A Law Reasoning Benchmark for LLM with Tree-Organized Structures including Factum Probandum, Evidence and Experiences

Jiaxin Shen [1,2], **Jinan Xu** [1,2], **Huiqi Hu** [1,2], **Luyi Lin** [2], **Fei Zheng** [3]
Guoyang Ma [4], Fandong Meng [5], Jie Zhou [5], Wenjuan Han* [1,2]

1. Key Laboratory of Big Data & Artificial Intelligence in Transportation (Beijing Jiaotong University), Ministry of Education;
2. School of Computer Science and Technology, Beijing Jiaotong University, Beijing 100044, China;
3. Key Laboratory of Evidence Science, China University of Political Science and Law, Ministry of Education;
4. Law School, Beijing Jiaotong University, Beijing, China;
5. Pattern Recognition Center, WeChat AI, Tencent Inc, China

Abstract

While progress has been made in legal applications, law reasoning, crucial for fair adjudication, remains unexplored. We propose a transparent law reasoning schema enriched with hierarchical factum probandum, evidence, and implicit experience, enabling public scrutiny and preventing bias. Inspired by this schema, we introduce the challenging task, which takes a textual case description and outputs a hierarchical structure justifying the final decision. We also create the first crowd-sourced dataset for this task, enabling comprehensive evaluation. Simultaneously, we propose an agent framework that employs a comprehensive suite of legal analysis tools to address the challenge task. This benchmark paves the way for transparent and accountable AI-assisted law reasoning in the "Intelligent Court"[1].

1 Introduction

In recent times, Artificial Intelligence (AI) has demonstrated a profound impact on legal applications, including the generation of legal document summarization (Jain et al., 2023), argument mining, (Xu et al., 2021) and legal case retrieval (Ma et al., 2023; Liu et al., 2023). While recent advances focus on generating impartial and interpretable judicial judgments based on established criminal fact (T.y.s.s. et al., 2024; He et al., 2024; Han et al., 2024). However, the premise for ensuring this process is the accurate determination of the ultimate criminal facts. The fundamental challenge remains: how to construct logically rigorous, evidence-backed ultimate criminal facts from evidentiary materials and inferred interim facts.

Accurate criminal fact determination forms the cornerstone of judicial fairness Allen (2010); Anderson et al. (2005); Chafee Jr (1931). However,

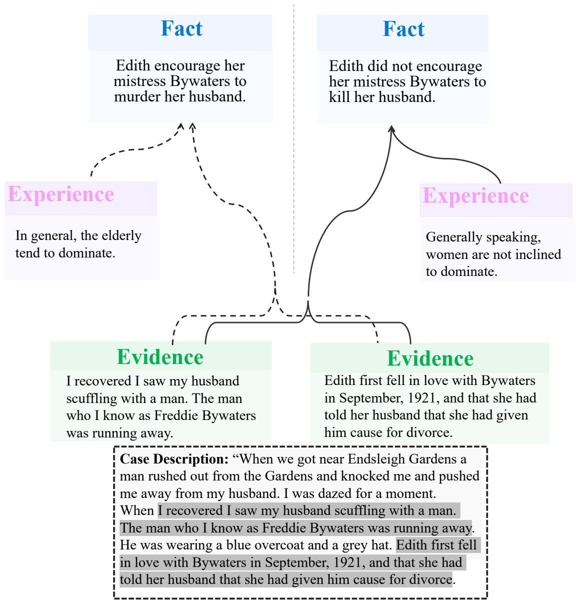

Figure 1: Case "Rex v. Bywaters and Thompson" that demonstrates different experiences have impacted different results (LEFT vs. RIGHT). The case description and evidence are shared, but the experiences of both sides are different, which leads to different ultimate probandum.

existing AI judges primarily address post-fact legal procedures rather than simulating comprehensive court processes. the fairness of adjudication fundamentally depends on systematic evidence analysis and fact reasoning during fact-finding phases. Therefore, we shift focus to an underexplored frontier: **Law Reasoning**[2], aiming to bridge the gap between evidence interpretation and judicial decision-making.

To highlight the significance of Law Reasoning, we provide examples that are widely recognized where different evidence and human experience lead to different criminal facts. Recognizing these instances is crucial for maintaining judicial justice and public trust. Here is a notable example in Figure 1.

* Wenjuan Han is the corresponding author. The email address is wjhan@bjtu.edu.cn

[1] The code and data are available at `https://github.com/cocacola-lab/LawReasoningBenchmark`

[2] Law Reasoning is also known as evidence reasoning and evidence analysis.

Findings of the Association for Computational Linguistics: ACL 2025, pages 17252–17274
July 27 - August 1, 2025

In these cases and many others before them, it is evident that wrongful judgments often arise due to the misuse of experience. To mitigate this risk, we aim to make the law reasoning procedure **transparent** and make the details visible through the law reasoning process employed by judges to subject judicial activities to social supervision, prevent the influence of prejudice, promote social fairness and justice, and enhance public trust in the judiciary.

In light of the essential of transparent law reasoning, a schema that accurately simulates law reasoning process is desired. Wigmore (1937) has long proposed a diagram method for law reasoning. However, these diagram methods remain at a theoretical level due to their complex structure and numerous elements. To address this, Anderson et al. (2005) have enhanced Wigmore's diagrammatic method to make it more user-friendly. Taking inspiration from these iconographic methods, we have adopted a modified version that enriches the schema by incorporating implicit experience. The modified schema shows a justification procedure of facts made by the fact finder (Jury or judge) at a trial. Section 2.1 provides a visual representation and detailed explanation of the schema.

Then with the designed schema as a foundation, we introduce a new challenging task — **T**ransparent **L**aw Reasoning with Tree-Organized Structures (TL for short), which aims to generate the hierarchical and layered law reasoning structure following the schema (for ease of understanding, we explain the legal terms involved in **Table 3** and use them later to describe our work). In this challenge, the textual case description is input, and the TL task is to output the law reasoning structure where the top node represents the terminal fact. Specifically, we formalize the TL procedure as a tree-like structure. Each node involves a step of reasoning from evidence to interim probandum that need to be proven, and then from interim probandum to ultimate probandum. Additionally, we conduct the first comprehensive and qualitative study on law reasoning simulation at a trial by introducing a crowd-sourcing development/test dataset for evaluation (Section 3).

In summary, our contributions are three-fold: (i) A schema enhanced with hierarchical factum probandum[3], evidence, and experiences fusion; (ii) A new challenging task – TL with crowdsourcing data and corresponding metrics; (iii) The TL agent utilizes a comprehensive suite of legal analysis tools to construct tree-organized structures.

[3]The factum probandum (pl. facta probanda) refers to the fact to be proved.

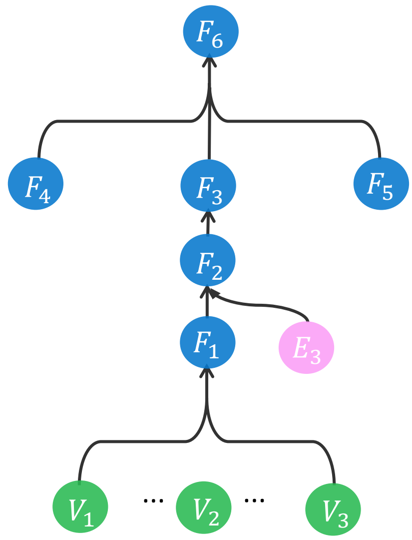

Figure 2: Illustration of the schema.

2 Task Definition

We start by briefly introducing our schema formulation in Section 2.1. Then we present our task TL from task formulation (Section 2.2) to metrics (Section A in appendix) in detail.

2.1 Schema Formulation

The term "law reasoning" is used to describe the process of reasoning from the case description $\mathbf{x}$ to the ultimate probandum to be proved, which determines the inductive logical nature of judicial proof. The designed schema rigorously represent this process, showing how ultimate probandum are finally obtained from original evidences. The process starts from evidences, goes through different granularity of factum probandum, and gets the ultimate probandum. We introduce the schema formulation in Figure 2. To make the implicit law reasoning process transparent, all factum probandum, evidence, experiences and the supportive relationships between them need to be revealed. The schema is formed in a nested tree similar to Anderson et al. (2005), including the following four elements:

• **Evidence.** Following the real judicial process, the basic e**V**idence block $\mathbf{V}$ of the schema includes documentary evidences, testimonial evidences, among others. The evidence node is the leaf node from which legal practitioners or intelligent models need to infer that certain factum probandum did occur. $\mathrm{v}_1, \mathrm{v}_2, \mathrm{v}_3 \in \mathbf{V}$.

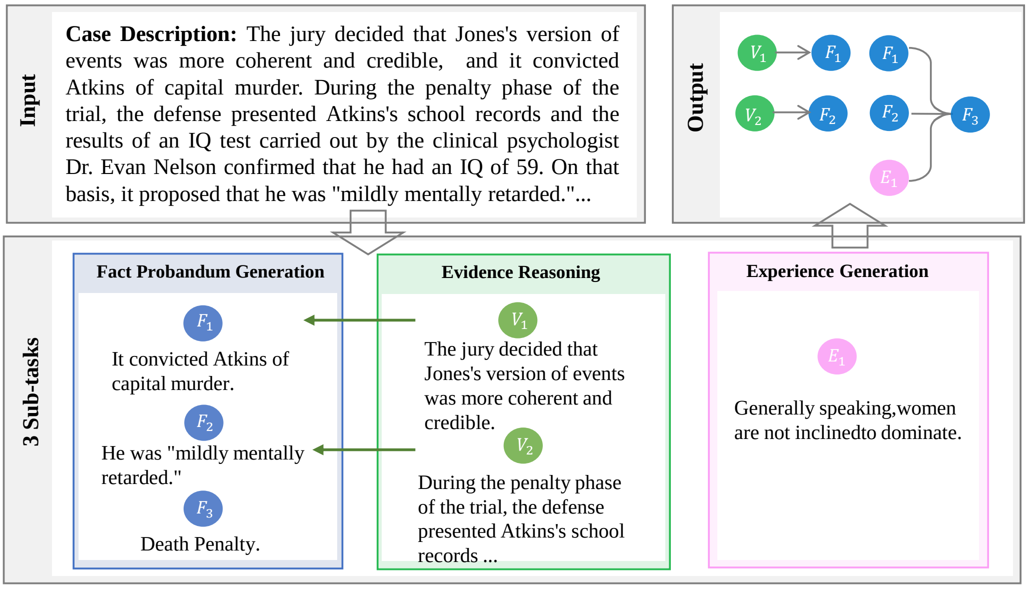

Figure 3: Illustration of the task. For convenience, we showcase examples for each sub-task. The output of the 3 sub-tasks is collected to form the complete law reasoning structure.

- **Factum Probandum.** Factum probandum have multiple levels of granularity, including interim probandum ($\mathrm{f}_1, \mathrm{f}_2$), penultimate probandas ($\mathrm{f}_3, \mathrm{f}_4, \mathrm{f}_5$), and ultimate probandum (f_6), from fine to coarse. More coarse ones are made up of fine ones. Fine-to-coarse factum probandum $[\mathrm{f}_1, \mathrm{f}_2, \mathrm{f}_3, \mathrm{f}_4, \mathrm{f}_5]$ guide a series of inference connecting the evidences $[\mathrm{v}_1, \mathrm{v}_2, \mathrm{v}_3..., \mathrm{v}_n]$ with the ultimate probandum $\mathrm{f}_6 . \mathrm{f}_\mathrm{i} \in \mathrm{F}$. [4]
- **Experiences.** Human Experience e used during connecting evidence v and fact f, and forming coarse factum probandum. Practitioners or intelligent models may need personal experiences for reasoning. The experiences help to explain why the decision maker inference like this, making the process more explicit to understand.
- **Inferences.** The edges r in the reasoning process, support each reasoning step and authorize the inference from the bottom up. Inferences exist between evidences v and factum probandum f, as well as between different granularity of factum probandum. Formally, $\mathrm{r} : \mathrm{v} \longrightarrow \mathrm{f}$ under e.

