GLEAR: A Graph Logic-Enhanced RAG Framework for Legal QA

Jingyun Sun
Northeast Forestry University
Harbin, China
sunjingyun@nefu.edu.cn

Jiaming Tian
Northeast Forestry University
Harbin, China
minigenepig@nefu.edu.cn

Jie Shi
Northeast Forestry University
Harbin, China
shi_jie@nefu.edu.cn

Yixin Zhang
iFLYTEK Co., Ltd.
Harbin, China
yxzhang26@iflytek.com

Wenxi Sheng
Northeast Forestry University
Harbin, China
shengwenxi@nefu.edu.cn

Yang Li*
Northeast Forestry University
Harbin, China
yli@nefu.edu.cn

ABSTRACT

Existing Legal Large Language Models (3LMs) can answer user queries due to their parameterized ability to understand and generate legal text. However, they lack effective utilization of logical legal knowledge, limiting their performance in legal Question Answering (QA). To leverage the semantic understanding capability of 3LMs while effectively capturing the logical relationships between legal knowledge, we propose a graph logic-enhanced RAG framework for legal QA, named GLEAR. The framework first structures various legal knowledge into a multi-source heterogeneous knowledge graph, and then enhances the model's response quality through three core processes: dual-driven legal knowledge retrieval, key logical path mining, and inference enhancement. Experimental results show that GLEAR outperforms the baselines by an average of 14 percentage points across five traditional legal NLP tasks. In the free-form legal QA task, GLEAR also significantly surpasses the baselines in terms of response accuracy, professionalism, and comprehensiveness. Additionally, experiments demonstrate that GLEAR outperforms the standard RAG method in both performance and computational efficiency.

CCS CONCEPTS

• **Computing methodologies** → **Natural language processing**; *Knowledge representation and reasoning*; • **Information systems** → *Information retrieval*; • **Applied computing** → *Law*.

KEYWORDS

Legal AI; Graph RAG; Knowledge Graph; Legal Reasoning; GAAI

ACM Reference Format:
Jingyun Sun, Jiaming Tian, Jie Shi, Yixin Zhang, Wenxi Sheng, and Yang Li. 2026. GLEAR: A Graph Logic-Enhanced RAG Framework for Legal QA. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 9 pages. https://doi.org/10.65109/ZNVM5881

*Corresponding author.

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/ZNVM5881

1 INTRODUCTION

Recently, Large Legal Language Models (3LMs) have gained widespread academic attention for their ability to understand and generate legal text [7, 32]. Research shows that they demonstrate significant value in tasks such as legal charge prediction [20], legal information retrieval [17], and contract review [10, 21]. Additionally, their potential in legal consulting services [13] and judicial decision-making support has further advanced the development of legal intelligence.

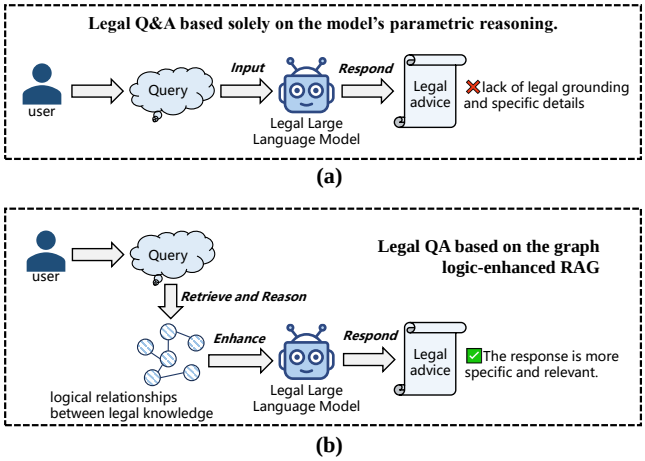

Figure 1: (a) shows responses generated solely using the parametric knowledge of the legal large model; (b) demonstrates enhanced reasoning by incorporating logical knowledge from the graph structure of legal knowledge, yielding more accurate, specific, and relevant answers.

Existing research primarily uses large-scale unsupervised legal texts to continuously pre-train general LLMs, then combines (or separately uses) supervised instruction data for fine-tuning. This approach enhances the models' semantic understanding and instruction-following abilities in the legal domain. Representative works include InternLM-Law [5], Fuzi-Mingcha [26], Wisdom-Interrogatory [27], Law-Neo [15], LawGPT [31], and LexiLaw [14].

However, existing 3LMs typically rely on parametric reasoning to address legal inquiries, lacking a structured logical utilization of legal knowledge. For example, when a user inquires about handling a divorce dispute, current 3LMs fail to establish a clear, logical

reasoning path between the user's query, similar cases, and relevant legal articles. Instead, they provide direct responses based solely on their parametric reasoning, as shown in Figure 1. This leads to responses that lack legal grounding and specific details, failing to meet the user's practical needs.

Although some studies have attempted to enhance model responses by introducing RAG-based methods (Retrieval-Augmented Generation) to retrieve external legal knowledge—such as Fuzi-Mingcha, Wisdom-Interrogatory, CBR-RAG [25], and HyPA-RAG [12]—they treat legal knowledge as fragmented text chunck and fail to fully explore the underlying connections, limiting further improvements in the model's reasoning performance.

To fully leverage the advantages of 3LMs in parametric semantic understanding while effectively capturing the logical relationships between legal knowledge, we propose a graph logic-enhanced RAG framework for legal QA, referred to as **GLEAR**. The framework first structures various legal knowledge sources into a **multi-source heterogeneous knowledge graph**, establishing connections across multiple dimensions, including case-case, article-article, case-article, query-case, and query-article associations.

Building upon this structured knowledge, we enhance legal knowledge logical utilization and model reasoning through three core modules: **First**, a dual-driven retrieval mechanism efficiently matches relevant entities from the legal knowledge graph based on user queries. **Second**, we explicitly mine the logical relationships between legal entities, providing the model with transparent and precise reasoning foundations. **Finally**, a knowledge augmentation module dynamically integrates explicit reasoning evidence into the model's generation process, improving the accuracy and specificity of the generated responses.

