Fine-tuning Large Language Models for Improving Factuality in Legal Question Answering

Yinghao Hu[1], Leilei Gan[1,*], Wenyi Xiao[1], Kun Kuang[2], Fei Wu[2]
{huyinghao, leileigan, wenyixiao, kunkuang, wufei}@zju.edu.cn,
[1]School of Software Technology, Zhejiang University
[2]College of Computer Science and Technology, Zhejiang University

Abstract

Hallucination, or the generation of incorrect or fabricated information, remains a critical challenge in large language models (LLMs), particularly in high-stake domains such as legal question answering (QA). In order to mitigate the hallucination rate in legal QA, we first introduce a benchmark called LegalHalBench and three automatic metrics to evaluate the common hallucinations when LLMs answer legal questions. We then propose a hallucination mitigation method that integrates behavior cloning and a novel Hard Sample-aware Iterative Direct Preference Optimization (HIPO). We conduct extensive real-data experiments to validate the effectiveness of our approach. Our results demonstrate remarkable improvements in various metrics, including the newly proposed Non-Hallucinated Statute Rate, Statute Relevance Rate, Legal Claim Truthfulness, as well as traditional metrics such as METEOR, BERTScore, ROUGE-L, and win rates. Our benchmark and model are available at https://github.com/YinghaoHu/LegalHalBench.

1 Introduction

Recently, large language models (LLMs) have demonstrated notable advancement and have been applied to a broad spectrum of natural language process (NLP) tasks across different domains, such as medical (Zhou et al., 2023), financial (Wang et al., 2023) and legal domains (Trautmann et al., 2022). Among them, the application of LLMs in the legal field can help legal practitioners substantially improve their productivity in completing daunting tasks such as legal judgment prediction (Gan et al., 2021a; Masala et al., 2024; Li et al., 2024a), legal question answering (Gan et al., 2021b; Louis et al., 2024), generation of judicial opinions (Zhou et al., 2024; Li et al., 2024b; Liu et al., 2024) and legal document drafting (Blair-Stanek et al., 2023; Choi et al., 2021; Hargreaves, 2023).

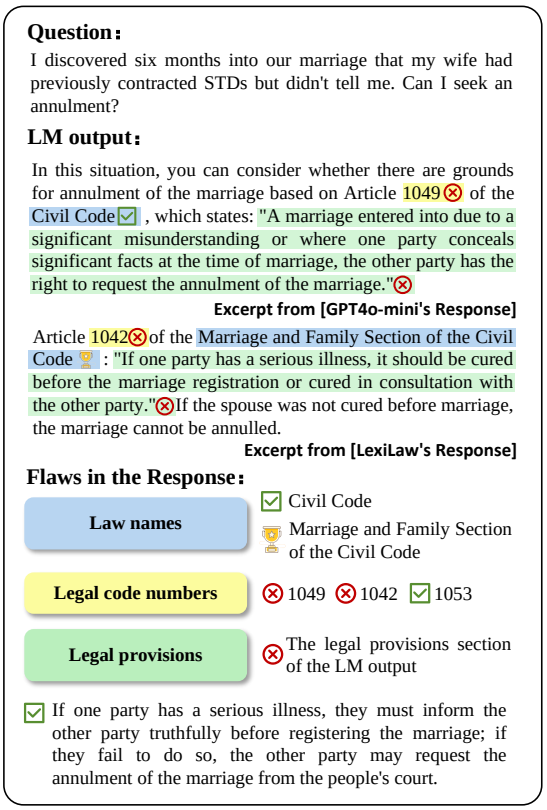

Figure 1: Hallucinations of LLMs in the legal question answering task.

Despite showing promising benefits, general LLMs are usually troubled by the so-called hallucination problem which refers to the scenario of LLM generating responses that are inconsistent with the real-world legal facts (Dahl et al., 2024). To improve legal factuality of LLMs, some specialized LLMs for the legal domain have been proposed (Pengxiao Song, 2023; Li, 2023; Yiquan et al., 2024) by either continually training general LLMs (Yiquan et al., 2024) on massive legal datasets or by employing retrieval-augmented generation technique (Huang et al., 2023b). In this study, we focus on the former approach as it can inherently mitigate hallucinations by injecting legal

*Corresponding author

Proceedings of the 31st International Conference on Computational Linguistics, pages 4410–4427
January 19–24, 2025.

knowledge into the models. Successful progress has been made in this direction. For example, the Lawyer LLaMA (Huang et al., 2023b) model introduces a legal statute retriever for marriage-related issues, while ChatLaw2-MoE (Cui et al., 2023) employs a mixture of experts (MoE) model and a multi-agent system to improve factuality for legal domain applications.

However, the issue of hallucination in LLMs within the legal domain, particularly in the task of legal question answering (QA), remains a significant challenge. Legal QA requires not only the memorization of legal knowledge but also the ability to synthesize this knowledge into factual responses. For example, as shown in Fig. 1, when answering a legal question, both the general LLM GPT-4o-mini and the specialized LLM LexiLaw models make fabricated content on the citation of law provisions and generate claims which were not in line with the actual law provisions. The inaccuracy of these LLM-based legal QA substantially impedes their practical application, as the high-stakes legal domain demands faithful and accurate interpretations of the law, along with harm-free and precise legal advice. Surprisingly, few studies have thoroughly addressed the LLM hallucination problem in legal QA. Moreover, there is a clear lack of tailored hallucination evaluation metrics and benchmarks in the literature.

In this paper, to improve the factuality in legal QA, we first develop a legal hallucination evaluation benchmark (LegalHalBench), including various automatic metrics for detecting five types of common hallucinations which are often generated by LLMs. Then, a hallucination mitigation method is proposed to help LLMs cite correct law provisions and generate factual claims. This method includes two stages: (1) a supervised fine-tuning (SFT) stage and (2) a hard sample-aware iterative direct preference optimization stage. Current studies have demonstrated that both the SFT (Huang et al., 2023a) and preference learning techniques (Saeidi et al., 2024; Schulman, 2023) can help mitigate hallucinations in LLMs. Additionally, in order to generate a large-scale legal QA dataset for training LLMs, we propose an automatic legal QA dataset curation approach to help alleviate the manual annotation cost and improve the scalablity of the datasets.

We conduct extensive experiments on the curated LegalHalBench using our newly introduced hallucination evaluation metrics. The experimental results demonstrate the effectiveness of the proposed two-stage fine-tuning strategy. Our method achieves a non-hallucinated statute rate of 38.353%, significantly surpassing specialized LLMs such as WisdomInterrogatory and LexiLaw, as well as general models like Llama3.1 405B and GPT-4o. Furthermore, our model exhibits improvements of 37.13% in statute relevance rate and 6.56% in legal claim truthfulness compared to the vanilla version of the base model. Additionally, helpfulness evaluation experiments reveal that our method achieves a dominant win rate compared to existing legal LLMs and general-purpose LLMs. The meticulously designed ablation studies further demonstrate the effectiveness of each proposed component. The human consistency experiment, conducted to evaluate the legal hallucination metrics, supports the reliability of our proposed metrics.

