STARD: A Chinese Statute Retrieval Dataset with Real Queries Issued by Non-professionals

Anonymous ACL submission

Abstract

Statute retrieval aims to find relevant statutory articles for specific queries. This process is the basis of a wide range of legal applications such as legal advice, automated judicial decisions, legal document drafting, etc. Existing statute retrieval benchmarks focus on formal and professional queries from sources like bar exams and legal case documents, thereby neglecting non-professional queries from the general public, which often lack precise legal terminology and references. To address this gap, we introduce the STAtute Retrieval Dataset (STARD), a Chinese dataset comprising 1,543 query cases collected from real-world legal consultations and 55,348 candidate statutory articles[1]. Unlike existing statute retrieval datasets, which primarily focus on professional legal queries, STARD captures the complexity and diversity of real queries from the general public. Through a comprehensive evaluation of various retrieval baselines, we reveal that existing retrieval approaches all fall short of these real queries issued by non-professional users. The best method only achieves a Recall@100 of 0.907, suggesting the necessity for further exploration and additional research in this area.

1 Introduction

Statutes are written laws formally created and approved by a legislative body, such as a parliament or congress (Livingston, 1990). They set out specific rules and guidelines within a certain area or jurisdiction. Therefore, statutes are the primary source of legal authority in civil law countries and also play a significant role in common law jurisdictions.

Statute retrieval involves finding relevant statutory articles or sections of laws for a specific query. This process is vital in the legal field and supports a wide range of applications, including legal advice services, automated judicial decisions, and logical legal analysis. This task is challenging for the following reasons: **(1)** Statutes use complex legal terminology and linguistic structures rarely found in open-domain corpus. As a result, traditional retrieval models that lack domain-specific knowledge may struggle to accurately capture the meanings of these specialized terms. **(2)** The criteria for assessing information relevance in the legal domain differ greatly from those used in open-domain search tasks. General search tasks focus mainly on textual similarity, while legal tasks involve legal reasoning that requires the understanding of different areas of law, the relations between them, as well as the relevance of specific legal principles and their practical applications.

Question: Is disclosing the medical case information of a patient considered an invasion of privacy?
Relevant Statute Articles **Personal Information Protection Law, Article 28:** Sensitive personal information refers to information that, if leaked or illegally used, could easily harm an individual's dignity or endanger their personal or property safety. This includes biometric data, religious beliefs, specific identities, medical health, financial accounts, tracking information, and personal information of minors under the age of fourteen.
Civil Code, Article 1032: Individuals have the right to privacy. No organization or individual may infringe upon another's privacy rights through snooping, harassment, disclosure, or publicization. Privacy encompasses the tranquility of an individual's private life and the private spaces, activities, and information they wish to keep unknown to others.
Civil Code, Article 1226: Medical institutions and their medical personnel must keep patients' privacy and personal information confidential. Those who disclose patients' private and personal information or publish their medical records without the patient's consent must bear infringement liability.

Table 1: An example of the query and relevant statute articles in the STARD dataset.

Due to the challenging nature of statute retrieval and its paramount importance in civil law systems, significant progress has been made in this field. For example, the annual COLIEE competitions introduce a series of statute retrieval tasks using the

[1]All the codes and datasets are available at: https://anonymous.4open.science/r/STARD/

questions extracted from the Japanese legal bar exams (Goebel et al., 2023; Kim et al., 2022; Rabelo et al., 2022). These tasks aim to retrieve relevant statute law from the Japanese Civil Code Article according to the question from bar exams. AILA (Bhattacharya et al., 2019) competitions also introduce a series of statute retrieval datasets. The queries from AILA are case documents that were judged by the Supreme Court of India. The candidate statutes are part of the set of statutes from Indian law.

Despite these advancements, a significant gap persists in addressing real queries from non-professional people, who represent a large population of legal advice service users. The current statute retrieval benchmarks are primarily based on queries from formal legal documents, such as bar exam questions or Supreme Court case documents, which differ significantly from the everyday language used by the general public. However, queries from non-professionals often lack precise legal terminology and may include ambiguous references to legal concepts, which significantly complicate the task of statute retrieval.

To address the limitations of existing benchmarks, we propose STAtute Retrieval Dataset (STARD) i.e. STARD, a Chinese statute retrieval dataset based on real legal consultation questions from the general public. The STARD dataset comprises 1,543 query cases collected from real-world legal consultations and 55,348 candidate statutory articles extracted from all official Chinese legal regulations and judicial interpretations. Table 1 shows an example of our dataset. To the best of our knowledge, STARD is the first statute retrieval dataset where queries are from real-world legal consulting proposed by the general public.

We conduct experiments on a wide range of information retrieval (IR) baselines on the STARD dataset, including traditional lexical matching models, open-domain neural retrieval models, legal domain neural retrieval models, and a dense retriever trained with data annotated by GPT-4. The experimental results show that all existing baselines fall short of accurately and comprehensively retrieving the relevant statutes, leaving significant room for future work. Additionally, our experimental results show that employing STARD as an external knowledge source for Retrieval-Augmented Generation (RAG) significantly enhances the performance of large generative language models (LLMs) on legal tasks. This indicates that STARD is useful for developing more accessible and efficient legal systems.

In conclusion, the contributions of this paper are as follows:

- We propose STARD, a statute retrieval dataset derived from real-world legal consultation posed by non-professionals, with 1,543 queries and their corresponding relevant statutes.
- We propose a comprehensive annotation framework specifically designed for the statute retrieval task based on non-professional queries, which provides references and insights for future annotation in the legal field.
- We conduct experiments on a wide range of retrieval baselines and find that statute retrieval with queries issued by non-professionals is still a difficult task that requires further investigation.
- We present experiments on LLMs solving legal tasks with and without the STARD dataset. Experiments show that STARD can notably enhance the performance of LLMs in legal tasks.

