \mathcal{H}^{(0)} \oplus \hat{y}^{(0)}$
for $t = 0$ **to** $T - 1$ **do**
  $(\ell^{(t)}, f^{(t)}) \leftarrow J(q, \widetilde{F}, \hat{y}^{(t)})$
  if $\ell^{(t)} =$ FULLY-COMPLIANT **then**
    return $\hat{y}^{(t)}$
  end if
  $\mathcal{H}^{(t+1)} \leftarrow \mathcal{H}^{(t)} \oplus f^{(t)}$
  $\hat{y}^{(t+1)} \leftarrow G(\mathcal{H}^{(t+1)})$
  $\mathcal{H}^{(t+1)} \leftarrow \mathcal{H}^{(t+1)} \oplus \hat{y}^{(t+1)}$
end for
return $\hat{y}^{(T)}$

5.3 LLM-based Evidence Extraction

Beyond answer generation and verification, we explicitly extract textual *evidence* from the retrieved statutory context. This follows evidence-based QA paradigms (DeYoung et al., 2020; Thorne et al., 2018) and is particularly relevant in law, where verbatim, source-linked excerpts improve transparency and support post-hoc auditing.

Let q be a question, $C = \{a_1, \ldots, a_N\}$ the article corpus, and $\widetilde{F} = R(q, C; k)$ the retrieved set. Given the selected answer $y^\star$, we define an evidence extractor as

$$E : (q, \widetilde{F}, y^\star) \rightarrow \widehat{\mathcal{S}}, \tag{7}$$

where $\widehat{\mathcal{S}}$ is a set of predicted evidence spans. Each span is represented as $s = (id(a), \tau)$, where $a \in \widetilde{F}$ and τ is a verbatim excerpt from a; thus $\widehat{\mathcal{S}} = \{\hat{s}_1, \ldots, \hat{s}_m\}$. Gold evidence is defined analogously as $\mathcal{S} = \{s_1, \ldots, s_n\}$ with $s_i = (id(a_i), \tau_i)$, matching the dataset annotation protocol and enabling direct set-level comparison. Operationally, E returns a structured list of $(id(a), \tau)$ pairs covering all excerpts needed to justify $y^\star$. This set-based formulation captures the many-to-many structure of legal justification, while restricting extraction to $\widetilde{F}$ aligns evidence with the same sources used for answering and limits unsupported citations. Combined with judge-based refinement, this

yields answers that are both context-consistent and grounded, a requirement emphasized in long-form legal QA (Louis et al., 2024).

6 Experiments

We evaluate our framework by measuring its ability to generate legally compliant answers and accurately identify supporting evidence. The experiments aim to quantify the impact of iterative judge feedback on answer quality and to assess the precision of the extraction and retrieval components.

6.1 Answer Evaluation

Expert evaluation. To validate whether judge feedback improves answer quality in practice, we conduct an expert evaluation on 181 generated answers from 133 randomly sampled questions. This includes one judge-selected answer for each question and, for 48 of them, the corresponding initial answer before feedback, enabling paired comparison. These 48 questions cover roughly 25% of the cases in which feedback-based refinement was applied. Legal annotators assess each answer using the same three-level compliance scale and criteria adopted by the judge (§ 5.2), given the question, the generated answer, and the gold relevant articles, and are instructed to rely only on those articles. We use these labels to characterize overall compliance and to measure the effect of feedback against the no-feedback baseline.

Automatic metrics. We measure reference-based similarity between generated and expert answers with METEOR (Banerjee and Lavie, 2005) and BERTScore (Zhang et al., 2020). Pure n-gram overlap metrics such as BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) are less suitable for long-form, open-ended answers (Wang et al., 2022; Xu et al., 2023). METEOR mitigates surface-form mismatch via stemming and synonym matching and has shown stronger correlation with human judgments than overlap-based measures (Sharma et al., 2017; Chen et al., 2022), while BERTScore captures semantic similarity through contextual embeddings and is therefore more robust to paraphrasing and variation in detail level.

6.2 Evidence Extraction Evaluation

Exact string matching between predicted and annotated evidence is brittle in legal texts, where harmless boundary differences or adjacent context may lead to mismatches. We therefore adopt a containment-based criterion computed *per sample* and then macro-averaged over the dataset. For the i-th question, let $\hat{T}_i = \{\hat{\tau}_{i,1}, \ldots, \hat{\tau}_{i,m_i}\}$ be the set of predicted excerpts and $T_i = \{\tau_{i,1}, \ldots, \tau_{i,n_i}\}$ the corresponding gold excerpts, pooled across retrieved articles. For a predicted excerpt $\hat{\tau} \in \hat{T}_i$ and a gold excerpt $\tau \in T_i$, we define:

$$\mathrm{GoldCont}(\hat{\tau}, \tau) = \frac{|Tok(\hat{\tau}) \cap Tok(\tau)|}{|Tok(\tau)|}, \qquad (8)$$

where $Tok(\cdot)$ denotes the set of normalized tokens (lowercasing and punctuation removal). A match is counted when $\mathrm{GoldCont}(\hat{\tau}, \tau) \geq 0.95$, with one-to-one matching to prevent multiple predictions from claiming the same gold excerpt. This yields per-sample TP_i, FP_i, and FN_i, from which we compute precision, recall, and F1, and finally macro-average over all samples.

6.3 Experimental Setup

Retriever and re-ranking. We fine-tune *jina-embeddings-v5-text*[4] (Akram et al., 2026), an encoder-only embedding model based on EuroBERT (Boizard et al., 2025), as the dense retriever used throughout the pipeline. The model has 239M parameters and supports inputs up to 8,192 tokens, making it suitable for long legislative texts. We train it on a mixture of real and synthetic questions: following prior work on statutory retrieval (Louis and Spanakis, 2022; Noce et al., 2026), we generate one synthetic question per article and combine these queries with the expert-authored questions as synthetic query generation has been shown to improve retrieval quality (Lawrie et al., 2025). Training uses a learning rate of 3.5×10^{-5} with linear decay, weight decay of 10^{-2}, batch size 768, and 35 epochs. At inference time, we first retrieve an initial candidate set and then re-rank the top-20 articles with *Qwen3-Reranker-4B* (Zhang et al., 2025); a more detailed description and comparison with other models is provided in Appendix C.

Metrics. We evaluate retrieval with standard rank-aware metrics over the top-k list. Recall@k measures the fraction of gold relevant articles retrieved in the top-k, MRR@k emphasizes how early the first relevant article appears, and nDCG@k and MAP@k reward systems that rank multiple relevant provisions higher in the list.

Generation, judging, and evidence extraction. We use Qwen3-32B[5] (Team, 2025) in no-thinking

[4] jinaai/jina-embeddings-v5-text-nano [5] Qwen3.x models.

Retriever	*Params*	*R@20*	*nDCG@20*	*MRR@20*	*MAP@20*
DAR	305M	83.16	74.59	77.26	68.50
DARv2	239M	**85.88**	**75.97**	**77.55**	**69.28**

Table 4: First-stage retriever performance on the JuriFindIT validation set at k=20.