2 Related Work

2.1 Large Language Models in Legal Domain

Based on the idea of LLM training, various pre-trained LLMs for legal QA have been proposed to help general LLMs comprehend legal terminology and learn legal knowledge (Chalkidis et al., 2020; Zheng et al., 2021; Pengxiao Song, 2023; Li, 2023; Huang et al., 2023b; Cui et al., 2023). After training, these specialized models have been employed to complete various legal tasks such as predicting legal judgments, analyzing legal documents, and drafting legal texts (Iu and Wong, 2023; Macey-Dare, 2023; Oltz, 2023; Gan et al., 2023). For example, LawGPT (Pengxiao Song, 2023) is built by continually training Llama (Touvron et al., 2023) on extensive legal QA datasets. Based on ChatGLM-6B, LexiLaw (Li, 2023) is trained on a mixup dataset from general domain and legal domain. Nevertheless, these legal-specific LLMs such as LawGPT, LexiLaw, Lawyer LLaMA, and ChatLaw, as well as the general-purpose LLMs like Qwen2 (Yang et al., 2024), GLM4 (GLM et al., 2024), GPT-4 (OpenAI, 2024), and Llama 3.1 (Dubey et al., 2024b) still occasionally produce misleading or erroneous responses.

2.2 Hallucinations in Large Language Models

LLM Hallucination (Schulman, 2023; Zhang et al., 2023) refers to the phenomenon where the generated responses of LLMs are factually incorrect, or not grounded with given contexts. To alleviate

the LLM hallucination problem, there are mainly two lines of research work. One is based on the retrieval-augmented generation techniques (Lewis et al., 2020; Guu et al., 2020) which can retrieve relevant information to address the knowledge gap. The other is training-based approaches which seek to reduce hallucinations in LLMs via further training, such as instruction fine-tuning (Elaraby et al., 2023) or preference learning (Tian et al., 2023; Yuan et al., 2023; Ethayarajh et al., 2024; Xiao et al., 2024b,a). For example, Elaraby et al. (2023) proposes to use an SFT model with a curated, domain-specific dataset to alleviate hallucinations. In preference learning, Proximal Policy Optimization (PPO) (Schulman et al., 2017) and Direct Preference Optimization (DPO) (Rafailov et al., 2023) have been exploited to enhance the factuality of LLMs (Lee et al., 2022) as well as various adaptations, including Self-Rewarding (Yuan et al., 2024), Iterative DPO (Xu et al., 2023; Xiong et al., 2023), and SimPO (Meng et al., 2024). In this work, we particularly focus on the factuality issues of the LLMs in legal QA and propose a two-stage fine-tuning model combining the SFT and hard sample-aware iterative DPO techniques to effectively mitigate hallucinations of the LLMs.

3 Legal Hallucination Definition, Benchmark and Evaluation Metrics

To improve the factuality of the LLMs when addressing legal queries, we first introduce five commonly generated hallucination types. Then, we design the respective metrics for evaluating these specific hallucinations.

3.1 Hallucinations in Legal Question Answering

We define five hallucination types which are often generated by LLMs.

Incorrect law name. This error occurs when the name of a law is incorrectly stated. For example, mistakenly referring to the "Clean Air Act" as the "Air Protection Act."

Incorrect legal code number. This happens when a specific section or provision of the law is cited with the wrong number. For example, citing "Section 302" when it should be "Section 303."

Fabrication of legal provision. This involves making up a law or legal provision that does not actually exist. For example, referencing a non-existent "Article 15 on Data Privacy" in a law where no such article exists.

Incorrect citation of legal provision. This type of error occurs when the cited legal provision is correct but irrelevant to the issue at hand. For example, citing a provision about road safety in a question that pertains to inheritance.

Suggestions that contradict regulations. This type of error involves providing advice or recommendations that directly contradict the existing legal rules or regulations, for example, proposing a course of action that is explicitly prohibited by the law.

3.2 Legal Hallucination Benchmark

To thoroughly evaluate the phenomenon of hallucination in LLMs when addressing legal queries, we introduce the first benchmark for legal hallucination detection, designated LegalHalBench. This benchmark encompasses a range of scenarios from both civil and criminal law, including *Inheritance Law*, *Road Traffic Safety Law*, *Marriage Law*, *Tort Liability Law*, *Real Rights Law*, *Lending Law*, and *Criminal Law*. These laws are selected based on their prevalence according to the statistical distribution reported by Chen (2023).

The questions and reference answers in LegalHalBench are constructed as follows: (1) For civil questions, we primarily gather inquiries from phone consultations and online queries conducted at a law firm. The reference answers are initially generated by an LLM (e.g., GPT-4-turbo) and subsequently subjected to rigorous review and refinement by legal experts. Next, legal professionals are asked to provide authoritative legal statutes to be included in the reference answers. Additionally, questions shorter than 20 words are filtered out, and unclear inquiries are rephrased to ensure they are answerable. (2) For criminal questions, we source them from criminal judgment documents issued by the courts. Given the highly structured nature of criminal judgments, we can directly use regular expressions to extract necessary question information from sections such as "Defendant's Basic Information" and "Facts of the Prosecution", and answers from sections like "Court's Opinion" and "Judgment", including relevant legal articles and judgment outcomes, respectively. The extracted information is then combined with our designed instructions.

Finally, LegalHalBench consists of a total of 1,988 questions, covering over 800 distinct legal provisions. The dataset is structured in the format question, reference statutes, reference answers. The statistical distribution of LegalHalBench is presented in Tab. 9. The cost of constructing this dataset is detailed in Sec. A.13.

3.3 Legal Hallucination Evaluation Metrics

Manually evaluating model-generated responses for hallucinations is costly and time-consuming. Inspired by Min et al. (2023), we propose the following metrics to automatically detect hallucinations in the responses.

Non-Hallucinated Statute Rate. We define the Non-Hallucinated Statute Rate (NHSR) as follows: Let $\hat{y}$ denote the statutes generated by the model. The NHSR is the proportion of non-hallucinated statutes among all statutes in $\hat{y}$. A statute is considered non-hallucinated if its name, number, and content are entirely accurate when compared to the golden reference statute.

$$\text{NHSR} = \frac{\sum_{i=1}^{N} \mathbb{I}(S_{\text{name},i} \wedge S_{\text{number},i} \wedge S_{\text{content},i})}{N} \tag{1}$$

Here, $\mathbb{I}()$ and N represent an indicator function and the total number of statutes in $\hat{y}$. $S_{\text{name},i}$, $S_{\text{number},i}$, and $S_{\text{content},i}$ represent whether the statute's name, number, and content are correct, respectively. The values of $S_{\text{name},i}$, $S_{\text{number},i}$ and $S_{\text{content},i}$ are obtained by first extracting statutes from $\hat{y}$ via prompting LLMs and then comparing them with each part of the most similar golden statute. A more formal calculation process is provided in the Sec. A.3.

Statute Relevance Rate. This metric measures the relevance between the knowledge contained in legal statutes and the legal question. We denote this metric as *Rel*. The value of *Rel* ranges from 0 to 10, where values closer to 0 indicate that the knowledge in the statutes is less relevant to the question, while values closer to 10 suggest a higher relevance. The technical details and calculation formula can be found in the Sec. A.4.