We evaluated the effectiveness of the GLEAR framework on multiple traditional legal NLP tasks, as well as a free-form legal Q&A task. Experimental results show that GLEAR significantly outperforms baselines in legal charge prediction, prison term prediction, legal article prediction, dispute focus identification, and legal element extraction, with gains of 13, 24, 14, 26, and 5 percentage points, respectively. In free-form legal Q&A, GLEAR substantially improves the accuracy, professionalism, and comprehensiveness of the responses generated by existing LLMs, while enhancing the stability of their response quality. Moreover, experiments demonstrate that compared to standard RAG methods, GLEAR also offers significant advantages in both performance and computational efficiency.

2 RELATED WORK

Many researchers have conducted in-depth studies on Legal Large Language Models (3LMs). Most efforts focus on using massive and diverse legal corpora to train general LLMs, thereby enhancing their semantic understanding and instruction-following capabilities in legal contexts.

Notable works include LawyerGPT [31], LexiLaw [14], Law-Neo [15], and Wisdom-Interrogatory [27]. Besides, InternLM-Law [5] is trained on data from 22 different legal NLP tasks, making it more adept at traditional legal NLP challenges. Meanwhile, Fuzi-Mingcha [26] adds syllogistic legal judgments to its training data, improving its performance on judgment reasoning tasks such as charge prediction and prison term prediction.

In addition to training standalone 3LMs, some studies leverage external legal knowledge to boost the accuracy and interpretability of generated responses. Louis et al.'s retrieve-then-read pipeline [16] is a representative approach that guides the model to produce interpretable long-form answers by retrieving relevant legal provisions. Besides, CBR-RAG [25], HyPA-RAG [12], and Wisdom-Interrogatory [27] follow a similar strategy, drawing on legal knowledge bases to deliver more accurate, detailed responses. However, these RAG methods treat legal knowledge merely as scattered text chunks, without fully exploring the inherent connections between them, leading to fragmented and inefficient knowledge utilization.

In contrast, our framework unifies various forms of legal knowledge—including historical queries, cases, and legal articles—into a heterogeneous graph. During retrieval, it connects related knowledge nodes into a logical reasoning path, thus avoiding the fragmentation issue in traditional RAG methods. Moreover, most legal knowledge in our framework is obtained through key logical path mining, which significantly reduces computational overhead during retrieval.

Furthermore, our research is highly relevant to Graph RAG. Currently, Graph RAG has become a significant research direction in mitigating hallucinations of large language models, with numerous related works, such as KetRAG[9], G-retriever[8], Think-on-graph[22], and ODA[23]. However, existing studies almost exclusively focus on enhancing large models with knowledge graphs in general domains. In the legal field, it is challenging to construct knowledge graphs that represent the relationships between entities and concepts similarly to those in general domains. Additionally, there is a lack of well-developed knowledge graph reasoning methods tailored for legal scenarios.

To address these issues, this paper presents the first attempt to construct a heterogeneous legal knowledge graph by integrating similar cases, the hierarchical structure of legal provisions, and historical queries, and proposes a query-to-query reasoning path to enhance large language models.

3 FRAMEWORK

Figure 2 illustrates the structure of our framework, which consists of two core processes: ***graph construction*** and ***logic enhanced inference***. The logic enhanced inference process includes three modules: legal knowledge retrieval, key logical path mining, and dynamic knowledge augmentation. These will be detailed in the following sections.

3.1 Graph Construction

We use a heterogeneous knowledge graph to represent the multidimensional legal knowledge system, covering five structured relationships among three types of entities: cases, legal articles, and historical queries. **Note** that historical queries are not constructed in relation to each other. The construction methods for each relational network are detailed in the supplementary material[1]. We summarize the approaches as follows:

CaseCase: The case association network is constructed by calculating the mixed similarity between two cases, as shown in Equation 1:

[1]Supplementary material is publicly available at https://github.com/NEFUJing/GLEAR.

