A Framework to Retrieve Relevant Laws for Will Execution

Md Asiful Islam[1], Alice Saebom Kwak[2], Derek E. Bambauer[3], Clayton T. Morrison[4], Mihai Surdeanu[1]

[1] Department of Computer Science, University of Arizona
[2] Department of Linguistics, University of Arizona
[3] Levin College of Law, University of Florida
[4] College of Information Science, University of Arizona
{asifulislam, alicekwak, claytonm, msurdeanu}@arizona.edu, bambauer@law.ufl.edu

Abstract

Wills must comply with jurisdiction-specific statutory provisions to be valid, but retrieving the relevant laws for execution, validation, and probate remains labor-intensive and error-prone. Prior legal information retrieval (LIR) research has addressed contracts, criminal law, and judicial decisions, but wills and probate law remain largely unexplored, with no prior work on retrieving statutes for will validity assessment. We propose a legal information retrieval framework that combines lexical and semantic retrieval in a hybrid pipeline with large language model (LLM) reasoning to retrieve the most relevant provisions for a will statement. Evaluations on annotated will-statement datasets from the U.S. states of Tennessee and Idaho using six LLMs show that our hybrid framework consistently outperforms zero-shot baselines. Notably, when paired with our hybrid retrieval pipeline, GPT-5-mini achieves the largest relative accuracy gains, improving by 41.09 points on the Tennessee and 48.68 points on the Idaho test set. We observed similarly strong improvements across all models and datasets.

1 Introduction

A will is a legal document that articulates an individual's final intentions, including the distribution of assets, the administration of the estate, and appointment of guardians for dependents. For a will to be legally valid, its execution, validation, and probate must comply with the statutory provisions of the governing jurisdiction (Moy, 2024). Validation generally requires confirming that the document satisfies formal legal criteria, such as being in writing, signed by the testator, and witnessed by competent individuals, and that real-world conditions, such as the testator's mental capacity or the eligibility of witnesses, align with statutory definitions (Langbein, 1974).

These requirements are codified in jurisdiction-specific statutes, and failure to comply with even a single provision may render a will partially or entirely invalid (Hirsch, 1996). Therefore, the first step in assessing validity is to retrieve the statutory provisions relevant to the specific will statement. This task is challenging because the applicable provisions are dispersed across large and complex collections of statutes covering a wide range of legal issues. Traditionally, this lookup process has been conducted manually by legal professionals, who must examine statutory codes to identify the relevant laws. Such manual retrieval is time-consuming, costly, and prone to error.

The growing digitization of legal texts and the adoption of computational methods in law present an opportunity to automate this process. Advances in artificial intelligence (AI) and natural language processing (NLP), particularly in information retrieval (IR), have demonstrated strong performance across diverse legal tasks (Quevedo et al., 2024). However, the domain of wills and probate law remains largely underexplored. To the best of our knowledge, no prior work has directly addressed the retrieval of statutory provisions specifically relevant to the validity of wills.

Motivated by this gap, we propose a legal information retrieval framework that automatically retrieves relevant statutory provisions to verify the validity of will statements. Our approach integrates lexical and semantic retrieval in a hybrid pipeline, enhanced with large language model (LLM) reasoning, and is evaluated on comprehensive statutory corpora and annotated datasets of real-world will statements. This work makes two key contributions:

(1) We present a framework for retrieving statutory provisions relevant to will validation by formulat-

Code and dataset are available at https://github.com/asiful109/will-law-retrieval

Proceedings of the Natural Legal Language Processing Workshop 2025, pages 338–350

ing the task as an open-domain legal information retrieval problem. We use this framework in a setting that, to our knowledge, has not been explored before, making both our problem formulation and application domain (wills and probate law) novel.

(2) We evaluate this framework on two public-domain datasets of will statements from U.S. jurisdictions (Tennessee and Idaho) and compare its performance against traditional information retrieval baselines. Experimental results show that our method significantly outperforms these baselines, underscoring its effectiveness in automating the retrieval of statutory provisions for will validation.

2 Related work

Legal information retrieval (LIR) has been widely applied across diverse legal tasks, including case law retrieval for identifying relevant precedents (Cao et al., 2024), statutory law retrieval for finding applicable statutes or regulations (Louis et al., 2023), contract clause analysis for extracting legal obligations such as confidentiality and termination (Wang et al., 2025), regulatory compliance by linking business activities to statutory requirements (Sun et al., 2025), and legal question answering (Hu et al., 2025).

Early LIR systems relied on lexical retrieval techniques such as TF–IDF (Salton and Buckley, 1988) and BM25 (Robertson and Zaragoza, 2009), as well as symbolic legal ontologies (Benjamins et al., 2005). Although effective for exact keyword matching, these approaches struggled with lexical variation and synonymy (e.g., "minor heir" versus "underage beneficiary"), limiting their robustness in complex legal tasks (Chen et al., 2013; Murata et al., 2005; Saravanan et al., 2009). These challenges motivated a shift toward semantic and neural retrieval methods. The advent of transformer-based encoder models such as BERT (Devlin et al., 2019) and legal-domain variants like LEGAL-BERT (Chalkidis et al., 2020) revolutionized legal retrieval by enabling context-aware embeddings. Transformer based encoder models now consistently outperform traditional lexical baselines like BM25 and TF–IDF across legal retrieval tasks (Rabelo et al., 2020).

The arrival of Large Language Models (LLMs) has elevated the legal domain to new heights due to their strong ability to understand and reason with complex legal language. One approach is domain-adapted legal LLMs such as SaulLM (Colombo et al., 2024) and DeepLegal-CN (Guo, 2025), which improve legal reasoning through targeted pretraining and fine-tuning. Another increasingly popular technique is Retrieval Augmented Generation (RAG), where relevant external knowledge, such as statutes, court cases, or legal precedents, is retrieved and incorporated into the input context before the LLM generates a response. Several domain-specific RAG variants have been proposed to better serve the legal information retrieval tasks. For instance, CBR-RAG (Wiratunga et al., 2024) incorporates case-based reasoning to retrieve precedent cases for legal question answering, while UniLR (Li et al., 2025) introduces a unified retriever for multiple legal retrieval tasks using attention supervision and knowledge graphs. Eval-RAG (Ryu et al., 2023) adapts the RAG framework to improve the evaluation of LLM outputs by comparing them with retrieved legal references. HyPA-RAG (Kalra et al., 2024) introduces parameter-adaptive control to handle dynamically changing legal and policy environments.