Legal Claim Truthfulness. This metric measures the truthfulness of claims in the model-generated answers. We denote this metric as T_{LC}. The T_{LC} ranges from 0 to 10, where a score closer to 10 indicates higher truthfulness, and a lower score suggests a greater likelihood of unfounded or incorrect legal claims. The technical details and calculation formula can be found in the Sec. A.5.

4 Method

To improve the factuality of LLMs for legal question answering, we propose a two-stage fine-tuning algorithm, including SFT and hard sample-aware iterative direct preference optimization techniques. Current studies have demonstrated that both SFT (Huang et al., 2023a) and preference learning techniques (Saeidi et al., 2024; Schulman, 2023) can help mitigate LLM hallucinations.

4.1 Training Dataset Construction

At present, there is a lack of large-scale QA datasets that include accurate citations of legal provisions suitable for fine-tuning. Therefore, as illustrated in Fig. 2, we propose a two-step automated construction method to initially curate such a dataset, which is not only cost-effective but also scalable.

Legal QA data from publicly available datasets. We use the CAIL2018 legal QA dataset (Xiao et al., 2018) as one of the data source, which comprises questions derived from real-world scenarios. However, we find that the provided answers are neither informative nor cite relevant statutes. To address this issue, we turn to proprietary LLMs to generate high-quality responses, as shown in Fig. 2. Nonetheless, due to lack of legal knowledge, proprietary LLMs such as GPT-4-turbo may also fabricate legal statutes. To mitigate this, we first extract statutes from the LLM-generated response and use these extracted statutes as queries to search the most similar statute in an authentic statute database. We then prompt GPT-4-turbo with the most similar statute to revise its generated response accordingly. Detailed prompts for extracting statute are provided in Sec. A.2.

Through the aforementioned procedures, we obtain 12,149 legal QA samples with answers that are not only informative but also include accurate citations of legal provisions. For a detailed evaluation of the quality of the training data, refer to Sec. A.1.

Legal Provisions based Legal QA dataset Construction. While the initial portion of the dataset is of high quality, it inherently lacks comprehensive coverage of all legal provisions due to the extensive number of provisions. Additionally, we aim to instruct the model on citing specific legal provisions. To address this, we propose a legal provision-based

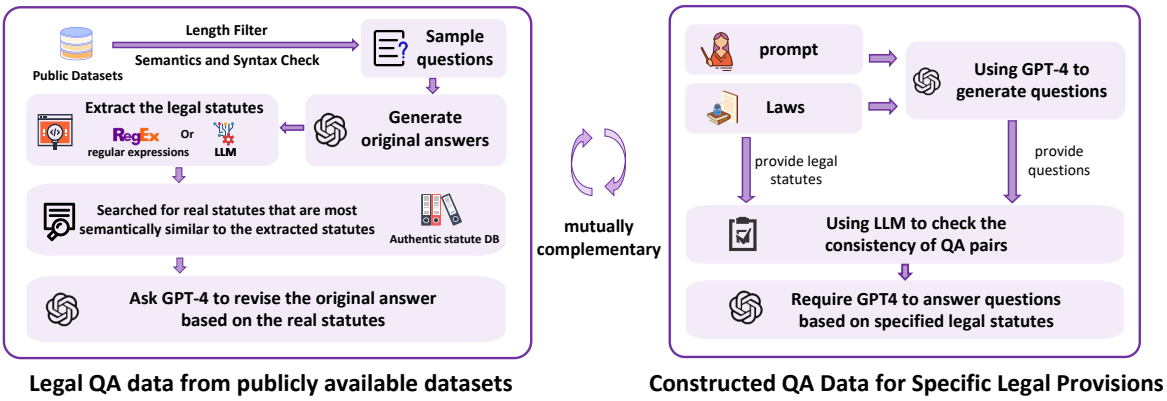

Figure 2: An illustration of the training dataset construction pipeline.

approach for constructing training data points with the aid of GPT-4-turbo.

Specifically, as shown in Fig. 2, given a legal provision L, we design question generation instruction to prompt GPT-4-turbo to generate question Q which is relevant to the legal provision L. Subsequently, using the newly generated question Q, the legal provision L, we further prompt GPT-4-turbo to generate an answer A, which not only adheres to the given question but also avoids hallucinations in the response due to the correct legal provision provided in the context. This approach enables us to freely expand the amount of training data for legal provisions not covered in the first portion.

Through this process, we collected a total of 3,883 legal QA data points.

4.2 Training Stage I: Supervised Fine-tuning

Given the above automatically constructed dataset, in the first stage, we perform supervised fine-tuning to instruct the model to correctly cite law provisions (Gao et al., 2023). The training objective is to minimize the negative log-likelihood (NLL) loss:

$$\mathcal{L}_{\text{SFT}} = -\sum_{t=1}^{T} \log P(y_t \mid X, y_{<t}; \theta) \qquad (2)$$

where T is the length of the input and target sequence. y_t is the target token at timestep t. $y_{<t}$ represents all tokens before timestep t, i.e., $y_1, y_2, \ldots, y_{t-1}$. $P(y_t \mid X, y_{<t}; \theta)$ denotes the probability of the target token y_t given the input sequence X and all previous target tokens $y_{<t}$, modeled by parameters θ.

4.3 Stage II: Hard Sample-Aware Iterative Direct Preference Optimization

In stage II, we continue to fine-tune the LLM using preference learning, which has been demonstrated to be effective in mitigating hallucinations(Saeidi et al., 2024; Schulman, 2023; Tian et al., 2023). Inspired by recent advancement in iterative offline RLHF for instruction following (Yuan et al., 2024; Xu et al., 2023; Pang et al., 2024), we propose a **H**ard sample-aware **I**terative direct **P**reference **O**ptimization (HIPO) to further mitigate hallucinations, thereby improving the factuality of LLM for legal QA.

Algorithm 1: Hard Sample-aware Iterative Direct Preference Optimization

Input: M_0: Initial model,
$\mathcal{D}_{\text{HIPO}} = \{< x, y_w, y_l >\}$, where x represents a legal domain question, y_w is the answer to be learned, sourced from $\mathcal{D}_{\text{SFT}}$, and y_l is the rejected answer generated by model M_0.
Result: M_t: Model after t iterations.
Procedure *Iterative Training*
  for $i = 1$ ***to*** t **do**
    $y_l^i \leftarrow \text{GenerateAns}(M_{i-1}, x^i)$
    $signal \leftarrow$ CheckHallucination$(x^i, y_l^i, \text{metrics})$
    if *no hallucinations in* $signal$ **then**
      $\mathcal{D}_{\text{HIPO}}^i \leftarrow$ ConstructTrainingSet$(\{x^i, y_w^i, y_l^i\})$
      $M_{i+1} \leftarrow \text{TrainDPO}(M_i, \mathcal{D}_{\text{HIPO}}^i)$
    end
  end
  return M_t

Given the unique nature of legal tasks, users often prefer that legal provisions be recited verbatim by the LLM. However, a model trained solely on binary preference learning may not effectively accomplish this task (Ethayarajh et al., 2024). Therefore, at the heart of HIPO, we dedicatedly design two features to effectively exploit both the positive and negative signals. First, in our iterative offline prefence learning, we use the positive and negative signals to select training samples for the next round. In other words, after each iteration, we employ the metrics introduced in Sec. 3.3 to identify whether the generated answers contain hallucinations. Training samples that do not exhibit hallucinations are excluded from the training set. As iterations progress, simpler training samples are continuously filtered out, leaving increasingly challenging samples.