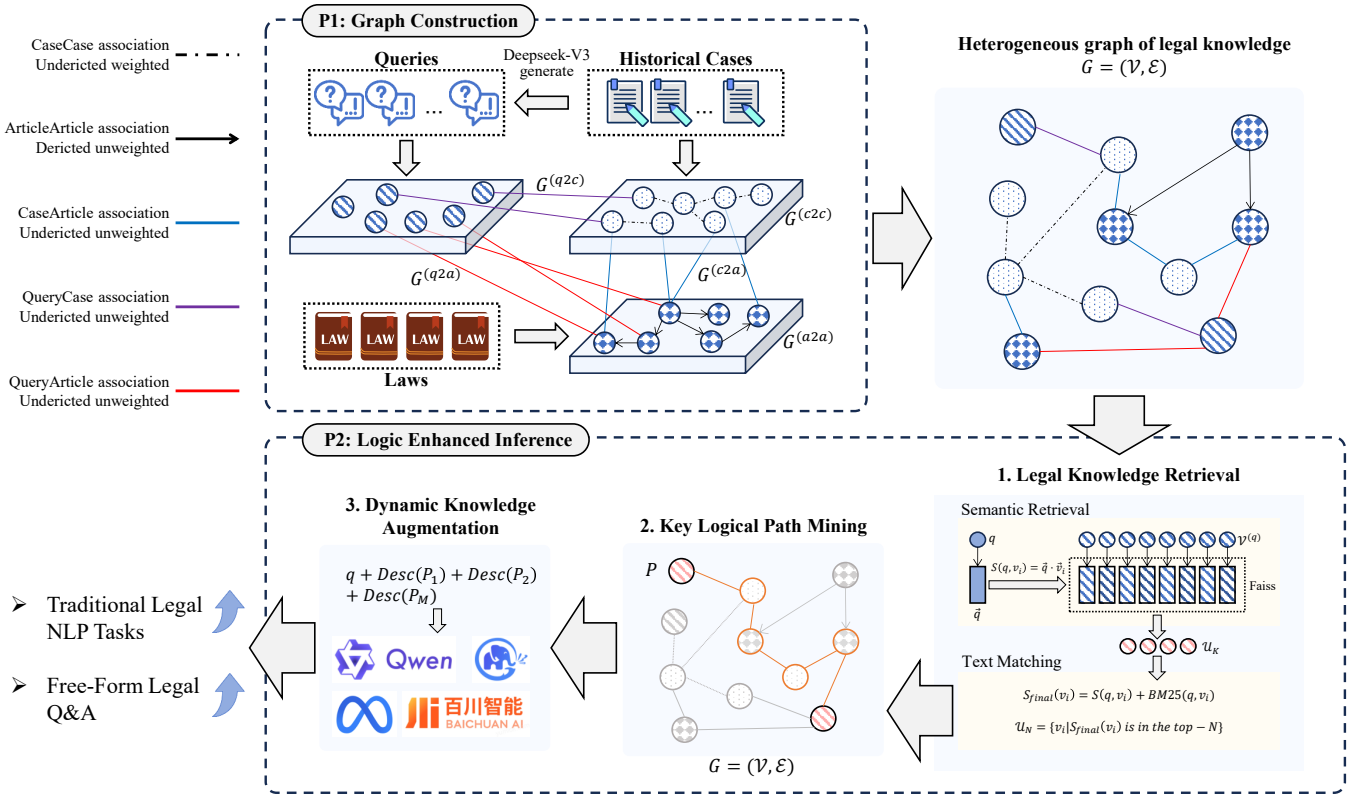

Figure 2: Architecture of the GLEAR framework.

$$sim(c_i, c_j) = \alpha \cdot \frac{\vec{c}_i \cdot \vec{c}_j}{\|\vec{c}_i\| \|\vec{c}_j\|} + (1-\alpha) \cdot \frac{|K_i \cap K_j|}{|K_i \cup K_j|} \quad (1)$$

where K_i is the set of legal terms in case c_i, $\vec{c}_i$ is the semantic vector obtained using the Lawformer [28] model, and α is the fusion coefficient. An association edge is established when the similarity exceeds the threshold $\theta = 0.85$, resulting in an undirected weighted graph $G^{(c2c)} = (\mathcal{V}^{(c)}, \mathcal{E}^{(c2c)})$.

ArticleArticle: The article-article association primarily reflects the hierarchical structure and logical relationships between legal provisions. To achieve this, we perform a fine-grained structural segmentation of the *Criminal Law of the People's Republic of China*, the *Civil Code of the People's Republic of China*, the *Criminal Procedure Law of the People's Republic of China*, and the *Civil Procedure Law of the People's Republic of China*. After data cleaning and association merging, a non-weighted directed graph $G^{(a2a)} = (\mathcal{V}^{(a)}, \mathcal{E}^{(a2a)})$ is constructed.

CaseArticle: The case-article association links cases to relevant legal articles to facilitate finding legal grounds when generating responses. We directly use the predefined case-article mappings from the CAIL2018 training set [33] to construct a undirected unweighted graph, denoted $G^{(c2a)} = (\mathcal{V}^{(c)} \cup \mathcal{V}^{(a)}, \mathcal{E}^{(c2a)})$.

QueryCase: The query-case association maps users' historical queries to relevant cases, facilitating case retrieval for more targeted legal advice. We use DeepSeek-V3[2] to generate virtual historical queries based on real cases, allowing the rapid construction of an undirected, unweighted graph $G^{(q2c)} = (\mathcal{V}^{(q)} \cup \mathcal{V}^{(c)}, \mathcal{E}^{(q2c)})$ that links queries and cases.

QueryArticle: The query-article association links users' historical queries to relevant legal articles, enabling the model to efficiently retrieve applicable laws. We collect user queries related to legal articles from Hualv.com[3] and apply DeepSeek-V3 for data augmentation. Based on the correspondence between user queries and legal articles, we construct an undirected, unweighted graph $G^{(q2a)} = (\mathcal{V}^{(q)} \cup \mathcal{V}^{(a)}, \mathcal{E}^{(q2a)})$.

Finally, we merge the five association networks to form a heterogeneous graph of legal knowledge $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \mathcal{V}^{(c)} \cup \mathcal{V}^{(a)} \cup \mathcal{V}^{(q)}$ and $\mathcal{E} = \mathcal{E}^{(c2c)} \cup \mathcal{E}^{(a2a)} \cup \mathcal{E}^{(c2a)} \cup \mathcal{E}^{(q2c)} \cup \mathcal{E}^{(q2a)}$. We use Neo4j Browser to visualize the constructed knowledge graph, as shown in the supplementary material.

3.2 Logic Enhanced Inference

The logic enhanced inference process consists of three modules: ***legal knowledge retrieval***, ***key logical path mining***, and ***dynamic knowledge augmentation***, aiming to leverage the heterogeneous legal knowledge graph to support the model in structured logical reasoning.

[2]https://www.deepseek.com/
[3]https://www.66law.cn/

3.2.1 Legal Knowledge Retrieval. Given a user query q, the legal knowledge retrieval module matches relevant historical query entities from the heterogeneous legal knowledge graph G. To achieve this, we propose a dual-driven retrieval mechanism that simultaneously evaluates semantic and textual similarity.

Semantic Retrieval: We first utilize Sentence-BERT [18] to encode historical queries in G into semantic vectors, obtaining $\vec{v}_i = BERT(v_i)$ for all $v_i \in \mathcal{V}^{(q)}$. These entity vectors are then stored in the FAISS database [11]. Given the user query q, we apply the same vectorization process to obtain $\vec{q}$. Next, we compute the inner product similarity between $\vec{q}$ and the historical query entity vectors, enabling the retrieval of the top K most similar historical queries from the FAISS, forming the initial recall set $\mathcal{U}_K$, as shown in Equations 2 and 3.

$$S(q, v_i) = \vec{q} \cdot \vec{v}_i \tag{2}$$

$$\mathcal{U}_K = FAISS.search(\vec{q}, K) \tag{3}$$

Text Matching: After obtaining the initial recall set $\mathcal{U}_K$, we compute the textual relevance between the user query and each candidate entity using the BM25 [19] algorithm, denoted as $BM25(q, v_i)$, where $v_i \in \mathcal{U}_k$.