Despite these advancements, legal will verification remains an underexplored area in legal NLP. Prior work by Kwak et al. (2022) introduced the first datasets and models for this task, framing will validation as a supervised natural language inference (NLI) problem, analyzing cross-jurisdictional transferability (Kwak et al., 2023a), and exploring prompt-based extraction and structured information annotation from wills (Kwak et al., 2023b, 2024). These studies laid important groundwork for understanding the linguistic and legal complexities of wills, but they primarily operated over small, curated, human-annotated law sets and assumed access to the relevant laws at training and inference time. Their models were designed to classify a triplet (will statement, condition, and relevant law) into support, refute, or unrelated categories. So, their model needs to know the relevant law ahead of time and does not handle retrieving relevant laws; instead, it simply classifies whether a given law supports, refutes, or is unrelated.

In contrast, we formulate will validation as an open-domain retrieval task: given a will statement and its associated condition, the goal is to identify the most relevant provision from the full statutory corpus of a jurisdiction. This retrieval-centric formulation better reflects real-world scenarios, where the applicable law must first be retrieved before any reasoning can be applied. To the best of our

Will statement	Condition	Relevant law	Type
The foregoing instrument, consisting of four (4) pages, including the page signed by the undersigned witnesses, was, on the thereof signed, published and declared by the above-named [Person-1], to be his Last Will and Testament, in the presence of us, who, at his request and in his presence and in the presence of each other, and on the same date, have subscribed our names as witnesses thereto.	Two or more eligible witnesses have witnessed the testator signing his/her will and signed their names in the presence of the testator and in the presence of each other.	15-2-502. EXECUTION. Except as provided for holographic wills, writings within section 15-2-513 of this part, and wills within section 15-2-506 of this part, or except as provided in section 51-109, Idaho Code, every will shall be in writing signed by the testator or in the testator's name by some other person in the testator's presence and by his direction, and shall be signed by at least two (2) persons each of whom witnessed either the signing or the testator's acknowledgment of the signature or of the will.	**support**
	One out or two witnesses was under 18 years old at the time of the execution.	15-2-505. WHO MAY WITNESS. (a) Any person eighteen (18) or more years of age generally competent to be a witness may act as a witness to a will. (b) A will or any provision thereof is not invalid because the will is signed by an interested witness.	**refute**

Table 1: Example of a will statement with a condition and relevant laws. A law is considered relevant if it either supports or refutes the will statement for a given condition. If a law neither supports nor refutes a will statement–condition pair, it is considered unrelated. The example is taken from the Idaho will validity dataset introduced by Kwak et al. (2023a).

knowledge, we are the first to propose an automated framework for retrieving relevant laws to support will statement validation. This represents a novel and practical step toward AI-assisted will validation and advances robust legal information retrieval in the domain of wills and probate law.

3 Task description

To determine the validity of a will, two aspects must be considered: (i) whether the statements within the will comply with the statutory laws of the jurisdiction, and (ii) whether the external conditions related to the individuals involved in the will (e.g., the testator, beneficiaries, executor, or witnesses) satisfy the legal requirements. Table 1 illustrates this with an example. The will statement in the example specifies that the will was signed by witnesses. The "Relevant law" column lists two laws from the U.S. state of Idaho that must be satisfied for the will to be valid. Under Idaho law, a valid will requires at least two witnesses (Idaho Code 15-2-502), and each witness must be at least 18 years old at the time of execution (Idaho Code 15-2-505). These requirements represent external conditions that may not be explicitly stated in the will itself but are nonetheless necessary to establish its validity. Thus, by considering both the will statement and its associated external condition, one can identify the statutory laws that govern the validity of the will. To support automated will validation, we propose an information retrieval (IR) framework that, given a will statement and its associated condition, retrieves the statutory laws necessary to evaluate the validity of the statement.

Formally, let w denote a will statement (a short excerpt from a will), c denote a condition (a real-world external scenario related to the validity of the will statement), and $\mathcal{L} = \{l_1, l_2, \ldots, l_n\}$ denote the set of statutory laws of a given jurisdiction that govern the validity of wills, where each l_i corresponds to a statutory provision. Given the input pair (w, c), the objective is to select the law $l^\star \in \mathcal{L}$ that is most relevant for assessing the validity of w under condition c. A law $l \in \mathcal{L}$ is considered *relevant* to (w, c) if it either supports or refutes the pair, while laws that do not address the subject matter of (w, c) are considered *unrelated*.

4 Proposed method

We propose an information retrieval framework that retrieves the laws necessary to assess the validity of will statements. Our framework combines hybrid retrieval (which integrates keyword-overlap based lexical search with embedding-similarity based semantic search), with LLM based reasoning, and operates in three steps: (i) preprocessing the law dataset, (ii) retrieving top K candidate laws using a hybrid search strategy, and (iii) selecting the single most relevant law with a large language model (LLM). An overview of the framework is shown in Figure 1, and a detailed description of each component is provided below.

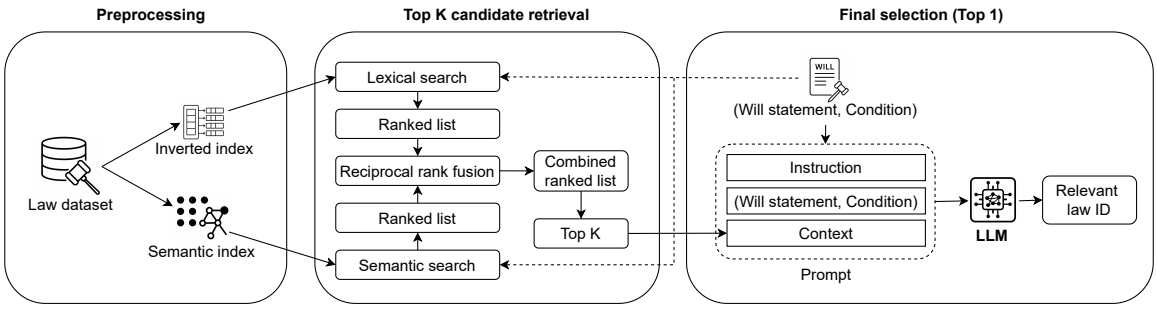

Figure 1: An overview of our proposed method for retrieving the most relevant law. In the first step, we preprocess the law dataset to create an inverted index and a semantic index. In the second step, we apply a hybrid search strategy to extract the top K candidate laws. In the third step, we use an LLM to select the most relevant law from these candidates, using the top K as context.