2.2 Task Formulation

We propose our task, **T**ransparent **L**aw-Reasoning with Tree-Organized Structures (TL for short), which aims to generate the hierarchical and layered fact-finding structure from the unstructured textual case description, as shown in Figure 3. The law reasoning structure should follow our designed schema, but we limit facts to only the two dimensions of Interim probandum and ultimate probandum due to the difficulty of identification and labeling. Formally, we aim to find a model $\mathcal{M}$, which takes the textual case description[5] $\mathbf{x} = [x_1, ..., x_n]$ with n tokens as input and predicts the law reasoning structure $\mathbf{y}$, i.e., $\mathbf{y} = \mathcal{M}(\mathbf{x})$. Note that, the ground-truth structure is labeled following the schema defined in Section 2.1. In detail, TL includes four sub-tasks according to its three elements (i.e., factum probandum, evidences, experiences, and inferences). We introduce each sub-task as follows:

Sub-task I: Factum Probandum Generation Aim to generate the factum probandum $\mathbf{F}$ that comes from a case description $\mathbf{x}$, including interim probandum, and ultimate probandum. Among them interim probandum can be extracted from the case description and ultimate probandum should be generated in other ways. Figure 4 shows an example to locate interim probandum in a case description.

Sub-task II: Evidence Reasoning Aim to specify the evidence that supports the interim probandum. For each interim probandum, multiple pieces of evidences are directly extracted from the case description.

The sub-task aims to find a model $\mathcal{V}$, which takes the case description $\mathbf{x}$ and factum probandum query $\mathbf{q_f}$ as input to extract the corresponding v_i, i.e., $\mathbf{v}_i = \mathcal{V}(\mathbf{x}, \mathbf{q_f})$. So this sub-task actually corre-

[4] Facts are renamed as propositions in some legal articles.

[5] The case description is a brief account of the case, usually including the times, events, actions, or behavior of each party, and any other important details that are relevant to the case.

Input

Case Description: it is necessary, as shortly as possible, to review some of the facts of this essentially commonplace and unedifying case. **The appellant, Edith Jessie Thompson, is twenty-nine years of age. She is the daughter of a Mr. Graydon, and seven years ago she married Mr. Percy Thompson, the man now dead, the only person who in this case excites any sympathy**…

Output

Interim Factum Probandum: The appellant, Edith Jessie Thompson, is twenty-nine years of age. She is the daughter of a Mr. Graydon, and seven years ago she married Mr. Percy Thompson, the man now dead, the only person who in this case excites any sympathy.
Location in Description: [15, 16]

Figure 4: Illustration of the factum probandum generation.

Input

Case Description: Mr. Clevely struck a match, and Miss Pittard asked Mrs.Thompson what had happened; and the appellant answered, "Oh, do not ask me; I do not know. Somebody flew past, and when I turned to speak to him blood was pouring out of his mouth." **A few minutes later Dr. Maudsley arrived, and he found that Mr. Thompson was dead.**…

Output

Evidence: The appellant, Edith Jessie Thompson, is twenty-nine years of age. She is the daughter of a Mr. Graydon, and seven years ago she married Mr. Percy Thompson, the man now dead, the only person who in this case excites any sympathy.
Location in Description: [43, 43]

Figure 5: Illustration of the evidence extraction in sub-task 2.

sponds to the evidences and references elements in the schema.

This task can be divided into two sequential steps: the first step involves extracting evidence from the case description, and the second step entails linking the extracted evidence to the interim probandum.

Figure 5 shows an example of step 1. Each evidence v_i is a span $[p_s, p_e]$, with p_s, p_e indicating the beginning and ending position in the case description. The evidence is localized at the sentence level.

The process of step 2 is shown in Figure 6. We contribute the relationship between the evidences extracted from the case description in the previous step and the interim probandum. If the interim probandum can be inferred from the evidence, we consider that a connection exists between the evidence and the interim probandum.

Sub-task III: Experience Generation Aim to reveal the human experiences e between the evidences v and the interim probandum f. Figure 7 shows an example.

3 Dataset Construction

We construct a high-quality dataset of case descriptions with multiple levels of annotated factum probandum, evidence and links with correlative factum probandum, and the involved experiences, which follow our schema and show the explicit path of the law reasoning process. This section

Input

Interim Factum Probandum: Edith incited Freddie to murder her husband.
Evidence:I wrote to Freddie, and were written to him without my husband's consent. When he was at home in England, we were in the habit of going out occasionally together without my husband's knowledge.

Output

Can the Interim Factum Probandum be inferred from the evidence?
True

Figure 6: Illustration of the evidence reasoning in sub-task2.

Input

Interim Factum Probandum: Edith incited Freddie to murder her husband.
Evidence:I wrote to Freddie, and were written to him without my husband's consent. When he was at home in England, we were in the habit of going out occasionally together without my husband's knowledge.

Output

Did experience help in the reasoning?
Yes
What the experience is used in the reasoning?
(a) older women tend to dominate, (b) older women are insecure.

Figure 7: Illustration of the experience generation.

delves into the details of the crowd-sourcing construction, statistical analysis and quality control(in Section B and Section C). We utilize publicly available data and implement strict controls over the annotation process. You can get more details on bias and ethical considerations in Section D from the appendix.

3.1 Crowd-Sourcing Dataset

We collect the unannotated data from China Judgement Online Website and each sample from the unannotated data describes a real-world case. Then we employ a two-phase methodology inspired by the Wizard-of-Oz technique (Kelley, 1984; Dahlbäck et al., 1993) to gather the annotated dataset.

In the initial phase, we employ law professionals to pre-annotate 15 cases and assess the quality of their annotations. This helps us ensure the reliability of our labeling methods and summarizes a set of tips for labeling and testing quality. In the second phase, we train and certify a large group of crowd-workers through tutorials and qualification tests developed during the first phase. We pay them the local average hourly wage.

We create a web-based survey using Label Studio[6],a data labeling platform that allows workers to annotate and refine unannotated data. We train workers with a comprehensive set of instructions and expert-level annotation cases to fulfill annotation standards.

During the refinement stage, we will present workers with the original description of a single case along with the corresponding factum proban-

[6]https://labelstud.io

dum, evidences and links, and experiences obtained from Stage I. Workers are requested to relabel the data with the inference of labeled data annotated by LLMs. This helps accelerate the labeling speed and does not cause any false negatives.

We utilize an automated mechanism and adhere to the schema concept outlined in Section2.1. We employ a prompt-based approach to effectively simulate human experiences in inferring interim probandum from criminal evidence in sub-task3. Subsequently, annotators refine the human-like annotations generated by the LLMs to enhance the accuracy and reliability of the results.

3.2 Dataset Statistical Analysis

We randomly select cases and split training/validation sets. Specifically, to select original cases. We organized the data based on the judicial case reference level and randomly chose cases ranging from guiding cases (higher quality) to typical cases (high quality) to maintain the dataset's quality and diversity. Prior to annotation, we divided the dataset into a training set, a validation set, and a test set. Texts included in the test set were randomly picked from both guiding and typical cases, while the remaining cases were randomly assigned to either the training or validation sets. The data in the test set underwent comprehensive manual annotation to guarantee the accuracy of the annotated information.

The collected data comprises 453 cases, 2,627 factun probandum, 14,578 pieces of evidence, and 16,414 experiences. The total number of tokens in the dataset is 6,234,443. The data statistics of the dataset are shown in Table 1. It is noteworthy that, we can construct an instruction dataset with a scale exceeding 40,000 samples, which can be utilized for fine-tuning LLMs.

	Train	*Val*	*Test*
# Instances	253	100	100
# Tokens	3,877,780	897,916	1,458,747
# Ave. Evidences	36.05	15.30	39.28
# Ave. Facts	6.77	3.47	5.67
# Ave. Experiences	37.77	18.63	44.5
# Ttl. Evidences	9,120	1,530	3,928
# Ttl. Facts	1,713	347	567
# Ttl. Experiences	9,550	1,863	4,450

Table 1: Data analysis of the collected data. Ave.: Average. Ttl.:Total. #: The number of ·.

4 Approach

For the task, we propose our Knowledge-enhanced Transparent Law Reasoning Agent (*TL Agent*). This approach, which see the whole law reasoning process as a tree structure, adheres to the established analytical approach employed by legal professionals.

As illustrated in Figure 8, the left side of the diagram depicts the user's input and the agent's output, the middle section outlines the fundamental workflow of the agent, and the right side present the main toolkit of the agent.

The task objective and corresponding case information are input into the agent, which, based on the tool manual's instructions, progresses through the stages of thinking, reasoning, and planning. Subsequently, the agent selects the appropriate tools from the toolkit to accomplish the task. To enhance the quality of the outcome, the agent analyzes the returned results to determine if additional tools are necessary to improve accuracy. This process continues until the agent deems the results satisfactory, at which point it invokes the finish tool to conclude the task.

In the following sections, we will delve into the details of our agent toolkit in Section 4.1 and agent strategy in Section 4.2. The prompt design details of each tool can be referred to in Section E in the appendix.

4.1 Designed Toolkits

Fact Finding Head The toolkit is designed with various tools for handling legal documents based on different task objectives in Section 2.2, including fact extraction tool, evidence extraction tool, fact-evidence linking tool, experience generation tool, etc. These tools define the task objectives, task rules, and json output format in the prompt to ensure that the task results are output in parallel when the tools are called, to ensure the accuracy and efficiency of the output results, and at the same time to facilitate the subsequent analysis and processing of the results by the model or the user.

Knowledge Search This toolkit contains different vector databases, which can retrieve similar texts in the vector database based on the input query to assist in determining whether the text meets the task objective. The tools in this toolkit have two steps. The first step is to retrieve similar texts based on the query, and the second step is to input the similar texts and the query into the LLM to compre-

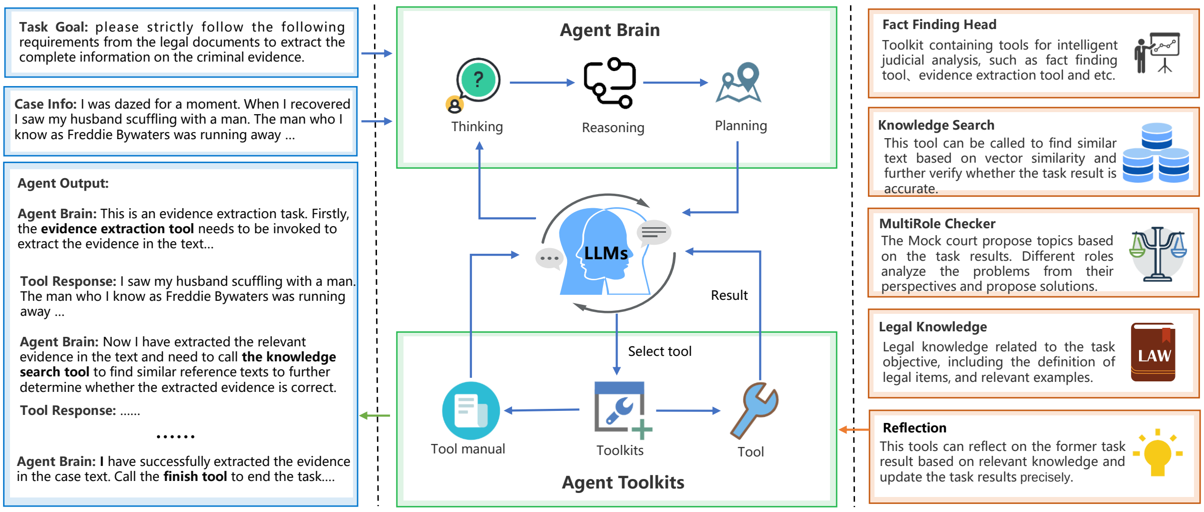

Figure 8: Illustration of our approach.

hensively determine whether the query conforms to the task objective.