2 Problem Formulation

2.1 Statute of Civil Law System

Civil law is a legal system primarily based on codified laws rather than case precedents, making written statutes the main source of legal authority. This contrasts with common law systems, where previous judicial decisions also play a central role. In civil law, statutes are created and enacted by legislative bodies, such as parliaments, and are organized into systematic collections known as codes, which cover various areas of law like contracts, torts, and property. A statute is a formal written law that provides specific rules and guidelines to be followed within a jurisdiction. Within statutes, there are sections known as statutory articles, which detail individual provisions or clauses of the law, addressing particular aspects or requirements. These statutes and their articles are fundamental in civil law systems to ensure that the legal framework is clear, predictable, and accessible, thereby facilitating order and defining societal rights and responsibilities.

2.2 Definition of Statute Retrieval

The statute retrieval task aims to accurately retrieve relevant statutory articles in response to a query. To be specific, given a query q that describes a legal issue or situation and a corpus of statutory articles

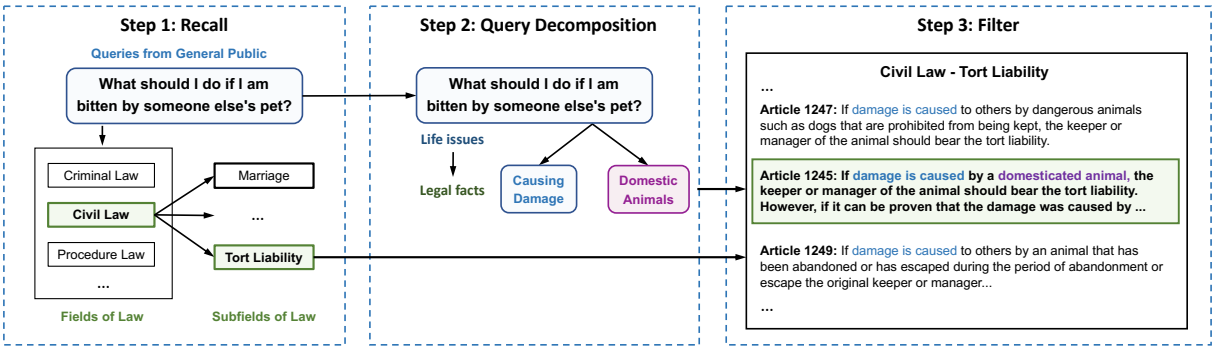

Figure 1: A schematic of our annotation framework with green boxes highlighting query-relevant elements.

$S = \{s_1, s_2, ..., s_n\}$, $n \in N^+$. For each statute s_i in the corpus, there is a Bernoulli variable r_i indicating whether s_i is relevant[2] to the query q. The goal of the statute retrieval task is to retrieve a set of statutes $R = \{s_j | r_j = 1\}$, including all statutes relevant to the query.

3 Annotation Framework

This section explains how annotators transform general questions into professional legal questions submitted by non-professionals and then identify the most relevant legal statutes to support these questions. To be specific, annotators use a three-step method: recall, query decomposition, and filtering (illustrated in Figure 1). This method mirrors the structured approach commonly used in legal reasoning, which involves three logical steps: establishing a broad legal principle (major premise), applying it to the specific facts of a case (minor premise), and then concluding. This section is organized into three subsections, each detailing a part of the annotation process that is designed to mirror these logical steps in legal reasoning.

3.1 Step 1: Recall

When initiating the annotation of legal statutes pertinent to a query, our annotators first narrow down the scope of the relevant statutes. Specifically, they start by identifying the most pertinent areas of law within the entire legal system. The process uses a top-down refining method. Annotators begin with broad departmental categories of law, such as civil, criminal law, and administrative law. Upon encountering a specific issue, annotators first determine which category of departmental law it falls under, then progressively refine the issue to more specific aspects of the law. For instance, if the issue pertains to civil law, the annotator assesses whether it relates to contract law or tort law. If it is a matter of contract law, a further determination is made regarding the specific type of contract involved. Similarly, for tort law, the specific type of tort is identified. This step effectively narrows the scope of legal statute retrieval to particular chapters within the relevant departmental law.

3.2 Step 2: Query Decomposition

Given the specialized nature of legal knowledge, individuals without a formal education in law often frame their queries with informal language rather than professional legal terminology. These queries typically consist of straightforward semantic expressions that do not directly correspond to established legal norms. For instance, consider the question "What should I do if I am bitten by someone's pet?". Here, "pet bite" represents a common, non-technical description of an incident. Searching for legal norms based solely on such descriptions might lead to irrelevant or imprecise results. Therefore, when annotators perform legal statute retrieval, they should transform the informal fact descriptions written by the questioner into legal facts through interpretation. This is the step to find the minor premise in the legal logic syllogism. In this transformation process, the annotator evaluates the life facts according to the provisions of the law and selects the legal norms corresponding to these life facts. For example, for the aforementioned issue of a pet biting a person, the annotators will transform "pet bites a person" into the legal fact of "causing damage to other" and "domestic animals" according to the provisions of Chapter 9 of the Tort Liability Compilation of the Civil Code.

3.3 Step 3: Filter

The filtering process is a critical step where annotators refine and finalize the selection of relevant legal statutes. This is accomplished by employing a "subsumption" method, integral to the syllo-

[2]The definition of "relevant" is discussed in detail in Section 3.

gistic reasoning in law. In this method, the legal facts, which have been interpreted and transformed from real-life scenarios in the previous steps, are matched against the smallest possible subset of legal statutes that adequately address the query.

