

Evidence-Grounded Contract Clause Classification with Small Language Models

Dina Mohamed Ali Hager Hamed Abd Elrady Alaam Hayam Abdelbaset Rezk
Rahatara Ferdousi Mohammed Anber Abdelhamed

School of Computing, Queen's University, Kingston, Ontario, Canada
{dina.m.ali, hager.h.alaam, hayam.a.rezk,
rahatara.ferdousi, mohammed.anber}@queensu.ca

Abstract

Commercial contract clauses exhibit substantial linguistic variation and semantic overlap, making fine-grained classification challenging for small language models (SLMs). We investigate whether retrieving semantically similar labeled clauses can provide task-specific evidence at inference time and improve classification without modifying the underlying SLM. Using the Contract Understanding Atticus Dataset (CUAD), we formulate clause classification as a retrieval-conditioned prediction task in which relevant training clauses are retrieved and provided as in-context demonstrations to a locally deployed SLM. We evaluate prompting-only, retrieval-only, and retrieval-conditioned configurations on the same contract-disjoint held-out set of 1,679 clauses. The proposed retrieval-conditioned SLM achieves 66.41% accuracy and 51.72% macro-F1, compared with 35.80% and 31.41% for the zero-shot SLM and 50.98% and 40.93% for the random few-shot SLM, respectively. Retrieval-only nearest-neighbor baselines achieve 47.48–47.82% macro-F1, while conditioning the SLM on the same retrieved evidence increases macro-F1 to 51.72%. These results show that retrieval provides substantial task-specific context and that SLM conditioning provides additional improvement beyond direct nearest-neighbor label assignment, without task-specific parameter fine-tuning. The findings support retrieval-conditioned SLMs as a promising direction for evidence-grounded and locally deployable legal NLP.

Keywords: Legal NLP, Contract Review, CUAD, Retrieval-Augmented Generation, Clause Classification, Evidence Grounding, Vector Retrieval, Qdrant, Small Language Models

1 Introduction

Commercial contracts encode rights, obligations, restrictions, and risk allocations through language

that is both domain-specific and highly variable. Automatically identifying such provisions is therefore a central problem in legal NLP, with applications to contract review, due diligence, compliance analysis, and legal information extraction. Benchmarks such as the Contract Understanding Atticus Dataset (CUAD) (Hendrycks et al., 2021) have enabled systematic evaluation of models for identifying legally salient contract clauses, while subsequent work has explored contextual modeling and large language models (LLMs) for contract understanding.

Recent work has explored the use of large language models for contract analysis, including structured question answering over specific contractual clauses (Roegiest et al., 2023). However, legal documents pose particular challenges for language models due to specialized terminology, complex syntax, and long-context requirements (Hua et al., 2023; Blair-Stanek et al., 2024). Studies evaluating general-purpose LLMs on legal contract classification have also reported limitations compared with models adapted to the legal domain (Jayakumar et al., 2023). LLMs have also been applied to practical contract-analysis tasks such as comparing contracts against templates and identifying deviations between contractual provisions (Narendra et al., 2024). These limitations motivate investigating whether external contractual evidence retrieved from a legal knowledge base can improve the performance of a small language model (SLM) for clause classification.

Despite this progress, clause classification remains challenging for small, locally deployable SLMs. A target provision may be difficult to classify from its text and a set of abstract category names alone, particularly when clause categories are semantically related or expressed using heterogeneous contractual language. Larger models may encode more task-relevant knowledge in their parameters, but relying on increasingly large

models is not always desirable in legal settings where computational cost, data governance and local deployment are important. This motivates a different question: **can relevant contractual examples retrieved at inference time provide the task-specific evidence that an SLM lacks in its parametric representation?**

Retrieval offers a natural mechanism for addressing this problem. Rather than asking a classifier to infer a clause category in isolation, a retrieval-conditioned model can compare the target provision with semantically related, previously labeled contractual clauses. In this setting, retrieval is not merely a mechanism for locating documents. The retrieved examples provide instance-level task grounding by exposing the model to concrete realizations of legal categories at inference time. This perspective is particularly relevant to contract analysis, where similar legal functions can be expressed through substantially different lexical and syntactic forms.