For example, in the extraction task of factum probandum generation, we first use the extracted interim probandums as the query to retrieve similar texts in the vector database. Each similar text has a corresponding binary classification label. True indicates that this text belongs to the interim probandums, and False indicates that this text does not belong to the interim probandums. If there are more texts belonging to the interim probandums among the similar texts, the LLMs is more inclined to consider the input query as a interim probandums.

MultiRole Checker The agent will throw out issues based on the task objectives and task results in the previous step and provide them to this toolkit. The LLMs in this toolkit will respectively play different roles such as lawyers, judges, and police officers, analyze the issues from different perspectives, provide suggestions for solving the issues, and vote to determine whether the quality of the task completion is excellent.

Legal Knowledge Legal knowledge related to the task objective, including the definition of legal items, legal provisions and relevant examples.

Reflection This toolkit can reflect on whether the task result is accurate based on the task objective and the knowledge of other tools. It is mainly used as a middleware for the invocation of other tools, which can effectively ensure the consistency of the task output result and the accuracy of the task result.

There are some other tools used by our agent. Compared with the above-mentioned tools, these tools may be used less frequently.

Emotion Check This tool can determine the sentiment of the input text. There are three labels: positive, negative, and neutral. In factum probandum generation, it can be used to discriminate whether the generated facts contain sentiment to ensure the objectivity and neutrality of the generated facts.

Pattern Match This tool can automatically pre-analyze the required knowledge, rules, and text features that meet the task objective.

Finish When the agent judges that the generated result meets the task, this tool is called to finish the task.

4.2 Agent Strategy

To guide LLMs to leverage these powerful tools properly, we develop agent strategies to efficiently complete our task objective. Our agent's strategy is a ReAct-like (Yao et al., 2022) strategy. This strategy prompts large language models (LLMs) to generate reasoning traces and task-related actions in an interleaved manner. Depending on these actions, LLMs choose appropriate external tools and call them by supplying relevant inputs. Subsequently, this strategy regards the outputs of the tools as additional knowledge and determines whether to call a final tool or other tools for further processing.

Specifically, after each invocation of knowledge-based tools (excluding the reflection tool) 1 to 2 times, The LLMs will call the reflection tool to conduct reflection and update the task results by integrating the previously obtained knowledge. This approach can not only ensure that the results returned by the knowledge-based tools are fully utilized, thereby improving the accuracy of the task results, but also maintain the consistency of the format of the task results.

Approach	Task I			Task II			Task III			All
	S_{fact}-*1*	S_{fact}-*2*	S_{fact}-*l*	Pre	Rec	F_{evi}	S_{exp}-*1*	S_{exp}-*2*	S_{exp}-*l*	S_c
ChatGLM-6B	18.26	6.70	15.8	3.65	7.42	4.89	20.69	4.76	12.2	11.54
LexiLaw	18.65	7.59	15.98	2.51	12.97	4.20	16.60	3.58	13.80	12.56
Lawyer Llama v2	21.52	9.60	18.89	1.45	5.80	2.23	11.55	2.18	5.40	10.56
ChatGLM-6B finetune	29.30	19.11	26.82	5.95	23.56	9.50	23.12	4.26	19.17	14.37
Lexilaw finetune	29.91	20.40	26.57	8.87	27.09	13.37	19.37	2.41	16.69	23.46
Qwen-6B finetune	30.6	21.3	27.54	8.02	11.21	9.34	11.21	9.34	13.45	20.52
Spark 4.0 Ultra	25.61	13.33	22.33	7.62	6.66	7.11	23.54	5.44	18.31	24.63
ERNIE-4.0 Turbo-8k	26.83	13.16	22.37	5.26	7.66	6.24	28.7	8.53	22.31	26.38
Qwen-max	25.01	12.60	21.53	12.28	15.90	13.85	27.84	6.83	21.25	30.94
GLM-4-plus	23.23	10.33	19.70	9.65	18.96	12.78	25.75	5.61	20.60	26.43
Deepseek-v3	29.47	14.89	25.73	10.74	19.10	13.75	**31.61**	**9.21**	**25.53**	30.35
Claude-3.5	28.69	14.47	25.43	2.94	4.79	3.64	19.89	1.82	15.54	23.92
GPT-4o-mini	28.98	14.92	25.16	4.48	13.04	6.69	27.6	5.77	21.71	24.69
GPT-4o	29.86	16.43	26.44	9.72	19.84	13.05	28.71	7.31	22.36	25.74
TL Agent	**32.99**	**18.03**	**28.75**	**10.38**	**40.73**	**16.53**	30.92	8.66	24.81	**31.50**

Table 2: Comparison between our approach and baseline models. We use the comprehensive score to assess the whole structure. S_c: Comprehensive score. We also list the performance of three sub-tasks. The numbers *-1*, *-2*, and *-l* after *S* correspond to Rouge-1, Rouge-2, and Rouge-l in the formula respectively.

5 Experiments

Our agent has been rigorously compared against several classic LLMs available on the market, as detailed in Section 5.2. Additionally, we conducted a comprehensive comparison with state-of-the-art reasoning models, including o1 and r1, as discussed in Section F. To further validate the effectiveness of our tools, we performed an ablation study, the results of which are presented in Section G. These comparisons and analyses collectively demonstrate the robustness and superior performance of our agent in various tasks and scenarios.

5.1 Setup

Dataset The test set in our experiments uses the dataset we constructed in Section 3.2. The constructed training and validation set is used for fine-tuning the model. We split each case content into multiple fragments, with the length of the content being no more than 1500 tokens, and then constructed an instruction dataset including 5w samples with the corresponding evidence, factum probandum and experiences. The constructed dataset is used to finetune LLM.

Metrics For task 1, task 3, and the comprehensive evaluation (All), the results are assessed using a modified version of the rouge scoreḞor task 2, the evaluation is conducted based on precision, recall, and f1 metrics. The definitions of these metrics are provided in Section A.

Agent Setting The base model used by the agent is 4o-mini[7]. The basic parameters of the model are temperature of 0.6, max_tokens of 8096, top_p of 1, frequency_penalty of 0, presence_penalty of 0, and number_of_results of 1. The vector database used by the Agent is chroma[8], the vector model is bge-large-zh-v1.5[9], and the database is Postgre SQL 16[10].

Baselines We compare our approach with strong baselines:

The first group of baseline models comprises models with fewer than 13B parameters, which have not undergone fine-tuning using the task-specific TL dataset. Notably, the **ChatGLM-6B** (Du et al., 2022) model has not been fine-tuned on the legal domain dataset. The **Lexilaw** (Haitao Li, 2024) model, however, is a variant of ChatGLM-6B that has been fine-tuned with legal domain-specific data. Similarly, the **Lawyer Llama v2** (Huang et al., 2023) model is an outstanding open-source Chinese legal LLMs and the model is a fine-tuned version of the Llama3 model, adapted to the legal dataset.

The second group encompasses models that have been fine-tuned using the TL dataset. Specifically,

[7]https://platform.openai.com/docs/models/gpt-4o-mini-2024-07-18
[8]https://www.trychroma.com/
[9]https://huggingface.co/BAAI/bge-large-zh-v1.5
[10]https://www.postgresql.org/

instruction datasets, as described in Section 5.1, were used to fine-tune the ChatGLM-6B, Lexilaw, and Qwen (Yang et al., 2024) models, producing their respective fine-tuned variants.

The third group consists primarily of API-accessible LLMs, which have been trained predominantly on Chinese language corpora. This group includes models such as **Spark 4.0 Ultra**[11], **ERNIE-4.0 Turbo-8k**, **Qwen-max**[12], **GLM-4-plus**[13], and **Deepseek-v3** (DeepSeek-AI et al., 2024).

The fourth group features API-based LLMs trained primarily on English corpora, including **Claude-3.5**, **GPT-4o-mini**, and **GPT-4o** [14].

To ensure optimal performance for various tasks, specific prompts have been designed to guide these API-based LLMs in completing TL tasks efficiently.

5.2 Results

Comprehensive Score. It can be observed from Table 2 that TL agent not only addresses the issue of producing the irrelevant experiences but also enhances the precision of the extracted evidence and generated factum probandum by incorporating supplementary legal knowledge and legal processes.

Factum Probandum Generation. As shown in Table 2, our agent model, through multi-step reasoning and tool utilization, effectively extracts interim probandum from text and generates ultimate probandum. By employing our agent, even with the base model being 4o-mini, we achieve performance surpassing that of the gpt-4o model. Notably, a smaller model with 6B parameters, after fine-tuning on the TL dataset, demonstrates capabilities comparable to, or even exceeding, those of LLMs. Additionally, we observe that among the 6B parameter models not fine-tuned on the TL dataset, those fine-tuned with legal knowledge, such as Lexilaw and Lawyer Llama, outperform the ChatGLM model, which lacks such legal knowledge fine-tuning.

Evidence Extraction. TL agent demonstrates enhanced precision in extracting evidentiary statements from text and establishing accurate correlations between evidence and interim probandum. Furthermore, the results indicate that all models face significant challenges in tasks involving linkage identification between evidence and interim probandum. Our agent and baseline models exhibit a propensity to associate evidence with interim probandum redundantly, irrespective of the existence of a substantive inference relations, which consequently results in lower precision metrics across all models. This phenomenon underscores the inherent complexity and challenge of the task at hand.

Experience Generation. Our agent model is capable of generating precise human-experience-based information necessary for inferring interim probandum from evidence, achieving performance comparable to that of DeepSeek-V3.

From the experimental results, it is observed that although some models (such as ChatGLM-6B fine-tune) have been fine-tuned for Task 3, their performance on Task 3 still does not surpass that of LLMs accessed via APIs (such as Deepseek-V3, GPT-4o-mini, and GPT-4o). This suggests that Task 3 relies on extensive commonsense knowledge and social experience, which are inherently embedded in larger-scale LLMs.

6 Conclusion

Artificial Intelligence legal systems currently face challenges in law reasoning. To address this issue, we propose TL agent for law reasoning. By following two key steps: schema design and establishing tree-organized structures process (i.e., evidential reasoning), we can develop an abstract, systematic, and formalized reasoning process for law reasoning tree based on unstructured data. This law reasoning system can serve as a foundation for the advancement of AI legal systems, enabling them to make judgments transparent.

To ensure transparency in the judge's decision-making process, it is important to visualize the experience relied upon and the intermediate conclusions reached at each step of reasoning and judgment. This serves as a helpful reminder to the judge of which experience was utilized in each step, thereby mitigating the inherent risk of personal bias and enhancing the accuracy of law reasoning along with the final judgment. Our contribution in terms of task formulation, dataset and modeling pave the way for transparent and accountable AI-assisted law reasoning.