To be specific, consider a set of legal statutes $S = \{S_1, S_2, S_3\}$ recalled in the first step. Through the transformation process, the query is deconstructed into distinct legal facts F_1, F_2, F_3. Each fact corresponds to a subset of statutes that it implies, denoted as $S_{F_1} = \{S_1, S_4, S_5\}$, $S_{F_2} = \{S_1, S_5\}$, and $S_{F_3} = \{S_3, S_6\}$. The objective in the filtering stage is to intersect these subsets with the initially recalled set S to determine the most relevant statutes. This is represented as $S_{Golden} = (S_{F_1} \cup S_{F_2} \cup S_{F_3}) \cap S$, yielding $S_{Golden} = \{S_1, S_3\}$.

These statutes in S_{Golden} are considered the "golden" legal statutes for the dataset, as they encompass all the legal implications drawn from the facts of the query. This step ensures that the selected statutes are not only relevant but also comprehensive in covering the legal issues presented in the query, thereby providing a solid legal foundation to support the resolution of the query.

3.4 Generalizability of Our Framework

In this section, we discuss the generalizability of the STARD dataset and our annotation framework, discussing the following two research questions (RQs): **RQ1:** Can the STARD dataset be applied to the legal systems of other countries through direct translation of our dataset? **RQ2:** Can our Three-Step Annotation Framework be applied to other legal systems?

For RQ1, directly translating the STARD dataset into other languages does not guarantee its applicability in foreign legal systems. Each country possesses unique legal statutes; articles selected from one jurisdiction may not exist or may have entirely different implications in another. Thus, the nuances of local laws must be considered, making straightforward translation inadequate for cross-national applications. For RQ2, our proposed Three-Step Annotation Framework is potentially generalizable to other countries under the civil law system. Countries with civil law systems, such as Germany, France, and Japan, typically share a similar process for retrieving law statutes. This process can generally be structured into three steps: Recall, Query Decomposition, and Filtering. Therefore, our framework could be adapted to these environments, supporting the construction of statute retrieval datasets and the application of legal statutes across various civil law jurisdictions.

4 Dataset Construction

4.1 Data Sources

All queries in our dataset derive from real legal consultations. Specifically, our legal team creates legal questions from the 12348 China Legal Service Website[3], followed by a manual anonymization of each question, which involved removing any potential identifiers associated with entities, corporations, or individuals.

To obtain the 55,348 candidate statutory articles, our legal team conducted extensive research and discussions to compile a comprehensive list of currently valid Chinese statutory laws and regulations[4]. We then manually downloaded the most up-to-date versions of these laws from the government's official website. These laws were subsequently divided into the smallest searchable units based on articles using automated scripts.

4.2 Recruitment and Payment of Annotators

For recruitment, we sourced annotators from prominent law schools. The annotation team initially consisted of 16 members. Although three members departed during the project, their positions were quickly filled to maintain the team size. Our salary plan remunerates participants based on the number of annotations they complete, with a fixed rate of approximately 10 CNY per annotation. On average, annotators processed four queries per hour, resulting in an average hourly wage of 40 CNY. This pay rate significantly exceeds the minimum hourly wage mandated in Beijing.

4.3 Annotation Process

Annotators are tasked with identifying relevant articles of statutes in response to actual legal queries posed by the general public. The specifics of the annotation framework are detailed in Section 3. Additionally, annotators are instructed not to use generative models, such as ChatGPT, for assistance. The annotation process starts with the manual anonymization of each question within the STARD dataset, involving the removal of any potential identifiers associated with entities, corporations, or individuals. Subsequently, annotators are

[3]This is the Chinese government's official website for online legal services: http://www.12348.gov.cn/

[4]The entire list of statutes we selected can be found on our official GitHub.

required to locate relevant statutes for each question, following the three-step principle introduced in Section 3.

4.4 Annotation Consistency

For each question, two annotators were assigned. **The final gold standard for each question was established only when both annotators agreed on the same legal provisions[5].**

To evaluate the reliability of agreement among human annotators, we utilized Cohen's Kappa (Cohen, 1960) $\mathcal{K}$ coefficient in a binary classification context. Each query-statute article pair corresponds to a binary classification task, where annotators judge whether the query is related or unrelated to the statute. This analysis, conducted on a dataset comprising 1,543 annotated instances, yielded a $\mathcal{K}$ value of 0.5312. This indicates moderate agreement. Achieving such a $\mathcal{K}$ value is considered satisfactory for a complex task involving fifty thousand classifications with multiple possible correct labels.

4.5 Ethics Discussion

We have thoroughly addressed the following ethical considerations: **(1) Privacy and Anonymity:** Given the sensitive nature of legal consultations, we have rigorously anonymized all queries in the STARD dataset. **(2) Transparency:** To promote reproducibility and transparency, we have made the dataset, associated models, and the codes publicly available[6]. This allows other researchers to verify, replicate, and expand upon our work, advancing the field of legal informatics. **(3) Accountability:** Recognizing the dynamic nature of legal statutes, we commit to regularly updating the STARD dataset to reflect the latest changes in law. This ensures the dataset remains accurate and reliable for ongoing research and application. **(4) Accessibility:** The STARD dataset is freely available for download from the official website under the MIT license, facilitating easy access for researchers and practitioners alike. This promotes broader usage and supports innovation across various fields.

5 Dataset Statistics and Analysis

The basic statistics of our proposed dataset are shown in Table 2. STARD comprises a total of 1,543 queries and a large-scale corpus of 55,348 candidate statutory articles. Among these candidate statutory articles, 1,445 articles are relevant to at least one of the queries in the dataset. The average length of a query is 27.3 words, and the average length of a statute article is nearly 120 words.

Table 2: Basic statistics of our proposed STARD dataset.