Finally, we combine the semantic retrieval score with the BM25 relevance score to derive the final comprehensive score for each candidate entity, as shown in Equation 4:

$$S_{final}(v_i) = S(q, v_i) + BM25(q, v_i) \tag{4}$$

We then select the top N entities with the highest comprehensive scores as the final matching results, as defined in Equation 5:

$$\mathcal{U}_N = \{v_i \mid S_{final}(v_i) \text{ is in the top-}N\} \tag{5}$$

3.2.2 Key Logical Path Mining. Given the query q and its related historical query entities $\mathcal{U}_N$, a logical path P consists of nodes and edges in the graph G, where the start node (head node) u_h and the end node (tail node) u_t are both from $\mathcal{U}_N$. Thus, the path P takes the form of Equation 6:

$$P = (u_h, e_{h\to i}, v_i, \ldots, v_j, e_{j\to t}, u_t) \tag{6}$$

where $u_h, u_t \in \mathcal{U}_N$ denote the starting and ending nodes of the path, $v_i, v_j \in \mathcal{V}$ represent intermediate nodes, and $e_{h\to i}, e_{j\to t} \in \mathcal{E}$ are edges connecting adjacent nodes. Since there are no direct edges between historical query entities, any valid path must traverse at least one intermediate node, meaning the path length must be greater than 1.

We employ a greedy algorithm to compute the topological connectivity between any head node u_h and any tail node u_t, selecting the optimal paths as described in Algorithm 1. Specifically, during path construction, the algorithm iteratively selects the highest-weight reachable edge at each step to maximize the overall path weight (i.e., topological connectivity). For edges without predefined weights, a default weight of 1.0 is assigned. Finally, we identify M optimal paths to form the set $\mathcal{P} = \{P_1, P_2, \ldots, P_M\}$, where each path P_m represents a key (high topological connectivity) path in the legal knowledge graph G.

Algorithm 1 Key Logical Path Mining

Require: Graph G, relevant historical query set $\mathcal{U}_N$, number of optimal paths M
Ensure: Set of M highest-weighted logical paths $\mathcal{P} = \{P_1, P_2, ..., P_M\}$

Initialize $\mathcal{P} \leftarrow \emptyset$, $C \leftarrow \emptyset$ {Final paths, candidates}
for each $(u_h, u_t) \in \mathcal{U}_N \times \mathcal{U}_N, u_h \neq u_t$ **do**
  $P_m \leftarrow [u_h]$, $c \leftarrow u_h$
  Initialize $W(P_m) \leftarrow 0$
  while $c \neq u_t$ **do**
    Select $e_{c\to i}$ with maximum weight from the edges connected to c
    if $e_{c\to i}$ does not exist **then**
      break
    end if
    $W(P_m) += W(e_{c\to i})$
    Append $e_{c\to i}$ and v_i to P_m, set $c \leftarrow v_i$
  end while
  if $c == u_t$ **then**
    add (...) to C
  end if
end for
$\mathcal{P} \leftarrow$ Top M paths in C sorted by weight

return $\mathcal{P}$

3.2.3 Dynamic Knowledge Augmentation. To inject the knowledge contained in logical paths into the inference process of the legal large language model, we convert each optimal path into an explicit natural language representation. Specifically, we sequentially concatenate the descriptions of all nodes within the path P_m to generate a natural language description $Desc(P_m)$. Finally, we use the natural language descriptions of the optimal paths as additional contextual information to assist the model in question answering, as formulated in Equation 7:

$$\hat{a} = LLM(q + Desc(P_1) + Desc(P_2) + \cdots + Desc(P_M)) \tag{7}$$

This process is referred to as dynamic augmentation because the optimal path set $\mathcal{P}$ varies for each user query q. The entire process of dynamic knowledge augmentation can be formulated as 8:

$$\hat{a} = DynamicEnhance(q, G \mapsto \mathcal{P}, LLM) \tag{8}$$

where q represents the user query, G is the heterogeneous legal knowledge graph constructed in Section 3.1, $\mathcal{P}$ consists of the optimal logical paths obtained in Sections 3.2.1 and 3.2.2, and LLM refers to the large language model.

4 EXPERIMENTAL SETUP

4.1 Tasks, Datasets and Evaluations

We set up two types of tasks to evaluate the effectiveness of GLEAR.

The first type includes five classic legal NLP applications: 1) **Legal charge prediction**, using CAIL2018 [29] as test data and F1 as the evaluation metric; 2) **Prison term prediction**, using CAIL2018 as the test data and nLog distance as evaluation metric;

3) **Dispute focus identification**, using LAIC2021 [3] as the test data and F1 as the evaluation metric; 4) **Legal article prediction**, using CAIL2018 as the test data and F1 as the evaluation metric; 5) **Element detection**, using the LEVEN [30] dataset for testing and F1 as the evaluation metric.

The second type of evaluation task is **free-form legal Q&A**, where users input any question in natural language, and the model generates the corresponding response. For this task, we follow widely accepted evaluation methods: human expert scoring and model scoring [2]. We set the scoring range from 1 to 10, with the experts and Deepseek-R1 evaluating the generated responses based on accuracy, professionalism, and comprehensiveness. The prompt templates used for model scoring are provided in the supplementary material[4].

4.2 Baselines

We select two types of models as baselines in our experiments. **The first** category includes widely used general Chinese models, including ChatGLM3-6B [6], LLaMa3-8B [4], Qwen2.5-7B [24], and Baichuan2-7B [1]. **The second** category consists of recent Chinese legal model, including InterLM-Law [5], LexiLaw [14], Lawyer-LLaMA [31], Fuzi-Mingcha [26], and Wisdom-Interrogatory [27]. We focus on analyzing their performance in legal scenarios after integrating the GLEAR framework.