[11] https://xinghuo.xfyun.cn/

[12] https://qwenlm.github.io/blog/qwen2.5-max/

[13] https://bigmodel.cn/

[14] https://platform.openai.com/docs/models/gpt-4o-2024-08-06

Limitations

Although TL agent has yielded impressive results, the underlying reasons for these outcomes have not been thoroughly investigated. Moreover, the use of open-ended natural language as prompts presents both advantages and challenges. Successful extraction often necessitates domain expertise to design schema and can be a time-intensive process.

Ethics Statement

This study strictly adheres to the ethical principles outlined in the Declaration of Helsinki, which serves as a guiding framework for conducting research involving human subjects. It is of utmost importance to ensure that all participants in this study are treated with respect, dignity, and fairness.

To ensure transparency and informed decision-making, all participants will receive comprehensive information regarding the nature and purpose of the study. They will have the opportunity to ask questions and clarify any concerns they may have before providing their written informed consent. It is essential to emphasize that participation in this study is completely voluntary, and individuals have the right to withdraw their involvement at any point in time without facing any negative consequences or penalties.

In compliance with applicable laws and regulations, the confidentiality and privacy of all participants will be diligently protected. Measures will be implemented to safeguard their personal information and ensure that only authorized personnel have access to it. Any data collected throughout the study will be anonymized, ensuring that the identities of participants remain confidential.

By upholding these ethical principles and safeguards, we aim to conduct a study that upholds the highest standards of integrity and respects the rights and well-being of every participant involved.

Acknowledgement

This study is supported by National Nature Science Foundation of China (No.62406020) and the Major Program of Science and Technology Development of China State Railway Group Co., Ltd.(K2024G017).

References

Layman E Allen. 2013. Symbolic logic: A razor-edged tool for drafting and interpreting legal documents. In *Logic, Probability, and Presumptions in Legal Reasoning*, pages 1–48. Routledge.

Ronald J Allen. 2010. Evidence and inference/probability and plausibility. *Evidence Science*, 19:112–120.

Terence Anderson, David Schum, and William Twining. 2005. *Analysis of evidence*. Cambridge University Press.

Kevin D Ashley. 1991. Reasoning with cases and hypotheticals in hypo. *International journal of man-machine studies*, 34(6):753–796.

Michalis Avgerinos Loutsaris, Zoi Lachana, Charalampos Alexopoulos, and Yannis Charalabidis. 2021. Legal text processing: Combing two legal ontological approaches through text mining. In *DG. O2021: The 22nd Annual International Conference on Digital Government Research*, pages 522–532.

Dor Bernsohn, Gil Semo, Yaron Vazana, Gila Hayat, Ben Hagag, Joel Niklaus, Rohit Saha, and Kyryl Truskovskyi. 2024. LegalLens: Leveraging LLMs for legal violation identification in unstructured text. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2129–2145, St. Julian's, Malta. Association for Computational Linguistics.

Floris J Bex. 2011. *Arguments, stories and criminal evidence: A formal hybrid theory*, volume 92. Springer Science & Business Media.

L Karl Branting. 2017. Data-centric and logic-based models for automated legal problem solving. *Artificial Intelligence and Law*, 25(1):5–27.

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

Z Chafee Jr. 1931. The principles of judicial proof: Or the process of proof as given by logic, psychology and general experience and illustrated in judicial trials.

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

Pierre Colombo, Telmo Pessoa Pires, Malik Boudiaf, Dominic Culver, Rui Melo, Caio Corro, Andre F. T. Martins, Fabrizio Esposito, Vera Lúcia Raposo, Sofia Morgado, and Michael Desa. 2024. Saullm-7b: A pioneering large language model for law.

Nils Dahlbäck, Arne Jönsson, and Lars Ahrenberg. 1993. Wizard of oz studies—why and how. *Knowledge-based systems*, 6(4):258–266.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. 2024. Deepseek-v3 technical report.

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.

Neil Duxbury. 1995. *Patterns of American jurisprudence*. Clarendon Press.

Zhiwei Fei, Songyang Zhang, Xiaoyu Shen, Dawei Zhu, Xiao Wang, Maosong Cao, Fengzhe Zhou, Yining Li, Wenwei Zhang, Dahua Lin, Kai Chen, and Jidong Ge. 2024. Internlm-law: An open source chinese legal large language model.

Enrico Francesconi. 2014. A description logic framework for advanced accessing and reasoning over normative provisions. *Artificial intelligence and law*, 22(3):291–311.

Leilei Gan, Baokui Li, Kun Kuang, Yating Zhang, Lei Wang, Anh Luu, Yi Yang, and Fei Wu. 2023. Exploiting contrastive learning and numerical evidence for confusing legal judgment prediction. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12174–12185, Singapore. Association for Computational Linguistics.

Anne von der Lieth Gardner. 1987. *An artificial intelligence approach to legal reasoning*. MIT press.

Guido Governatori and Antonino Rotolo. 2010. Changing legal systems: Legal abrogations and annulments in defeasible logic. *Logic Journal of the IGPL*, 18(1):157–194.

Guido Governatori, Antonino Rotolo, Régis Riveret, Monica Palmirani, and Giovanni Sartor. 2007. Variants of temporal defeasible logics for modelling norm modifications. In *Proceedings of the 11th international conference on artificial intelligence and law*, pages 155–159.

Guido Governatori, Antonino Rotolo, and Giovanni Sartor. 2005. Temporalised normative positions in defeasible logic. In *Proceedings of the 10th international conference on Artificial intelligence and law*, pages 25–34.

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

Wenjuan Han, Jiaxin Shen, Yanyao Liu, Zhan Shi, Jinan Xu, Fangxu Hu, Hao Chen, Yan Gong, Xueli Yu, Huaqing Wang, Zhijing Liu, Yajie Yang, Tianshui Shi, and Mengyao Ge. 2024. Legalasst: Human-centered and ai-empowered machine to enhance court productivity and legal assistance. *Information Sciences*, 679:121052.

Zhitao He, Pengfei Cao, Chenhao Wang, Zhuoran Jin, Yubo Chen, Jiexin Xu, Huaijun Li, Kang Liu, and Jun Zhao. 2024. AgentsCourt: Building judicial decision-making agents with court debate simulation and legal knowledge augmentation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9399–9416. Association for Computational Linguistics.

Quzhe Huang, Mingxu Tao, Chen Zhang, Zhenwei An, Cong Jiang, Zhibin Chen, Zirui Wu, and Yansong Feng. 2023. Lawyer llama technical report. *arXiv preprint arXiv:2305.15062*.

Weijing Huang, Xianfeng Liao, Zhiqiang Xie, Jiang Qian, Shaojun Wang, Bojin Zhuang, and Jing Xiao. 2020. Generating reasonable legal text through the combination of language modeling and question answering. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2020)*. Accessed: 2025-02-16.

Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2023. A sentence is known by the company it keeps: Improving legal document summarization using deep clustering. *Artificial Intelligence and Law*, pages 1–36.

John F Kelley. 1984. An iterative design methodology for user-friendly natural language office information applications. *ACM Transactions on Information Systems (TOIS)*, 2(1):26–41.

R Keown. 1980. Mathematical models for legal prediction. *Computer/lj*, 2:829.

Fred Kort. 1957. Predicting supreme court decisions mathematically: A quantitative analysis of the "right to counsel" cases. *American Political Science Review*, 51(1):1–12.

Benjamin E Lauderdale and Tom S Clark. 2012. The supreme court's many median justices. *American Political Science Review*, 106(4):847–866.

Bulou Liu, Yiran Hu, Qingyao Ai, Yiqun Liu, Yueyue Wu, Chenliang Li, and Weixing Shen. 2023. Leveraging event schema to ask clarifying questions for conversational legal case retrieval. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 1513–1522.

Yixiao Ma, Yueyue Wu, Qingyao Ai, Yiqun Liu, Yunqiu Shao, Min Zhang, and Shaoping Ma. 2023. Incorporating structural information into legal case retrieval. 42(2).

Kaiz Merchant and Yash Pande. 2018. Nlp based latent semantic analysis for legal text summarization. In *2018 international conference on advances in computing, communications and informatics (ICACCI)*, pages 1803–1807. IEEE.

John Merryman and Rogelio Pérez-Perdomo. 2018. *The civil law tradition: an introduction to the legal systems of Europe and Latin America*. Stanford University Press.

Ephraim Nissan. 2012. *Computer Applications for Handling Legal Evidence, Police Investigation and Case Argumentation*, volume 5. Springer Science & Business Media.

Henry Prakken and Giovanni Sartor. 1997. *Logical models of legal argumentation*. Springer.

Marc Queudot, Éric Charton, and Marie-Jean Meurs. 2020. Improving access to justice with legal chatbots. *Stats*, 3(3):356–375.

Adam Roegiest, Radha Chitta, Jonathan Donnelly, Maya Lash, Alexandra Vtyurina, and Francois Longtin. 2023. Questions about contracts: Prompt templates for structured answer generation. In *Proceedings of the Natural Legal Language Processing Workshop 2023*, pages 62–72, Singapore. Association for Computational Linguistics.

Carlo Sansone and Giancarlo Sperlí. 2022. Legal information retrieval systems: State-of-the-art and open issues. *Information Systems*, 106:101967.

Pengxiao Song. 2023. Lawgpt: Legal applications of gpt models. `https://github.com/pengxiao-song/LaWGPT/tree/main?tab=readme-ov-file#readme`. Accessed: 2025-02-16.

Santosh T.y.s.s., Nina Baumgartner, Matthias Stürmer, Matthias Grabmair, and Joel Niklaus. 2024. Towards explainability and fairness in Swiss judgement prediction: Benchmarking on a multilingual dataset. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 16500–16513. ELRA and ICCL.

Douglas Walton. 2009. Hendrik kaptein, henry prakken and bart verheij (eds): Review of legal evidence and proof: statistics, stories, logic: Farnham, ashgate, applied legal philosophy series, 2009, 288 pp.

J.H. Wigmore. 1937. *The Science of Judicial Proof: As Given by Logic, Psychology, and General Experience, and Illustrated in Judicial Trials*. Little, Brown.

Yiquan Wu, Siying Zhou, Yifei Liu, Weiming Lu, Xiaozhong Liu, Yating Zhang, Changlong Sun, Fei Wu, and Kun Kuang. 2023. Precedent-enhanced legal judgment prediction with llm and domain-model collaboration. *arXiv preprint arXiv:2310.09241*.

Adam Wyner and Giovanni Casini. 2017. *Legal Knowledge and Information Systems: Jurix 2017: the Thirtieth Annual Conference*, volume 302. IOS Press.

Huihui Xu, Jaromir Savelka, and Kevin D. Ashley. 2021. Toward summarizing case decisions via extracting argument issues, reasons, and conclusions. page 250–254. Association for Computing Machinery.

Nuo Xu, Pinghui Wang, Long Chen, Li Pan, Xiaoyan Wang, and Junzhou Zhao. 2020. Distinguish confusing law articles for legal judgment prediction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3086–3095, Online. Association for Computational Linguistics.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

PKU Yuan Group. 2023. Chatlaw: A conversational legal assistant. https://github.com/PKU-YuanGroup/ChatLaw. Accessed: 2025-02-16.