Setting	*Precision*	*Recall*	*F1*
Main correctness	87.16	84.31	85.71
Side correctness	79.84	72.79	76.15

Table 5: Judge (Qwen3-32B) performance against expert labels, treating correctness as the positive class.

mode for both answer generation and for judging within the feedback loop. For compliance judging, we evaluate Qwen3 family of models[6], GPT-oss[7] (OpenAI, 2025) variants, and Gemma 4 models[8]. The generator produces answers conditioned on the question and retrieved articles, while judges assign compliance labels and provide feedback grounded in the same statutory context. Due to memory constraints, we cap the context to the top-5 retrieved articles and set the maximum number of feedback iterations to 2. For evidence extraction, we treat the task as structured prediction of verbatim supporting excerpts paired with article identifiers, comparing Qwen3 and Gemma 4 models. Text generation parameters are reported in Appendix I.

7 Results

7.1 Retriever Performance

Table 4 reports first-stage retrieval performance on the candidate pool used by the reranker. DARv2 improves over DAR across all metrics, with the largest gain on Recall. Paired statistical tests over validation queries show that improvements are significant for Recall at $k \geq 20$; full results are reported in Appendix B. This result is especially relevant for our pipeline, where the first-stage retriever should maximize coverage of relevant legal provisions before reranking. We also tested an iterative LLM-based filtering strategy, where the model scans retrieved texts through a sliding window and decides whether they are sufficient or more should be inspected, retaining only the relevant ones. Under our constrained input setting, however, its gains over traditional re-ranking were limited and outweighed by higher latency. We therefore adopt the standard re-ranker and report the full comparison and more detailed description in Appendix D.

7.2 Judge-Based Evaluation and Feedback

Agreement with expert judgments. We first assess the judge as a proxy for expert evaluation by treating correctness as the positive class and reporting precision, recall, and F1 in Table 5; raw counts are provided in Appendix E. The judge is substantially more reliable on *main* content than on *side* information, with a better precision–recall trade-off on the core legal substance of the answer. In legal QA, precision is especially important because false positives correspond to answers that the judge approves despite being considered incorrect by experts, thereby increasing the risk of endorsing legally unreliable content. An error analysis suggests that retrieval recall is a key driver of expert-rated quality: answers judged incorrect by experts are associated with markedly lower average article recall than correct ones, both for main and side content, whereas false negatives exhibit recall levels comparable to true positives. This indicates that missing relevant context primarily harms answer quality, while many false negatives reflect conservative judge behaviour. Further details such as qualitative analysis are reported in Appendix F.

Judging with access to expert answers. We further evaluate judges in a reference-based setting, where they receive the expert-written answer, labelled articles, and generated answer. Results are reported in Table 6, with counts in Appendix E. Providing the reference answer improves alignment with experts, but reveals distinct operating profiles. Qwen3-32B achieves the best F1 and highest recall, making it the strongest option when preserving expert-correct answers is important. Gemma 4 31B attains the highest precision, while Gemma 4 26B-A4B offers the best balance among the newly tested models. GPT-oss variants and Qwen3.6 27B are precise but more conservative, and Qwen3.6 35B-3A shows the lowest recall. These results highlight a legal-domain safety–coverage trade-off: precision reduces false acceptance of incorrect answers, whereas recall preserves valid ones.

Effect of judge feedback. Finally, we examine whether judge feedback translates into better answers under expert assessment. Table 7 compares the proportion of expert-rated correct answers before and after refinement. On the subset of questions with non-zero retrieval recall, feedback im-

[6] Qwen3, 3.5, 3.6 [7] GPT-oss. [8] Gemma 4.

Judge model	*Precision*	*Recall*	*F1*
Qwen3 32B	89.57	**95.42**	**92.40**
Qwen3.5 27B	93.89	80.39	86.62
Qwen3.5 27B (thinking)	94.57	79.74	86.52
Qwen3.6 27B	95.38	81.05	87.63
Qwen3.6 35B-3A	93.64	67.32	78.33
GPT-oss 20b (thinking med.)	92.37	79.08	85.21
GPT-oss 20b (thinking low)	92.65	82.35	87.20
Gemma 4 31B	**95.71**	87.58	91.47
Gemma 4 31B (thinking)	94.85	84.31	89.27
Gemma 4 26B-A4B	93.20	89.54	91.33

Table 6: Judge performance comparison on main correctness when provided with the expert-written reference answer and labelled articles.

Correct answers	*No feedback*	*With feedback*
Main correctness	76.74	83.72
Side correctness	69.77	74.42

Table 7: Percentage of expert-rated correct answers, comparing the initial answer (*No feedback*) and the judge-refined answer (*With feedback*).

proves correctness on both main content and side details, indicating that judge-guided revision is effective when relevant statutory support is available. At the same time, the benefit is contingent on retrieval quality: when recall is zero, we found feedbacks counterproductive, as the model has no reliable evidence to ground the revision and producing unsupported changes. This confirms that the usefulness of iterative refinement depends on the availability of relevant context, and therefore on the quality of the retriever upstream.

7.3 Automatic Evaluations

We also compare generated answers against expert-written references using METEOR and BERTScore. Since expert references mainly capture the core legal content, we stratify the analysis by expert judgments of *main* correctness (Table 8). Both metrics are higher for answers judged correct by experts, but the separation is limited, especially for BERTScore. This suggests that reference-based similarity captures only partial alignment with expert judgments. These metrics should therefore be viewed as descriptive rather than correctness-based: incorrect answers may still score high, while correct ones may differ in wording or detail.

7.4 Evidence Extraction

Table 9 reports evidence extraction results on the JustifIT-QA validation set using the same prompting setup and the top-5 retrieved articles. Qwen3-32B achieves the best overall performance, followed closely by Qwen3.5-27B. Within the Qwen family, larger models generally improve precision, recall, and F1, whereas the Gemma models remain less competitive despite gains from E4B to 31B. Across all systems, recall is consistently higher than precision, suggesting that models often cover relevant evidence but include overly broad or unmatched excerpts. Evidence extraction benefits from stronger models but remains challenging under strict near-verbatim matching.

Score	*Expert=0*	*Expert=1*
METEOR	25.34 ± 6.70	32.12 ± 11.34
BERTScore	66.80 ± 2.84	70.27 ± 4.11

Table 8: Similarity between generated answers and expert references, stratified by expert main correctness, reported as mean $\pm$ standard deviation.

Model	*Precision*	*Recall*	*F1*
Qwen3 8B	25.75	36.37	27.39
Qwen3 14B	34.70	46.64	36.26
Qwen3 32B	**41.07**	**53.08**	**42.55**
Qwen3.5 9B	28.73	45.13	32.13
Qwen3.5 27B	40.09	51.20	41.92
Gemma 4 E4B	23.32	37.03	26.11
Gemma 4 31B	29.78	41.41	31.83

Table 9: Evidence extraction results on the JustifIT-QA validation set. Scores are macro-averaged across samples.

8 Conclusion

We introduced *JustifIT-QA*, an Italian statutory QA dataset with expert-written answers and fine-grained evidence spans, enabling evaluation beyond article-level retrieval. We also presented *DARv2*, which improves retrieval performance over the original JuriFindIT retriever. On top of this retrieval layer, we studied a judge-based feedback loop for answer refinement and showed that, when relevant context is available, it improves expert-rated correctness. We further found that LLM-based judging is a feasible proxy for evaluating core legal content, although performance drops on ancillary details, and that evidence extraction quality improves with model scale. In the end, our results highlight the value of combining retrieval, verification, and source-linked evidence for grounded and explainable legal QA, providing resource for future research directions.

Limitations

Our study has several limitations. First, due to memory constraints, all LLM-based components operate on at most the top-5 retrieved articles. While this budget keeps the pipeline tractable, it inevitably excludes relevant provisions for some questions, especially in domains such as AML/CTF and Privacy, where evidence is often distributed across multiple statutory texts. This constraint is particularly relevant for downstream generation and judging, since the feedback loop can only verify answers against the evidence that is actually retrieved and included in the context.