Prior work has established the importance of evidence and contextual information for contract understanding. ContractNLI (Koreeda and Manning, 2021) explicitly associates contractual inference with supporting evidence, while ConReader (Xu et al., 2022) models relations extending beyond isolated clauses. Retrieval-augmented approaches have further demonstrated the value of grounding legal question answering in relevant legal provisions (Redelaar et al., 2024), while recent work has highlighted retrieval reliability as an important source of downstream error (Reuter et al., 2025). ContractEval further shows that LLM performance varies substantially across models on clause-level legal analysis (Liu et al., 2025).

However, less is known about the extent to which retrieved labeled contractual examples can improve fine-grained clause classification for capacity-constrained SLMs under a controlled inference setting. In particular, separating the contribution of retrieved evidence from changes in the underlying language model is important for determining whether improvements arise from retrieval itself.

We study this question through retrieval-conditioned clause classification on CUAD. We construct the retrieval corpus exclusively from training contracts and retrieve semantically similar labeled clauses for each unseen target clause. These examples are supplied as in-context demonstrations to a locally deployed Qwen2.5-1.5B classifier. To isolate the effect of retrieval, we com-

pare prompting-only, retrieval-only, and retrieval-conditioned settings while keeping the evaluation instances, label space and generation configuration fixed. Contracts are partitioned before clause-level restructuring, ensuring that retrieval cannot draw evidence from the contract containing an evaluation instance and annotation labels are excluded from retrieval representations. The key contributions of this work are as follows.

- We provide a controlled empirical study of retrieval-conditioned SLM classification for legal text, explicitly separating the contributions of random few-shot context, semantic retrieval, nearest-neighbor voting and SLM-based evidence conditioning under a contract-disjoint evaluation setting.
- We show that retrieval-conditioned SLM inference achieves 66.41% accuracy and 51.72% macro-F1, outperforming zero-shot, random few-shot, and direct retrieval baselines while keeping the underlying SLM unchanged.

All code, vector retrieval pipelines, and evaluation scripts are publicly available.¹ The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 presents the retrieval-conditioned methodology. Section 4 describes the experimental setup. Section 5 reports the results and Section 6 concludes the paper with limitations and future work.

2 Related Work

Prior work related to this study falls into three connected lines: evidence-centered contract understanding, retrieval-grounded legal NLP and LLM-based clause analysis. Table 1 summarizes the studies that directly inform our experimental setting.

These studies establish two properties that are central to our setting. First, contractual decisions often depend on evidence or context beyond the target clause itself (Hendrycks et al., 2021; Koreeda and Manning, 2021; Xu et al., 2022). Second, retrieval can provide such context but must itself be treated as an uncertain component of the pipeline: legal RAG has benefited from attributed retrieval (Redelaar et al., 2024), while retrieval failures can propagate to downstream predictions (Reuter et

¹<https://github.com/MohammedAnber/CUAD-Contract-Review-Baselines>

Table 1: Prior work on evidence, contractual context, retrieval, and language-model-based contract analysis.

Study	Task	Relevant Contribution
CUAD (Hendrycks et al., 2021)	Contract review	Expert-annotated clause-level contract review across 41 categories, providing a benchmark for identifying legally salient provisions.
ContractNLI (Koreeda and Manning, 2021)	Contract inference	Document-level inference paired with explicit identification of supporting contractual evidence.
ConReader (Xu et al., 2022)	Clause extraction	Models long-range context, term-definition relations, and related contractual provisions beyond the target clause.
Redelaar et al. (Redelaar et al., 2024)	Legal QA	Retrieval-augmented legal QA with attribution to supporting legal provisions.
Reuter et al. (Reuter et al., 2025)	Legal RAG	Studies retrieval reliability and identifies document-level retrieval mismatch as a source of downstream retrieval errors.
ContractEval (Liu et al., 2025)	LLM clause analysis	Benchmarks proprietary and open-source LLMs for clause-level legal-risk identification on CUAD.