4.1 Preprocessing

As a preprocessing step, we built a law dataset by extracting statutory provisions from the web using a custom web crawler. The law dataset contains the set of statutory laws $\mathcal{L}$ (defined in Section 3) from the jurisdiction that governs the validity of wills. Each entry includes both the law code (ID) and its corresponding text. Our method can generalize to any jurisdiction, provided that the relevant statutory laws from that jurisdiction are supplied. In this work, however, we evaluate it on two datasets constructed from the statutory laws of two U.S. jurisdictions. A detailed description of these two datasets is provided in Section 5.1.2.

From the extracted provisions, we constructed two indices: an inverted index and a semantic index. The inverted index supports lexical search by mapping each term to the list of law provisions where it appears. The semantic index, on the other hand, encodes each provision into a dense vector representation using a transformer-based model, allowing retrieval based on semantic similarity rather than exact word matches. Together, these indices enable complementary search capabilities that are later combined in our hybrid retrieval step. The exact tools and models used to build the inverted and semantic indices are described in Section 5.2.

4.2 Top K candidate retrieval

Given a will statement and condition pair (w, c), our objective in this step is to retrieve the top K most relevant candidate laws from the set $\mathcal{L}$. To achieve this, we adopt a hybrid retrieval strategy that combines the strengths of lexical search and semantic search.

The lexical search operates over the inverted index and ranks laws using the BM25 scoring function (Robertson and Zaragoza, 2009). BM25 is a probabilistic ranking function that estimates the relevance of a law to a query by combining term frequency, inverse document frequency, and document length normalization. Given a query $q = (w, c)$, BM25 is applied to the inverted index to produce a ranked list of laws, where higher scores correspond to stronger lexical matches between the statutory provisions and the will statement–condition pair.

We apply semantic search over the semantic index, where each law provision is encoded into a dense vector representation using a transformer-based encoder model. We take the output from the final encoder layer and apply mean pooling across all token embeddings to obtain a single fixed-size vector for each law provision. If a law provision exceeds the model's maximum context length, it is truncated to fit within the limit before encoding. We compute the embedding of the query $q = (w, c)$ using the same encoder and pooling strategy as the law provisions and measure their similarity using cosine similarity. This produces a ranked list where higher scores correspond to stronger semantic similarity.

We query both the lexical index and the semantic index independently with (w, c) and then combine their ranked lists using Reciprocal Rank Fusion (RRF) (Cormack et al., 2009). RRF is a rank aggregation method that merges results from multiple retrieval systems by assigning higher scores to items that appear closer to the top in any list. We adopt RRF as our aggregation method since it is simple, unsupervised, and has been shown to outperform alternatives such as Condorcet and CombMNZ (Cormack et al., 2009). Supervised alternatives, such as learning-to-rank and meta classifier based fusion, require training a reranking model on labeled data

that captures ranking quality. Since obtaining such labels requires manual expert annotation, which is both expensive and time-consuming, we leave the exploration of these approaches to future work. Our experimental results (see Section 6) show that RRF delivers consistently strong performance, making it a highly effective choice in this setting.

Formally, the RRF score of a law l_i is computed as:

$$\text{RRF}(l_i) = \sum_{s \in S} \frac{1}{d + \text{rank}_s(l_i)}$$

where S is the set of retrieval systems (e.g., lexical and semantic), $\text{rank}_s(l_i)$ is the rank position of law l_i in system s, and d is a constant (typically set to 60) that controls the influence of lower-ranked items. This hybrid fusion strategy yields a ranked list of top K candidate laws that combine the precision of keyword-based retrieval with the generalization power of semantic similarity. Using this approach, we ensure that the candidate set for each input pair (w, c) captures both explicitly mentioned and implicitly related statutory provisions.

4.3 Final selection (top 1)

The goal of the final stage is to select the single most relevant law from the top K candidate set for the input pair (w, c). We formulate this step as a retrieval-augmented generation (RAG) problem, where the top K candidate laws are provided as context to a large language model (LLM).

We build a prompt for the LLM by combining (i) a task-specific instruction, (ii) the will statement–condition pair (w, c), and (iii) the top K candidate laws (IDs and law text) retrieved in the previous step. The prompt instructs the LLM to analyze the top K candidate laws and output the ID of the single most relevant law. Appendix B provide the full prompt template that we used for our experiment.

This approach addresses the limitations of alternative methods such as zero-shot prompting or hybrid retrieval. A zero-shot LLM without a retrieval context may hallucinate legal knowledge, rely on outdated information, or fail to align with jurisdiction-specific laws. Alternatively, hybrid retrieval allows us to control the data source, ensuring that searches rely on up-to-date and jurisdiction-specific laws. While hybrid retrieval is effective at producing a strong candidate set, it often fails to rank the single most relevant law at the top. This limitation arises because lexical search relies on surface-level similarity, whereas semantic search captures broader meaning but still falls short of handling the nuance and complexity of legal language. As a result, the correct provision may appear in the candidate set but not as the highest-ranked law.

To overcome this, we adopt the RAG formulation introduced above, which combines the complementary strengths of hybrid retrieval and LLM reasoning. Hybrid retrieval provides a small candidate set from the statutory laws of the target jurisdiction, while the LLM identifies the single most relevant law by disambiguating subtle differences and interpreting nuanced legal language.

5 Experiment setup

5.1 Datasets

5.1.1 Will statement dataset

We evaluate our framework on two datasets introduced by Kwak et al. (2022, 2023a), which contain legal wills from the U.S. states of Tennessee and Idaho. Both datasets originate from the public-domain U.S. Wills and Probates dataset from Ancestry[1]. The authors followed the same construction methodology for both datasets. They restricted the datasets to typewritten wills executed on or after 1970 and probated on or after 2000.

The dataset is annotated by two students (a law student and a student from another department) under the supervision of a law professor. They (1) extracted will text via OCR and segmented it into statements; (2) mapped each statement to five state laws (one supporting, one refuting, and three unrelated); (3) added hypothetical external conditions that altered whether a law supports, refutes, or is unrelated; and (4) anonymized all personally identifiable information.

The annotators achieved high inter-annotator agreement, with Cohen's kappa scores of 0.91 for Tennessee and 0.89 for Idaho. The datasets include 1,014 and 609 annotated statements, respectively, and use standard train/dev/test splits. Since our framework focuses on retrieving relevant laws, we evaluate only on statements labeled *support* or *refute*, excluding unrelated cases. Appendix A provides full dataset statistics.