Second, the proposed feedback loop introduces additional computational cost and latency compared with a single-pass RAG pipeline. Each refinement step requires both answer generation and judge-based verification, and our experiments use large open-weight models. This design is useful for studying evidence-grounded self-correction, but it may be expensive or slow in practical deployments.

Third, the refinement process is capped at two feedback iterations. This choice prevents uncontrolled generation cycles and keeps the evaluation feasible. Moreover, the system does not simply return the answer produced at the last iteration; rather, it selects the best answer according to the judge among the generated candidates. With more iterations, answers might become overly conservative, or degrade if the judge introduces misleading feedback. Since changing the number of iterations can change the set of candidate answers and therefore the selected output, a rigorous analysis would require additional expert evaluation.

Fourth, the evaluation of answer quality remains partly constrained by the use of expert judgments on a sampled subset rather than on the full dataset. Although this is common in high-cost legal annotation settings, it limits the granularity with which we can analyse domain-specific failure modes. In addition, while we provide an auxiliary LLM-based agreement analysis as a consistency check, this does not replace independent human inter-annotator agreement. A future extension should include multiple legal experts judging the same answers and evidence spans, in order to better quantify the difficulty and reliability of the annotation task.

Finally, evidence extraction is evaluated with a containment-based matching criterion that is more robust than exact string matching, but still remains surface-oriented. It does not fully capture deeper equivalence relations such as paraphrastic justifications, alternative but legally valid excerpt boundaries, or implicit support distributed across multiple passages. More semantically informed evaluation of legal evidence extraction remains an open problem.

Ethical considerations

All questions, answers, and evidence annotations in JustifIT-QA were produced by legal experts, and the dataset does not contain personal or sensitive user data. Nevertheless, errors may arise from preprocessing, article segmentation, evidence selection, or annotation inconsistencies, and may propagate to downstream training, generation, or evaluation.

The pipeline is intended as research infrastructure for grounded legal QA, not as a substitute for professional legal advice. Although retrieval, judging, and evidence extraction improve transparency, generated answers may still omit relevant provisions, misapply cited norms, or contain unsupported statements, especially when retrieval fails. Source-linked evidence may also induce automation bias, as users could over-trust answers accompanied by statutory excerpts. Any practical use should therefore keep legal experts in the loop and treat retrieved evidence as material for verification rather than as a guarantee of correctness.

Finally, legal QA systems carry broader risks. They may improve access to legal information, but could also be misused to identify minimal compliance strategies or regulatory loopholes, or reinforce inequalities if available only to well-resourced actors. These risks further support framing JustifIT-QA as a research resource rather than a deployable legal-advice system.

References

Mohammad Kalim Akram, Saba Sturua, Nastia Havriushenko, Quentin Herreros, Michael Günther, Maximilian Werk, and Han Xiao. 2026. jina-embeddings-v5-text: Task-targeted embedding distillation. *Preprint*, arXiv:2602.15547.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization@ACL 2005, Ann Arbor, Michigan, USA, June 29, 2005*, pages 65–72. Association for Computational Linguistics.

Nicolas Boizard, Hippolyte Gisserot-Boukhlef, Duarte M. Alves, André F. T. Martins, Ayoub Hammal, Caio Corro, Céline Hudelot, Emmanuel Malherbe, Etienne Malaboeuf, Fanny Jourdan, Gabriel Hautreux, João Alves, Kevin El Haddad, Manuel Faysse, Maxime Peyrard, Nuno Miguel Guerreiro, Patrick Fernandes, Ricardo Rei, and Pierre Colombo. 2025. Eurobert: Scaling multilingual encoders for european languages. *CoRR*, abs/2503.05500.

Dominique Brunato, Andrea Cimino, Felice Dell'Orletta, Giulia Venturi, and Simonetta Montemagni. 2020. Profiling-ud: a tool for linguistic profiling of texts. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 7145–7151. European Language Resources Association.

Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael J. Bommarito II, Ion Androutsopoulos, Daniel Martin Katz, and Nikolaos Aletras. 2022. Lexglue: A benchmark dataset for legal language understanding in english. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 4310–4330. Association for Computational Linguistics.

Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. 2024. Humans or LLMs as the judge? a study on judgement bias. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8301–8327, Miami, Florida, USA. Association for Computational Linguistics.

Yiran Chen, Zhenqiao Song, Xianze Wu, Danqing Wang, Jingjing Xu, Jiaze Chen, Hao Zhou, and Lei Li. 2022. MTG: A benchmark suite for multilingual text generation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2508–2527, Seattle, United States. Association for Computational Linguistics.

Matthew Dahl, Varun Magesh, Mirac Suzgun, and Daniel E. Ho. 2024. Large legal fictions: Profiling legal hallucinations in large language models. *CoRR*, abs/2401.01301.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2020. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4443–4458. Association for Computational Linguistics.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2025. A survey on llm-as-a-judge. *Preprint*, arXiv:2411.15594.

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, Adam Chilton, Aditya Narayana, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel N. Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory M. Dickinson, Haggai Porat, Jason Hegland, and 21 others. 2023. Legalbench: a collaboratively built benchmark for measuring legal reasoning in large language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Jiafeng Guo, Yinqiong Cai, Yixing Fan, Fei Sun, Ruqing Zhang, and Xueqi Cheng. 2022. Semantic models for the first-stage retrieval: A comprehensive review. *ACM Trans. Inf. Syst.*, 40(4):66:1–66:42.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. CUAD: an expert-annotated NLP dataset for legal contract review. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*, 43(2):42:1–42:55.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12):248:1–248:38.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6769–6781. Association for Computational Linguistics.

Dawn J. Lawrie, Efsun Kayi, Eugene Yang, James Mayfield, Douglas W. Oard, and Scott Miller. 2025. Generate-distill: Training cross-language IR models with synthetically-generated data. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2025, Padua, Italy, July 13-18, 2025*, pages 2926–2930. ACM.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 2511–2522. Association for Computational Linguistics.

Antoine Louis and Gerasimos Spanakis. 2022. A statutory article retrieval dataset in french. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 6789–6803. Association for Computational Linguistics.

Antoine Louis, Gijs van Dijck, and Gerasimos Spanakis. 2024. Interpretable long-form legal question answering with retrieval-augmented large language models. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2024, February 20-27, 2024, Vancouver, Canada*, pages 22266–22275. AAAI Press.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan T. McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 1906–1919. Association for Computational Linguistics.

Niko Dalla Noce, Davide Colla, Sina Farhang Doust, Lorenzo De Mattei, and Davide Bacciu. 2026. Jurifindit: an italian legal retrieval dataset. In *Findings of the Association for Computational Linguistics: EACL 2026, Rabat, Morocco, March 24-29, 2026*, Findings of ACL, pages 4223–4241. Association for Computational Linguistics.

OpenAI. 2025. gpt-oss-120b & gpt-oss-20b model card. *CoRR*, abs/2508.10925.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Shikhar Sharma, Layla El Asri, Hannes Schulz, and Jeremie Zumer. 2017. Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation. *Preprint*, arXiv:1706.09799.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Alessandro Stolfo. 2024. Groundedness in retrieval-augmented long-form generation: An empirical study. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1537–1552, Mexico City, Mexico. Association for Computational Linguistics.