Statistic	# Number
Total Queries	1,543
Total Candidate Statutory Articles	55,348
Total Num of Relevant Statutory Articles	1,445
Occurrences of Relevant Articles	2,717
Avg. Relevant Articles per Query	1.76
Avg. Query Length	27.30
Avg. Article Length	119.93

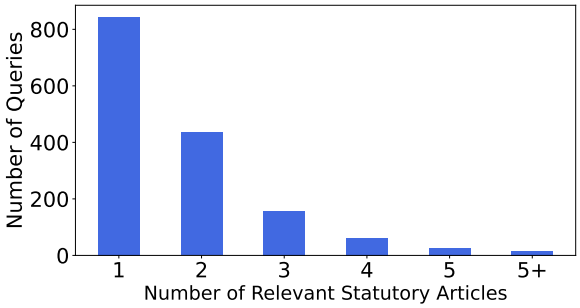

Figure 2: Distribution of relevant statutory article numbers for each query.

Figure 2 presents the distribution of queries across the number of relevant statutory articles, highlighting the varied complexity within the dataset. A substantial majority of the queries, 843 out of 1,543, correspond to just one relevant statutory article, indicating a significant number of queries can be addressed with a single, specific legal reference. This could suggest that many of the non-professional queries are focused and pertain to specific legal issues that require straightforward statute retrieval. However, 45% of queries require multiple statutory articles which indicates some of the questions are more complex, involving multiple references of law. This diversity in query complexity demonstrates that our dataset is capable of accommodating a wide range of legal questions, from straightforward to highly intricate.

6 Statute Retrieval Experiment

6.1 Selected Retrieval Baselines

We consider four types of baselines for comparison, including traditional IR methods, pre-trained Language models on general domain data, PLMs tailored for IR, and pre-trained language models built with legal documents. **The implementation details of these baselines are provided in Appendix B**

[5]In cases where annotators had differing opinions, the question would not be included in the final dataset.

[6]https://anonymous.4open.science/r/STARD/

Table 3: The overall experimental results of multiple baselines on STARD. The best results are in bold, and the second-best results are underlined. "R" stands for Recall, and "M" stands for MRR. General PLM and Legal PLM are all in the zero-shot setting. Note that LSI-STARD is a classification model where each statute is treated as a unique label; we report its ranking performance based on the probability for each statute.

		R@5	R@10	R@20	R@30	R@50	R@100	R@200	M@3	M@5	M@10
Lexical Matching	**QL**	0.3363	0.4020	0.4651	0.4839	0.5537	0.6515	0.7224	0.3052	0.3167	0.3304
	BM25	0.3349	0.3943	0.4504	0.4773	0.5240	0.6493	0.7035	0.3176	0.3251	0.3369
Open-Domain PLM	**Roberta**	0.3216	0.3908	0.4646	0.5042	0.5715	0.6633	0.7351	0.2766	0.2905	0.3010
	SEED	0.2897	0.3555	0.4264	0.4589	0.4975	0.5626	0.6260	0.2607	0.2708	0.2816
	coCondenser	0.1120	0.1598	0.2223	0.2659	0.3288	0.4292	0.5246	0.0847	0.0922	0.1004
Legal PLM	**SAILER**	0.2330	0.3050	0.3790	0.4286	0.4885	0.5674	0.6463	0.2006	0.2115	0.2234
	Lawformer	0.2411	0.2989	0.3720	0.4137	0.4733	0.5478	0.6309	0.2205	0.2313	0.2412
Fine-tuned PLM	**Dense-STARD**	**0.5206**	**0.6061**	**0.7064**	**0.7485**	**0.8107**	**0.9065**	**0.9531**	**0.4372**	**0.4543**	**0.4724**
	Dense-GPT4	0.4382	0.5174	0.5961	0.6471	0.6810	0.7984	0.8521	0.3842	0.3948	0.4106
	Dense-CAIL	0.0887	0.1272	0.1832	0.2341	0.2712	0.3281	0.3819	0.0660	0.0719	0.0842
	LSI-STARD	0.1861	0.2069	0.2386	0.2564	0.3004	0.3410	0.3956	0.2062	0.2093	0.2156

- **Traditional IR Methods**
  - **QL** (Ponte and Croft, 2017) is a language model based on Dirichlet smoothing and has good performance on retrieval tasks.
  - **BM25** (Robertson et al., 2009) is an effective retrieval model based on lexical matching that achieves good performance in retrieval tasks.
- **General Domain Pre-trained Models**
  - **Chinese-RoBERTa-WWM** (Cui et al., 2021) is a language model pre-trained with the Whole Word Masking strategy.
  - **SEED** (Lu et al., 2021) is a pre-trained text encoder for dense retrieval that achieves state-of-the-art performance.
  - **coCondenser** (Gao and Callan, 2021b) is an enhanced version of Condenser(Gao and Callan, 2021a) that adds an unsupervised corpus-level contrastive loss to warm up the passage embedding space.
- **Legal Domain Pre-trained Models**
  - **Lawformer** (Xiao et al., 2021) apply Longformer(Beltagy et al., 2020) to initialize and train with the MLM task on the legal domain.
  - **SAILER** (Li et al., 2023) is a structure-aware pre-trained language model for tailored legal document representation. It utilizes the logical connections between different sections within a legal document.
- **Fine-tuned Dense Retrieval Model**
  - **Dense-CAIL** is a dense retrieval model trained on the CAIL2018 dataset (Xiao et al., 2018). We choose this baseline to verify whether the existing dataset based on formal professional questions is sufficient for addressing statute retrieval tasks based on non-professional queries.
  - **Dense-STARD** employs a five-fold cross-validation technique on the STARD dataset.