5.1.2 Law dataset

The will statement datasets of Kwak et al. (2022, 2023a) include only the laws mapped to statements

[1] https://www.ancestry.com/search/categories/us_willsprobate

Method	refute					support					overall				
	K=1	K=5	K=10	K=20	K=40	K=1	K=5	K=10	K=20	K=40	K=1	K=5	K=10	K=20	K=40
Lexical Search	15.09	33.96	37.74	45.28	49.06	20.63	44.44	49.21	57.14	63.49	18.10	39.66	43.97	51.72	56.90
Semantic Search	16.98	37.74	41.51	62.26	**75.47**	17.46	36.51	53.97	71.43	80.95	17.24	37.07	48.28	67.24	78.45
Hybrid Search	**28.30**	**45.28**	**60.38**	**66.04**	**75.47**	**30.16**	**57.14**	**74.60**	**80.95**	**88.89**	**29.31**	**51.72**	**68.10**	**74.14**	**82.76**

Table 2: This table compares the performance of three methods for top K candidate retrieval on the **Tennessee test set**. Values show $Recall@K$ for *refute*, *support*, and overall (computed over the full test set). Hybrid search consistently outperforms lexical and semantic search, demonstrating the benefit of combining semantic and lexical search.

Method	refute					support					overall				
	K=1	K=5	K=10	K=20	K=40	K=1	K=5	K=10	K=20	K=40	K=1	K=5	K=10	K=20	K=40
Lexical search	**29.03**	48.39	61.29	74.19	**83.87**	18.75	46.88	**68.75**	71.88	78.12	**23.81**	47.62	65.05	73.02	80.95
Semantic search	16.13	41.94	51.61	64.52	77.42	15.62	40.62	43.75	59.38	**81.25**	15.87	41.27	47.62	61.90	79.37
Hybrid search	25.81	**70.97**	**77.42**	**83.87**	**83.87**	**21.88**	**50.00**	**68.75**	**75.00**	**81.25**	**23.81**	**60.32**	**73.02**	**79.37**	**82.54**

Table 3: Top K candidate retrieval performance on the **Idaho test set**. Values show $Recall@K$ for *refute*, *support*, and overall (computed over the full test set). Overall, hybrid search outperforms or matches both lexical and semantic search. Looking at *refute* and *support* separately, hybrid search outperforms or matches in all cases except one ($K = 1$) case of refute.

and not the complete set of statutes on wills and probate in Tennessee and Idaho. To support a more realistic evaluation, we built two comprehensive law datasets, one for Tennessee and one for Idaho, covering all titles that contain provisions related to wills and probate. We identified Titles 30, 31, 32, 35, and 40 of the Tennessee Code and Titles 15 and 68 of the Idaho Code as containing at least one section that addresses wills or probate. Although not every section under these titles is directly relevant to will and probate, including all sections creates a larger pool of laws, making the retrieval task more challenging and realistic. We built a custom web crawler to extract the laws from the 2024 versions of the *Idaho Code*[2] and the *Tennessee Code*[3]. We obtained both codes from Justia, a publicly accessible legal resource that permits web crawling. The final Tennessee dataset consists of 1,579 statutory provisions, whereas the Idaho dataset consists of 676.

5.2 Implementation details

We implemented the inverted index using Elasticsearch[4], and the semantic index using FAISS (Douze et al., 2025). For semantic embeddings, we use the `Stern5497/sbert-legal-xlm-roberta-base` model[5] from HuggingFace.

For reciprocal rank fusion we followed the original formulation and set the parameter d to its default value of 60. For top K retrieval, we tuned the hyperparameter K on the training split of the Tennessee dataset and found that $K = 20$ yielded the best performance on the downstream top 1 retrieval task. We fixed this value for all experiments on the test partitions of both the Tennessee and Idaho datasets. For the final top 1 selection, we set the temperature to 0 for all LLMs that support temperature control.

6 Results and analysis

6.1 Top K candidate retrieval performance

We evaluate three retrieval strategies for top K candidate law selection: lexical search, semantic search, and our proposed hybrid search method. Details of these methods are provided in Section 4.2. Our method adopts hybrid search as the top K retrieval strategy, with lexical and semantic search serving as baselines for comparison. Tables 2 and 3 present the results for various values of K, with performance reported separately for the *refute* and *support* subsets and for the full test set (*overall*). Since all methods are deterministic, a single run suffices.

The tables report $Recall@K$, which measures whether the gold (i.e., ground truth) law appears

[2] https://law.justia.com/codes/idaho/2024
[3] https://law.justia.com/codes/tennessee/2024
[4] https://www.elastic.co/elasticsearch
[5] https://huggingface.co/Stern5497/sbert-legal-xlm-roberta-base

Method	Model	Tennessee test set			Idaho test set		
		refute	support	overall	refute	support	overall
Baseline (Zero-shot)	Llama-3.1-8B	0.00 ± 0.00	1.59 ± 0.00	0.86 ± 0.00	3.23 ± 0.00	3.12 ± 0.00	3.17 ± 0.00
	SaulLM-54B	3.14 ± 1.09	5.82 ± 2.43	4.60 ± 0.99	7.53 ± 3.06	5.21 ± 3.06	6.35 ± 0.00
	GPT-4o-mini	11.32±1.89	14.82±0.91	13.22±0.50	40.86 ± 1.52	35.42 ± 1.47	38.10 ± 1.29
	GPT-5-mini	13.21 ± 8.65	14.81 ± 3.30	14.08 ± 2.63	18.28 ± 3.72	17.71 ± 4.78	17.99 ± 0.92
	GPT-4o	49.06 ± 0.00	38.10 ± 0.00	43.10 ± 0.00	49.46 ± 4.93	48.96 ± 1.80	49.21 ± 3.18
	GPT-5	44.02 ± 3.93	43.38 ± 3.30	43.68 ± 3.48	73.12 ± 1.86	58.34 ± 1.81	65.61 ± 0.92
Our method (Hybrid RAG)	Llama-3.1-8B	25.79 ± 2.18	34.39 ± 1.84	30.46 ± 0.99	54.84 ± 3.23	40.62 ± 3.13	47.62 ± 1.59
	SaulLM-54B	24.53 ± 3.27	31.75 ± 3.18	28.45 ± 0.86	40.86 ± 4.93	36.46 ± 3.60	38.63 ± 0.91
	GPT-4o-mini	49.68 ± 2.18	**57.67 ± 2.42**	54.02 ± 0.50	62.37 ± 1.86	52.08 ± 1.80	57.14 ± 1.59
	GPT-5-mini	57.86 ± 1.09	52.91 ± 1.84	**55.17 ± 1.49**	75.27 ± 1.86	58.34 ± 1.81	66.67 ± 1.59
	GPT-4o	52.83 ± 1.89	47.62 ± 0.00	50.00 ± 0.86	67.74 ± 0.00	59.38 ± 0.00	63.49 ± 0.00
	GPT-5	**60.38 ± 1.89**	50.79 ± 1.59	**55.17 ± 1.73**	**76.34 ± 1.86**	**61.46 ± 1.80**	**68.78 ± 0.92**