Chongyang Tao, Jiazhan Feng, Tao Shen, Chang Liu, Juntao Li, Xiubo Geng, and Daxin Jiang. 2023. CORE: cooperative training of retriever-reranker for effective dialogue response selection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 3102–3114. Association for Computational Linguistics.

Qwen Team. 2025. Qwen3 technical report. *CoRR*, abs/2505.09388.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and verification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 809–819. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Shufan Wang, Fangyuan Xu, Laure Thompson, Eunsol Choi, and Mohit Iyyer. 2022. Modeling exemplification in long-form question answering via retrieval. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2079–2092, Seattle, United States. Association for Computational Linguistics.

Fangyuan Xu, Yixiao Song, Mohit Iyyer, and Eunsol Choi. 2023. A critical evaluation of evaluations for long-form question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3225–3245, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. JEC-QA: A legal-domain question answering dataset. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9701–9708. AAAI Press.

A Dataset

A.1 LLM-based Agreement Analysis

We conduct an auxiliary LLM-based agreement analysis to further assess the consistency of the answer correctness judgments in our dataset. This analysis is not intended to replace human expert inter-annotator agreement, but rather to provide an additional model-based consistency check on a sample of legal question-answer pairs.

For this analysis, we used two frontier LLM-based evaluators accessed through their respective APIs: GPT-5.5 with medium reasoning and Gemini 3.5 Flash with medium reasoning. For both models, web search was enabled. The goal is to verify whether two independently operating model-based evaluators reach comparable assessments when judging the same legal question-answer pairs under a shared evaluation protocol. We sampled 10% of the dataset, preserving the heterogeneity of the resource across the covered legal domains, including civil law, criminal law, anti-money laundering, and privacy. Each example was evaluated independently by both models using the same structured prompt, the same label set, and the same evaluation criteria.

For each question-answer pair, the models were asked to assign one of three labels: CORRECT, PARTIALLY CORRECT, or INCORRECT. The prompt also required a confidence score, a short rationale, and a categorical indication of the main critical issue, if any. The evaluation was performed in an open-book setting: both models were allowed to use web search to consult relevant legal sources when needed, in addition to their internal legal knowledge. At the same time, the prompt instructed the models not to introduce additional factual assumptions beyond those contained in the question-answer pair. Since web search was enabled independently for each model, the external sources consulted by the two evaluators were not necessarily identical. Accordingly, the analysis reflects agreement between two independently operating LLM-based evaluators in a realistic open-book evaluation setting, rather than agreement under a fixed shared-source condition.

The prompt is reported in Appendix G.4. Concise answers were not penalized when legally sufficient, but substantial legal errors, misleading generalizations, missing conditions, relevant exceptions, and unsupported claims were treated as correctness issues. This choice reflects the nature of legal question answering, where an answer may be directionally correct but still problematic if it omits a necessary legal qualification, condition, or exception.

ChatGPT marks 78 answers as CORRECT (84.8%) and Gemini 76 answers as CORRECT (82.6%), while both models assign PARTIALLY CORRECT to 12 answers (13.0%). Fully negative judgments are rare, with only 2 INCORRECT labels from ChatGPT (2.2%) and 4 from Gemini (4.3%). In the two cases marked as INCORRECT by both models, manual inspection showed that the expert answer referred to a legal provision that had been recently updated, highlighting the temporal sensitivity of legal QA rather than indicating systematic annotation errors.

We compare the judgments produced by the two LLM-based evaluators using raw agreement, Cohen's κ, and weighted versions of Cohen's κ. Raw agreement measures the proportion of examples for which the two models assign the same label, while Cohen's κ corrects this agreement for chance. Since the labels form an ordered scale from INCORRECT to PARTIALLY CORRECT to CORRECT, we also report linear and quadratic weighted κ, which penalize distant-label disagreements more strongly than adjacent-label disagreements. For all metrics, we compute 95% bootstrap confidence intervals.

Metric	Value	95% CI
Raw agreement	0.880	[0.815, 0.946]
Cohen's κ	0.618	[0.412, 0.792]
Linear weighted κ	0.644	[0.447, 0.809]
Quadratic weighted κ	0.679	[0.456, 0.845]

Table 10: Agreement between the two LLM-based evaluators on the 10% dataset sample. Confidence intervals are computed through bootstrap resampling.

Table 10 shows that the two LLM-based evaluators reach a raw agreement of 88.0%, assigning the same correctness label in almost nine out of ten cases. Cohen's κ is 0.618. The point estimate falls within the range often interpreted as substantial agreement, although the confidence interval also includes values typically interpreted as moderate agreement. This suggests that the agreement remains meaningful after correcting for chance, while also reflecting the uncertainty inherent in the sample-based analysis.

The weighted coefficients are higher than the unweighted κ, with a linear weighted κ of 0.644 and a quadratic weighted κ of 0.679. This indicates that the remaining disagreements are mostly mild rather than radical: the evaluators more often differ between adjacent labels, such as CORRECT and PARTIALLY CORRECT, than between opposite labels, such as CORRECT and INCORRECT. This pattern is particularly relevant in legal answer assessment, where borderline cases may depend on how strictly an evaluator interprets completeness, implicit legal conditions, or the need for explicit qualifications.

Critical issue	GPT-5.5	Gemini 3.5 Flash
`none`	78	76
`missing_condition`	9	4
`wrong_law`	2	4
`overgeneralization`	1	1
`ambiguity`	1	2
`unsupported_claim`	1	0
`other`	0	5

Table 11: Marginal distribution of critical issues assigned by the two LLM-based evaluators.

Table 11 provides a qualitative view of the residual issues identified by the two LLM-based evaluators. The predominance of `none` suggests that most sampled answers are not associated with substantive legal flaws. When issues are detected, they are concentrated in a limited set of legally meaningful categories, rather than being broadly distributed across heterogeneous error types.

The main pattern concerns missing legal conditions, exceptions, or qualifications, which are especially relevant in legal QA because an answer may be broadly correct while still incomplete with respect to the applicable rule. Gemini 3.5 Flash appears to use a slightly different categorization strategy, more often assigning problems to the residual `other` category. By contrast, issues such as `overgeneralization`, `ambiguity`, and `unsupported_claim` remain marginal, suggesting that the problematic cases mostly concern legal completeness and norm selection rather than vague or unsupported answers.

Finally, the agreement metrics and the critical-issue distribution suggest that the two LLM-based evaluators provide broadly consistent judgments

on the sampled question-answer pairs. Most detected issues concern missing legal qualifications or overly broad formulations, rather than systematic answer unreliability.

A.2 Linguistic profiling of answers

To characterize the linguistic properties of the answers, we compute Profiling-UD features separately for the four normative areas (AML/CTF, Civil Code, Criminal Law, Privacy). Table 12 reports a concise set of indicators covering lexical complexity (chars/token, lexical density), POS distribution (%NOUN, %VERB), and syntactic complexity (dependency depth, dependency length, %subordinate clauses). The trends are consistent across measures: *Privacy* answers are syntactically more articulated, with longer sentences and stronger embedding, whereas *Criminal Law* displays the highest nominal share, reflecting a more nominalized style. *AML/CTF* and *Civil Code* show broadly similar POS profiles and overall syntactic patterns.

A.3 Synthetic questions

Linguistic profiling of real vs. synthetic questions To characterize the linguistic gap between human-authored queries and our synthetic supervision, we report Profiling-UD statistics (Brunato et al., 2020) for (i) real, expert-annotated questions, (ii) JuriFindIT synthetic questions distributed with the dataset, and (iii) our new synthetic questions used in the new training runs. Table 13 summarizes the features (mean $\pm$ std).