  We initialize the above two models with Chinese-Roberta-WWM (Cui et al., 2021). For the setting of cross-validation, the dataset is randomly divided into five subsets, where one subset serves as the test set and the remaining four are used as training sets. The details of our fine-tuning process are introduced in Appendix G.
- **Dense-GPT4**: We distill a dense retrieval model from GPT-4. The process involved using GPT-4 to generate legal questions based on statute articles within a given corpus. Specifically, we prompted GPT-4 to create a legal question q that is closely related to a specific statute article a_i^+, resulting in a query-statute pair (q, a_i^+). Then, we employ a contrastive learning approach utilizing these query-statute pairs to train the dense retriever. Details are provided in Appendix H.
- **LSI-STARD** is a Transformer based classifier fine-tuned on STARD. In the Legal Statute Identification (LSI) field (Zhong et al., 2018; Paul et al., 2022; Chalkidis et al., 2021), the statute retrieval task is approached as a classification problem, where each statute is treated as a unique label. This method transforms the task into classifying legal documents or queries against a set of labels, each representing a different statute. Following this methodology, we finetune a transformer-based classification model on the STARD dataset, employing the same five-fold cross-validation setting. We initialize the transformer-based model

with Chinese-Roberta-WWM (Cui et al., 2021) and randomly initialize the outermost MLP Layer. Details are provided in Appendix I.

6.2 Evaluation Metrics

We use Mean Reciprocal Rank and Recall as evaluation metrics. By using both MRR and Recall, we can gain insights into both the accuracy of the top-ranked results and the comprehensiveness of the relevant statutory articles retrieved by the retrieval model. Detailed definitions of these metrics are provided in Appendix C.

6.3 Experimental Results

In this subsection, we provide a detailed analysis of the performance of various retrieval baselines evaluated on our proposed STARD dataset. We have the following insights into the effectiveness of different retrieval methods:

(1) Under the zero-shot setting, traditional lexical matching techniques surpass both general and legal-domain pre-trained language models (PLMs). This demonstrates that lexical matching methods are still very strong baselines in retrieval tasks. **(2)** Among all the methods that do not use human annotation, the performance of Dense-GPT4 stands out, exceeding that of all unsupervised methods tested. This indicates that distilling GPT4 to train task-specific models is a good choice in scenarios without human annotations. **(3)** Domain-specific models like SAILER are optimized for particular tasks, thus resulting in underperformance compared to general domain models. Specifically, SAILER is tailored for legal case retrieval involving long documents as queries. Consequently, it struggles with tasks that involve short queries and medium-length articles, unlike the model STARD. **(4)** The retrieval model fine-tuned on the CAIL2018 dataset performed sub-optimally on the STARD dataset. This suggests significant differences between the non-professional queries in STARD and the formal legal queries in existing datasets. Consequently, it underscores the unique nature of STARD, necessitating specialized models for effective statute retrieval. **(5)** While the LSI classifier performs well in existing studies for tasks involving the classification of a few dozen statutes, it struggles with the STARD dataset, which contains over 50,000 labels, resulting in suboptimal performance. As a result, retrieval methods are more effective than the LSI approach for large-scale statute retrieval tasks. **(6)** The performance of both the lexical matching method and the non-finetuned models is less effective than that of the Dense-STARD model. This arises because the former models lack the capacity to interpret life issues as legal facts, a capability that Dense-STARD has acquired through fine-tuning. It has been trained to associate the life issues presented in queries with relevant legal articles. However, Dense-STARD's training set is confined to just over one thousand query-article pairs. Consequently, its recall rates remain suboptimal, with Recall@100 at only 90.65%. These findings underscore the necessity for further exploration in this field.

7 Retrieval Augmented Generation Experiment

7.1 Selected Benchmark

We select two datasets encompassing three tasks for our RAG experiment:

- **JecQA** (Zhong et al., 2020) is the most extensive multiple-choice dataset within the Chinese legal field. This dataset includes two distinct tasks: Knowledge-Driven Questions (KD-questions) and Case-Analysis Questions (CA-questions), encompassing a total of 26,365 questions. All the questions are multi-select, meaning that more than one option can be correct.

- **CAIL 2018** (Xiao et al., 2018) is a large-scale Chinese legal dataset designed for judgment prediction with over 2.6 million criminal cases. This dataset contains detailed annotations of judgment results, including applicable law articles, specific charges, and prescribed prison terms. We select the Charge Prediction task of CAIL 2018 and use prediction Accuracy as the evaluation metric.

7.2 Selected LLMs and Settings

Our selected LLMs are introduced in Appendix D. The generation configuration is detailed in Appendix E. The prompt template for LLMs is detailed in Appendix F.

7.3 Experimental Results

Table 4 presents the results of the LLM's performance with and without the use of Retrieval-Augmented Generation (RAG). In the scenario without RAG, the LLM directly outputs the correct options based on the question. In the RAG scenario, the retrieval model (BM25 or Dense-STARD) recalls the top 10 relevant statutory articles from the

Table 4: The overall experimental results of three LLMs on the JecQA benchmark. We report accuracy as the evaluation metric. The best results are in bold and the second best results are underlined.

	Retriever	JQA-CA	JQA-KD	CAIL
Baichuan	**w/o RAG**	0.231	0.266	**0.850**
	BM25	0.233	0.288	0.766
	Dense-STARD	**0.238**	**0.291**	0.816
chatGLM	**w/o RAG**	0.185	0.194	0.636
	BM25	0.189	0.224	0.646
	Dense-STARD	**0.200**	**0.237**	**0.684**
chatGPT	**w/o RAG**	0.187	0.206	0.496
	BM25	**0.233**	**0.293**	**0.528**
	Dense-STARD	0.193	0.252	0.503

corpus based on the question. The retrieved statutory articles are then integrated into a meticulously designed prompt template (detailed in Appendix F).