Table 4: Final top 1 selection performance of our method on the **Tennessee** and **Idaho** test sets. The values represent accuracy ($Recall@1$). Results are reported as the mean ± standard deviation over three runs.

among the top K retrieved candidates. This metric is appropriate for our setting because the subsequent top 1 selection stage only requires the gold law to be present in the candidate pool, its exact position within the list is not critical. Thus, $Recall@K$ serves as a reliable indicator of retrieval quality.

Across both the Tennessee and Idaho test sets, hybrid search consistently outperforms the lexical and semantic search in the *overall* test sets. To quantify this improvement, we compute the average gains achieved by hybrid search over each baseline across all K values in the overall columns. On the Tennessee test set, hybrid search achieves average improvements of 19.14 points over lexical search and 11.55 over semantic search. On the Idaho test set, the corresponding improvements are 5.72 and 14.61 points, respectively. Similar trends are observed in the refute and support subsets.

At $K = 20$ (which we use as the candidate set size for the final top 1 selection stage), hybrid search retrieves the gold law in 74.14% of cases on the Tennessee test set and 79.37% on the Idaho test set (both overall). Considering that the Tennessee law dataset contains 1579 laws and the Idaho law dataset 676 laws, these results confirm that hybrid search effectively reduces the search space to a small candidate pool while maintaining high recall. Moreover, the consistently strong performance across both *refute* and *support* subsets indicates that hybrid search is robust and not biased toward any particular label category.

The superior performance of hybrid search demonstrates that lexical and semantic retrieval are complementary. Lexical search ensures precision by retrieving exact statutory matches, while semantic search enhances recall by identifying provisions expressed with different wording. Importantly, semantic retrieval does not fully subsume lexical retrieval, since exact statutory terms often carry binding legal significance that semantic similarity alone may overlook. Their combination therefore yields broader coverage and higher retrieval quality than either method alone.

6.2 Final top 1 selection performance

While top K retrieval ensures that the gold law is included in a candidate set, legal applications ultimately require top 1 selection, since practitioners must reference the exact statutory provision governing a will's validity for compliance and citation. We therefore evaluate the final top 1 law selection performance of our hybrid RAG method using six LLMs spanning diverse categories. Among open-source models, we consider Llama-3.1-8B (Grattafiori et al., 2024), a small general-purpose LLM, and SaulLM-54B (Colombo et al., 2024), a large legal-domain-adapted model. Both are used in their instruction-tuned versions. For closed-source models, we evaluate four GPT variants (GPT-4o-mini, GPT-5-mini, GPT-4o, and GPT-5)[6]. This setup enables comparisons across open vs. closed-source, small vs. large-scale, and general-purpose vs. legal-specialized LLMs.

As a baseline, we use the zero-shot setting, where the LLM directly predicts the most relevant law given only the will statement–condition pair, without any retrieved context. This evaluates the model's ability to rely solely on its internal (para-

[6] https://platform.openai.com/docs/models

metric) knowledge. We compare this to our proposed hybrid RAG method (Section 4.3), where the LLM is provided with the top $K = 20$ candidate laws retrieved via hybrid search as contextual input. The value of K is selected by hyperparameter tuning in the Tennessee training set. The full prompt template used for all LLMs is available in Appendix B.

Table 4 presents the results on both the Tennessee and Idaho test sets. The reported values are mean $\pm$ standard deviation of accuracy ($Recall@1$) across three runs. Across all models and both datasets, the hybrid RAG approach consistently outperforms the zero-shot baseline. Hybrid RAG substantially improves performance for both open-source models. Llama-3.1-8B improves by 29.6 points on the *overall* Tennessee test set and by 44.45 points on *overall* Idaho test set. SaulLM-54B improves by 30.65 and 32.77 points, respectively.

Among the GPT models, the smaller variants show the largest gains. GPT-4o-mini improves by 39.22 points on Tennessee and 19.04 points on Idaho, while GPT-5-mini achieves the highest overall improvement: 41.09 points on Tennessee and 48.68 points on Idaho. While the larger GPT variants already perform well in the zero-shot setting, hybrid RAG still leads to notable gains. GPT-4o improves by 6.99 points on *overall* Tennessee test set and 14.28 on Idaho, while GPT-5 gains 11.49 and 3.17 points, respectively. A likely reason for the smaller gains is that these stronger models already perform well in the zero-shot setting, potentially due to indirect exposure to the Tennessee and Idaho datasets, which were publicly available on GitHub prior to their training cutoff. Nevertheless, the improvements confirm that hybrid RAG remains beneficial even for strong LLMs.

These trends hold consistently across both the *support* and *refute* subsets. Overall, the results in Table 4 demonstrate that our hybrid RAG method significantly improves retrieval accuracy across LLM families and datasets.

6.3 Title–chapter retrieval performance

The statutory codes of Tennessee and Idaho are hierarchically organized into title–chapter–section structures. Titles represent broad legal domains, chapters denote specific subdomains within those titles, and sections correspond to individual statutory provisions. For instance, Tennessee Code 32-1-105 falls under Title 32 (Wills), Chapter 1 (Execution of wills), and Section 105 (Holographic will). This structure inherently clusters related provisions, allowing us to assess whether a retrieval method can at least localize laws to the correct domain, even when it fails to identify the precise section.