Across features, the new synthetic questions are closer to real queries in length than the JuriFindIT synthetic set (higher tokens_per_sent), while remaining less subordinated (lower subordinate_proposition_dist and subordinate chain length). We hypothesize that this shift yields more retrieval-effective supervision for dense encoders: (i) slightly longer queries provide additional disambiguating context against near-duplicate statutory provisions, and (ii) a stronger nominal footprint (higher NOUN distribution) aligns better with the noun-heavy terminology that distinguishes legal articles (entities, institutions, and procedural objects). At the same time, reduced subordination may remove syntactic "noise" (incidental clauses) that is weakly predictive of relevance, stabilizing query representations during contrastive training and improving separation from hard negatives.

Synthetic Question Generation. Synthetic questions were generated with QWEN3.5-27B, prompted to produce one professional-style Italian legal question for each article. The prompt was designed to encourage technically precise, closed-form questions grounded only in the normative text, without introducing unsupported conditions or external legal knowledge.

> You are an expert lawyer in Italian law. Your task is to formulate **one single legal question** in the style of a professional legal memorandum or opinion, applying the normative text to a minimal abstract fact pattern without inventing elements not present in the text.
>
> Follow these constraints:
>
> **(1) Use only the provided text.** Do not introduce definitions, thresholds, time limits, sanctions, exceptions, or subjects not explicitly supported by the text.
>
> **(2) Ask a precise legal question.** The question should target a specific legal issue, such as conditions, obligations, scope of application, exceptions, procedures, deadlines, addressees, or legal effects.
>
> **(3) Keep the question closed and non-generic.** Avoid requests for general explanation or summary (e.g., "what does the article say"). Produce a single interrogative sentence, without subquestions, lists, or preambles.
>
> **(4) Use professional legal style.** Prefer formulations such as: "under what conditions", "in which cases", "who is required to", "what requirement applies", "what legal effect follows", or "whether an exception applies". Use precise legal vocabulary and keep the wording concise (roughly 20–45 words).
>
> **(5) Avoid article references.** Do not mention article, paragraph, or section numbers, and do not use expressions such as "in this article" or "in the present text."
>
> **Input:** Normative text: `{text}`
> Style examples: `{examples}`
>
> **Output:** Return only a valid Python dictionary, with no additional text, in the following format:
>
> `{"question": "<generated question>"}`

A.4 Annotated Examples

This appendix provides two examples from the dataset, showing the question, the annotated relevant evidence span(s), and the corresponding expert-written answer.

Example 1. `Question:` If Anna gives Marco a watch to thank him for significant help, is this considered a donation?

`Relevant evidence:` *Article_id: 19778*, *"A gift made out of gratitude, in consideration of the*

Area	tok/sent	chars/tok	lex. dens.	%NOUN	%VERB	depth	dep. len	%subord.
AML/CTF	36.69	5.12	0.480	20.68	9.40	6.17	2.80	58.30
Civil Code	35.81	4.76	0.470	21.17	9.40	5.71	2.95	59.65
Criminal Law	44.06	4.91	0.480	24.04	8.14	6.43	2.95	62.60
Privacy	72.96	5.18	0.484	22.29	8.46	8.03	3.31	75.07

Table 12: Selected Profiling-UD statistics (means) computed on answer texts. *tok/sent*: tokens per sentence; *chars/tok*: characters per token; *lex. dens.*: content-word ratio; *depth*: average maximum dependency-tree depth; *dep. len*: average dependency link length (tokens); *%subord.*: subordinate proposition ratio.

Feature	Real (annot.)	JuriFindIT synth.	New synth.
tokens_per_sent	34.27 ± 12.00	24.82 ± 6.41	27.14 ± 7.05
char_per_tok	5.00 ± 0.51	5.03 ± 0.62	5.35 ± 0.73
lexical_density	0.50 ± 0.06	0.45 ± 0.06	0.47 ± 0.12
upos_dist_VERB	10.85 ± 3.99	6.66 ± 3.81	5.00 ± 4.07
upos_dist_NOUN	20.34 ± 4.60	22.94 ± 4.43	26.28 ± 9.49
avg_max_depth	5.45 ± 1.63	5.68 ± 1.75	5.44 ± 1.83
avg_links_len	2.92 ± 0.66	2.21 ± 0.41	3.02 ± 1.49
avg_subordinate_chain_len	1.12 ± 0.63	0.82 ± 0.61	0.60 ± 0.63
subordinate_proposition_dist	57.78 ± 25.45	40.81 ± 27.79	31.95 ± 32.70

Table 13: Profiling-UD features (mean $\pm$ std) for real and synthetic questions. "JuriFindIT synth." refers to the synthetic questions originally distributed with JuriFindIT; "New synth." refers to our newly generated synthetic questions.

recipient's merits, or as a special reward is also a donation."

`Expert-written answer:` Yes. A gift made out of gratitude or in consideration of the recipient's merits is also considered a donation.

Example 2. `Question:` When the lease expires, Laura continues to live in the property and the landlord does not object. Can the contract be considered renewed?

`Relevant evidence:` *Article_id: 20758*, *"The lease is deemed renewed if, after its term has expired, the tenant remains and is allowed to retain possession of the leased property, or, in the case of a lease of indefinite duration, if notice has not been given in accordance with the previous article. The new lease is governed by the same conditions as the previous one, but its duration is that established for leases of indefinite duration."*

`Expert-written answer:` Yes. If, after the lease expires, the tenant remains in possession of the property and the landlord does not object, the lease is tacitly renewed under the same conditions as the previous one, but with indefinite duration, unless notice to vacate or termination had already been given.

B Retriever Statistical Significance

We compare DAR and DARv2 on the JuriFindIT validation set using paired tests over queries. For each metric, we compute the per-query difference between DARv2 and DAR, estimate a 95% paired bootstrap confidence interval with 10,000 samples, and apply a one-sided paired approximate randomization test with 10,000 sign permutations. Table 14 reports results at multiple cutoffs. DARv2 improves all metrics across all cutoffs, but statistically significant gains are observed only for Recall at $k \geq 20$. This supports the use of DARv2 as a first-stage retriever: its main benefit is improved coverage of relevant provisions before reranking.

C Re-ranking

To isolate the contribution of reranking, we evaluate a two-stage pipeline in which DARv2 first retrieves the Top-20 candidate articles and a reranker reorders only this fixed pool. The DARv2 row in Table 15 reports the original dense-retriever results, while the Qwen3 rows report the effect of reranking. Since the pool is fixed, the comparison measures ranking quality conditional on candidate generation: rerankers can promote relevant articles already present in the Top-20 list, but cannot recover missing ones.