The experimental results reveal that using the STARD corpus as the external knowledge base for the RAG significantly enhances the performance of large language models (LLMs) and underscores the value of our proposed dataset in improving the effectiveness of LLMs on legal tasks. The results also reveal that different LLMs have unique preferences for retrievers. For the Baichuan and ChatGLM models, a fine-tuned dense retriever surpasses BM25, indicating that these models benefit from dense retrievers' high recall rates. However, this advantage is not observed with the ChatGPT model, where BM25 outperforms the fine-tuned dense retriever. This suggests that the performance of RAG is highly dependent on the preferences of the LLM regarding the retriever. The experimental results on the CAIL 2018 dataset align with those observed for JecQA, with one notable exception: the performance of the Baichuan model without RAG. In this setting, Baichuan's performance is markedly superior to that of chatGLM, chatGPT, and Baichuan with RAG. We hypothesize that this exception arises from the Baichuan model's utilization of the CAIL 2018 dataset during its pretraining phase, leading to a direct answer accuracy rate that is even 81% higher than that of chatGPT.

8 Related Work

CAIL 2018 (Xiao et al., 2018; Zhong et al., 2018) competitions conduct law statute retrieval work using formal legal judgment documents. The queries in the dataset originate from the "Court's Findings" part of the judgments, and the candidates are statute articles of Chinese Criminal Law. The annual COLIEE competitions introduce a series of statute retrieval datasets using the questions extracted from the Japanese legal bar exams (Goebel et al., 2023; Kim et al., 2022; Rabelo et al., 2022). These tasks aim to retrieve relevant statute law from the Japanese Civil Code Article according to questions from bar exams. AILA (Bhattacharya et al., 2019) competitions also introduce a series of statute retrieval datasets. The queries from AILA are legal judgment documents from the Supreme Court of India. The candidate statutes are part of the set of statute articles from Indian law. BSARD (Louis and Spanakis, 2021) is a statutory article retrieval dataset in French with candidate articles from a 22,600+ Belgian law articles corpus.

In the studies of the Legal Statute Identification (LSI) (Zhong et al., 2018; Paul et al., 2022; Chalkidis et al., 2021), finding the relevant statute is approached as a classification problem, where each statute is treated as a unique label. LADAN (Xu et al., 2020) is an LSI method that uses a graph neural network and attention mechanism to distinguish confusing law articles. LeSICiN (Paul et al., 2022) utilizes both textual content and legal citation networks to identify relevant legal statutes.

Legal QA tasks also aim to fulfill the public's demand for legal information (Do et al., 2017). LLeQA (Louis et al., 2024) is a French long-form legal QA dataset comprising 1,868 expert-annotated legal questions. GerLayQA (Büttner and Habernal, 2024) is a question-answering dataset comprising 21k laymen's legal questions paired with answers from lawyers and grounded in concrete law book paragraphs.

9 Conclusion

We present STARD, a new benchmark consisting of 1,543 questions from the general public. To the best of our knowledge, STARD is the first Chinese statutes retrieval dataset tailored for the general public. Moreover, we propose an annotation framework to improve the accuracy and relevance of statute retrieval annotation, which offers valuable guidelines for future legal annotations. Our experiments across various retrieval models highlighted the complexities of non-professional statute retrieval, indicating the necessity for further exploration. Additionally, we demonstrated that integrating the STARD dataset significantly boosts the performance of LLMs in legal tasks, showcasing its potential to enhance legal AI applications.

10 Limitations

We acknowledge the limitations of this paper. One of the primary limitations is that our dataset is specifically designed around the Chinese legal system, inherently limiting its direct applicability to legal systems outside of this context. Despite our discussions on potential methodologies for adapting STARD to other civil law systems, such an expansion necessitates creating and annotating new datasets tailored to those systems' distinct legal frameworks and statutes. Thus, our future work will be dedicated to developing additional datasets that encompass a broader range of civil law systems. This endeavor aims to extend the utility of our work and foster further research and development in the domain of legal statute retrieval, ensuring broader applicability and relevance across different legal landscapes.

11 Ethics Statement

In the framework of this research, ethical considerations have been paramount from the initial stages, underscoring our commitment to the responsible advancement and application of artificial intelligence technologies. Our adherence to the principles of open research and the critical importance of reproducibility have compelled us to make all associated models, datasets, and codebases publicly available on GitHub.

Moreover, in the development of our dataset, we have paid scrupulous attention to privacy and respect for individuals' rights. Given the inherently sensitive nature of legal consultations, we have diligently anonymized every query within the STARD dataset. This process involved the removal of any potential identifiers related to entities, corporations, or individuals, thereby safeguarding privacy and preempting the possibility of data misuse.

References

Qingyao Ai, Keping Bi, Jiafeng Guo, and W Bruce Croft. 2018. Learning a deep listwise context model for ranking refinement. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 135–144.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Paheli Bhattacharya, Kripabandhu Ghosh, Saptarshi Ghosh, Arindam Pal, Parth Mehta, Arnab Bhattacharya, and Prasenjit Majumder. 2019. Fire 2019 aila track: Artificial intelligence for legal assistance. In *Proceedings of the 11th annual meeting of the forum for information retrieval evaluation*, pages 4–6.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Marius Büttner and Ivan Habernal. 2024. Answering legal questions from laymen in german civil law system. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2015–2027.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136.

Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Martin Katz, and Nikolaos Aletras. 2021. Lexglue: A benchmark dataset for legal language understanding in english. *arXiv preprint arXiv:2110.00976*.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2021. Pre-training with whole word masking for chinese bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3504–3514.

Phong-Khac Do, Huy-Tien Nguyen, Chien-Xuan Tran, Minh-Tien Nguyen, and Minh-Le Nguyen. 2017. Legal question answering using ranking svm and deep convolutional neural network. *arXiv preprint arXiv:1703.05320*.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.

Luyu Gao and Jamie Callan. 2021a. Condenser: a pre-training architecture for dense retrieval. *arXiv preprint arXiv:2104.08253*.

Luyu Gao and Jamie Callan. 2021b. Unsupervised corpus aware language model pre-training for dense passage retrieval. *arXiv preprint arXiv:2108.05540*.