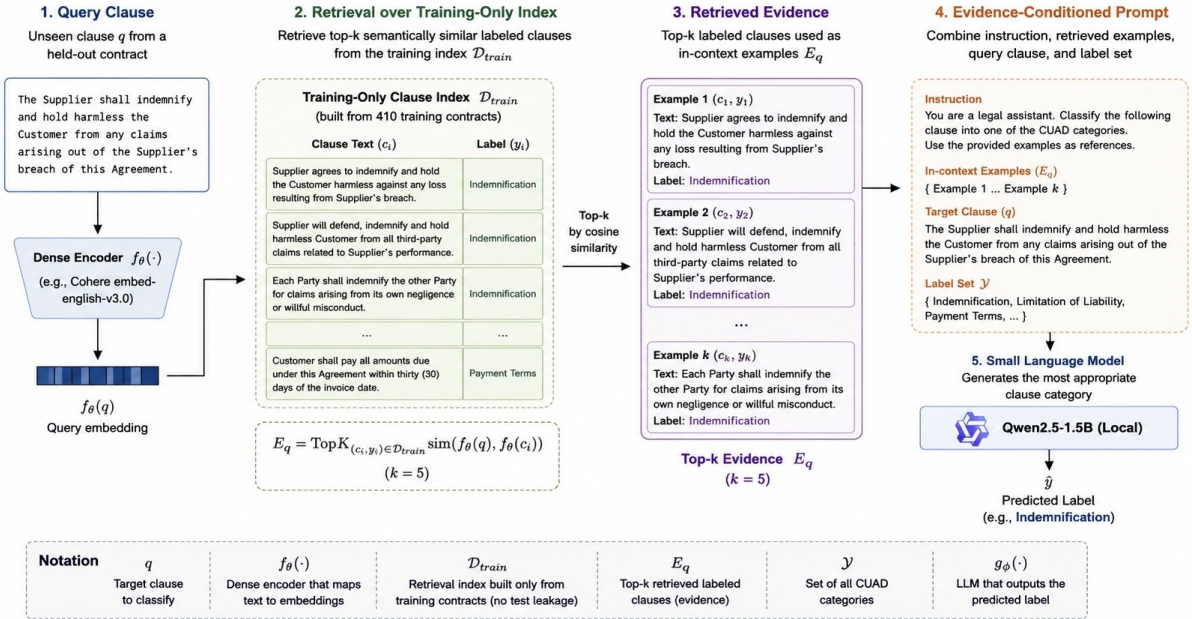


Figure 1: Overview of the retrieval-conditioned clause classification framework. A target clause is encoded and used to retrieve the top- k semantically similar labeled clauses from a training-only index. The retrieved clauses serve as in-context evidence for a small language model, which predicts the corresponding CUAD clause category.

al., 2025). ContractEval further shows that LLM performance varies substantially across models on clause-level legal analysis (Liu et al., 2025).

Taken together, these trends expose a specific gap: it remains unclear whether improvements in fine-grained contract classification can be obtained by supplying a capacity-constrained language model with relevant labeled contractual evidence at inference time, rather than increasing model capacity. Addressing this gap requires an experimental design that (i) retrieves evidence exclusively from training contracts, (ii) prevents document- and label-level leakage, (iii) evaluates retrieval-based classification independently from

evidence-conditioned prediction, and (iv) compares evidence-free and evidence-conditioned inference while holding the underlying classifier and evaluation setting fixed. Accordingly, we formulate clause classification as a retrieval-conditioned prediction problem, in which semantically related labeled clauses are retrieved from a contract-disjoint training corpus and supplied as in-context demonstrations to the same small language model. The following section describes this methodology.

3 Methodology

The proposed framework uses semantic retrieval to provide relevant contractual evidence before clause

classification. Given an input clause, the retrieval component identifies similar clauses from the training corpus, and Qwen2.5-1.5B uses these labeled examples to assign the final CUAD category. The separation between retrieval and prediction also permits the contribution of retrieved evidence to be evaluated independently.

3.1 Task Formulation

Let q denote a target contractual clause and $\mathcal{Y} = \{y_1, \dots, y_m\}$ the set of candidate clause categories. Standard clause classification predicts a label using only the target clause and the predefined label space:

$$\hat{y}_{\text{base}} = g_\phi(q, \mathcal{Y}), \quad (1)$$

where g_ϕ denotes the language-model classifier.

We instead formulate clause classification as *retrieval-conditioned inference*. Let $\mathcal{D}_{\text{train}} = \{(c_i, y_i)\}_{i=1}^N$ denote the training-only corpus of labeled contractual clauses, and let $f_\theta(\cdot)$ be a dense clause encoder. For a target clause q , the retrieval stage constructs an evidence set

$$E_q = \text{TopK}_{(c_i, y_i) \in \mathcal{D}_{\text{train}}} \text{sim}(f_\theta(q), f_\theta(c_i)), \quad (2)$$

where $\text{sim}(\cdot, \cdot)$ denotes cosine similarity and E_q contains the k highest-ranked labeled clauses.