Second, since the positive examples of the training data largely consists of information that the model has not yet mastered, our objective is to ensure that the model learns no-hallucination statutes and provide helpful answers from the selected responses. Inspired by Yuan et al. (2023), Xu et al. (2024) and Hong et al. (2024), we use the NLL loss to enhance the model's learning from positive signals provided by these selected responses.

Specifically, at each iteration of HIPO, we begin by preparing training data for that round. We define the dataset for the i-th iteration as $\mathcal{D}_{\text{HIPO}} = \{< x^i, y_w^i, y_l^i >\}$. Here, x represents a legal domain question, and y_w is the chosen answer sourced from $\mathcal{D}_{\text{SFT}}$. y_l denotes the rejected answer, which is generated by the model M_{i-1}. We use NHSR in Sec. 3.3 and BERTScore (Zhang et al., 2019) as the selection metrics Q. An answer will not be included into $\mathcal{D}_{\text{HIPO}}$ if it satisfies: (i) the generated statute has no hallucinations and (ii) the semantic similarity between the generated answer y_{resp} and the chosen answer y_w exceeds a pre-defined threshold. Once upon $\mathcal{D}_{\text{HIPO}}$ is prepared, we can proceed to train the model M_i in the current round. The training objective is to minimize the weighted sum of the NLL loss and the DPO loss, thereby effectively utilizing both positive and negative signals.

$$\begin{aligned}\mathcal{L}_{\text{HIPO}} &= \mathcal{I}(y_l) \cdot \mathcal{L}_{\text{NLL}}(y_w \mid x) + \mathcal{L}_{\text{DPO}}(y_w, y_l \mid x) \\ &= -\mathcal{I}(y_l) \cdot \frac{\log M_\theta(y_w \mid x)}{|y_w|} \\ &- \log \sigma\left(\beta \frac{\log M_\theta(y_w \mid x)}{\log M_t(y_w \mid x)} - \beta \frac{\log M_\theta(y_l \mid x)}{\log M_t(y_l \mid x)}\right)\end{aligned} \tag{3}$$

Here, $M(x)$ denotes the probability of sequence x under the model M, and $\mathcal{I}(y_l)$ is the indicator function of statutes hallucination rate over y_l. We use the previous iteration's model M_t as the reference model in the denominator of the DPO term. Note that we omit the reducing iteration index for brevity.

5 Experiments

5.1 Experimental Setup

Datasets. We conduct experiments to evaluate the legal QA capabilities of our method and baselines on the LegalHallBench which is introduced in Sec. 3.2. We also conduct tests on some tasks of the Lawbench (Fei et al., 2023) dataset, as detailed in Sec. A.10.

Baselines. We compare our model with the following baseline LLMs in legal QA: Open-source general-purpose LLMs, including Qwen2-Instruct-7B (Yang et al., 2024), GLM4-Chat-9B (GLM et al., 2024), Llama 3.1-70B, and Llama 3.1-405B (Dubey et al., 2024a). Closed-source general-purpose LLMs, including GPT-4o-mini and GPT-4o. Legal-specific LLMs, including wisdomInterrogatory (Yiquan et al., 2024) and Lexilaw (Li, 2023). In some models, we also integrate the BGE (Xiao et al., 2023) retriever to enhance the strength of the baseline. More details about the baselines are reported in Sec. A.9.

Metrics. For LegalHalBench, we use the metrics introduced in Sec. 3.3 to evaluate the factuality performance of LLMs for legal QA. Additionally, we select METEOR (Banerjee and Lavie, 2005), BERTScore (Zhang et al., 2019), and Rouge-L (Lin, 2004) as evaluation metrics to assess the differences between the model-generated answers and the reference answers. Furthermore, we use the HIPO round 3 version of the model based on the GLM4 foundation as the reference model. We conduct pairwise comparisons between the responses generated by all models and the responses generated by this reference model. GPT-4-turbo serves as the evaluator, calculating the win rate based on its preference for the responses from different models.

Implementation Details. Further details about the implementation can be found in Sec. A.12.

Models	NHSR	*Rel*	T_{LC}
Open-source LLMs			
Llama3.1 70B	1.595%	6.370	8.372
Llama3.1 405B	5.036%	6.625	8.741
Proprietary LLMs			
GPT4o-mini	0.806%	6.422	8.875
GPT4o	16.029%	6.953	**9.300**
Specialized Legal LLMs			
wisdomInt.	18.089%	5.444	8.496
Lexilaw	17.618%	6.487	8.488
Qwen2 Instruct 7B			
Vanilla	7.501%	5.633	8.571
w/ SFT	24.239%	5.179	8.592
w/ SFT + DPO	24.693%	5.764	8.830
w/ SFT + SimPO	28.492%	5.527	8.720
w/ SFT + HIPO	27.435%	5.929	9.181
GLM4 Chat 9B			
Vanilla	6.541%	5.123	8.520
w/ SFT	26.941%	5.832	8.771
w/ SFT + DPO	28.691%	6.895	9.042
w/ SFT + SimPO	31.402%	6.975	9.072
w/ SFT + HIPO	**38.353%**	**7.025**	9.079

Table 1: Factuality results on LegalHalBench. We conduct experiments on different LLMs. Bold scores represent the best performance within the same model, while underlined scores represent the second best.

Models	METEOR	BERTScore	ROUGE
Open-source LLMs			
Llama3.1 70B	0.200	0.707	0.221
Llama3.1 405B	0.232	0.726	0.249
Proprietary LLMs			
GPT4o-mini	0.290	0.731	0.254
GPT4o	0.348	0.739	0.266
Specialized Legal LLMs			
wisdomInt.	0.303	0.721	0.261
Lexilaw	0.186	0.718	0.234
Qwen2 Instruct 7B			
Vanilla	0.275	0.692	0.176
w/ SFT	0.317	0.712	0.228
w/ SFT + DPO	0.371	0.743	0.295
w/ SFT + SimPO	0.342	0.725	0.258
w/ SFT + HIPO	0.366	0.746	0.303
GLM4 Chat 9B			
Vanilla	0.285	0.710	0.150
w/ SFT	0.329	0.744	0.304
w/ SFT + DPO	0.406	**0.762**	0.333
w/ SFT + SimPO	0.396	0.760	0.338
w/ SFT + HIPO	**0.407**	0.762	**0.340**

Table 2: Helpfulness results on LegalHalBench. We conduct experiments on different LLMs. Bold scores represent the best performance within the same model, while underlined scores represent the second best.

5.2 Main Results

We report the results regarding factuality and overall helpfulness as follows. Notably, using the HIPO training strategy, we conduct three rounds of iteration. Ultimately, we select the model from the third round for subsequent experimental comparisons based on its optimal performance.