A Metrics

Metrics for Factum Probandum Generation. We use the Rouge F_1 score, which is commonly used for text summarization tasks. For each case, the fact set provided as ground truth is represented as $\mathrm{F}^* = [\mathrm{f}_1^*, \mathrm{f}_2^*, ..., \mathrm{f}_n^*]$, while the prediction set generated by a model is denoted as $\mathrm{F} = [\mathrm{f}_1, \mathrm{f}_2, ..., \mathrm{f}_m]$. The metric can be defined using the following formula:

$$S_{fact} = \frac{1}{n}\sum_{i=1}^{n} \max_{\mathrm{f}_j^* \in F}(Rouge(\mathrm{f}_i, \mathrm{f}_j^*))$$

Metrics for Evidence Reasoning. The F_{evi} metric measures how well a model extracts relevant evidences to support its factum probandum. It does this by comparing the model's predicted evidence spans to the actual ground-truth evidence spans and penalizing for both missing important evidence and including irrelevant information. Thus, each piece of evidence can be linked to specific factum probanda it supports. These connections are represented by triples (factum probandum, relation, evidence). Think of an arrow pointing from the evidence to the factum probandum. The model predicts some evidence to support the ground-truth facta probanda, resulting in triples. F_{evi} focuses on the macro F_1-like metrics, meaning it only cares about how accurate the model's chosen triple is. The more overlap between predicted and ground-truth triples, the higher the score. F_{evi} is formulated as:

$$Pre = \frac{\sum_{i=1}^{k} |\mathrm{L}_i^* \cap \mathrm{L}_i|}{\sum_{i=1}^{k} |\mathrm{L}_i|}$$

$$Rec = \frac{\sum_{i=1}^{k} |\mathrm{L}_i^* \cap \mathrm{L}_i|}{\sum_{i=1}^{k} |\mathrm{L}_i^*|}$$

$$F_{evi} = \frac{2 \cdot Pre \cdot Rec}{Pre + Rec}$$

For each case, the triple set provided as ground truth is represented as L_i^*. The prediction set generated by a model is denoted as L_i. We use set intersection ($\cap$) to identify the overlap between the predicted and ground-truth set. α is a hyper-parameter to balance between Pre and Rec. k is the number of cases. $|\,|$ returns the number of the element in the set.

Metrics for Experience Generation. The metric for experience generation considers two aspects. First, we should consider whether the experience needs to be generated as a component to achieve the interim probandum. It is a binary classification problem and we measure accuracy as the metric. Then, we consider the quality of the generated experience using Rouge F_1. The experience alone does not support an interim probandum. The following formula defines the process.

$$R_{exp}(\mathrm{e}_i^*, \mathrm{e}_i) = \begin{cases} 1 & \mathrm{e}_i^* = \mathrm{e}_i = None \\ Rouge(\mathrm{e}_i^*, \mathrm{e}_i) & else \end{cases}$$

$$S_{exp} = \frac{1}{t}\sum_{i=1}^{t} R_{exp}(\mathrm{e}_i^*, \mathrm{e}_i)$$

t is the number of generated experiences. e_i^* is the ground-truth experience quadruple (fact, relation, evidence, experience). e_i is predicted experience quadruple. $\mathrm{e}_i = None$ means that the relation from the evidence to the interim probandum doesn't require additional experience. If either e_i^* or e_i is not equal to $None$, R_{exp} is set to 0.

Comprehensive Score. The three metrics mentioned above pertain to the sub-task level. To evaluate the comprehensive score, it is important to consider the overall quality of the structure, in addition to the necessity of each sub-task. The Comprehensive Score (S_c) is calculated as follows:

$$Rouge_{sum} = \frac{1}{2}(Rouge(\mathrm{d}_m, \mathrm{d}_m^*) + Rouge(\mathrm{d}_n, \mathrm{d}_n^*))$$

$$\hat{r}_q = arg \max_{\mathrm{r}_q^* \in \mathrm{y}_i^*}(Rouge_{sum}(\mathrm{r}_p, \mathrm{r}_q^*))$$

$$S = \frac{1}{max([\mathrm{y}_i], [\mathrm{y}_i^*])}\sum_{p=1}^{[\mathrm{y}_i]}(Rouge_{sum}(\mathrm{r}_p, \hat{\mathrm{r}}_q) + R_{exp}(\mathrm{e}_p, \hat{\mathrm{e}}_q))$$

Term	Definition
`Factum Probandum`	The fact that must be proven. It's used in legal contexts to refer to a fact or set of facts that one party in a case must establish in order to prove their claim or defense.
`Interim Probandum`	The provisional or temporary facts to be proven. It refers to facts that are temporarily or provisionally considered to be established for the purposes of an ongoing legal proceeding, pending further evidence or a final ruling.
`Ultimate Probandum`	The "ultimate fact" or the final fact that must be proven in a case. It is the core fact or facts that are central to the resolution of the legal issue at hand. The ultimate probandum is the fact that, if proven, will ultimately decide the outcome of the case.
`Criminal Evidence`	The information, objects, or testimony presented in a court of law to prove or disprove the factum probandum.
`Human Experience`	The understanding of human behavior, societal norms, and practical reasoning to resolve disputes and administer justice. It play significant roles in evaluating evidence, determination of factum probandum and making judicial decisions.

Table 3: Legal Terms

$$S_c = \frac{1}{k}\sum_{i=1}^{k} S$$

y_i is the predicted fact-finding structure. y_i^* is ground-truth structure. Each y_i include two basic elements, nodes d ($\mathrm{d} \in \{\mathrm{f}, \mathrm{v}\}$) and relation r, which connect between node d_m and node d_n. $[]$ denotes the number of relations in structure y.

B Quality control

Since labeling is a task without formulaic standards, we employ multiple methods to control the annotation quality.

Data Source We use data from China Judgement Online Website[15], which assures our case descriptions are following a relatively fixed logical structure. This reduces the difficulty of labeling, even that amateurs can also understand the idea of the annotation after receiving a little training.

Workers and Payment We restrict the workers to those in law schools. Their research direction is highly aligned with the topic of our paper.

In particular, we recruited a total of 45 students, with an hourly labor compensation of $ 7.5. The average labeling time for each annotation is 55 minutes, and the verification time is 20 minutes. On average, each annotator was assigned to 15.1 annotations, and the reviewer was assigned to 30.2 annotations. The total labeling duration is 566.25 hours, and the total labor cost paid is $ 4246.9.

Training and Pre-labeling Prior to annotation, we referenced examples from three expert-level academic papers for theoretical guidance. We refered to the book(Anderson et al., 2005) for annotation. Additionally, we enlisted law professors to pre-annotate 15 cases as examples for annotators to follow. Pre-annotation training sessions were also organized to help annotators understand the tasks and perform accurate annotations. During the initial stages, we identified and addressed issues in the annotation process to improve overall quality. We implemented strict validation checks and trained annotators on expert-defined guidelines to ensure consistency in future iterations. Then, we scaled the annotation process to larger datasets while maintaining accuracy.

Annotation Process On average, there are 3 workers assigned to each annotation. During the labeling process, two annotators are responsible for labeling, while the third skilled worker verifies the evaluation results. In cases where there are disagreements among the two labeling workers, we collect the data and mark the disagreements. Any samples that have conflicting results are reviewed manually by another law professional. We also control the quality of workers by excluding the results made from workers whose work constantly appears to be inadequate. Other quality control measures

[15] `https://wenshu.court.gov.cn`

are implemented too. Prior to survey submission, the refined data will be examined by the Label Studio to verify the edits made by the workers. If incomplete sentences are identified, a message will be displayed, prompting workers to review their work.

In these ways, our labeled results show great reliability. After careful review, the inter-annotator agreement scores (specifically, inter-rater Spearman and Pearson Correlations) are found to be above 0.93, indicating a strong consensus.

C Dataset Reliability Analysis

We assess the reliability of our new dataset through manual review. Human workers were enlisted to assess whether the labeled data is aligned with the requirements and adhered to the schema. A multiple-choice questionnaire was created, consisting of fifteen labeled samples for scoring. The questionnaire included three possible responses: correct, neutral, and incorrect. Workers were asked to indicate whether the labeled samples were correct. Each participant received a compensation of approximately $8.5 per hour. The results showed that the majority of the workers found the labeled samples to be correct (ratio of correct 95%). This indicates that the labeled data aligns with the requirements and adheres to the schema.

D Bias and Ethics Statement

We use data sources that have been published on the official website. Although the case descriptions involve names, they do not contain key privacy information such as user contact information and identity ID. Note that if there is private information in the cases data, the parties can apply to the court to prevent its disclosure. Therefore, our dataset, based on publicly available data, does not involve an invasion of personal privacy.

All the participation in this study is completely voluntary, and the confidentiality and privacy of all participants will be diligently protected. In addition, in the process of manual labeling, we show the full content of the case description to the manual annotator, in order to prevent potential biases resulted from the LLM automatic labeling. We used legal professionals to conduct a test label and legal experts to assess whether bias was included. All annotators are trained to consciously avoid bias and unfairness.

We aim to use our methods to assist in case adjudication and to support the public. While we explore a clear process for determining facts, this does not imply that we encourage the use of LLMs to make final case decisions.

E The Toolkits Detail

E.1 Thought Process

The thought process and tool selection of the Agent are primarily controlled by LLMs, with the corresponding prompt illustrated in Figure 9. The first line clearly defines the problem the Agentis solving, the second line outlines the tool selection strategy, and the third line determines the termination conditions for TL Agent. Following this, the `{Goals}` field is used to input the objectives of the task along with relevant textual content. The `{Tools}` field enumerates the tools available for selection by the large model. The descriptions of these tools are generated by converting tool classes into textual representations using the Pydantic module, which includes the tool class name, class function, and the required input arguments.

Finally, the output format of the model is defined as a JSON-compliant string that can be successfully parsed by Pydantic. The returned JSON string include a "thinking" field for the model's reflections, which encompasses the thought content (text), reasoning (reasoning), and planning (plan) among other fields. The "tool" field specifies the name of the tool (name) to be invoked at the current step and the parameters to be passed (args).

This structured approach ensures a systematic and efficient decision-making process within the realms of artificial intelligence and deep learning, facilitating advanced computational tasks and analyses.

E.2 Fact Finding Head

The Fact Finding Toolkits comprise five distinct tools, each specifically designed for different TL subtasks. These tools are capable of generating results in parallel and formatting the outputs, thereby enhancing the operational efficiency of the Agent and improving the quality of task results. Moreover, they facilitate the effective utilization of the Agent's results in experimental testing scenarios.

The first two tools are utilized in Task 1. The Interim Probandum Finding Tool generates an Interim Probandum based on the content of legal documents, while the Ultimate Probandum Generation

```
You are TF agent, an AI assistant to solve complex legal problems.
It is necessary to call various tools according to different task objectives and complete the
objective as accurately as possible.
If you have completed all your tasks or reached end state, make sure to use the "finish" tool.
GOALS:
{Goals}

TOOLS:
{Tools}

Respond with only valid JSON conforming to the following schema, You must generate JSON as
output and not JSON schema:

{
    "type": "object",
    "properties": {
        "thoughts": {
            "type": "object",
            "properties": {
                "text": ......,
                "reasoning": ......,
                "plan": ......,
                ......
            },
            ......
        },
        "tool": {
            "type": "object",
            "properties": {
                "name": ......,
                "args": ......,
            },
            .......
        }
    },
}
```

Figure 9: The thought prompt of TL agent .

```
You are a professional legal documents analysis assistant,
please strictly follow the following requirements from the legal documents to extract the
complete information on the criminal evidence:

1. Position principle:
{Position-Principle}

2. Specification:
{Specification}

{Goals}

......

Respond with only valid JSON conforming to the following json schema. You should generate JSON
as output and not JSON schema.
{Json-Format}
```

Figure 10: The evidence extraction prompt of TL agent .