The results show that sufficiently large Qwen3 rerankers substantially improve the ordering of the retrieved candidates, especially in early-rank metrics. This indicates that relevant legal articles are more often promoted to the top positions after reranking. Qwen3-Reranker-8B achieves

Metric	DAR	DARv2	Δ	95% CI	p-value
R@5	76.11	77.38	+1.27	[-2.49, 5.09]	0.2571
nDCG@5	73.29	74.94	+1.65	[-1.87, 5.25]	0.1844
MRR@5	76.50	76.88	+0.38	[-3.87, 4.72]	0.4350
MAP@5	68.56	70.09	+1.53	[-2.42, 5.48]	0.2232
R@20	83.16	85.88	**+2.71**	[0.22, 5.33]	**0.0169***
nDCG@20	74.59	75.97	+1.38	[-1.66, 4.38]	0.1830
MRR@20	77.26	77.55	+0.28	[-3.81, 4.45]	0.4517
MAP@20	68.50	69.28	+0.78	[-2.85, 4.33]	0.3339
R@60	88.25	91.64	**+3.38**	[1.19, 5.99]	**0.0014***
nDCG@60	76.05	77.15	+1.10	[-1.78, 3.96]	0.2250
MRR@60	77.29	77.64	+0.35	[-3.76, 4.51]	0.4380
MAP@60	68.92	69.35	+0.44	[-3.12, 3.92]	0.4066
R@100	91.18	93.79	**+2.61**	[0.16, 5.45]	**0.0274***
nDCG@100	76.86	77.80	+0.93	[-1.89, 3.71]	0.2555
MRR@100	77.30	77.65	+0.35	[-3.74, 4.48]	0.4394
MAP@100	69.10	69.52	+0.42	[-3.13, 3.89]	0.4112

Table 14: Paired statistical comparison between DAR and DARv2 on the JuriFindIT validation set. Scores are percentages. Confidence intervals are 95% paired bootstrap intervals over validation queries. p-values are computed with a one-sided paired approximate randomization test using 10,000 sign permutations. * indicates $p < 0.05$.

Model	R@5	MAP@5	MRR@5	nDCG@5
DARv2	77.38	70.09	76.88	74.94
Qwen3-Reranker-0.6B	73.73	64.78	73.63	70.03
Qwen3-Reranker-4B	**79.23**	74.78	85.59	79.51
Qwen3-Reranker-8B	78.63	**75.36**	**86.45**	**79.77**

Table 15: Pool-based reranking on the Top-20 candidates returned by DARv2. The DARv2 row reports the original ranking; Qwen3 reranker rows report effectiveness after reordering the same candidate lists.

the strongest overall performance, with Qwen3-Reranker-4B close behind, suggesting that both models are effective at refining the dense retrieval output. In contrast, the 0.6B variant consistently underperforms the original DARv2 ordering, showing that smaller rerankers may be unable to model the fine-grained legal relevance distinctions required in this setting and can even disrupt an already strong candidate ranking.

D Comparison between traditional and iterative retrieval

We compare a traditional reranker with an iterative LLM-based document selection strategy on the JuriFindIT validation set. In the traditional setting, the reranker scores and reorders the retrieved documents. In the iterative setting, the LLM receives documents in batches and decides which ones should be retained as relevant. We report the iterative results both in the unconstrained setting, where the model can select any number of documents, and in a constrained setting, where the final selection is capped at five documents.

Top-5 document selection: traditional and iterative retrieval. Table 16 compares dense retrieval, traditional reranking, and iterative LLM-based selection under the same top-5 constraint. In this setting, all methods return at most five documents, making the comparison directly aligned with the context size used before answer generation. The traditional reranker only marginally improves over dense retrieval, suggesting that the initial dense ranking is already strong. The iterative models, especially Qwen3.5-9B and Qwen3-8B, substantially improve precision while maintaining recall close to the traditional reranker. This indicates that iterative selection can produce a cleaner top-5 context, although the traditional reranker remains stronger in recall and offers a simpler, more stable pipeline.

Runtime also differs substantially across methods. Dense retrieval is the fastest option, while traditional reranking adds only moderate overhead. Iterative LLM-based selection is much slower, requiring roughly 45 minutes to 1.5 hours over the 179 validation queries. As suggested by Table 17, this cost is not determined only by model size: smaller models may iterate more often because they more frequently judge the current window insufficient, offsetting the expected speed advantage of fewer parameters.

Adaptive document selection over the top-20 retrieved candidates. In this setting, each iterative model receives up to 20 documents retrieved by

Model	Precision@5	Recall@5	Time
DARv2	21.46	77.38	∼1m:30s
Qwen3-4B Rr.	22.69	**79.23**	∼5m
Qwen3.5-9B	**63.81**	76.45	∼45m
Qwen3-8B	52.46	77.54	∼1h
Qwen3.5-4B	29.50	74.91	∼1h:30m
Gemma 4-E4B	49.85	75.74	∼1h:30m
Gemma 4-E2B	38.02	69.97	∼50m

Table 16: Top-5 document selection on the JuriFindIT validation set. We compare the original DARv2 ranking, traditional reranking (Rr.), and iterative LLM-based selection on the 20 documents pool retrieved by DARv2. Time denotes the approximate runtime over the 179 validation queries.

the dense retriever and can select any subset of them. This differs from the top-5 setting because the model is not only ranking documents, but also estimating how many legal texts should be retained.

Model	Precision@20	Recall@20	Insuf.
Qwen3-4B Rr.	8.85	85.88	-
Qwen3.5-9B	**61.97**	79.54	**7**
Qwen3-8B	51.09	79.28	15
Qwen3.5-4B	22.00	**80.54**	26
Gemma 4-E4B	55.24	78.27	24
Gemma 4-E2B	36.35	70.69	53

Table 17: Adaptive iterative document selection over the top-20 retrieved candidates. Rr. means traditional reranker. 'Insuff' indicates how many times the model deemed the text in the window insufficient for answering a question.

The results show that stronger models are better at controlling the size and quality of the selected context. Qwen3.5-9B provides the best trade-off, combining high precision with high recall and few insufficient cases. Qwen3-8B behaves more inclusively, preserving recall but selecting a noisier context. Qwen3.5-4B further illustrates this tendency: its high recall comes at the cost of very low precision, suggesting substantial over-selection. The Gemma models are less competitive overall, with Gemma 4-E4B showing reasonable performance and Gemma 4-E2B producing the weakest selection. Insufficiency decisions tend to be more frequent for smaller models, suggesting that limited model capacity may make it harder to determine whether the relevant legal evidence is already present in the top-10 window. Larger models appear more selective and less likely to request additional evidence.

Error in the number of selected documents. Table 18 reports how accurately each iterative model estimates the number of documents to retain. The average K error measures the absolute difference between the number of selected documents and the number of gold-relevant documents, while over- and under-retrieval errors distinguish whether the model selects too many or too few documents.

The results confirm that Qwen3.5-9B provides the best control over the size of the selected context, with the lowest average error and very few under-retrieval cases. Qwen3-8B is close, although it over-selects in more examples. Qwen3.5-4B shows the clearest over-selection pattern: it almost never selects too few documents, but it retains too many in most cases, explaining its high recall and low precision in Table 17. The Gemma models fall between these extremes, with Gemma 4-E4B showing moderate control and Gemma 4-E2B remaining less reliable overall.

Discussion. The iterative approach is useful because it can build a cleaner context for answer generation. In particular, Qwen3.5-9B greatly improves precision compared with the traditional reranker, while keeping recall close to the same level. This suggests that LLM-based selection can reduce noisy legal context before generation.

However, the traditional reranker remains simpler and more robust. It produces a deterministic ranking, is easier to evaluate with standard IR metrics, and avoids JSON parsing errors, iterative failures, and intermediate document-selection mistakes. It is also easier to batch and less expensive computationally.

For these reasons, although the iterative method is promising for adaptive context construction, the traditional reranker was preferred in the final pipeline because it offers a more stable, reproducible, and standard retrieval component.