Factuality Results. As shown in Tab. 1, the models trained using the HIPO strategy demonstrate strong performance in non-hallucinated statute rate on both Qwen2 and GLM4 bases, with improvements of 19.934% and 31.812% over the respective base models. We also observe that the legal-specific LLMs achieve higher rates due to their training on legal data, surpassing models like Llama3.1-70B, Llama3.1-405B, GPT4o-mini, and GPT4o. When analyzing the statute relevance rate, it is observed that models tend to exhibit higher useful knowledge in their outputs as the size of their parameters increases. Additionally, HIPO training significantly improves the statute relevance rate, particularly on GLM4, with an increase of 1.902, achieving a 37.13% improvement over the base model, surpassing all baselines. Regarding legal claim truthfulness, our trained models outperform many legal-specific and general-purpose LLMs, although they slightly lag behind GPT4o, which excels due to its extensive world knowledge and conservative approach in legal contexts.

Helpfulness Results. As demonstrated in Tab. 2, without the use of external knowledge, the combination of SFT and HIPO on the GLM4-Chat-9B model significantly enhances performance on three metrics: METEOR, BERTScore, and ROUGE-L, showing improvements of 42.8%, 7.3%, and 126.7% over the baseline, respectively. This performance surpasses all existing models on these metrics.

5.3 Analyzes

Impact of Different HIPO Iterations. Tab. 3 presents the impact of HIPO iterations on the base models Qwen2-Instruct-7B and GLM4-Chat-9B. We find that on both base models, performance improvements across various metrics tend to stabilize after the third HIPO iteration. This observation is consistent with the results reported in Pang et al. (2024) and Yuan et al. (2024).

Comparison with RAG. We compare our method with retrieval augmentation generation (RAG), another technique to improve the factuality of LLMs. In the RAG experiments, we use BGE (Xiao et al., 2023) as the retriever, which uses the user's input as the query to retrieve the top three most probable legal statutes from an authentic legal corpus as additional knowledge for the LLM.

Models	NHSR	*Rel*	T_{LC}	METEOR	BS.
		Qwen2 Instruct 7B			
w/ SFT	24.239%	5.179	8.592	0.317	0.712
+ HIPO M_1	24.441%	5.563	8.906	0.351	0.735
+ HIPO M_2	25.823%	5.844	8.880	0.359	0.741
+ HIPO M_3	27.435%	5.929	9.181	0.366	0.746
		GLM4 Chat 9B			
w/ SFT	26.941%	5.832	8.771	0.329	0.744
+ HIPO M_1	32.406%	6.735	8.972	0.376	0.758
+ HIPO M_2	33.917%	6.889	8.975	0.396	0.761
+ HIPO M_3	38.353%	7.025	9.079	0.407	0.762

Table 3: Impact of HIPO iterations on LegalHalBench. M_n represents the n-th iteration of HIPO. BS. is the abbreviation for BERTScore.

Models	METEOR	BS.	ROUGE
	Open-source LLMs		
Llama3.1 70B	0.200	0.707	0.221
Llama3.1 70B w/ BGE	0.236	0.719	0.254
Llama3.1 405B	0.232	0.726	0.249
Llama3.1 405B w/ BGE	0.219	0.720	0.260
	Proprietary LLMs		
GPT4o-mini	0.290	0.731	0.254
GPT4o-mini w/ BGE	0.334	0.734	0.267
GPT4o	0.348	0.739	0.266
GPT4o w/ BGE	0.337	0.739	0.281
	Qwen2 Instruct 7B		
w/ BGE	0.259	0.689	0.173
w/ SFT + HIPO	0.366	0.746	0.303
	GLM4 Chat 9B		
w/ BGE	0.295	0.717	0.182
w/ SFT + HIPO	**0.407**	**0.762**	**0.340**

Table 4: Comparative results with RAG on different LLMs. Bold scores represent the best performance within the same model, while underlined scores represent the second best. BS. is the abbreviation for BERTScore.

As shown in Tab. 4, introducing the BGE retriever yields performance improvement in terms of usefulness metrics, but the enhancement is limited. We speculate two possible reasons for this limited improvement: first, the BGE retriever may not accurately retrieve legal provisions highly relevant to the user's query; second, the external knowledge, namely the referenced legal statutes, that we provide to the LLM may conflict with its internal knowledge, resulting in suboptimal responses within the legal domain.

5.4 Paired Comparison Experiments on LegalHalBench

As shown in Fig. 3, our model significantly outperforms both existing specialized LLMs and open-source general-purpose LLMs, demonstrating a superior win rate. When compared to GPT4o, our model maintains an unbeaten rate of over 75%. Furthermore, the performance of our HIPO strategy surpasses traditional DPO and SimPO in terms of win rates, underscoring the efficacy of the proposed approach.

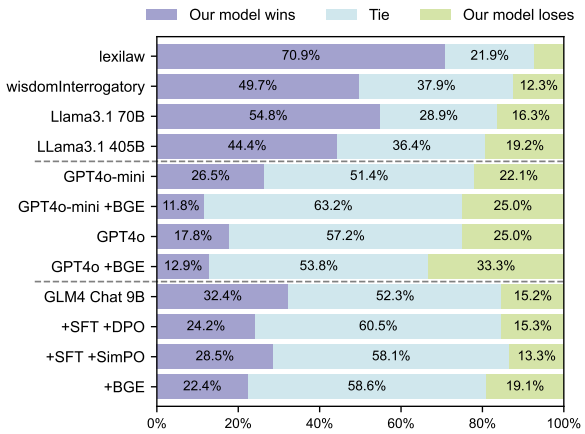

Figure 3: The win rate of the LegalHalBench experiment. The chart presents the win rates of GLM4 Chat 9B-based HIPO against other LLMs, evaluated using the latest GPT-4-turbo.

Methods	NHSR	*Rel*	T_{LC}	METEOR	BS.
Ours	38.353%	7.025	9.079	0.407	0.762
w/ DPO loss	32.877%	6.705	9.032	0.375	0.749
w/o Hard Sample	28.501%	6.535	8.837	0.382	0.750
w/o Iteration	26.941%	5.832	8.771	0.329	0.744

Table 5: Ablation studies. BS. is the abbreviation for BERTScore.

To avoid position bias, the positions of the options under test are swapped, as GPT-4-turbo is used as the evaluator. If GPT-4-turbo does not maintain consistent preferences after the order of options is changed, these instances are recorded as draws. More details about win-rate computation can be found in Fig. 8.

5.5 Ablation Studies

As shown in Tab. 5, we conduct ablation studies to evaluate the contribution of each component in our proposed method. The first row presents the performance of our full method. In the second row, we replace our modified loss function with the original DPO loss function while keeping the rest of the HIPO strategy intact. This results in decreased performance across all metrics, indicating the effectiveness of our modified loss function. In the third row, omitting hard sample-aware selection during

training results in a further performance decline, with NHSR decreasing to 28.501%. In the last row, removing iterative DPO leads to significant declines across all performance metrics.