Method	Tennessee test set		Idaho test set	
	title	chapter	title	chapter
Zero-shot				
Llama-3.1-8B	15.81 ± 0.99	4.60 ± 0.50	38.10 ± 4.20	16.93 ± 3.30
SaulLM-54B	47.70 ± 5.05	26.15 ± 2.77	50.79 ± 3.18	39.68 ± 3.23
GPT-4o-mini	71.55±1.50	55.46±0.99	98.41 ± 0.00	90.48 ± 0.00
GPT-5-mini	73.27 ± 1.50	57.76 ± 3.45	97.35 ± 1.83	85.71 ± 3.18
GPT-4o	77.87 ± 1.32	**72.70 ± 1.32**	96.83 ± 1.59	88.89 ± 1.59
GPT-5	76.72 ± 1.73	70.69 ± 0.86	**97.36 ± 0.91**	**92.06 ± 0.00**
Hybrid RAG				
Llama-3.1-8B	74.42 ± 1.80	63.50 ± 0.50	95.24 ± 0.00	86.77 ± 2.42
SaulLM-54B	71.26 ± 2.77	62.07 ± 2.28	93.12 ± 0.92	82.54 ± 1.59
GPT-4o-mini	79.31 ± 1.49	69.54 ± 0.50	92.59 ± 0.92	88.89 ± 0.00
GPT-5-mini	80.74 ± 0.50	70.69 ± 0.00	93.12 ± 0.92	89.95 ± 0.92
GPT-4o	80.17 ± 1.73	72.41 ± 1.73	**97.36 ± 0.91**	89.42 ± 0.92
GPT-5	**81.03 ± 0.00**	71.84 ± 0.50	94.18 ± 0.92	88.89 ± 0.00

Table 5: Title-Chapter retrieval performance. The values represent accuracy ($Recall@1$). Results are reported as the mean $\pm$ standard deviation over three runs.

To support this analysis, we re-evaluated our framework by relaxing the evaluation criteria from exact section-level matches to title and chapter level matches. Table 5 presents title and chapter retrieval results. On the Tennessee test set, hybrid RAG achieves title-level accuracies between 71.26% and 81.03%, and chapter-level accuracies between 63.50% and 72.41%. On the Idaho test set, performance is even stronger: title-level accuracies range from 92.59% to 97.36%, and chapter-level accuracies from 82.54% to 89.95%. These results suggest that hybrid RAG consistently selects laws from the correct domain, even when it misses the exact provision.

Compared to their zero-shot counterparts, open-source models (Llama-3.1-8B and SaulLM-54B) show clear improvements in both title and chapter retrieval when using hybrid RAG. For the closed-source GPT models, the results are more mixed: hybrid RAG outperforms zero-shot prompting on the Tennessee test set, while zero-shot variants perform slightly better on the Idaho test set. However, as shown in Table 4, zero-shot models are less reliable in retrieving the exact section (law code), whereas hybrid RAG is effective not only in identifying the correct domain (title and chapter) but also in pinpointing the exact section. Since real-world legal applications require retrieval at the section level for compliance and citation, accurate section retrieval remains the most critical measure of system performance.

6.4 Error analysis

In this section, we outline several limitations related to both the models and datasets that may constrain the performance of our method. While our approach consistently enhances the accuracy of all LLMs compared to their zero-shot counterparts, the following factors contribute to remaining sources of error and variability:

SaulLM-54B underperforming Llama-3.1-8B: Despite its larger size and legal-domain continued pretraining, SaulLM-54B performs worse than Llama-3.1-8B. A likely reason is its effective context length. Although SaulLM-54B supports up to 32,768 tokens, its continued pretraining was limited to 8,192 tokens, potentially hindering its ability to utilize long-context inputs. In contrast, Llama-3.1-8B supports up to 128K tokens, enabling it to better exploit retrieved context. Nonetheless, our method improves both models over their respective zero-shot baselines.

Smaller gains for GPT-5: GPT-5 achieves very high zero-shot accuracy, leaving less room for improvement. One probable explanation is that the Tennessee and Idaho datasets used in our evaluation were publicly available on GitHub prior to GPT-5's training cutoff date. Given GPT-5's extensive pretraining on GitHub data to enhance its coding capabilities, it is possible that the Tennessee and Idaho datasets were included in its training corpus, potentially inflating its zero-shot performance. However, our method still improves GPT-5's performance over its baseline, even though the relative gain is smaller compared to other models.

Limitations in dataset labeling: The Tennessee and Idaho datasets annotate only one supporting and one refuting statute per will statement–condition pair. However, several statutes could reasonably be relevant to a given will statement–condition pair. As a result, some predictions counted as errors in our evaluation may in fact correspond to legally relevant provisions that were simply not labeled. Therefore, the results in Table 4 should be interpreted as a lower bound on model performance. A more comprehensive annotation of relevant laws could provide a more accurate evaluation and potentially reveal higher accuracy, which we leave for future work.

Lexical vs. Hybrid at $K = 1$: In the *refute* subset of the Idaho test set (Table 3), lexical search outperforms hybrid retrieval at $K = 1$. A possible explanation is that some will statements may contain distinctive legal terms that closely overlap with statutory text, which allows lexical search (BM25) to retrieve the correct statute at the top rank. In such situations, hybrid fusion might dilute this advantage by balancing lexical and semantic cues. However, at higher K, hybrid search consistently provides stronger performance in Idaho, and in the Tennessee test set hybrid retrieval outperforms lexical search across all K, indicating that these refute cases are exceptions rather than the norm.

7 Conclusion

In this paper, we introduced a legal information retrieval framework for will validation, combining hybrid retrieval with large language model reasoning. To the best of our knowledge, our approach is the first to tackle statutory retrieval in the domain of wills and probate law. Experiments on real-world datasets from two U.S. states demonstrate significant gains over traditional information retrieval baselines. By advancing automated statutory retrieval in this underexplored domain, our framework contributes to assisting legal professionals and others involved in executing, validating, or probating wills by delivering faster and more reliable access to the relevant laws.

Limitations

Our approach is designed to generalize to any jurisdiction as long as the relevant statutory laws are provided. However, in this work we evaluated it only on two U.S. jurisdictions (Tennessee and Idaho). While the results are strong, further experiments on additional jurisdictions, including those outside the United States, are needed to more fully verify this generalizability.

Acknowledgments

We thank the reviewers for their thoughtful comments and suggestions. This work was partially supported by the National Science Foundation (NSF) under grant #2217215, and by University of Arizona's Provost Investment Fund. Mihai Surdeanu and Clayton Morrison declare a financial interest in lum.ai. This interest has been properly disclosed to the University of Arizona Institutional Review Committee and is managed in accordance with its conflict of interest policies.