The retrieved evidence is subsequently provided to the same classifier as in-context demonstrations:

$$\hat{y}_{\text{ret}} = g_\phi(q, E_q, \mathcal{Y}). \quad (3)$$

Thus, the prompting-only and retrieval-conditioned settings differ only in the availability of E_q ; the underlying classifier remains unchanged. This formulation allows us to test whether retrieved contractual examples provide instance-level task grounding beyond the parametric knowledge of a small language model.

Figure 1 illustrates the resulting framework, consisting of dense retrieval over a training-only clause index, construction of an evidence set from the top- k labeled clauses, and evidence-conditioned clause prediction.

3.2 Evidence Retrieval

The retrieval corpus $\mathcal{D}_{\text{train}}$ contains labeled clauses derived exclusively from the training contracts. Each clause c_i is mapped to a dense representation $f_\theta(c_i)$ and stored in a vector index. Importantly, clause labels and annotation answers are excluded from the embedding input, ensuring that retrieval is

based solely on contractual text. For a target clause q , we compute $f_\theta(q)$ and rank training clauses by cosine similarity:

$$s(q, c_i) = \frac{f_\theta(q)^\top f_\theta(c_i)}{\|f_\theta(q)\| \|f_\theta(c_i)\|}. \quad (4)$$

The top $k = 5$ clauses and their associated labels form the evidence set E_q used for downstream classification. Because the index contains only training contracts, no clause from a held-out contract can be retrieved as evidence.

3.3 Evidence-Conditioned Classification

The retrieved evidence set $E_q = \{(c_j, y_j)\}_{j=1}^k$ is incorporated into the classifier prompt as a set of labeled in-context demonstrations. Each retrieved pair provides a concrete example of contractual language associated with a candidate clause category. Given an instruction I , the permitted label set $\mathcal{Y}$, and target clause q , the evidence-conditioned prompt is defined as

$$P(q, E_q, \mathcal{Y}) = [I; E_q; \mathcal{Y}; q]. \quad (5)$$

The language model then predicts

$$\hat{y} = g_\phi(P(q, E_q, \mathcal{Y})), \quad (6)$$

where the output is restricted to a category in $\mathcal{Y}$. In contrast, the evidence-free baseline receives the same instruction, label space, and target clause but excludes E_q . This design enables the effect of retrieved contractual evidence to be assessed without modifying the parameters of the underlying classifier.

3.4 Controlled Evaluation and Leakage Prevention

To isolate the effect of retrieved evidence, we compare inference conditions using the same SLM:

$$\hat{y}_{\text{base}} = g_\phi(q, \mathcal{Y}), \quad (7)$$

$$\hat{y}_{\text{ret}} = g_\phi(q, E_q, \mathcal{Y}). \quad (8)$$

The model, evaluation instances, label space, and generation configuration are held fixed while the source of the contextual demonstrations changes.

To prevent information leakage, contracts are partitioned before clause-level restructuring, and the retrieval index is constructed exclusively from training contracts. Consequently, clauses from held-out contracts cannot appear as retrieved evidence. Annotation labels are also excluded from

the embedding representations, ensuring that retrieval similarity is determined only by clause text.

This controlled design separates the contribution of retrieval from changes to the underlying classifier and provides a direct test of whether retrieved contractual evidence improves clause classification.

4 Experimental Setup

In this section, we describe the dataset and experimental procedure in detail.

4.1 Data

We use CUAD v1, an expert-annotated benchmark for commercial contract review containing 510 contracts and 41 annotation categories (Hendrycks et al., 2021). These 41 categories comprise five contract-level metadata fields and 36 substantive clause categories. The five metadata fields are excluded from the classification label space, so the experimental task considers 36 clause categories.

To prevent document-level leakage, the dataset is partitioned at the contract level before clause-level restructuring. Using random seed 42, 410 contracts are assigned to training and 100 to the held-out set, with zero overlap between the two partitions. We treat each annotated clause as one classification instance, with its CUAD category assigned as the class label. The training contracts are then converted into clause-level records containing the source contract, clause category, clause text, annotation answer, and metadata flag. Annotation answers remain separate from clause text and are not used as embedding input. The resulting training representation contains 7,004 clause records, which form the retrieval corpus.