Additionally, we use the standard RAG method [16] as a baseline to compare the superiority of graph logic-based retrieval enhancement over text chunk-based retrieval enhancement. For this, we chunk all historical cases and legal articles, embed them into vectors, and then store vectors into the FAISS database as candidate knowledge. The maximum number of chunks retrieved is set to 5, which is the optimal value determined through experiments.

4.3 Implementation Detail

We use HuggingFace Transformers[5] and ModelScope[6] to invoke the general language models. For the legal language models, we obtain their weights directly from GitHub. In addition, we use the Neo4j database to store and visualize the knowledge graph and the Haystack[7] framework to implement RAG. All experiments are conducted on an A100 GPU with 80GB of memory.

The GLEAR framework includes four manually set hyperparameters. In Section 3.1, the case similarity threshold θ is set to 0.85. In Section 3.2.1, the numbers of retrieved entities K and N are set to 12 and 5, respectively. In Section 3.2.2, the number of optimal logic paths M is set to 3.

5 RESULTS AND ANALYSIS

We conducted extensive experiments to address the following research questions:

- **RQ1**: Can GLEAR improve the performance of existing LLMs on legal NLP tasks?
- **RQ2**: Can GLEAR enhance the legal Q&A ability of existing LLMs?
- **RQ3**: Is GLEAR more advantageous than standard RAG methods?
- **RQ4**: Does each component in GLEAR contribute to its performance?
- **RQ5**: Does GLEAR offer benefits in terms of time complexity?

5.1 Performance on Legal Applications

In this section, we evaluate the zero-shot reasoning capability of the GLEAR framework on five traditional legal NLP tasks: legal charge prediction, prison term prediction, dispute focus identification, legal article prediction, and element detection. The experimental setup is detailed in Section 4.1. To mitigate the impact of model hallucinations and ensure statistical reliability, we conducted 20 independent trials on each dataset and report the average results.

As shown in Table 1, GLEAR significantly enhances the performance of all baseline models across all five tasks. Specifically, compared to the base models without any augmentation, GLEAR achieves average improvements of 0.13, 0.24, 0.14, 0.19, and 0.04 in F1 score or nLog-distance for legal charge prediction, prison term prediction, dispute focus identification, legal article prediction, and element detection, respectively. These results demonstrate that the structured logical reasoning provided by the heterogeneous knowledge graph and the key logical path mining mechanism effectively complement the parametric knowledge of LLMs, leading to more accurate and reliable predictions.

Notably, GLEAR also outperforms the standard RAG method, which retrieves scattered text chunks from legal documents. The improvements over RAG are 0.06, 0.11, 0.04, 0.07, and 0.03 for the five tasks, respectively. This indicates that retrieving and integrating graph-structured knowledge with explicit logical connections is more effective than retrieving isolated text segments, as it provides the model with a coherent reasoning chain rather than fragmented information.

These findings address **Research Questions 1 and 3**, showing GLEAR's ability to enhance LLMs' performance on legal tasks and outperform the RAG method. Sections 5.2 and 5.4 further explore GLEAR's advantages in legal Q&A and time complexity.

5.2 Performance on Free-Form Q&A

In this section, we evaluate GLEAR's advantages in free-form legal Q&A. We created 300 legal questions and assessed responses on accuracy, professionalism, and comprehensiveness, as shown in Figure 3. The first row presents expert scores, and the second row shows scores from the Deepseek-R1 model, with the red line indicating mean score changes.

GLEAR shows significant improvements across all three dimensions: accuracy increases by 1.7 points, professionalism by 1.9 points, and comprehensiveness by 2.1 points, with comprehensiveness improving the most due to the integration of relevant legal knowledge. Additionally, GLEAR reduces the number of low-scoring outliers, with scores of 3 or below, compared to baselines, indicating improved average performance and more consistent, higher-quality results.

[4]Supplementary material is publicly available at https://github.com/NEFUJing/GLEAR
[5]https://huggingface.co/docs/transformers/v4.17.0/en/index
[6]https://www.modelscope.cn/home
[7]https://github.com/deepset-ai/haystack

Model	Legal Charge Prediction (F1)	Prison Term Prediction (nLog-distance)	Dispute Focus Identification (F1)	Legal Article Prediction (F1)	Element Detection (F1)
General LLM					
ChatGLM3-6B [6]	0.31	0.73	0.27	0.52	0.13
+GLEAR	**0.42**	0.79	**0.34**	**0.62**	**0.17**
LLaMa3-8B [4]	0.21	0.51	0.51	0.30	0.11
+RAG	0.28	0.64	0.61	0.37	0.12
+GLEAR	**0.34**	**0.75**	**0.65**	**0.39**	**0.15**
Qwen2.5-7B [24]	0.50	0.81	0.37	0.72	0.12
+RAG	0.51	0.87	0.43	0.75	0.13
+GLEAR	**0.54**	**0.89**	**0.46**	0.75	0.12
Baichuan2-7B [1]	0.42	0.65	0.18	0.27	0.08
+RAG	0.45	0.73	0.25	0.39	0.10
+GLEAR	**0.51**	**0.79**	0.25	**0.46**	**0.12**
Chinese Legal LLM					
InternLM-Law [5]	0.40	0.76	0.28	0.21	0.12
+RAG	0.43	0.77	0.30	0.23	0.13
+GLEAR	**0.50**	**0.79**	**0.31**	**0.29**	**0.17**
LexiLaw [14]	0.37	0.66	0.26	0.20	0.10
+RAG	0.41	0.70	0.31	0.25	0.13
+GLEAR	**0.45**	**0.75**	0.31	**0.26**	0.13
Lawyer-LLaMa [31]	0.42	0.70	0.26	0.21	0.09
+RAG	0.44	0.71	0.32	0.34	0.11
+GLEAR	**0.52**	**0.74**	**0.34**	**0.47**	**0.13**
Fuzi-Mingcha [26]	0.50	0.76	0.31	0.18	0.15
+RAG	0.53	0.79	0.32	0.23	0.15
+GLEAR	**0.59**	0.79	**0.33**	**0.28**	**0.17**
Wisdom-Interrogatory [27]	0.33	0.73	0.27	0.19	0.14
+RAG	0.36	0.75	0.31	0.21	0.14
+GLEAR	**0.41**	**0.79**	**0.38**	**0.26**	**0.16**

Table 1: Evaluation results on legal NLP tasks. Boldface values indicate the best performance. The background color in cells indicate the performance gain of GLEAR over RAG, with darker shades signifying larger improvements.