Luyu Gao, Zhuyun Dai, and Jamie Callan. 2021. Rethink training of bert rerankers in multi-stage retrieval pipeline. In *European Conference on Information Retrieval*, pages 280–286. Springer.

Randy Goebel, Yoshinobu Kano, Mi-Young Kim, Juliano Rabelo, Ken Satoh, and Masaharu Yoshioka. 2023. Summary of the competition on legal information, extraction/entailment (coliee) 2023. In *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*, pages 472–480.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Mi-Young Kim, Juliano Rabelo, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. 2022. Coliee 2022 summary: Methods for legal document retrieval and entailment. In *JSAI International Symposium on Artificial Intelligence*, pages 51–67. Springer.

Haitao Li, Qingyao Ai, Jia Chen, Qian Dong, Yueyue Wu, Yiqun Liu, Chong Chen, and Qi Tian. 2023. Sailer: structure-aware pre-trained language model for legal case retrieval. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1035–1044.

Michael Livingston. 1990. Congress, the courts, and the code: Legislative history and the interpretation of tax statutes. *Tex. L. Rev.*, 69:819.

Antoine Louis and Gerasimos Spanakis. 2021. A statutory article retrieval dataset in french. *arXiv preprint arXiv:2108.11792*.

Antoine Louis, Gijs van Dijck, and Gerasimos Spanakis. 2024. Interpretable long-form legal question answering with retrieval-augmented large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 22266–22275.

Shuqi Lu, Di He, Chenyan Xiong, Guolin Ke, Waleed Malik, Zhicheng Dou, Paul Bennett, Tie-Yan Liu, and Arnold Overwijk. 2021. Less is more: Pretrain a strong siamese encoder for dense text retrieval using a weak decoder. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2780–2791.

Shounak Paul, Pawan Goyal, and Saptarshi Ghosh. 2022. Lesicin: A heterogeneous graph-based approach for automatic legal statute identification from indian legal documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11139–11146.

Jay M Ponte and W Bruce Croft. 2017. A language modeling approach to information retrieval. In *ACM SIGIR Forum*, volume 51, pages 202–208. ACM New York, NY, USA.

Juliano Rabelo, Randy Goebel, Mi-Young Kim, Yoshinobu Kano, Masaharu Yoshioka, and Ken Satoh. 2022. Overview and discussion of the competition on legal information extraction/entailment (coliee) 2021. *The Review of Socionetwork Strategies*, 16(1):111–133.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Chaojun Xiao, Xueyu Hu, Zhiyuan Liu, Cunchao Tu, and Maosong Sun. 2021. Lawformer: A pre-trained language model for chinese legal long documents. *AI Open*, 2:79–84.

Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, et al. 2018. Cail2018: A large-scale legal dataset for judgment prediction. *arXiv preprint arXiv:1807.02478*.

Nuo Xu, Pinghui Wang, Long Chen, Li Pan, Xiaoyan Wang, and Junzhou Zhao. 2020. Distinguish confusing law articles for legal judgment prediction. *arXiv preprint arXiv:2004.02557*.

Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, et al. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

Haoxi Zhong, Chaojun Xiao, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, et al. 2018. Overview of cail2018: Legal judgment prediction competition. *arXiv preprint arXiv:1810.05851*.

Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. Jec-qa: a legal-domain question answering dataset. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 9701–9708.

A License and Permissions

STARD is available under the MIT License. This permissive license was chosen to encourage the widespread use and adaptation of our resources, allowing for both academic and commercial applications without significant restrictions. For detailed terms and conditions, including how the dataset, code, and models can be used, modified, and shared, please refer to the documentation provided in our GitHub repository[7].

B Implementation Details of Retrieval Baselines

- For the implementation of traditional IR methods QL and BM25, we use the Pyserini toolkit: https://github.com/castorini/pyserini.
- For the implementation of Chinese-RoBERTa-WWM, we directly use their models released on Huggingface[8]. As SEED and Condenser have no available Chinese versions, we reproduce their work on the Chinese Wikipedia based on their open-source training code and follow all settings provided in their paper (Lu et al., 2021; Gao and Callan, 2021a).
- For the implementation of Lawformer (Xiao et al., 2021) and SAILER (Li et al., 2023), we directly use the checkpoints released on the official GitHub[9].

C Evaluation Metrics

We use Mean Reciprocal Rank and Recall as evaluation metrics.

The Mean Reciprocal Rank is a statistical measure used to evaluate the performance of a query-based system, where the primary goal is to retrieve the highest-ranked item. MRR calculates the average of the reciprocal ranks of results for a sample of queries. The reciprocal rank of a query response is the multiplicative inverse of the rank of the first correct answer:

$$\text{MRR} = \frac{1}{Q}\sum_{i=1}^{Q}\frac{1}{\text{rank}_i} \quad (1)$$

where Q is the number of queries, and rank_i is the rank position of the first relevant document for the i-th query.

[7]https://anonymous.4open.science/r/STARD/

[8]https://huggingface.co/hfl/chinese-roberta-wwm-ext

[9]https://github.com/CSHaitao/SAILER/, https://github.com/thunlp/LegalPLMs

Recall measures the ability of a model to retrieve all relevant instances in a dataset. It is defined as the ratio of the number of relevant items correctly retrieved to the total number of relevant items in the database, which is critical in scenarios where missing any relevant item could be costly:

$$\text{Recall} = \frac{\text{Number of relevant items retrieved}}{\text{Total number of relevant items}} \quad (2)$$

D Selected LLMs

Our selected LLMs are listed as follows:

- **Baichuan** (Yang et al., 2023) is a series of large-scale multilingual language models, trained from scratch on 2.6 trillion tokens. We choose the **Baichuan-2-Base-13B** model which is widely used in bilingual Chinese-English scenarios.
- **ChatGLM** (Du et al., 2022) is a series of generative language models optimized for Chinese question answering and dialogue. We choose **ChatGLM3-6B** with 6.2 billion parameters.
- **ChatGPT** (Brown et al., 2020) is a series of large language models developed by OpenAI, including several versions. Among these, we choose **GPT-3.5-turbo**, which is identified as the most advanced GPT-3.5 model.