4.2 Experimental Configuration

All classification configurations are evaluated on the same 1,679 held-out clauses. The retrieval corpus contains clauses from the 410 training contracts only. Each test clause is encoded with embed-english-v3.0, and the five most similar training clauses are retrieved from a local Qdrant collection using cosine similarity.

The prompting-only and retrieval-conditioned configurations use the same Qwen2.5-1.5B model served through Ollama with deterministic generation (temperature=0). The zero-shot SLM baseline receives the target clause and the CUAD label set. The random few-shot SLM configuration receives randomly selected labeled examples, while

the proposed retrieval-conditioned configuration receives the five retrieved clauses and their labels as few-shot examples. Thus, the model, evaluation instances, label space, and generation configuration remain fixed while the source of the contextual demonstrations changes.

For the fine-tuned encoder baselines, BERT and LegalBERT were trained using the same CUAD-derived training partition and evaluated on the same 1,679-clause held-out test set. BERT uses bert-base-uncased, while LegalBERT uses nlpaueb/legal-bert-base-uncased; both models are fine-tuned for three epochs using class-weighted CrossEntropy loss.

4.3 Evaluation Protocol

Classification is evaluated using accuracy and macro-F1. The same 1,679 held-out clauses are used for the prompting-only, retrieval, and proposed RAG comparisons, allowing the effect of retrieved contractual context to be assessed under a matched evaluation protocol.

The retrieval baselines are evaluated using the same training-only retrieval corpus as the proposed method. Pure Top-1 retrieval assigns the label of the nearest retrieved clause directly. Majority-vote k -NN assigns the most frequent label among the five nearest clauses, while weighted k -NN uses similarity-weighted voting. The proposed RAG Few-Shot SLM instead supplies the five retrieved labeled clauses to the SLM as contextual demonstrations.

Macro-F1 is computed by first calculating the F1 score independently for each class and then taking the unweighted mean across the 36 clause categories. This metric is reported alongside accuracy to account for the class imbalance present in the CUAD-derived classification task.

5 Results

The results are presented in three groups: prompting-only approaches, retrieval baselines compared with the proposed retrieval-conditioned SLM, and fine-tuned parametric encoders.

5.1 Prompting-Only Approaches

Table 2 presents the prompting-only configurations evaluated on the same 1,679 held-out clauses.

The zero-shot SLM achieves 35.80% accuracy and 31.41% macro-F1. Providing randomly selected few-shot examples increases performance

Table 2: Prompting-only approaches without semantic retrieval on the 1,679 held-out clauses.

Methodology	Accuracy	Macro-F1
Zero-Shot SLM	35.80%	31.41%
Random Few-Shot SLM	50.98%	40.93%

to 50.98% accuracy and 40.93% macro-F1. This improvement indicates that example-based prompting can provide useful task context even when the demonstrations are not selected through semantic retrieval. However, random demonstrations do not explicitly target clauses that are semantically related to the test instance.

5.2 Comparison of RAG-based SLM with Baseline Models

The retrieval baselines and proposed method are compared in Table 3 using the same 1,679 held-out clauses.

To isolate the contribution of retrieval from that of the SLM, we added random few-shot prompting and three retrieval-only k -NN baselines. Random few-shot prompting improves macro-F1 from 31.41% to 40.93%, while retrieval-only methods reach 47.48–47.82%. Conditioning the SLM on retrieved evidence further increases macro-F1 to 51.72%, an improvement of 3.90–4.24 percentage points over the corresponding retrieval-only baselines. Thus, a substantial portion of the improvement originates from retrieval itself, while the SLM provides an additional gain beyond direct nearest-neighbor voting.

The retrieval-only experiments also clarify the distinction between nearest-neighbor classification and the proposed approach. In k -NN, the retrieved labels directly determine the prediction through Top-1, majority, or similarity-weighted voting. In our approach, retrieved clauses and their labels serve as evidence in the SLM context rather than determining the output directly. The 3.90–4.24 percentage-point macro-F1 improvement over the k -NN variants indicates that the SLM contributes additional classification capability beyond direct neighbor voting.