These findings address **Research Questions 2 and 3**, showing that GLEAR enhances the performance of existing LLMs and RAG methods in legal Q&A.

5.3 Ablation Study

To systematically evaluate the contribution of each component in the GLEAR framework, we conducted an ablation study using ChatGLM3-6B as the base model. The results demonstrate that each module, the graph structure, the dual-driven retrieval mechanism, and the key logical path mining, contributes notably to the overall performance. The complete GLEAR framework achieved the highest scores across all five legal NLP tasks, with particular gains in legal charge prediction (F1 = 0.42) and legal article prediction (F1 = 0.62). Removing the graph structure (i.e., reverting to standard RAG) led to the most significant decline, especially in charge prediction (−0.11) and article prediction (−0.10), underscoring the critical role of structured legal knowledge representation.

Furthermore, ablating the dual-driven retrieval mechanism resulted in a noticeable drop in performance across all tasks, confirming that combining semantic and lexical matching is essential for accurate entity retrieval. The key logical path mining module, while having a relatively smaller individual impact, still provided consistent improvements, particularly in prison term prediction and element detection. These findings affirm that all three components synergistically enhance the model's ability to perform logical legal

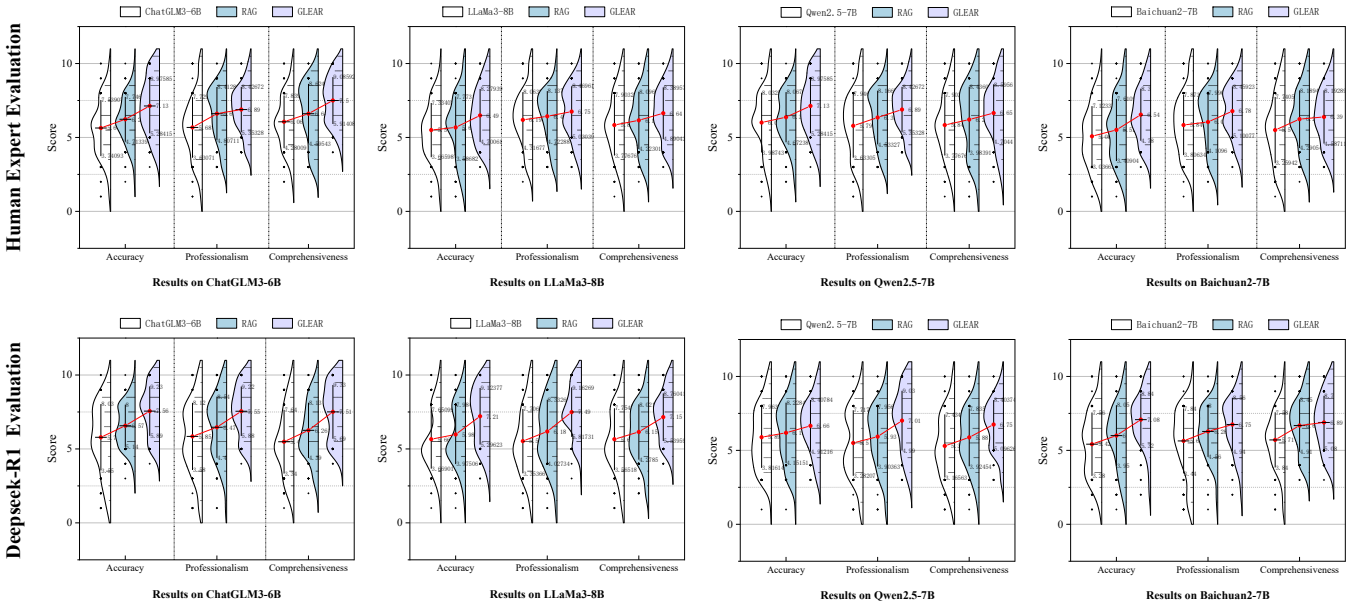

Figure 3: Evaluation results on the free-form legal Q&A task. We evaluated the responses generated for random selected 300 queries along three dimensions: accuracy, professionalism, and comprehensiveness. The first row shows the distribution of scores assigned by human experts, and the second row shows the distribution of scores generated by the Deepseek-R1 model. The red line indicates changes in the mean score.

reasoning, with the graph-based knowledge representation serving as the foundational element for improved retrieval and inference.

5.4 Time Complexity Analysis

This section presents a comparative experiment to analyze the response generation efficiency of GLEAR. Figure 4 shows the time (in seconds) required to generate responses, with the horizontal axis representing the number of legal queries. We record the response times for Qwen2.5-7B, LLaMa3-8B, ChatGLM3-6B, and Baichuan2-7B under three configurations: the original model, the model with the standard RAG framework, and the model with our proposed GLEAR framework.

As shown, both the RAG and GLEAR models take more time than the original models due to the additional overhead of knowledge retrieval and reasoning. However, GLEAR's time growth curve is notably flatter than that of RAG, indicating lower computational complexity.

These results address **Research Questions 3 and 5**, demonstrating GLEAR's superiority over existing RAG methods in computational efficiency.