References

Valerio R. Benjamins, Pompeu Casanovas, Joost Breuker, and Aldo Gangemi, editors. 2005. *Law and the Semantic Web: Legal Ontologies, Methodologies, Legal Information Retrieval, and Applications*, volume 3369 of *Lecture Notes in Computer Science*. Springer.

Lang Cao, Zifeng Wang, Cao Xiao, and Jimeng Sun. 2024. PILOT: Legal case outcome prediction with case law. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 609–621, Mexico City, Mexico. Association for Computational Linguistics.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Yen-Liang Chen, Yi-Hung Liu, and Wu-Liang Ho. 2013. A text mining approach to assist the general public in the retrieval of legal documents. *Journal of the American Society for Information Science and Technology*, 64(2):280–290.

Pierre Colombo, Telmo Pires, Malik Boudiaf, Rui Melo, Dominic Culver, Etienne Malaboeuf, Gabriel Hautreux, Johanne Charpentier, and Michael Desa. 2024. Saullm-54b & saullm-141b: Scaling up domain adaptation for the legal domain. In *Advances in Neural Information Processing Systems*, volume 37, pages 129672–129695. Curran Associates, Inc.

Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, page 758–759, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2025. The faiss library. *Preprint*, arXiv:2401.08281.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Shaopeng Guo. 2025. Deeplegal-cn: Research and application of a deepseek-based large language model for the legal domain. In *2025 IEEE 7th International Conference on Communications, Information System and Computer Engineering (CISCE)*, pages 944–947.

Adam J Hirsch. 1996. Inheritance and inconsistency. *Ohio St. LJ*, 57:1057.

Yinghao Hu, Leilei Gan, Wenyi Xiao, Kun Kuang, and Fei Wu. 2025. Fine-tuning large language models for improving factuality in legal question answering. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 4410–4427, Abu Dhabi, UAE. Association for Computational Linguistics.

Rishi Kalra, Zekun Wu, Ayesha Gulley, Airlie Hilliard, Xin Guan, Adriano Koshiyama, and Philip Colin Treleaven. 2024. HyPA-RAG: A hybrid parameter adaptive retrieval-augmented generation system for AI legal and policy applications. In *Proceedings of the 1st Workshop on Customizable NLP: Progress and Challenges in Customizing NLP for a Domain, Application, Group, or Individual (CustomNLP4U)*, pages 237–256, Miami, Florida, USA. Association for Computational Linguistics.

Alice Kwak, Gaetano Forte, Derek Bambauer, and Mihai Surdeanu. 2023a. Transferring legal natural language inference model from a US state to another: What makes it so hard? In *Proceedings of the Natural Legal Language Processing Workshop 2023*, pages 215–222, Singapore. Association for Computational Linguistics.

Alice Kwak, Jacob Israelsen, Clayton Morrison, Derek Bambauer, and Mihai Surdeanu. 2022. Validity assessment of legal will statements as natural language inference. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6047–6056, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Alice Kwak, Cheonkam Jeong, Gaetano Forte, Derek Bambauer, Clayton Morrison, and Mihai Surdeanu. 2023b. Information extraction from legal wills: How well does GPT-4 do? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4336–4353, Singapore. Association for Computational Linguistics.

Alice Kwak, Clayton Morrison, Derek Bambauer, and Mihai Surdeanu. 2024. Classify first, and then extract: Prompt chaining technique for information extraction. In *Proceedings of the Natural Legal Language Processing Workshop 2024*, pages 303–317,

Miami, FL, USA. Association for Computational Linguistics.

John H Langbein. 1974. Substantial compliance with the wills act. *Harvard Law Review*, 88:489.

Ang Li, Yiquan Wu, Yifei Liu, Ming Cai, Lizhi Qing, Shihang Wang, Yangyang Kang, Chengyuan Liu, Fei Wu, and Kun Kuang. 2025. UniLR: Unleashing the power of LLMs on multiple legal tasks with a unified legal retriever. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11953–11967, Vienna, Austria. Association for Computational Linguistics.

Antoine Louis, Gijs van Dijck, and Gerasimos Spanakis. 2023. Finding the law: Enhancing statutory article retrieval via graph neural networks. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2761–2776, Dubrovnik, Croatia. Association for Computational Linguistics.

Jo Yin Moy. 2024. Succession law: Essential guide to draft a valid will. *Available at SSRN 4990247*.

Masaki Murata, Toshiyuki Kanamaru, Tamotsu Shirado, and Hitoshi Isahara. 2005. Using the k nearest neighbor method and bm25 in the patent document categorization subtask at ntcir-5. In *NTCIR*.

Ernesto Quevedo, Tomas Cerny, Alejandro Rodriguez, Pablo Rivas, Jorge Yero, Korn Sooksatra, Alibek Zhakubayev, and Davide Taibi. 2024. Legal natural language processing from 2015 to 2022: A comprehensive systematic mapping study of advances and applications. *IEEE Access*, 12:145286–145317.

Juliano Rabelo, Mi-Young Kim, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. 2020. Coliee 2020: Methods for legal document retrieval and entailment. In *New Frontiers in Artificial Intelligence: JSAI-IsAI 2020 Workshops, JURISIN, LENLS 2020 Workshops, Virtual Event, November 15–17, 2020, Revised Selected Papers*, page 196–210, Berlin, Heidelberg. Springer-Verlag.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Cheol Ryu, Seolhwa Lee, Subeen Pang, Chanyeol Choi, Hojun Choi, Myeonggee Min, and Jy-Yong Sohn. 2023. Retrieval-based evaluation for LLMs: A case study in Korean legal QA. In *Proceedings of the Natural Legal Language Processing Workshop 2023*, pages 132–137, Singapore. Association for Computational Linguistics.

Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing Management*, 24(5):513–523.

Manavalan Saravanan, Balaraman Ravindran, and Shivani Raman. 2009. Improving legal information retrieval using an ontological framework. *Artificial Intelligence and Law*, 17(2):101–124.

Jingyun Sun, Zhongze Luo, and Yang Li. 2025. A compliance checking framework based on retrieval augmented generation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2603–2615, Abu Dhabi, UAE. Association for Computational Linguistics.

Steven H Wang, Maksim Zubkov, Kexin Fan, Sarah Harrell, Yuyang Sun, Wei Chen, Andreas Plesner, and Roger Wattenhofer. 2025. ACORD: An expert-annotated retrieval dataset for legal contract drafting. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 24739–24762, Vienna, Austria. Association for Computational Linguistics.