Relative to the zero-shot SLM baseline, the proposed method improves accuracy from 35.80% to 66.41% and macro-F1 from 31.41% to 51.72%. Relative to the random few-shot SLM, the corresponding improvements are 15.43 percentage points in accuracy and 10.79 percentage points in

macro-F1.

5.3 Fine-Tuned Parametric Encoders

As an additional reference point, we evaluated fully supervised parametric encoders using the same CUAD-derived training partition and the same 1,679 held-out clauses. The results are summarized in Table 4. Fine-tuned BERT achieves 71.47% accuracy and 64.08% macro-F1, while fine-tuned LegalBERT achieves 76.00% accuracy and 68.18% macro-F1.

Fine-tuned BERT and LegalBERT achieve higher absolute macro-F1 (64.08% and 68.18%, respectively), demonstrating the effectiveness of task-specific supervised optimization. However, this performance comes from updating model parameters using task-specific labeled training data. In contrast, our proposed RAG-SLM requires no task-specific parameter fine-tuning and performs adaptation at inference time by conditioning the model on retrieved labeled evidence. Consequently, newly available evidence can be incorporated into the retrieval store without retraining the SLM. This distinction is particularly relevant to our objective of developing a lightweight, locally deployable legal classification framework in which the evidence base can evolve independently of model parameters. Despite operating without task-specific fine-tuning, the proposed method achieves 51.72% macro-F1 and substantially improves over the zero-shot SLM (31.41%) and random few-shot SLM (40.93%). Thus, the proposed approach represents a complementary trade-off: fine-tuned encoders provide stronger absolute predictive performance, whereas retrieval-conditioned SLM inference provides greater flexibility for evidence-driven adaptation without repeated parameter updates.

5.4 Category-Level Recall Analysis

Category-level analysis. Figure 2 compares per-category recall for the k -NN Majority Vote baseline and the proposed RAG Few-Shot SLM across the 36 CUAD categories. Recall improved in 17 categories (47.2%), remained unchanged in 5 (13.9%), and decreased in 14 (38.9%). Improvements and declines occurred across categories with different levels of test support, while some low-support categories exhibited large percentage-point changes. We therefore interpret these results descriptively and avoid drawing conclusions from isolated per-category differences.

Table 3: Comparison of retrieval-free, retrieval-only, and retrieval-augmented baselines on the same 1,679-clause evaluation set.

Method	Retrieval	Inference	Accuracy (%)	Macro-F1 (%)
Zero-Shot SLM	No	Yes	35.80	31.41
Random Few-Shot SLM	Random	Yes	50.98	40.93
k NN Top-1	Yes	No	60.33	47.48
k NN Majority ($k = 5$)	Yes	No	63.43	47.52
Weighted k NN ($k = 5$)	Yes	No	63.43	47.82
RAG-based SLM (Proposed)	Yes	Yes	66.41	51.72

Table 4: Comparison with fine-tuned encoder baselines.

Method	Task-Specific Fine-Tuning	Inference-Time Evidence	Adaptation to New Evidence	Macro-F1 (%)
Fine-Tuned BERT	Required	No	Requires retraining	64.08
Fine-Tuned LegalBERT	Required	No	Requires retraining	68.18
Proposed	Not required	Retrieved labeled evidence	Update evidence store without parameter retraining	51.72

5.5 Error Analysis

The aggregate metrics show that retrieval-conditioned inference improves classification performance, but aggregate accuracy and macro-F1 do not fully describe how retrieved evidence affects individual clause categories.

The confusion matrices in Figure 3 and Figure 4 provide a category-level view of the prediction errors for the prompting-only baseline and the proposed retrieval-conditioned configuration. The comparison is used to examine whether retrieved contractual examples change the distribution of errors across CUAD categories and to identify categories that remain challenging despite the availability of additional contractual evidence.

The category-level error analysis is interpreted descriptively. In particular, retrieval may reduce errors for categories whose contractual language varies substantially across documents, because semantically related training clauses provide concrete examples of how the same category is expressed. At the same time, closely related contractual categories may remain difficult to distinguish when their retrieved examples share overlapping language. The confusion matrices therefore complement the aggregate results by showing how errors are distributed across the label space.