Tool produces the final Ultimate Probandum from the obtained Interim Probandum. Tools three and four are applied in Task 2; the Evidence Extraction Tool extracts criminal evidence from legal documents, and the Evidence Linking Tool connects the criminal evidence related to the Interim Probandum. The Experience Generation Tool is employed in Task 3, which generates human experience from evidence reasoning to Interim Probandum.

These tools are driven by LLMs, and the corresponding prompts are illustrated in Figure 10. Taking the Evidence Linking Tool as an example, the function of tool is clearly stated at the beginning. The `{Position Principle}` field informs the model of the potential positional and textual characteristics of the evidence, and the `{Specification}` field provides detailed specifications for the model's output. Finally, the model outputs a string that conforms to the `{Json-Format}` based on the objectives `{Goals}`.

This structured approach ensures that the outputs are precise and tailored for further analysis and application in the fields of law.

E.3 Multi-role Checker

The Multi-role Checker is designed to address issues raised by the Agent by providing solutions through analyses comment from different roles . These analyses are synthesized based on the task objectives and the results generated by the TL Agent in previous steps. The Multi-role Checker tool operates in two main phases: the

```
Given the following overall objective
Objective:
{Goals}

and the following issue that is the topics for the discussion, `{Issue}`.

and the following legal text that needs to be discussed, `{Legal_Text}`.


You are a professional lawyer who is familiar with and proficient in relevant laws
and regulations, case law and judicial interpretation. You are able to analyze
the facts of a case through logical reasoning, identify key points of application
of the law, and predict possible legal consequences.

You need to approach the objective from a lawyer's perspective to assess
whether the legal text is correct and reasonable,

If there is a mistake, please point out the wrong location and the reason issue for the
mistake in review.
```

Figure 11: The laywer prompt of TL agent .

```
Given the following overall objective
Objective:
{Goals}

and the following issue that is the topics for the discussion, `{Issue}`.

and the following legal text that needs to be discussed, `{Legal_Text}`.

From the police's perspective:
`{Police_Text}`

From the lawyer's perspective:
`{Lawyer_Text}`

From the people's perspective:
`{People_Text}`

You are a professional judge, you are well versed in the law and know the current laws
and regulations, judicial interpretations and case law. You have the good judgment to
weigh the evidence, analyze the facts, and make a fair decision. You've always been neutral.
You've always been impartial.

Please make a final decision based on the judgment of the above idea and discuss the final
results.
```

Figure 12: The judge decision prompt of TL agent .

first phase involves different roles analyzing the problem and proposing solutions, while the second phase involves a chief justice synthesizing these solutions to arrive at a final decision.

In the first phase, we have defined three distinct roles—lawyer, police office, and general public—to analyze the problem. The basic prompt format for this phase is illustrated in Figure 11. Here, the `{Issue}` field represents the question posed by the LLMs after deliberation based on previously generated results, and the `{Legal_text}` field contains the text content under discussion. Subsequent paragraph describe the characteristics associated with each role. The last paragraph details the requirements expected from each role.

Finally, as illustrated in Figure 12, we employ a prompt to consolidate the solutions proposed by the various roles. The judge role then synthesizes these inputs to deliver the final decision. This outcome is subsequently utilized to inform the Agent's subsequent thinking processes and tool selections, ensuring a coherent and well-considered approach to task execution.

E.4 Reflection

The Reflection Tool integrates the task objectives and the knowledge returned by relevant tools to analyze whether the input text can accurately fulfill the task objective. The specific design of the prompt is illustrated in Figure 13, where `{Goals}` represents our task objectives, `{Relevant_Tool_Response}` denotes the relevant knowledge returned by previously similar tools, and `{Input_Text}` is the text that requires reflection in conjunction with the task objectives and related knowledge. Additionally, the prompt emphasizes that the output must align with the format of `{Input_Text}`.

F Comparison with Advanced LLMs

We conduct a comprehensive comparison between our agent and state-of-the-art LLMs (including reasoning models). To further evaluate the performance of our agent, we also enhance the baseline LLMs by constructing few-shot prompts, which are

```
Given the following overall objective
Objective:
{Goals}

Relevant knowledge:
`{Relevant_Tool_Response}`

and the input text is
`{Input_Text}`

Please perform task by comprehends the Objective,
and according to the relevant knowledge to determine
whether the input text is accurate, If the  input text are inaccurate,
you need to be revised the input text to make it accurate.

the output must strictly follow the input text. Do not output any extra information.
eg. the input text is dict, the output text is also dict.
the input text is str, the output text is also str.
```

Figure 13: The reflection prompt of TL agent .

designed to improve their effectiveness.

We selected two LLMs as the foundational models for our agent: GPT-4o-mini and GPT-4o. The versions used are consistent with those specified in Section 5.1. For the evaluation, we randomly selected 10% of the data for evaluation.

As shown in Table 4, our agent achieves optimal results across all tasks. Specifically, the agent based on GPT-4 demonstrates outstanding performance in evidence reasoning (Task 2), indicating that models with larger parameter scales excel in tasks requiring logical reasoning and comprehension. However, in generative tasks such as fact probandum generation (Task 1), the agent based on GPT-4o-mini outperforms its got-4o counterpart, suggesting that smaller models may exhibit advantages in certain generation-oriented scenarios.

Furthermore, the experimental results reveal that the reasoning models (o1 and r1) outperform their base models (4o and v3) in Task 2, highlighting their enhanced capability in reasoning-intensive tasks. Conversely, in tasks more focused on generation (Task 1 and Task 3), the reasoning models underperform compared to their base models (4o and v3). This contrast underscores the importance of model architecture and scale in task-specific performance, particularly in balancing reasoning and generative capabilities.

Additionally, introducing examples to construct few-shot prompts effectively improves the model's performance on our benchmark. This demonstrates the utility of few-shot learning in enhancing task-specific adaptability and overall effectiveness.

G Ablation Study

In this section, we randomly selected 10% of the data from the test set for evaluation to the impact of removing different tools from the toolkit on the agent's performance. The base model of the TL agent is GPT-4o-mini. Additionally, since the Fact Finding Head and Reflection Tool serve as the core tools and foundational experimental tools for the agent, ensuring the formatted output of the agent's results, the absence of these two tools would prevent the agent from producing formatted JSON data necessary for experimental validation. Therefore, we did not conduct ablation studies on these two tools.

As shown in Table 5, the results of the ablation study indicate that each tool in our agent contributes positively to the task outcomes. Specifically, we observed that removing knowledge-based tools (Legal Knowledge and Knowledge Search) led to a significant decline in performance across all tasks, suggesting that the incorporation of domain-specific legal knowledge through these tools effectively enhances the agent's task execution. Furthermore, we found that removing the Multirole Checker tool also resulted in a noticeable decrease in the agent's task performance, indicating that the inclusion of multi-role judgment significantly improves the accuracy of task results.

H Future Work

Despite our first try, challenges persist in AI adoption within the legal domain. Issues such as data privacy, imperceptible bias, the interpretability of AI enhancement, and the impact on traditional legal practices warrant further investigation. Future research directions involve addressing these challenges, enhancing interpretability and fostering interdisciplinary collaborations between AI agents and legal professionals.

Approach	Task I			Task II			Task III			All
	S_{fact}-*1*	S_{fact}-*2*	S_{fact}-*l*	*Pre*	*Rec*	F_{evid}	S_{exp}-*1*	S_{exp}-*2*	S_{exp}-*l*	S_c
Claude3.5-sonnet-10	33.97	20.08	29.85	1.6	3.19	2.13	17.59	1.34	13.93	24.09
Claude3.5-sonnet-10 3 shot	36.69	20.86	32.25	2.87	5.64	3.8	17.77	2.63	12.54	27.03
ChatGPT-4o	31.93	18.32	28.43	3.65	8.51	5.12	27.37	8.08	21.99	32.52
ChatGPT-4o 3 shot	36.26	21.26	31.36	10.74	19.48	13.85	26.56	7.83	20.45	36.52
ChatGPT-o1	30.56	17.63	27.61	7.54	23.84	11.45	23.56	5.45	17.63	32.25
ChatGPT-o1 3 shot	34.7	19.18	29.43	11.97	32.3	17.48	25.71	6.96	19.94	35.46
Deepseek-V3	30.23	15.12	26.13	7.2	14.35	9.59	29.23	9.54	23.97	32.62
Deepseek-V3 3 shot	31.3	15.69	26.48	9.86	17.86	13.48	29.78	9.86	24.46	34.57
Deepseek-R1	29.74	12.3	25.46	10.78	21.25	14.3	21.58	2.94	16.53	32.38
Deepseek-R1 3 shot	31.94	14.55	27.28	14.34	27.43	18.83	22.5	3.7	17.39	36.01
TL Agent (4o-mini)	**37.92**	**21.60**	**33.83**	8.38	39.48	13.83	31.92	9.32	25.49	**36.62**
TL Agent (4o)	34.88	21.37	29.86	**16.61**	**25.38**	**20.08**	**32.87**	**10.94**	**25.98**	36.11

Table 4: The results of advanced model

Approach	Task I			Task II			Task III		
	S_{fact}-*1*	S_{fact}-*2*	S_{fact}-*l*	*Pre*	*Rec*	F_{evid}	S_{exp}-*1*	S_{exp}-*2*	S_{exp}-*l*
TL Agent	37.92	21.60	33.83	8.38	39.48	13.83	31.92	9.32	25.49
- Pattern Match	37.65	21.45	33.53	7.65	38.44	12.76	31.33	9.46	24.45
- Multirole Checker	36.83	20.55	32.12	6.23	37.67	10.69	31.94	10.03	25.33
- Legal Knowledge	35.70	19.56	31.98	8.35	37.23	13.64	30.56	8.68	23.80
- Knowledge Search	35.65	19.23	31.79	6.93	38.12	11.72	-	-	-
- Emotion Check	37.12	20.85	32.95	-	-	-	-	-	-

Table 5: Ablation Study.

I Related Work

In the realm of judicial proceedings, the process can often be categorized into two fundamental phases Duxbury (1995); Merryman and Pérez-Perdomo (2018): (1) **Law Reasoning**, involving the determination of factual circumstances within a case; and (2) the **Law Application**, which pertains to the utilization and application of relevant legal statutes and principles to those identified facts. Thus in this section, we review the current state of AI technology in these two subfields.

I.1 AI for Law Application

Law application refers to the process of applying the law. This involves determining the circumstances of the case, selecting the appropriate legal norm to be applied, interpreting the meaning of the chosen legal norm, and issuing a document that applies the legal norm to the relevant person or organization[16].