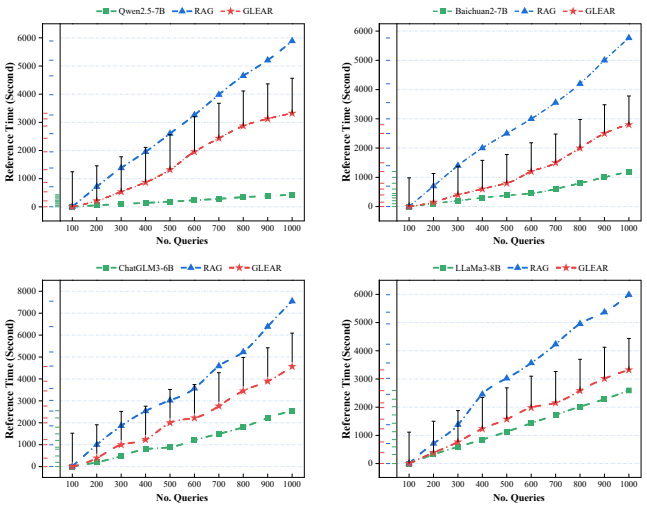

Figure 4: Quantitative results of the time complexity experiment. The x-axis denotes the number of legal queries, and the y-axis indicates the time (in seconds) required to generate responses.

5.5 Hyperparameter Analysis

This section examines the four hyperparameters in the GLEAR framework: θ, K, N, and M, to identify optimal settings. θ is the case similarity threshold for network construction, as described in Section 3.1. K is the maximum number of similar query entities retrieved during semantic retrieval. N is the maximum number retrieved after dual-driven search, as discussed in Section 3.2.1. M represents the number of best logical paths mined, as detailed in Section 3.2.2. We evaluate these hyperparameters using Qwen2.5-7B, Baichuan2-7B, LLaMa3-8B, and ChatGLM3-6B as base models.

Figure 5 illustrates how GLEAR's F1 score on the legal charge prediction task varies with changes in θ, K, N, and M. We observe that when θ is set to around 0.85, GLEAR achieves its best performance; when K ranges between 10 and 15, it attains the highest F1 score;

Version	Legal Charge Prediction (F1)	Prison Term Prediction (nLog-distance)	Dispute Focus Identification (F1)	Legal Article Prediction (F1)	Element Detection (F1)
GLEAR (Full)	0.42	0.79	0.34	0.62	0.17
w/o Path Mining	0.40	0.75	0.33	0.60	0.15
w/o Dual-driven	0.35	0.73	0.29	0.56	0.14
w/o Graph	0.31	0.73	0.27	0.52	0.13

Table 2: Ablation Study Results

and when N is about 5, its performance is optimal. Furthermore, we find that setting M to 3 or 4 both yields the best results.

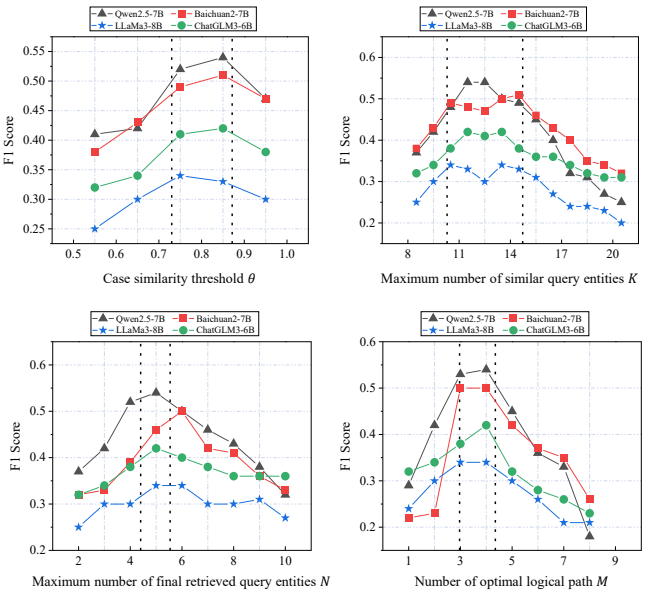

Figure 5: Hyperparameter analysis results. The four subfigures respectively illustrate how the F1 score on the legal charge prediction task varies under different settings of the four key hyperparameters (θ, K, N, and M) in GLEAR.

6 CONCLUSION

We propose a graph logic-enhanced RAG framework to address the current shortcomings of LLMs in structured legal knowledge utilization. Based on the heterogeneous legal knowledge graph we constructed, the framework injects explicit, logic-traced contextual information into the language model through three core modules: dual-driven legal knowledge retrieval, key logical path mining, and inference enhancement. Experimental results show that our framework achieves notable performance gains on both traditional legal NLP tasks and free-form legal Q&A, while also exhibiting advantages in computational efficiency. These findings open new avenues and possibilities for further integrating LLMs and RAG methods in the legal domain.

LIMITATIONS

Although the GLEAR framework demonstrates promising performance in traditional legal NLP tasks and free-form legal question answering, it still faces issues in knowledge representation and query interaction that need to be addressed.

Insufficient Knowledge Representation Granularity: Current strategies to construct heterogeneous legal knowledge graphs are limited to macro-level entities such as legal articles and cases, failing to provide explicit representations for core micro-elements of legal reasoning, such as "elements of a crime" or "sentencing factors". This coarse-grained knowledge organization may lead to semantic drift during the retrieval phase, affecting the effectiveness of complex legal reasoning tasks.

Lack of Query Tolerance Mechanism: The framework lacks an effective response mechanism for vague user inputs. When legal questions contain inaccurate phrasing or semantic ambiguity, the existing retrieval strategy struggles to accurately locate relevant knowledge nodes, resulting in a significant drop in knowledge recall rates.

To address these issues, we propose two key research directions for the future: First, we need to build a fine-grained legal element knowledge graph, enhancing the reasoning adaptability of knowledge representation by introducing a Legal Element Layer. Second, we should design query optimization algorithms based on semantic enhancement, establishing an interactive question-answering framework that includes legal term normalization and intent clarification mechanisms.

ETHICS STATEMENT

Currently, large language models still exhibit significant hallucination issues. Therefore, the legal advice generated by the GLEAR framework is not 100% accurate. The content produced by GLEAR is intended for academic research only. In the case of serious legal disputes, **please prioritize advice from a real human lawyer**.

ACKNOWLEDGMENTS

This research was supported by the Special Program of the National Natural Science Foundation of China (Grant No. L2424126), the Fundamental Research Funds for the Central Universities (Grant No. 2572024BR31), the National Natural Science Foundation of China (Grant No. 62276059), and the Heilongjiang Provincial Natural Science Foundation of China (Grant No. YQ2023F001).