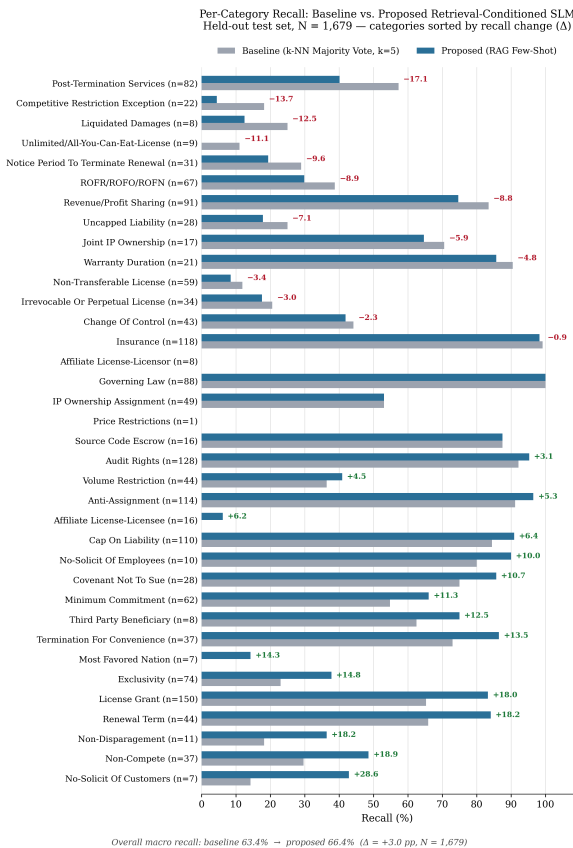


Figure 2: Per-category recall comparison between the k -NN Majority Vote baseline and the proposed RAG Few-Shot SLM, sorted by recall change (Δ).

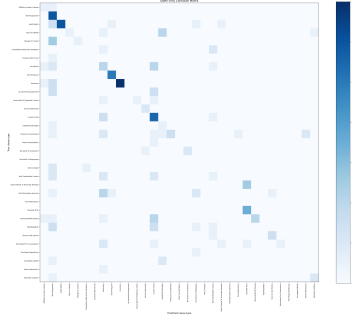


Figure 3: Confusion matrix for clause classification on the same held-out evaluation set using the zero-shot SLM without retrieved evidence.

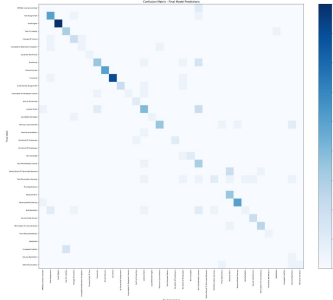


Figure 4: Confusion matrix for clause classification on the same held-out evaluation set using the proposed retrieval-conditioned SLM with five retrieved labeled contractual examples.

6 Conclusion

This work investigated whether retrieved labeled contractual examples can provide effective task-specific grounding for small language models in fine-grained contract clause classification. The findings demonstrate that retrieval-conditioned inference can substantially strengthen an SLM without task-specific parameter fine-tuning, and that retrieved evidence is most useful when incorporated as contextual demonstrations rather than used directly for nearest-neighbor label assignment. More broadly, the study highlights a complementary direction to parameter-based adaptation: keeping the language model fixed while allowing its evidence base to evolve independently. This flexibility is particularly relevant to legal NLP settings where local deployment, data governance, and efficient adaptation are important considerations. Overall, our findings establish retrieval-conditioned SLMs as a promising foundation for evidence-grounded legal language processing and motivate their evaluation across broader legal datasets and contract-analysis tasks.

Limitations

This study evaluates the proposed approach on a held-out set of 1,679 CUAD clauses. Although the contract-disjoint evaluation provides a controlled setting for measuring the effect of retrieved evidence, further evaluation on additional legal contract datasets would provide stronger evidence regarding generalization beyond CUAD.

The experiments also consider a fixed retrieval configuration using dense semantic retrieval with cosine similarity and $K = 5$. Accordingly, the reported results capture the effectiveness of the retrieval-conditioned framework under this setting; exploring alternative retrieval strategies and values of K remains an avenue for future work.

Finally, the current evaluation focuses on clause classification performance. The extent to which retrieval-conditioned small language models generalize to other contract analysis tasks and legal datasets remains to be investigated.