Applying automated techniques to address a legal issue has a rich history. It can be traced back to early systems based on mathematics, such as Kort (1957); Keown (1980); Lauderdale and Clark (2012), which focus on analyzing cases using mathematical tools. Besides that, there are two main categories of AI approaches applied to Law: logic-based and data-driven approaches. The logic-based approach was introduced by (Allen, 2013), with its first appearance dating back to the 1980s. Around the same time, data-driven approaches were demonstrated by the HYPO system (Ashley, 1991). Some research has concentrated on addressing logical issues in legal documents and aims to clarify legal terminology, thereby contributing to the field of logic and the interactive representation of legal texts (Branting, 2017). Additionally, a comprehensive description logic framework, proposed by Francesconi (2014), builds upon Hohfeldian relations and effectively represents legal norms. Given that new laws are constantly introduced and existing laws are modified, the legal system is inherently dynamic, necessitating the establishment of an adaptable and modifiable model. To address this, the extension of defeasible logic has been widely employed and has yielded promising results (Governatori and Rotolo, 2010; Governatori et al., 2007, 2005). Another significant challenge in the legal domain is analyzing vast repositories of case law, which poses an obstacle for legal pro-

[16] https://encyclopedia2.thefreedictionary.com/Application+of+Law

fessionals. Data-driven approaches are well-suited to tackle this issue. These approaches, employing techniques such as text mining (Avgerinos Loutsaris et al., 2021), information retrieval (Sansone and Sperlí, 2022), and semantic analysis (Merchant and Pande, 2018), strive to extract valuable insights from legal texts and make judgments. Furthermore, the rise of conversational AI technologies has resulted in the creation of legal chatbots (Queudot et al., 2020), which provide users with the ability to access legal information, receive help in completing legal documents, and receive guidance throughout legal processes through natural language interactions. In recent years, large language models (**LLMs**) have emerged as powerful tools, with several models specifically tailored to the legal domain being proposed (Chalkidis et al., 2020; Song, 2023; Yuan Group, 2023; Colombo et al., 2024; Fei et al., 2024; Haitao Li, 2024). Concurrently, there has been a surge in the exploration of legal applications, including legal text comprehension and generation, legal documents analysis, legal violations detection, legal judgment predictions and etc. (Huang et al., 2020; Xu et al., 2020; Gan et al., 2023; Wu et al., 2023; Roegiest et al., 2023; Bernsohn et al., 2024; Cao et al., 2024). Still, these applications can only offer assistance to judges in duanting legal tasks but cannot delve deeply into the core of the judical process, which is rigorously revealing the ultimate fact through law reasoning.

I.2 AI for Law Reasoning

AI technology for law reasoning is relatively less common compared to law application, due to the complexity of structured law reasoning information. Structured facts exist in unstructured formats; teaching AI systems to extract accurate structured information at various levels is a complex task that requires sophisticated algorithms. Neglecting the structured law reasoning stage and directly employing the law application may lead to regulatory and compliance issues. Moreover, making all the facts, and shreds of evidence, as well as the reasoning process visible, makes AI applications more reliable.

Compared to AI technology for law reasoning, the legal field has explored the simulation of evidential reasoning. As early as the late 1980s, Anne Gardner applied artificial intelligence techniques to the legal reasoning process and proposed a law reasoning program in Chapters 6&7 of (Gardner, 1987). In 1997, the book of (Prakken and Sartor, 1997) edited by Henry Prakken, provided a systematic account of how logical models of legal argumentation operate in legal reasoning. The book discusses various models of legal reasoning, including dialogical models, and provides a detailed analysis of the operation of different non-monotonic logics (such as logic programs, qualified reasoning, default logics, self-cognitive logics, etc.) in legal reasoning. According to Henry Prakken, these logics are of great importance for the development of artificially intelligent legal systems. In 2009, the book of (Walton, 2009), edited by Henrik Kaptein, Henry Prakken, and Bart Verheij, proposed three practical ways of evidentiary reasoning: the statistical approaches, the storytelling-based approaches, and the argumentative approaches. Of these, the exploration of evidential reasoning with AI focuses on the latter two. Chapters 2&3 illustrate the statistical approaches, Chapters 4&5&6 describe the storytelling-based approaches, Chapters 7&8 compares storytelling-based approaches and argumentative approaches, and eventually, Chapters 9&10 systematically describe the argumentative approaches. Floris J. Bex (Bex, 2011) attempt to construct a hybrid theory of evidence based on argumentation and explanation, raises the issue of legal reasoning in AI. All the above literature discusses the theoretical possibilities of combining AI technology with legal reasoning theory, and suggests that the analysis of these argumentation patterns could be the logical architecture of an AI legal system. Till 2012, Ephraim Nissan (especially Volume 1, Chapter 3 of (Nissan, 2012)) attempts to introduce the Wigmore diagram and the Turmin logical argumentation model into computer programs, and attempts to place them under the calculation of certain AI systems. Then in 2017, Floris J. Bex and Serena Villata (Wyner and Casini, 2017) introduced and summarized the application of AI technology in the field of legal systems, especially the integration of AI technology and legal argumentation. It is evident that the ongoing research on legal systems for artificial intelligence has now reached the stage of developing models for evidence reasoning.

J Legal Cases Examples

To highlight the significance of law reasoning, we provide examples that are widely recognized where different judicial facts, evidence, and experience have impacted different results. Recognizing these instances is crucial for maintaining public trust. We provide an notable example in Figure 1. The *Rex v. Bywaters and Thompson* is one of England's most famous causes (Anderson et al., 2005). The *Rex v. Bywaters and Thompson* is one of England's most famous causes c´el'ebres (Anderson et al., 2005). The case is an example of rough as well as speedy "justice." On January 9, 1923, Frederick Bywaters and Edith Thompson were hanged for the murder of Edith's husband Percy, just three months and six days after his death. Public opinion at the time and subsequent commentators have been divided on the question of whether Edith had instructed her mistress Bywaters to kill her husband. All evidence indicates that Edith's marriage with her husband was unhappy. They met a young steward named Bywater on a cruise ship by chance. After that, they exchanged letters frequently. Although some of the letters were burned, Edith met a young steward named Bywater on a cruise ship by chance. All evidence indicates that Edith and Bywater had already reached the most reprehensible intimate relationship. Before Bywater killed Edith's husband, although there is no direct evidence that Edith instigated Bywater to kill her husband, due to the possibility of their meeting before the crime and the evidence of their shared interests, with the usual experience of "the elderly tend to dominate", Edith was sentenced to death. However, some commentators think that if we consider that Edith is an older woman, according to common sense, she would often take on the role of a mother, and because she is a woman who is less inclined to dominate others, she would not make such an inciting behavior.

Similar to the case of "Rex v. Bywaters and Thompson", in the case of "Nanjing Pengyu"[17], the judge applied the wrong experience that a person would not help someone who had fallen, but only the person who caused the fall. This erroneous experience resulted in a miscarriage of justice.

K Case Study

We randomly selected a few representative cases (1–3) from the experimental results of sub-tasks 1 through 3 to evaluate the effectiveness of the TL Agent and analyze specific errors observed during the process.

Sub-task 1: Factum Probandum Generation

In the sub-task 1, we show some cases in Figure 14 and Figure 15. the TL Agent demonstrated certain limitations. For instance, in case 1 of F_2, the TL Agent extracted information related to the suspect that was not directly relevant to the interim probandum of the crime. Additionally, while generating crime-related interim probandums, the Agent included elements of evidence in the output, which should have been excluded. In case 2 of F_2, the portions marked in red illustrate a crime-related analysis but fail to represent the actual factum probandum of the crime. This suggests the need for stricter differentiation between crime-related facts and supporting evidence during the generation process.

Sub-task 2: Evidence Reasoning

The evidence reasoning task revealed another type of error. we show some cases in Figure 16 and Figure 17. In case 1 of V_1, the TL Agent incorrectly extracted text resembling evidence, which turned out to be an analysis of testimonies provided by individuals such as Ba X and others. This analysis should not be classified as evidence, as it does not align with the proper format or content expected of evidence. More accurate examples of the evidence format can be seen in cases 2 of V_1 and V_2, where the output adheres to the expected standards.

Sub-task 3: Experience Generation

In the Experience Generation task, a comparison between the TL Agent's generated results and the Ground Truth highlighted specific challenges. we show some cases in Figure 18 and Figure 19. For case 1, the TL Agent's output predominantly represented a reasoning process from evidence to facts, rather than human experiential knowledge. This limitation may stem from the model's current inability to effectively replicate human experiential reasoning. However, in case 2, the Agent demonstrated improved performance in producing experiential knowledge for typical DUI cases. In this scenario, the crime-related experiential knowledge generated by the Agent closely matched the Ground Truth, suggesting that the Agent has a degree of competency in generating experiential knowledge for common scenarios.

In summary, while the TL Agent has shown some promising results in certain cases, the observed errors underscore areas that require refinement. Addressing these limitations is critical for enhancing the Agent's ability to accurately differentiate between facts, evidence, and experiential knowledge.

[17] https://www.chinacourt.org/article/detail/2014/07/id/1352051.shtml

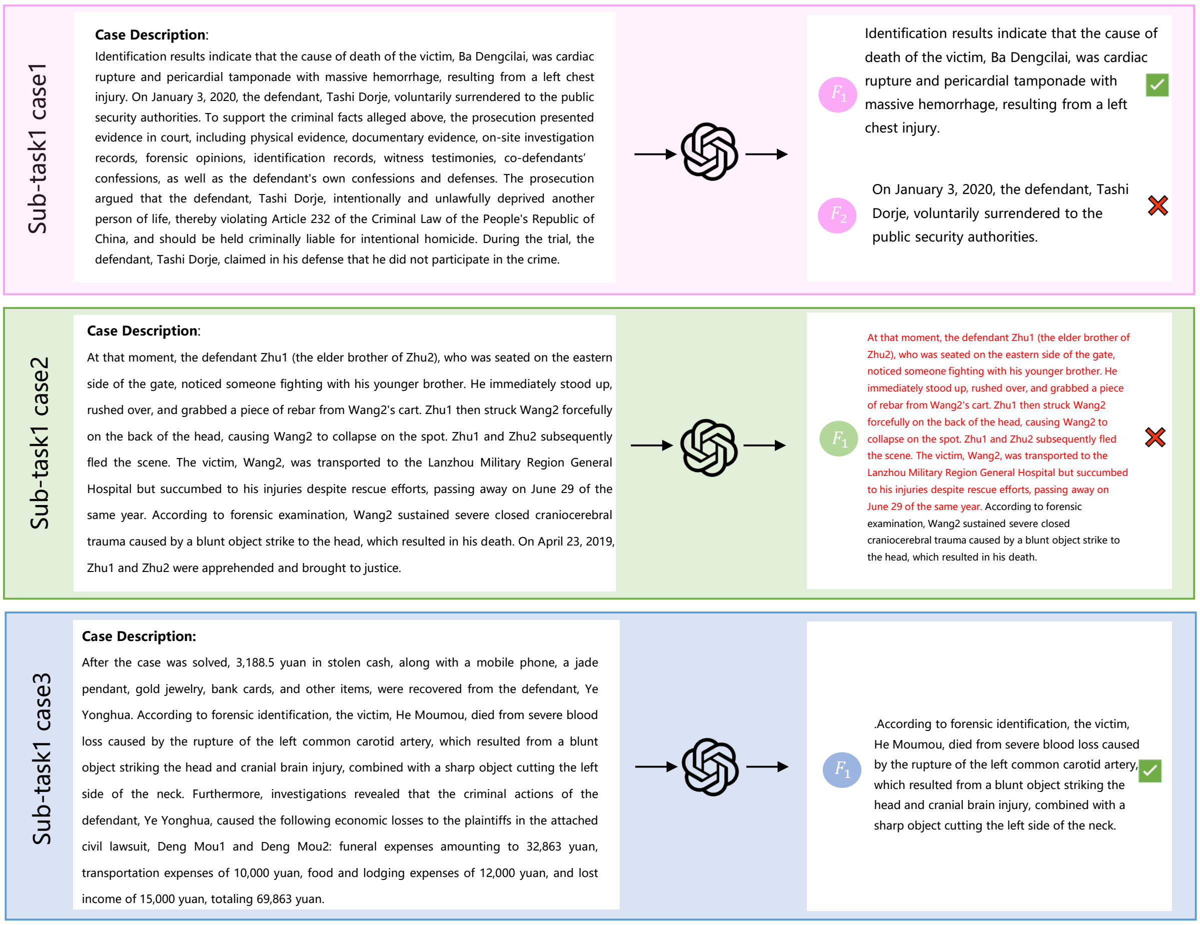

Figure 14: Case study of sub-task 1 in English.