E Generation Configuration

We obtain responses from chatGPT by accessing its official API [10]. For Baichuan and chatGLM, we directly download model parameters from each model's official Hugging Face repositories and use the official Python code provided by Hugging Face to obtain the response. We use the official default configurations provided by each model for the generation configuration.

F Prompt Template for RAG

In our RAG experiments, we employed the following prompt template for the LLM:

[10]https://platform.openai.com/docs/guides/text-generation/chat-completions-api

Prompt 1

Please answer the question based on the following statute articles:
Article 1: [Content]
.....
Article 10: [Content]
Please answer the following question based on the provided articles and your knowledge, prioritizing the provided knowledge. Note that the provided articles might not include those relevant to the question.
Question: xxx

G Fine-tuning Process

We initialize the model with Chinese-Roberta-WWM (Cui et al., 2021). We use the dual-encoder architecture (Karpukhin et al., 2020) to compute the dot product between two embedding vectors as the relevance score:

$$X(c) = [CLS]q[SEP], \tag{3}$$

$$X(s) = [CLS]s[SEP], \tag{4}$$

$$Emb(X) = transformer_{[CLS]}(X), \tag{5}$$

$$S(q,s) = Emb(X(q))^\top \cdot Emb(X(s)), \tag{6}$$

where q is the query, s is the statute, $transformer_{[CLS]}(\cdot)$ outputs a contextualized vector for each token and we select the "[CLS]" vector as the embedding vector of the input. In Equation 6, we regard the inner products of embeddings as the relevance score S.

For the loss function, we use the Softmax Cross Entropy Loss (Cao et al., 2007; Ai et al., 2018; Gao et al., 2021) to optimize the retrieval model, which is defined as:

$$\mathcal{L}(Q, s^+, N) = -\log \frac{exp(S(Q,s^+))}{exp(S(Q,s^+) + \sum_{s^- \in N} exp(S(Q,s^-))}, \tag{7}$$

where S is the relevance score function which is defined in Equation 6. Q is the query, s^+ is the relevant statute and N is the set of irrelevant statutes randomly sampled from the corpus.

H Training Process of the Dense Retrieval Model Distilled from GPT-4

We introduce an approach utilizing GPT-4 to generate labels for question-article pairs. Our methodology leverages GPT-4's capabilities to autonomously generate non-professional legal questions from statutory articles, thus enabling the pairing of these questions with their corresponding articles without the need for human supervision.

The process begins by selecting statutory articles from the corpus of STARD. GPT-4 is then tasked with generating a legal question based on the content of each article. This is achieved by providing GPT-4 with a specific prompt designed to simulate a scenario in which an individual without prior legal knowledge seeks advice. The prompt instructs GPT-4 to formulate a question that such an individual might ask, ensuring that the question is directly related to and explainable by the content of the statutory article provided. The prompt used in this study is structured as follows:

Prompt 1

Given the following known statutory article:

[Content of the statutory article]

Imagine a scenario in which a person without legal knowledge is seeking legal advice. Please generate a question that this party might ask.

Note: The question must be fully explainable using the statutory article mentioned above, and remember that the person who proposes this question has never read the legal articles mentioned before.

Each interaction with GPT-4 results in the creation of a query-statute pair (q, a_i^+), where q is the generated question and a_i^+ is the positive statute article to which the question is relevant.

Following the generation of query-statute pairs, we employ a contrastive learning framework to train a dense retriever model. We use the same relevance scoring function S, as detailed in Equation 6, which assesses the relevance of articles to the questions.

In the training phase, for each query Q paired with a positive article a_i^+, we also sample 8 negative articles from the corpus. These negative sam-

ples are not relevant to the query and serve as the negative set. The loss function employed, represented by Equation 8, is designed to maximize the score of the positive article relative to the scores of the negative samples, effectively training the model to distinguish between relevant and non-relevant articles accurately.

$$\mathcal{L}(Q, a_i^+, N) = -\log \frac{exp(S(Q, a_i^+))}{exp(S(Q, a_i^+) + \sum_{a^- \in N} exp(S(Q, a^-))}, \quad (8)$$

where S is the relevance score function, which is defined in Equation 6, and N is the set of irrelevant statutes randomly sampled from the corpus.

I Training Process of the LSI Classifier

We apply a fine-tuned classifier approach to evaluate the performance of Legal Statute Identification (LSI) methods, as defined in previous works (Zhong et al., 2018; Paul et al., 2022). LSI is framed as a classification task where each legal statute is treated as a distinct label. This transformation allows for the classification of legal documents or queries by associating them with the relevant statutory labels.

Our methodology utilizes a transformer-based classification model, specifically fine-tuned on the STARD dataset within a five-fold cross-validation framework. We initiate our model using the Chinese-Roberta-WWM (Cui et al., 2021) for the transformer's parameters, while the parameters for the outermost Multi-Layer Perceptron (MLP) layer are initialized randomly. The process of input transformation and subsequent classification is defined by the following equations:

$$X(q) = [CLS]q[SEP], \quad (9)$$

$$L(q) = MLP(transformer_{[CLS]}(X(q))), \quad (10)$$

where q represents a query from the STARD dataset. The function $transformer_{[CLS]}(\cdot)$ first encodes the input using the transformer architecture, focusing on the output of the [CLS] token's embedding vector. The $MLP(\cdot)$ then maps this embedding onto the space of statutory labels L.