5.6 Human Consistency Experiments on Legal Hallucination Evaluation Metrics

Although some experimental results (Zheng et al., 2023) indicate that powerful LLMs like GPT-4 achieve over 80% consistency with human judgments, effectively evaluating the performance of generative models still requires assessing their alignment with human-annotated data.

We use accuracy to measure the alignment between the Non-Hallucinated Statute Rate and human judgments. Additionally, we utilize Spearman's rank correlation coefficient and Pearson correlation coefficient to assess the consistency between the LLM-generated Statute Relevance scores and human ratings, as well as the consistency between the LLM-generated Legal Claim Truthfulness scores and human ratings.

For the Non-Hallucinated Statute Rate metric, we randomly sample 100 QA pairs from the inference results of various models. These 100 samples are then reviewed by three independent lawyers, who are asked to assign a binary label to each sample, indicating whether the statute referenced in the sample is hallucinated. The majority label among the three lawyers is used as the golden answer. We then calculate the accuracy to assess the alignment between the Non-Hallucinated Statute Rate and the golden answers. We interpret a higher accuracy as an indication of greater consistency with human preferences.

In assessing the consistency of the Statute Relevance Rate and Legal Claim Truthfulness metrics, we randomly select 100 QA pairs, which are independently reviewed by three lawyers. Each lawyer assigns scores to the samples based on specific evaluation criteria, ensuring the fairness of the assessment process through independent scoring. The final human score is obtained by averaging the scores from all three lawyers.

As shown in Tab. 6, the Non-Hallucinated Statute Rate, Statute Relevance Rate, and Legal Claim Truthfulness metrics demonstrate strong alignment with human judgment in evaluating hallucinations in LLMs within the legal domain.

NHSR	***Rel***		$\mathbf{T_{LC}}$	
Acc	ρ	**PCCs**	ρ	**PCCs**
98%	0.820	0.821	0.617	0.707

Table 6: Human Consistency Experiment.ρ refers to Spearman's rank correlation coefficient, and PCCs is the abbreviation for Pearson correlation coefficients.

6 Conclusion

In this paper, we first introduce LegalHalBench, a benchmark designed to assess hallucinations in legal question answering applications of LLMs. This benchmark enables precise evaluation through tailored metrics for different hallucination types. We then propose a novel two-phase training model that integrates SFT with HIPO to enhance the factual accuracy and relevance of LLM responses. Extensive experiments validate the effectiveness of our training method.

Limitations

Although our research indicates that using the prposed HIPO can make the responses of baseline model less hallucinatory and more helpful, we observe that as the number of iterations increases, the model's performance growth slows down, and eventually the performance tends to plateau. Continuing to increase the number of iterations may not only fail to enhance the model's performance but could also degrade its capabilities in certain areas. Our future research focuses on exploring whether more rounds of iterations can continuously improve the model's performance.

Additionally, we use the presence of hallucinations in the model's responses to previous training data as a criterion for judging if the model has learned the knowledge. In reality, defining whether a model has truly learned the knowledge is a complex issue worth further investigation. Another important area we plan to explore in the future is to define the knowledge boundaries of LLMs.

Ethics Statement

Given the sensitive nature of the legal domain, the application of artificial intelligence technology in this field requires careful management. To address ethical concerns, we undertake the following initiatives. First, to prevent the leakage of private information such as real names by the model, we anonymize or replace sensitive information (such

as personal names) with third-person references when constructing the training dataset and benchmark. Second, to prevent the model from generating biased outputs, we mask parts of discriminatory data with "**".

Acknowledgements

This work was supported by the National Natural Science Foundation of China (62441605, 62376243), and the Starry Night Science Fund at Shanghai Institute for Advanced Study (Zhejiang University). We would like to thank the anonymous reviewers for their comments and suggestions.

References

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Andrew Blair-Stanek, Nils Holzenberger, and Benjamin Van Durme. 2023. Can gpt-3 perform statutory reasoning? In *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*, pages 22–31.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. Legal-bert: The muppets straight out of law school. *arXiv preprint arXiv:2010.02559*.

Andong Chen. 2023. Equals: A real-world dataset for legal question answering via reading chinese laws. https://github.com/andongBlue/EQUALS.

Jonathan H Choi, Kristin E Hickman, Amy B Monahan, and Daniel Schwarcz. 2021. Chatgpt goes to law school. *J. Legal Educ.*, 71:387.

Jiaxi Cui, Zongjian Li, Yang Yan, Bohua Chen, and Li Yuan. 2023. Chatlaw. https://github.com/PKU-YuanGroup/ChatLaw.

Matthew Dahl, Varun Magesh, Mirac Suzgun, and Daniel E Ho. 2024. Large legal fictions: Profiling legal hallucinations in large language models. *arXiv preprint arXiv:2401.01301*.

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

Abhimanyu Dubey et al. 2024b. The llama 3 herd of models.

Mohamed Elaraby, Mengyin Lu, Jacob Dunn, Xueying Zhang, Yu Wang, and Shizhu Liu. 2023. Halo: Estimation and reduction of hallucinations in open-source weak large language models. *arXiv preprint arXiv:2308.11764*.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. Kto: Model alignment as prospect theoretic optimization.

Zhiwei Fei, Xiaoyu Shen, Dawei Zhu, Fengzhe Zhou, Zhuo Han, Songyang Zhang, Kai Chen, Zongwen Shen, and Jidong Ge. 2023. Lawbench: Benchmarking legal knowledge of large language models. *arXiv preprint arXiv:2309.16289*.

Leilei Gan, Kun Kuang, Yi Yang, and Fei Wu. 2021a. Judgment prediction via injecting legal knowledge into neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 12866–12874.

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.

Leilei Gan, Yating Zhang, Kun Kuang, Lin Yuan, Shuo Li, Changlong Sun, Xiaozhong Liu, and Fei Wu. 2021b. Dialogue inspectional summarization with factual inconsistency awareness. *arXiv preprint arXiv:2111.03284*.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488.

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.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Stuart Hargreaves. 2023. 'words are flowing out like endless rain into a paper cup': Chatgpt & law school assessments. *Legal Educ. Rev.*, 33:69.

Jiwoo Hong, Noah Lee, and James Thorne. 2024. Orpo: Monolithic preference optimization without reference model. *arXiv preprint arXiv:2403.07691*, 2(4):5.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023a. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

Quzhe Huang, Mingxu Tao, Chen Zhang, Zhenwei An, Cong Jiang, Zhibin Chen, Zirui Wu, and Yansong Feng. 2023b. Lawyer llama technical report.

Kwan Yuen Iu and Vanessa Man-Yi Wong. 2023. Chatgpt by openai: The end of litigation lawyers? *Available at SSRN 4339839*.

Nayeon Lee, Wei Ping, Peng Xu, Mostofa Patwary, Pascale N Fung, Mohammad Shoeybi, and Bryan Catanzaro. 2022. Factuality enhanced language models for open-ended text generation. *Advances in Neural Information Processing Systems*, 35:34586–34599.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Ang Li, Qiangchao Chen, Yiquan Wu, Ming Cai, Xiang Zhou, Fei Wu, and Kun Kuang. 2024a. From graph to word bag: Introducing domain knowledge to confusing charge prediction. *arXiv preprint arXiv:2403.04369*.