Nirmalie Wiratunga, Ramitha Abeyratne, Lasal Jayawardena, Kyle Martin, Stewart Massie, Ikechukwu Nkisi-Orji, Ruvan Weerasinghe, Anne Liret, and Bruno Fleisch. 2024. Cbr-rag: Case-based reasoning fornbsp;retrieval augmented generation innbsp;llms fornbsp;legal question answering. In *Case-Based Reasoning Research and Development: 32nd International Conference, ICCBR 2024, Merida, Mexico, July 1–4, 2024, Proceedings*, page 445–460, Berlin, Heidelberg. Springer-Verlag.

A Dataset statistics

Dataset	Split	refute	support	unrelated	total
Tennessee	train	107	116	281	504
	dev	52	61	142	255
	test	53	63	139	255
Idaho	train	62	64	179	305
	dev	31	31	88	150
	test	31	32	91	154

Table 6: Dataset statistics for Tennessee and Idaho across train, dev, and test splits.

Table 6 presents the statistics of will statement datasets proposed by Kwak et al. (2022, 2023a). The Tennessee dataset contains 1014 instances, and the Idaho dataset contains 609 instances. Both datasets are divided into train, dev, and test splits. Each instance, which is a triplet of a will statement, a condition, and a law, is labeled as either support, refute, or unrelated. Since our proposed method focuses only on retrieving relevant laws, we used only the support and refute subsets and excluded the unrelated cases. The Tennessee training set was used for hyperparameter tuning and LLM prompt design. All experimental results reported in this paper were generated using the test set split.

```
You are a legal assistant knowledgeable about {Tennessee/Idaho} state law. Always follow the steps and output exactly the requested JSON with no extra text. You are given a will statement and a set of conditions. Choose exactly one best matching most relevant {Tennessee/Idaho} law.
Will Statement:{statement}
Conditions or Assumptions: {condition}
Instructions:
1) Consider both the statement and the conditions.
2) Select the single most relevant law ID from the Idaho state code.
3) Respond in STRICT JSON on one line only, exactly with law code:{"best_law_id":"NNN-NNN-NNN",
"reason": "<brief justification>"}
4) Do not include any text outside the JSON. No backticks.
```

Figure 2: This is the template we used for zero-shot prompting. In this setting, the model is asked to generate a response based solely on its parametric knowledge without additional legal context. We applied this template to the two open-source models, Llama-3.1-8B and SaulLM-54B. In the figure, the text shown in blue represents placeholders to be filled with the jurisdiction name, will statement, and condition.

B Prompt template

We designed prompt templates to guide the large language models in selecting the most relevant statutory provision from the retrieved candidates. Since different models exhibit different levels of instruction-following ability, we adopted slightly different prompts for GPT models and open-source LLMs (Llama-3.1-8B and SaulLM-54B). GPT models, being more robustly instruction-tuned, could reliably generate structured outputs even with minimal prompting. In contrast, open-source LLMs required more explicit instructions and carefully crafted templates to ensure that the outputs followed the desired structured format.

We evaluated two prompting setups. In the zero-shot setting, models were prompted to generate a response based solely on their parametric knowledge without any external legal context. In the hybrid RAG setting, we supplied the top K candidate laws retrieved by our hybrid search as context, and the prompt instruct the model to select the most relevant law from this candidate set.

Figure 2, Figure 3, Figure 4, and Figure 5 illustrate the prompts and example outputs used for GPT and open-source models. These figures highlight the adjustments made in wording and instruction detail to accommodate the differences in instruction tuning quality between the two categories of models.

```
You are a legal assistant knowledgeable about {Tennessee/Idaho} state law. Always follow the steps and output exactly the requested JSON with no extra text. You are given a will statement and a set of conditions. Choose exactly one best matching most relevant {Tennessee/Idaho} law.
Will Statement:{statement}
Conditions or Assumptions: {condition}
Candidate Law IDs: {Law IDs}
Candidate Descriptions (ID: short description): Truncated law text
Instructions:
1) Consider both the statement and the conditions.
2) Select the single most relevant Idaho law ID for the given statement and condition based on your own knowledge and the candidate list.
3) Respond in STRICT JSON on one line only, exactly with law code:{"best_law_id":"NNN-NNN-NNN",
"reason": "<brief justification>"}
4) Do not include any text outside the JSON. No backticks.
```

Figure 3: Hybrid RAG + LLM prompt template for open-source models (Llama-3.1-8B and SaulLM-54B). In this setup, each candidate law is represented by its law ID concatenated with the corresponding law text in the 'candidate Description' field. Because the context length of open-source models is limited, each law text was truncated to 10,000 characters. We also observed that, in addition to providing concatenated law IDs and texts, including a separate list of candidate law IDs improved the models' ability to return the correct formatted law ID in a greater number of cases.

```
You are a legal assistant knowledgeable
about {Tennessee/Idaho} state law.
Consider the following will statement
and conditions together to determine
which {Tennessee/Idaho} state law is
most relevant.
Will Statement:{statement}
Conditions/Assumptions:{condition}
Question: Which one {Tennessee/Idaho}
state law is most relevant to the will
statement GIVEN the conditions?
Return only the Law ID. Do not include
any extra text.
```

Figure 4: This is the template we used for zero-shot prompting. In this setting, the model is asked to generate a response based solely on its parametric knowledge without additional legal context. We applied this template to four GPT variants. In the figure, the text shown in blue represents placeholders to be filled with the jurisdiction name, will statement, and condition.

```
You are a legal assistant knowledgeable
about {Tennessee/Idaho} state law.
Consider the following will statement
and conditions together to determine
which {Tennessee/Idaho} state law is
most relevant.
Will Statement:{statement}
Conditions/Assumptions:{condition}
Candidate Law Texts:{candidate laws}
Question: Which one {Tennessee/Idaho}
state law is most relevant to the will
statement GIVEN the conditions?
Return only the Law ID. Do not include
any extra text.
```

Figure 5: Hybrid RAG + LLM prompt template for GPT models. For GPT variants, each candidate law was represented by its law ID concatenated with the corresponding law text. Unlike open-source models, GPT models handled longer contexts reliably, so full candidate laws were included without truncation. Additionally, GPT variants consistently returned the correct formatted law ID without requiring an auxiliary list of candidate IDs.