E Confusion matrix evaluation

This appendix reports the confusion-matrix counts underlying the aggregate metrics discussed in § 7.2. While the main text focuses on precision, recall, and F1, the raw counts are useful to inspect the balance between false positives and false negatives, which have different implications in legal QA. In particular, false positives correspond to answers that the judge accepts although experts deem them incorrect, and are therefore the most safety-critical

Model	Avg. Error on K	Over Ret. Error	Over Ret. Cases	Under Ret. Error	Under Ret. Cases
Qwen3.5-9B	**2.32 ± 3.64**	4.40 ± 4.07	92	2.75 ± 1.26	4
Qwen3-8B	2.56 ± 3.47	**3.91 ± 3.72**	113	2.12 ± 0.83	8
Qwen3.5-4B	7.85 ± 4.75	8.55 ± 4.32	164	–	0
Gemma 4-E4B	3.06 ± 4.39	4.92 ± 4.84	107	**1.50 ± 0.65**	14
Gemma 4-E2B	3.14 ± 3.41	4.08 ± 3.45	133	1.90 ± 0.88	10

Table 18: Error in the number of selected documents for the iterative models. Experiments are computed on 179 samples.

Support	Expert=1	Expert=0
Main eval.		
Judge=1	129	19
Judge=0	24	9
Side eval.		
Judge=1	99	25
Judge=0	37	20

Table 19: Confusion-matrix counts comparing the judge against expert labels for main and side correctness.

errors. False negatives, by contrast, reflect conservative rejections and mainly affect coverage.

Table 19 provides the raw counts for the comparison between judge and expert labels on main and side correctness. Consistent with the aggregate metrics in the main text, the judge is more reliable on main content than on side information: the number of true positives is higher and the false-positive rate is lower relative to the more heterogeneous side-content setting. This supports the interpretation that the judge can more reliably assess the core legal claim than ancillary details.

Table 20 reports the corresponding counts when the judge is additionally provided with the expert-written reference answer and labelled articles. These counts clarify the different operating profiles of the evaluated models. Qwen3-32B achieves the strongest recall and the highest overall F1 by accepting most expert-correct answers, although it also produces more false positives than the most conservative alternatives. Gemma 4 31B yields the highest precision, with very few false positives and competitive recall, while Gemma 4 26B-A4B provides a strong balance between false positives and false negatives among the newly tested judges. GPT-oss variants and Qwen3.6 27B are also high-precision judges, but reject more expert-correct answers. Qwen3.6 35B-3A is the most conservative configuration, with the largest number of false negatives.

F Error Analysis on Model-Experts Agreement

We conducted a compact error analysis to better understand the disagreement patterns between the judge and expert annotators. The clearest signal is the role of retrieval recall. For *main* content, answers rated as incorrect by experts are associated with substantially lower average recall (approximately 0.5) than answers rated as correct (approximately 0.87). This suggests that, because the generator is explicitly constrained to rely on the retrieved statutory context, insufficient coverage of relevant articles directly harms answer quality. We observe the same trend for *side* content, where answers judged incorrect by experts have an average recall of about 0.55, compared with about 0.9 for correct answers.

A different pattern emerges for false negatives. Their average recall is comparable to that of true positives, both on main and side content. This indicates that these cases are less often due to missing evidence in the retrieved context, and more often to the judge being stricter than human annotators. In other words, low retrieval recall is primarily associated with genuinely poor answers, whereas false-negative judgments more typically reflect conservative model behaviour rather than lack of supporting statutory material.

Overall, the analysis points to retrieval coverage as the main upstream factor affecting answer correctness, while judge-specific errors are more often attributable to thresholding or calibration effects.

Qualitative error patterns. A qualitative inspection of annotator comments reveals a small number of recurring failure modes. The most frequent one is *incomplete legal grounding*: incorrect answers often omit decisive provisions, paragraphs, or legally relevant conditions, especially when the retrieved context does not contain the necessary source. In these cases, the model still produces a fluent answer, but lacks the normative basis needed

Support	Expert=1	Expert=0
Main eval. (Qwen3-32B)		
Judge=1	146	17
Judge=0	7	11
Main eval. (Qwen3.5-27B)		
Judge=1	123	8
Judge=0	30	20
Main eval. (Qwen3.5-27B, thinking)		
Judge=1	122	7
Judge=0	31	21
Main eval. (GPT-oss, thinking medium)		
Judge=1	121	10
Judge=0	32	18
Main eval. (GPT-oss, thinking low)		
Judge=1	126	10
Judge=0	27	18
Main eval. (Gemma 4 31B)		
Judge=1	134	6
Judge=0	19	22
Main eval. (Gemma 4 31B, thinking)		
Judge=1	129	7
Judge=0	24	21
Main eval. (Gemma 4 26B-A4B)		
Judge=1	137	10
Judge=0	16	18
Main eval. (Qwen3.6 35B-3A)		
Judge=1	103	7
Judge=0	50	21
Main eval. (Qwen3.6 27B)		
Judge=1	124	6
Judge=0	29	22

Table 20: Confusion-matrix counts for main correctness when the judge is provided with the expert-written reference answer and labelled articles.

to resolve the question, leading to partial or unsupported conclusions. This pattern also explains why feedback is unreliable when retrieval recall is zero: the judge can encourage revision, but cannot recover legal information absent from the input context.

A second common pattern is *misinterpretation of the applicable norm*, where the model cites a relevant provision but applies it incorrectly, narrows or broadens its scope, or reverses its legal effect. Annotators also report *hallucinated or unnecessary extensions*, i.e., answers that go beyond the provided legal basis by introducing speculative roles, conditions, or consequences not supported by the retrieved articles. Relatedly, several errors stem from *wrong legal qualification*, such as focusing on a non-pertinent institution, sanction, while missing the actually relevant legal basis. Finally, some answers are judged problematic because they are *too generic or evasive*: they restate parts of the retrieved text but fail to resolve the question concretely. These comments indicate that failures are driven less by surface-level phrasing than by missing or misapplied legal support, confirming that retrieval coverage is a prerequisite for effective generation and feedback.

G Prompt Templates

This appendix reports shortened English versions of the main prompts used in our pipeline. We omit repeated formatting instructions and minor implementation details, while preserving the core constraints that define each task.

G.1 Answer Generation Prompt

The generator is instructed to answer a legal question using only the retrieved statutory articles, with an emphasis on faithfulness, preservation of legal meaning, and explicit grounding in the provided documents.

> You are an expert in Italian law. Your task is to answer, in Italian, a legal question Q using **only** the relevant articles D provided in the context. Your answer must be legally precise, grounded in D, and free of unsupported additions or distortions.
>
> Follow these principles:
>
> **(1) Coverage and grounding.** Include every material element in D that directly answers Q. Derive the answer only from what is explicitly stated or clearly implied by D. If the articles are insufficient, state this explicitly.
>
> **(2) No contradiction.** Do not contradict the meaning, conditions, or consequences established in D. Do not mix unrelated provisions.
>
> **(3) Preserve legal force and precision.** Maintain the same deontic force (e.g., "must" vs. "may") and preserve quantitative or temporal details exactly.
>
> **(4) Preserve scope.** Do not alter who is bound, where the rule applies, when it applies, or the relevant exceptions and definitions.
>
> **(5) Faithful synthesis.** You may summarize or rephrase for clarity, but do not change the legal meaning. Do not add external facts, precedents, or interpretations not present in D.
>
> **(6) Transparency.** Explicitly mention the relevant legal facts and article identifiers when useful.
>
> If information is missing or ambiguous, do not speculate. Instead, state that the answer cannot be fully determined from the provided articles.
>
> **Input:** Question Q: `{question}`
> Relevant articles D: `{documents}`
>
> **Output:** Write a clear and legally precise answer in Italian that directly answers Q, remains strictly grounded in D, preserves legal meaning and scope, and explicitly refers to the relevant articles.