REFERENCES

[1] Baichuan. 2023. Baichuan 2: Open Large-scale Language Models. *arXiv preprint arXiv:2309.10305* (2023). https://arxiv.org/abs/2309.10305

[2] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2024. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology* 15, 3 (2024), 1–45.

[3] Junyun Cui, Xiaoyu Shen, and Shaochun Wen. 2023. A survey on legal judgment prediction: Datasets, metrics, models and challenges. *IEEE Access* 11 (2023), 102050–102071.

[4] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).

[5] Zhiwei Fei, Songyang Zhang, Xiaoyu Shen, Dawei Zhu, Xiao Wang, Jidong Ge, and Vincent Ng. 2025. InternLM-Law: An Open-Sourced Chinese Legal Large Language Model. In *Proceedings of the 31st International Conference on Computational Linguistics*. 9376–9392.

[6] Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. 2024. ChatGLM: A Family of Large Language Models from GLM-130B to GLM-4 All Tools. arXiv:2406.12793 [cs.CL]

[7] Neel Guha, Julian Nyarko, Daniel Ho, Christopher Ré, Adam Chilton, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, et al. 2023. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Advances in Neural Information Processing Systems* 36 (2023), 44123–44279.

[8] Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. 2024. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *Advances in Neural Information Processing Systems* 37 (2024), 132876–132907.

[9] Yiqian Huang, Shiqi Zhang, and Xiaokui Xiao. 2025. Ket-rag: A cost-efficient multi-granular indexing framework for graph-rag. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*. 1003–1012.

[10] Lohith JJ, Kunwar Singh, and Bharatesh Chakravarthi. 2024. Digital forensic framework for smart contract vulnerabilities using ensemble models. *Multimedia Tools and Applications* 83, 17 (2024), 51469–51512.

[11] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data* 7, 3 (2019), 535–547.

[12] Rishi Kalra, Zekun Wu, Ayesha Gulley, Airlie Hilliard, Xin Guan, Adriano Koshiyama, and Philip 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)*. 237–256.

[13] Bo Li, Shuang Fan, Shaolin Zhu, and Lijie Wen. 2025. CoLE: A collaborative legal expert prompting framework for large language models in law. *Knowledge-Based Systems* (2025), 113052.

[14] Haitao Li, Qingyao Ai, Qian Dong, and Yiqun Liu. 2024. Lexilaw: A scalable legal language model for comprehensive legal understanding.

[15] Zhihao Liu, Yanzhen Zhu, and Mengyuan Lu. 2024. Enhancing Legal Expertise in Large Language Models through Composite Model Integration: The Development and Evaluation of Law-Neo. In *Proceedings of the Natural Legal Language Processing Workshop 2024*. 33–41.

[16] 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*, Vol. 38. 22266–22275.

[17] Ha-Thanh Nguyen, Manh-Kien Phi, Xuan-Bach Ngo, Vu Tran, Le-Minh Nguyen, and Minh-Phuong Tu. 2024. Attentive deep neural networks for legal document retrieval. *Artificial Intelligence and Law* 32, 1 (2024), 57–86.

[18] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics. https://arxiv.org/abs/1908.10084

[19] Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and trends® in information retrieval* 3, 4 (2009), 333–389.

[20] Jingyun Sun, Shaobin Huang, and Chi Wei. 2024. Chinese legal judgment prediction via knowledgeable prompt learning. *Expert Systems with Applications* 238 (2024), 122177.

[21] 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*. 2603–2615.

[22] Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Lionel M. Ni, Heung-Yeung Shum, and Jian Guo. 2024. Think-on-Graph: Deep and Responsible Reasoning of Large Language Model on Knowledge Graph. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*.

[23] Lei Sun, Zhengwei Tao, Youdi Li, and Hiroshi Arakawa. 2024. ODA: Observation-Driven Agent for integrating LLMs and Knowledge Graphs. In *Findings of the Association for Computational Linguistics ACL 2024*. 7417–7431.

[24] Qwen Team. 2024. Qwen2.5: A Party of Foundation Models. https://qwenlm.github.io/blog/qwen2.5/

[25] 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 for retrieval augmented generation in LLMs for legal question answering. In *International Conference on Case-Based Reasoning*. Springer, 445–460.

[26] Shiguang Wu, Zhongkun Liu, Zhen Zhang, Zheng Chen, Wentao Deng, Wenhao Zhang, Jiyuan Yang, Zhitao Yao, Yougang Lyu, Xin Xin, Shen Gao, Pengjie Ren, Zhaochun Ren, and Zhumin Chen. 2023. *fuzi.mingcha*.

[27] Yiquan Wu, Yuhang Liu, Yifei Liu, Ang Li, Siying Zhou, and Kun Kuang. 2024. *wisdomInterrogatory*. https://github.com/zhihaiLLM/wisdomInterrogatory Available at GitHub.

[28] 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 (2021), 79–84.

[29] 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* (2018).

[30] Feng Yao, Chaojun Xiao, Xiaozhi Wang, Zhiyuan Liu, Lei Hou, Cunchao Tu, Juanzi Li, Yun Liu, Weixing Shen, and Maosong Sun. 2022. LEVEN: A Large-Scale Chinese Legal Event Detection Dataset. In *Findings of the Association for Computational Linguistics: ACL 2022*. 183–201.

[31] Shunyu Yao, Qingqing Ke, Qiwei Wang, Kangtong Li, and Jie Hu. 2024. Lawyer GPT: A legal large language model with enhanced domain knowledge and reasoning capabilities. In *Proceedings of the 2024 3rd International Symposium on Robotics, Artificial Intelligence and Information Engineering*. 108–112.

[32] Dell Zhang, Alina Petrova, Dietrich Trautmann, and Frank Schilder. 2023. Unleashing the power of large language models for legal applications. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 5257–5258.

[33] Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. How Does NLP Benefit Legal System: A Summary of Legal Artificial Intelligence. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 5218–5230.