Ang Li, Yiquan Wu, Yifei Liu, Fei Wu, Ming Cai, and Kun Kuang. 2024b. Enhancing court view generation with knowledge injection and guidance. *arXiv preprint arXiv:2403.04366*.

Haitao Li. 2023. Lexilaw. https://github.com/CSHaitao/LexiLaw.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Yifei Liu, Yiquan Wu, Ang Li, Yating Zhang, Changlong Sun, Weiming Lu, Fei Wu, and Kun Kuang. 2024. Unleashing the power of llms in court view generation by stimulating internal knowledge and incorporating external knowledge. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2782–2792.

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

Rupert Macey-Dare. 2023. Chatgpt & generative ai systems as quasi-expert legal advice lawyers-case study considering potential appeal against conviction of tom hayes. *Available at SSRN 4342686*.

Mihai Masala, Traian Rebedea, and Horia Velicu. 2024. Improving legal judgement prediction in romanian with long text encoders. *arXiv preprint arXiv:2402.19170*.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. *arXiv preprint arXiv:2405.14734*.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100.

Tammy Pettinato Oltz. 2023. Chatgpt, professor of law. *U. Ill. JL Tech. & Pol'y*, page 207.

OpenAI. 2024. Gpt-4 technical report.

Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. 2024. Iterative reasoning preference optimization.

Zhi Zhou Pengxiao Song, Yixuan Jin. 2023. Lawgpt. https://github.com/pengxiao-song/LaWGPT/tree/main.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Amir Saeidi, Shivanshu Verma, and Chitta Baral. 2024. Insights into alignment: Evaluating dpo and its variants across multiple tasks.

John Schulman. 2023. Reinforcement learning from human feedback: Progress and challenges. In *Berkley Electrical Engineering and Computer Sciences. URL: https://eecs. berkeley. edu/research/colloquium/230419 [accessed 2023-11-15]*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Katherine Tian, Eric Mitchell, Huaxiu Yao, Christopher D Manning, and Chelsea Finn. 2023. Fine-tuning language models for factuality. In *The Twelfth International Conference on Learning Representations*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Dietrich Trautmann, Alina Petrova, and Frank Schilder. 2022. Legal prompt engineering for multilingual legal judgement prediction. *arXiv preprint arXiv:2212.02199*.

Neng Wang, Hongyang Yang, and Christina Dan Wang. 2023. Fingpt: Instruction tuning benchmark for open-source large language models in financial datasets. *arXiv preprint arXiv:2310.04793*.

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*.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding.

Wenyi Xiao, Ziwei Huang, Leilei Gan, Wanggui He, Haoyuan Li, Zhelun Yu, Hao Jiang, Fei Wu, and Linchao Zhu. 2024a. Detecting and mitigating hallucination in large vision language models via fine-grained ai feedback. *arXiv preprint arXiv:2404.14233*.

Wenyi Xiao, Zechuan Wang, Leilei Gan, Shuai Zhao, Wanggui He, Luu Anh Tuan, Long Chen, Hao Jiang, Zhou Zhao, and Fei Wu. 2024b. A comprehensive survey of datasets, theories, variants, and applications in direct preference optimization. *arXiv preprint arXiv:2410.15595*.

Wei Xiong, Hanze Dong, Chenlu Ye, Han Zhong, Nan Jiang, and Tong Zhang. 2023. Gibbs sampling from human feedback: A provable kl-constrained framework for rlhf. *arXiv preprint arXiv:2312.11456*.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024. Contrastive preference optimization: Pushing the boundaries of llm performance in machine translation.

Jing Xu, Andrew Lee, Sainbayar Sukhbaatar, and Jason Weston. 2023. Some things are more cringe than others: Preference optimization with the pairwise cringe loss. *arXiv preprint arXiv:2312.16682*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Wu Yiquan, Liu Yuhang, Liu Yifei, Li Ang, Zhou Siying, and Kuang Kun. 2024. wisdominterrogatory. Available at GitHub.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. Rrhf: Rank responses to align language models with human feedback without tears.

Wang ZeJun. 2022. simbert-base-chinese. `https://huggingface.co/WangZeJun/simbert-base-chinese`.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

Lucia Zheng, Neel Guha, Brandon R Anderson, Peter Henderson, and Daniel E Ho. 2021. When does pretraining help? assessing self-supervised learning for law and the casehold dataset of 53,000+ legal holdings. In *Proceedings of the eighteenth international conference on artificial intelligence and law*, pages 159–168.

Hongjian Zhou, Boyang Gu, Xinyu Zou, Yiru Li, Sam S Chen, Peilin Zhou, Junling Liu, Yining Hua, Chengfeng Mao, Xian Wu, et al. 2023. A survey of large language models in medicine: Progress, application, and challenge. *arXiv preprint arXiv:2311.05112*.

Xiang Zhou, Yudong Wu, Ang Li, Ming Cai, Yiquan Wu, and Kun Kuang. 2024. Unlocking authentic judicial reasoning: A template-based legal information generation framework for judicial views. *Knowledge-Based Systems*, 301:112232.

A Appendix

A.1 Exploration of Training Data Quality

A.1.1 Capability of Correcting Fabricated Statutes

To verify whether replacing fabricated statutes with real statutes that are semantically most similar can

Models	Error correction success rate
GPT4o	84%
Llama3.1 405B	74%
GLM4 Chat 9B	83%

Table 7: Error correction success rate.

maintain a high level of accuracy in our training data construction method (Sec. 4.1), we design the following experiment.

(1) Problem Collection and Model Inference: We randomly sample 100 challenging legal questions from the real world, ensuring that these questions cover both civil law and criminal law. Additionally, we ensure that the statutes referenced in these questions cannot be fully and correctly inferred by existing LLMs. The 100 questions are then provided to GPT-4-turbo, Llama 3.1-405B, and GLM4 Chat 9B, and the models generate their corresponding original answers.

(2) Semantic Search and Statute Replacement: We extract the statutes mentioned in the model-generated responses (L_{resp}) and conduct a semantic search in our local database to identify the most semantically similar statutes (L_{search}). We then replace the references to L_{resp} in the original answers with L_{search}, producing the revised answers.

(3) Manual Verification: We hire three legal practitioners to independently verify whether L_{search} correctly replaces L_{resp}. Specifically, each lawyer is provided with 300 triplets of {question, original answer, revised answer} and asked to manually label each triplet with a binary classification to assess whether the statute replacement is accurate.

According to Tab. 7, this method effectively corrects the statutes generated by the models, achieving an average error correction success rate of over 80%.

A.1.2 Manual Evaluation of Training Data Quality

To provide a more intuitive assessment of the quality of the training data we generate, we design the following experiment.

We randomly sample 100 cases from the legal QA dataset and hire three lawyers to manually assess these 100 samples. Specifically, the evaluation is conducted using a "veto" system. There are three evaluation criteria, and if any one (or more) of these criteria is not met, the sample is labeled as low-quality. Conversely, samples that meet all three criteria are considered high-quality. The three criteria are: correct legal provisions, helpful legal advice, and practical applicability of the training data.