Figure 15: Case study of sub-task 1 in Chinese.

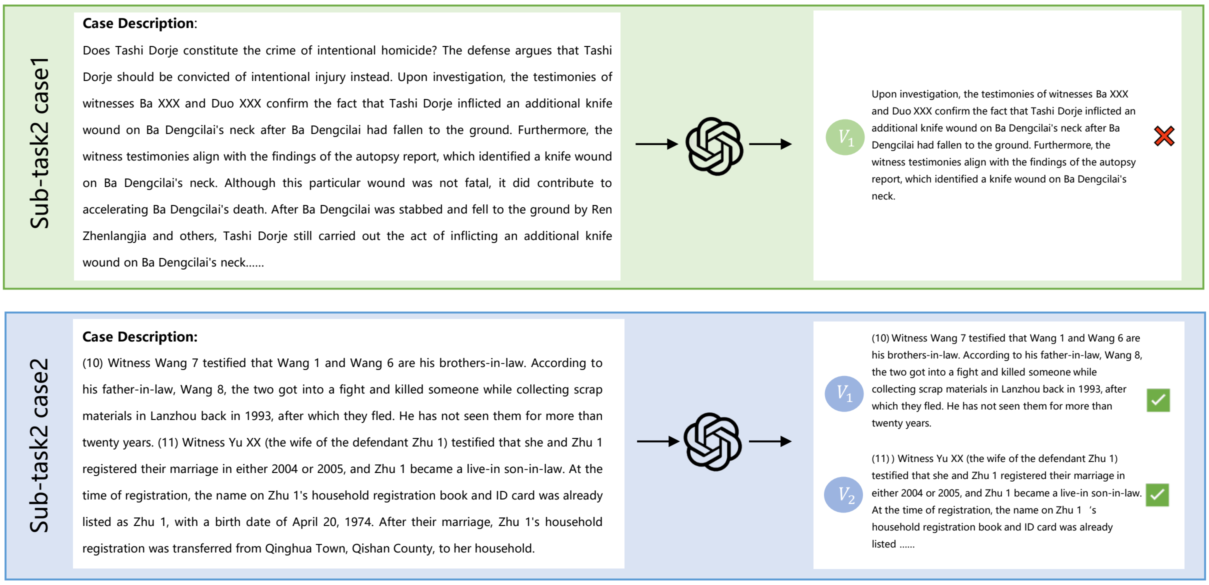

Figure 16: Case study of sub-task 2 in English.

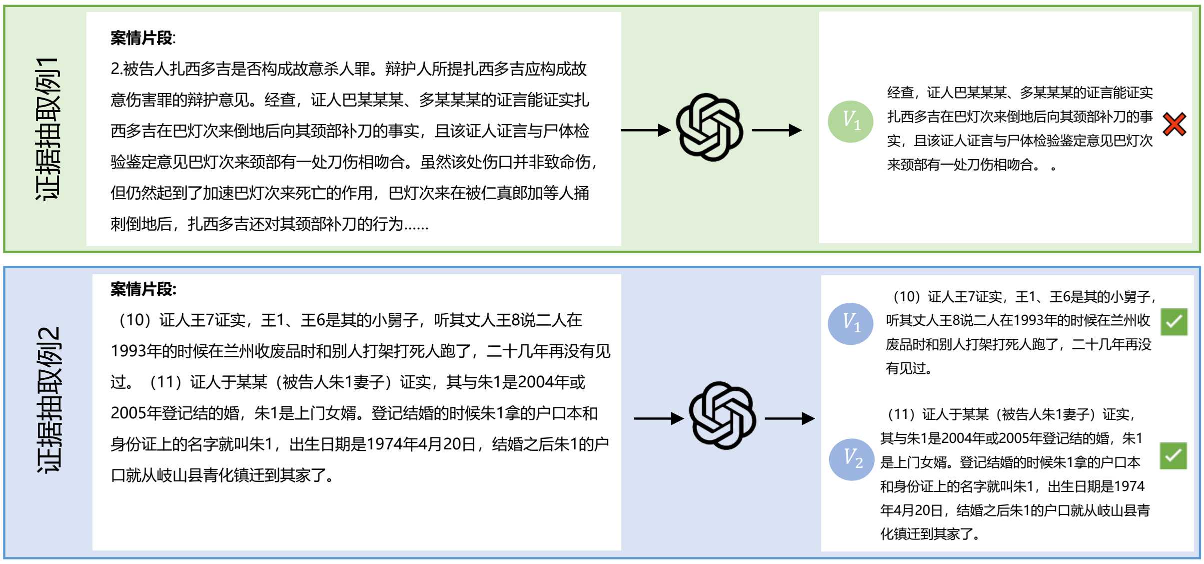

Figure 17: Case study of sub-task 2 in Chinese.

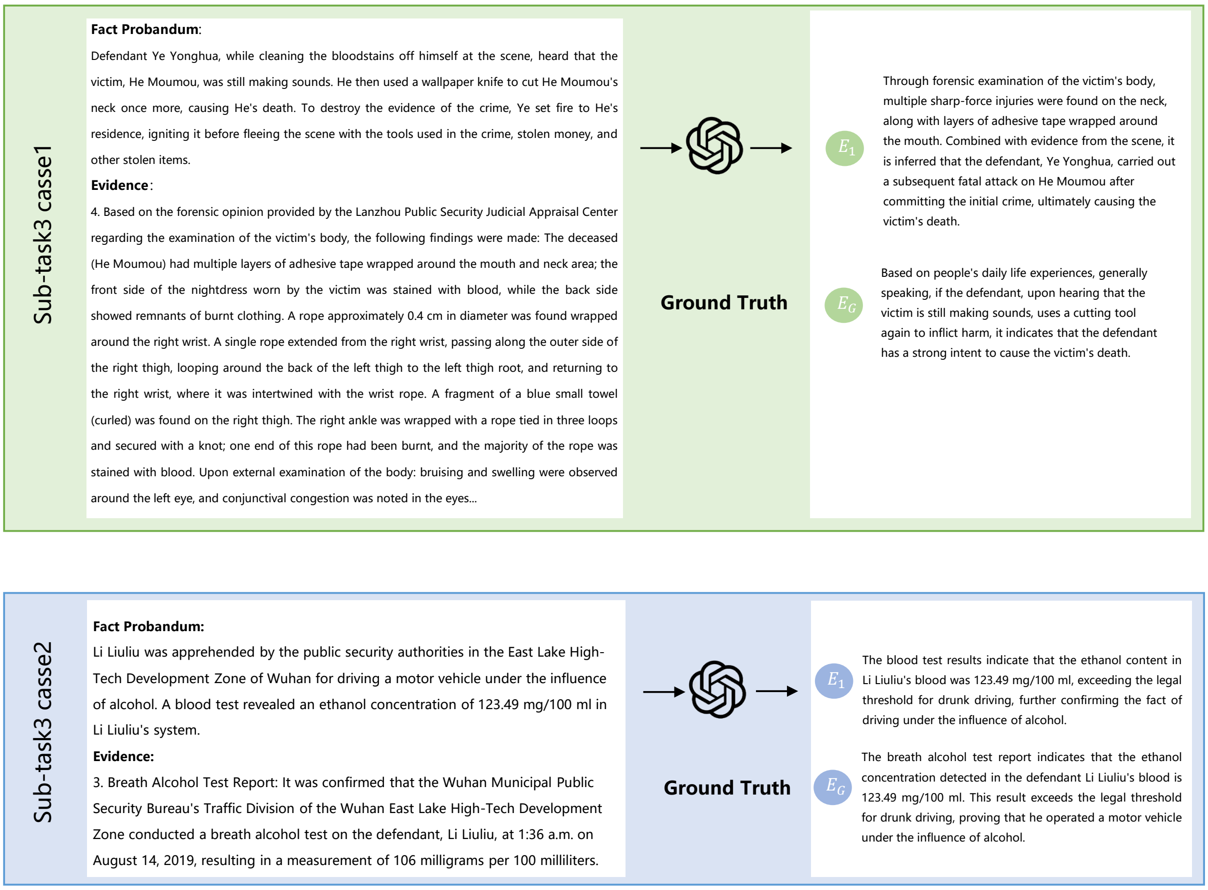

Figure 18: Case study of sub-task 3 in English.

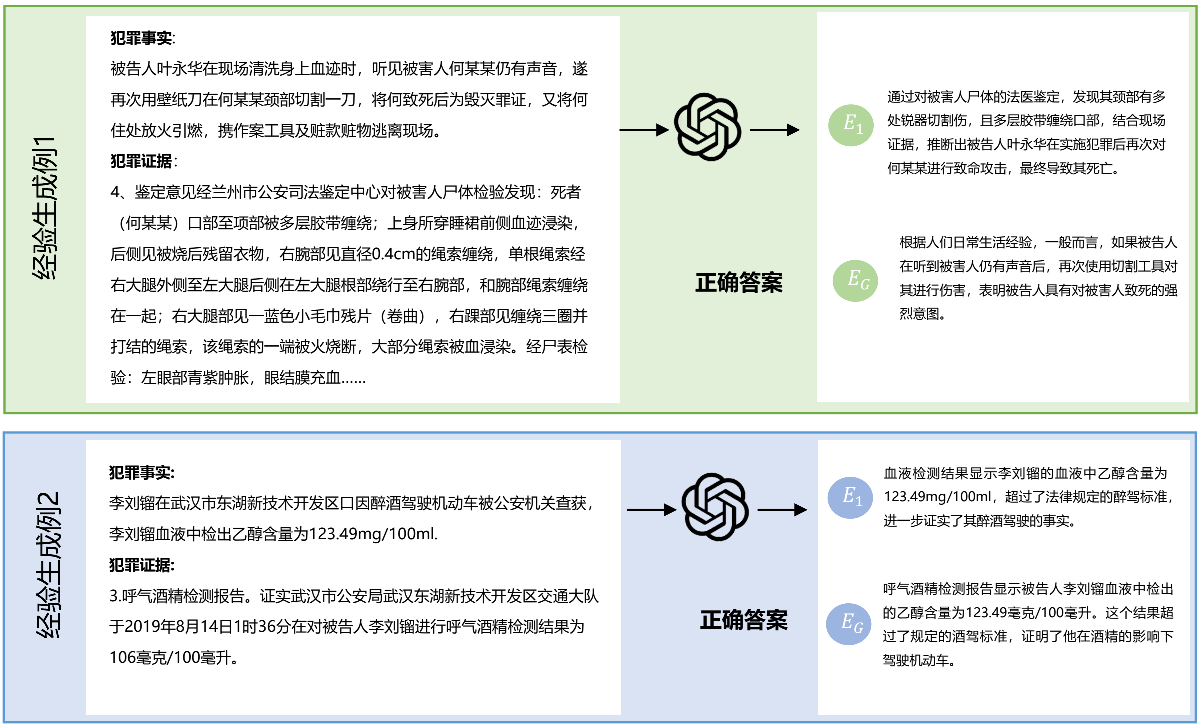

Figure 19: Case study of sub-task 3 in Chinese.