Acknowledgments

We would like to thank our industry partners at Capgemini Engineering, Eng. George Bishay and Eng. Ahmed Abdelghany, for their valuable collaboration, technical discussions, and insightful feedback throughout this project.

References

- Andrew Blair-Stanek, Nils Holzenberger, and Benjamin Van Durme. 2024. BLT: Can Large Language Models Handle Basic Legal Text? In *Proceedings of the Natural Legal Language Processing Workshop 2024*, pages 216–232. <https://aclanthology.org/2024.nllp-1.18/>. doi: 10.18653/v1/2024.nllp-1.18.
- Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. CUAD: An Expert-Annotated NLP Dataset for Legal Contract Review. In *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*, Datasets and Benchmarks Track. <https://arxiv.org/abs/2103.06268>. doi: 10.48550/arXiv.2103.06268.
- Wenyue Hua, Yuchen Zhang, Zhe Chen, Josie Li, and Melanie Weber. 2023. Mixed-domain Language Modeling for Processing Long Legal Documents. In *Proceedings of the Natural Legal Language Processing Workshop 2023*, pages 51–61. <https://aclanthology.org/2023.nllp-1.7/>. doi: 10.18653/v1/2023.nllp-1.7.
- Thanmay Jayakumar, Fauzan Farooqui, and Luqman Farooqui. 2023. Large Language Models are legal

- but they are not: Making the case for a powerful LegalLLM. In *Proceedings of the Natural Legal Language Processing Workshop 2023*, pages 223–229. <https://aclanthology.org/2023.nllp-1.22/>. doi: 10.18653/v1/2023.nllp-1.22.
- Yuta Koreeda and Christopher D. Manning. 2021. ContractNLI: A Dataset for Document-level Natural Language Inference for Contracts. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1907–1919. <https://aclanthology.org/2021.findings-emnlp.164/>. doi: 10.18653/v1/2021.findings-emnlp.164.
- Shuang Liu, Zelong Li, Ruoyun Ma, Haiyan Zhao, and Mengnan Du. 2025. ContractEval: Benchmarking LLMs for Clause-Level Legal Risk Identification in Commercial Contracts. In *Proceedings of the Natural Legal Language Processing Workshop 2025*. <https://aclanthology.org/2025.nllp-1.19/>. doi: 10.18653/v1/2025.nllp-1.19.
- Savinay Narendra, Kaushal Shetty, and Adwait Ratnaparkhi. 2024. Enhancing Contract Negotiations with LLM-Based Legal Document Comparison. In *Proceedings of the Natural Legal Language Processing Workshop 2024*, pages 143–153. <https://aclanthology.org/2024.nllp-1.11/>. doi: 10.18653/v1/2024.nllp-1.11.
- Felicia Redelaar, Romy van Drie, Suzan Verberne, and Maaïke de Boer. 2024. Attributed Question Answering for Preconditions in the Dutch Law. In *Proceedings of the Natural Legal Language Processing Workshop 2024*, pages 154–165. <https://aclanthology.org/2024.nllp-1.12/>. doi: 10.18653/v1/2024.nllp-1.12.
- Markus Reuter, Tobias Lingenberg, Rūta Liepiņa, Francesca Lagioia, Marco Lippi, Giovanni Sartor, Andrea Passerini, and Burcu Sayin. 2025. Towards Reliable Retrieval in RAG Systems for Large Legal Datasets. In *Proceedings of the Natural Legal Language Processing Workshop 2025*, pages 17–30. <https://aclanthology.org/2025.nllp-1.3/>. doi: 10.18653/v1/2025.nllp-1.3.
- Adam Roegiest, Radha Chitta, Jonathan Donnelly, Maya Lash, Alexandra Vtyurina, and François Longtin. 2023. Questions about Contracts: Prompt Templates for Structured Answer Generation. In *Proceedings of the Natural Legal Language Processing Workshop 2023*, pages 62–72. <https://aclanthology.org/2023.nllp-1.8/>. doi: 10.18653/v1/2023.nllp-1.8.
- Weiwen Xu, Yang Deng, Wenqiang Lei, Wenlong Zhao, Tat-Seng Chua, and Wai Lam. 2022. ConReader: Exploring Implicit Relations in Contracts for Contract Clause Extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2581–2594. <https://aclanthology.org/2022.emnlp-main.166/>. doi: 10.18653/v1/2022.emnlp-main.166.