G.2 Judge Prompt for Feedback Generation

The judge evaluates whether a generated answer is faithful to the retrieved articles and, when necessary, provides targeted feedback for revision.

> You are an expert legal verifier. Your task is to determine whether an answer R faithfully reflects the content and legal meaning of the relevant articles D.
>
> You must identify which parts of R are unsupported, contradictory, or legally distorted with respect to D, and provide clear feedback so that the answer can be corrected.
>
> Evaluate according to the following criteria:
>
> **(1) Coverage.** Every relevant claim in R must be supported or clearly implied by D.
>
> **(2) No contradiction.** R must not contradict any provision, condition, or rule in D.
>
> **(3) Normative force and precision.** R must preserve deontic force and semantic polarity (e.g., "must" $\neq$ "may") and keep quantitative and temporal details unchanged.
>
> **(4) Scope.** R must not alter subjects, jurisdiction, validity period, exceptions, definitions, or conditions stated in D.
>
> **(5) Faithful integration.** Paraphrases or extensions are allowed only if they remain fully consistent with D and introduce no arbitrary information.
>
> Assign one of the following labels:
>
> - **COMPLIANT**: all criteria are satisfied;
> - **PARTIALLY_COMPLIANT**: criteria (1)–(4) hold, but the answer includes non-essential unsupported additions;
> - **NON_COMPLIANT**: one or more of criteria (1)–(4) are violated.
>
> If the answer is not fully compliant, provide specific feedback indicating what must be corrected or removed. Avoid generic comments; each observation should be grounded in D.
>
> **Output (JSON only):**
>
> { "evaluation": "COMPLIANT" | "PARTIALLY_COMPLIANT" | "NON_COMPLIANT",
> "feedback": "<targeted correction instructions>"
> }

G.3 Judge Prompt with Access to Expert Answers

In the reference-based setting, the judge compares the generated answer against the expert-written answer while consulting the legal documents.

> You are a meticulous legal verifier. Your task is to determine whether a generated answer G is **COMPLIANT** with a reference answer L, while consulting the relevant documents D.
>
> G is:
>
> - **COMPLIANT** if it is consistent with L and may summarize or extend it without changing its legal meaning, normative force, or scope;
> - **NON_COMPLIANT** if it contradicts, distorts, or substantially alters L, or adds incompatible legal content.
>
> Use the following criteria: **(1) No contradiction.** G must not contradict any conclusion in L.
> **(2) No variation in normative force.** G must preserve deontic force, polarity, and all numerical/temporal details.
> **(3) No variation in scope.** G must not alter subjects, jurisdictions, validity conditions, exceptions, or definitions expressed in L.
>
> General rules:
>
> - Consider both L and G in light of the documents D.
> - A claim in G that is absent from L may still be compliant if it is supported by D and compatible with L.
> - Focus on substantive legal meaning, not stylistic or citation differences.
>
> **Input:** Reference answer L: `{reference}`
> Generated answer G: `{generated}`
> Relevant documents D: `{documents}`
>
> **Output (JSON only):**
>
> { "evaluation": "COMPLIANT" | "NON_COMPLIANT",
> "reason": "<brief justification based on criteria (1)–(3)>",
> "evidence": {
> "contradictions": [...],
> "incoherences": [...],
> "supporting_facts": [...]
> }
> }

G.4 LLM-based Agreement Analysis Prompt

For the auxiliary LLM-based agreement analysis, the evaluators were asked to judge whether an expert-written answer correctly addressed the corresponding legal question. Unlike the pipeline judge, this prompt was used in an open-book setting: evaluators could rely on general legal knowledge and external legal sources, while avoiding unsupported factual assumptions beyond the question.

> You are an expert evaluator tasked with assessing the correctness of an answer to a legal question.
>
> Your task is to determine whether the provided answer is correct with respect to the question. Do not rewrite the answer, do not propose an alternative answer, and do not add unnecessary information.
>
> Evaluate the answer based on:
>
> 1. the question;
> 2. the provided answer;
> 3. your general legal knowledge;
> 4. any available external knowledge, including relevant statutory, case-law, doctrinal, or institutional sources.

The evaluation is open-book: you may refer to external knowledge or relevant legal sources to verify the correctness of the answer. However, you must not assume specific facts that are not stated in the question, and you must not introduce additional factual hypotheses.

Do not penalize the answer only because it is concise, unless concision omits legally essential elements. Penalize answers that contain substantive legal errors, misleading generalizations, relevant omissions, or unsupported claims.

Use exactly one of the following labels:

- **CORRECT**: the answer is legally correct with respect to the question and sufficiently complete to answer it;
- **PARTIALLY_CORRECT**: the answer contains legally correct information, but is incomplete, ambiguous, overly general, or omits relevant conditions, exceptions, or qualifications;
- **INCORRECT**: the answer contains a substantive legal error, answers the question incorrectly, or is misleading overall.

Return only valid JSON, with no additional text before or after it.

Input: Question: `{question}`
Answer: `{answer}`

Output (JSON only):

{
"label": "CORRECT" | "PARTIALLY_CORRECT" | "INCORRECT",
"confidence": 1,
"rationale": "<brief explanation, maximum two sentences>",
"critical_issue": "none" | "missing_condition" | "wrong_law" | "overgeneralization" | "unsupported_claim" | "ambiguity" | "other"
}

Field instructions:

- `label`: choose exactly one label among `CORRECT`, `PARTIALLY_CORRECT`, and `INCORRECT`;
- `confidence`: assign an integer from 1 to 5, where 1 indicates low confidence and 5 indicates high confidence;
- `rationale`: briefly explain the reason for the judgment, in at most two sentences;
- `critical_issue`: choose none if there is no relevant issue; otherwise choose the main category of the problem.

H Hardware and Software Stack

All experiments were run on a workstation equipped with an NVIDIA A100 GPU with 80GB of memory. We used PyTorch as the main deep learning framework and Hugging Face Transformers for model inference. The dense retriever was trained using the SentenceTransformers library, while datasets and experimental splits were managed through Hugging Face Datasets.

I Text Generation Parameters

For all LLM-based experiments, we used the decoding configurations recommended by the corresponding model authors in the Hugging Face model cards or official usage guides. The main generation parameters are summarized in Table 21. When supported by the model, we distinguish between thinking (T) and no-thinking (NT) modes.

For GPT-oss, the model card emphasizes the Harmony response format and configurable reasoning effort rather than fixed sampling parameters.

Model / setting	Temp.	Top-p	Top-k	Min-p
Qwen3 models (T)	0.6	0.95	20	0
Qwen3 models (NT)	0.7	0.80	20	0
Gemma 4 (T & NT)	1.0	0.95	64	–

Table 21: Main text-generation parameters used for the open-weight LLMs. T and NT denote thinking and no-thinking modes, respectively. Settings follow the corresponding model-card recommendations.

J Dataset License

JustifIT-QA will be released under the Creative Commons Attribution 4.0 International license (CC BY 4.0). The license applies to the original annotations introduced in this work, including expert-written answers, span-level evidence annotations, metadata, and dataset splits. Users may share and adapt these annotations provided that appropriate credit is given and changes are indicated.

The dataset builds on JuriFindIT and links examples to statutory provisions from the underlying legal corpus. The original legal texts remain subject to their own source terms and public-sector reuse conditions, which users should verify when redistributing modified versions or combining the dataset with external legal corpora. The resource is intended for research on legal retrieval and grounded QA, and not for providing legal advice.