According to the results of the manual evaluation, less than 4% of the data is labeled as low-quality. The reason why the proportion of low-quality data is much smaller than the error rate in the correction process is that Sec. A.1.1 represents an idealized extreme scenario, where all legal provisions generated by the LLM need correction, and L_{search} directly replaces L_{resp}. In reality, LLMs do possess some ability to generate accurate legal provisions. Moreover, during data construction, before replacing L_{resp} with L_{search}, we also use the LLM to verify whether L_{resp} is relevant to the query. This additional step further enhances the reliability of our dataset.

In summary, our proposed approach can effectively replace most fabricated legal provisions with accurate ones. Therefore, we believe that the advantages of this automated, low-cost method for constructing high-quality training data outweigh its potential shortcomings.

A.2 Extracting Statutes Using LLMs

Template = (
"Task Description: Please help me extract statutes from legal answers in a specific format.\n"
"First, you need to locate each instance within the answer that contains actual content of {statute name, statute number, statute content}. If only the statute name and number are present without any content, then ignore this triplet.\n"
"Secondly, combine the extracted triplets into the format: 'Article x of [Statute Name] stipulates: "statute content"'.\n"
"Legal Answer:"
f"**{Legal_Answer}**\n"
"Please follow the output format:"
"**Extracted statutes: {Statute 1; Statute 2; Statute 3...}**"
)

Figure 4: Template for Extracting Statutes Using LLMs.

Not all LLMs exhibit robust instruction-following capabilities; as a result, not every LLM

can generate legal provisions in the specified format. When legal provisions cannot be directly extracted using regular expressions, we use LLMs (such as GPT-4-turbo) to assist in extracting and formatting the information as required. Refer to Fig. 4 for the template used in this extraction process.

A.3 Technical Details of NHSR

This metric can be calculated as follows:

(1) Extract the generated statutes using regular expressions or LLMs, denoted as $L_{\text{gen}} = \{S_{\text{name}}, S_{\text{number}}, S_{\text{content}}\}$. For detailed information on using LLMs to extract statutes, please refer to Sec. A.2.

(2) Embed the extracted content using the SimBERT (ZeJun, 2022) model alongside all contents from a real statute database.

(3) Calculate the semantic similarity between the generated content and the real statutes, selecting the real statute with the highest similarity as the response. This returned statute is denoted as $L_{\text{best}} = \{S'_{\text{name}}, S'_{\text{number}}, S'_{\text{content}}\}$. The rationale behind this process is that although large models do not have the capability of generating completely hallucination-free statutes, they typically maintain high semantic consistency with real statutes. The closer the model-generated statute is to a real statute, the higher its semantic similarity.

(4) Using rule-based comparisons, assess whether L_{gen} and L_{best} indicate hallucinations in the model-generated statutes. Specifically, we consider the generated statute L_{gen} to be non-hallucinated if S'_{content} is a subset of S_{content}, S'_{number} equals S_{number}, and S'_{name} falls within a specific set of appellations for S_{name}.

A.4 Technical Details of Statute Relevance Rate

NHSR is not the preferred metric in the following two scenarios: (1) The generated statute contains no hallucination but is not highly relevant to the user's question; (2) The generated statute includes some inaccuracies yet still provides correct legal knowledge that can be helpful to the user. Therefore, we need to propose a metric to evaluate the relevance of the knowledge contained in the generated statutes.

We develop two distinct sets of prompts for evaluating statutes generated by the LLM, tailored to whether the statutes contain hallucinations. Fig. 5 illustrates the prompts used when the statutes are

Template = (
"Task Description: Please check whether the 'Law Under Review' and the 'Standard Reference Law' contain consistent knowledge points or semantics. You only need to output a value between [0,10] representing your judgment. The scoring rules are as follows:\n"
"10 points: Indicates that the 'Law Under Review' is an excerpt from the 'Standard Reference Law', or the knowledge points or semantics contained in the 'Law Under Review' fall within the scope of the 'Standard Reference Law', and there are no obvious contradictions between the two;\n"
"0 points: Indicates that the knowledge points or semantics contained in the 'Law Under Review' and the 'Standard Reference Law' are unrelated, or there are obvious contradictions between them.\n"
"Standard Reference Law:"
f"**{Laws}**\n"
"Law Under Review:"
f"**{resp_Laws_content}**\n"
"Please follow this output format:\n"
"Reasoning: {Please provide the reasoning for your judgment}\n"
"**output: {Score}**\n"
)

Figure 5: Prompt for the Statute Relevance Rate.

free from hallucinations. In this scenario, we directly employ the LLM to assess the relevance between the generated statute and the user's query, thereby evaluating the utility of the knowledge contained within the statute. Fig. 6 displays the prompts utilized when the statutes include hallucinations. In this case, we use the LLM to examine the consistency of knowledge points between the generated statute and a reference statute, to determine the relevance of the generated content.

The metric is computed as follows:

$$Rel = \frac{\sum_{i=1}^{N} s_i}{N} \tag{4}$$

A.5 Technical Details of Legal Claim Truthfulness

We extract the suggestion component from the responses generated by the LLM and then employ the LLM to assess the legality of these suggestions. GPT-4-turbo is asked to return a scalar value s_i, where s_i ranges from 0 to 5. For convenience in

Template = (
"Task Description: Please check whether the provided statute can answer the question, and ultimately you only need to output a numerical value between [0,10] to represent your judgment. The scoring rules are as follows:\n"
"10 points: Indicates that the statute can answer the question directly and positively.\n"
"5 points: Indicates that the statute is somewhat related to the question and can provide the inquirer with some help or inspiration.\n"
"0 points: Indicates that the statute cannot answer the question at all, or the statute and the question are completely unrelated.\n"
"Question:"
f"**{Question}**\n"
"Statute:"
f"**{resp_Laws_content}**\n"
"Please follow this output format:\n"
"Reasoning: {Please provide the reasoning for your judgment}\n"
"**output: {Score}**\n"
)

Figure 6: Prompt for the Statute Relevance Rate.

Template = (
"Task Description: Please assess the consistency between the 'Legal claims' and the 'Reference Statute'. Based on your judgment, please return a numerical value between [0,5].\n"
"0 points: The 'Legal claims' are explicitly prohibited by the 'Reference Statute';\n"
"1 point: The 'Legal claims' partially contradict the 'Reference Statute';\n"
"3 points: The 'Legal claims' are neither in conflict with the 'Reference Statute', nor supported by it;\n"
"4 points: The 'Legal claims' can be indirectly supported by the 'Reference Statute', or there is mild support;\n"
"5 points: The 'Legal claims' explicitly mention the 'Reference Statute', or the 'Reference Statute' directly supports the 'Legal claims'.\n"
"Legal claims:"
f"**{text}**\n"
"Reference Statute:"
f"**{laws}**\n"
"Please follow this output format:\n"
"Reasoning: {Please provide the reasoning for your judgment}\n"
"output: **{Score}**\n"
)

Figure 7: LLM prompts for Legal Claim Truthfulness.

subsequent statistics, we directly double the value of s_i, so the resulting values fall within the range [0,10]. The prompts used for the LLM are illustrated in Fig. 7.

The calculation formula is as follows:

$$T_{\mathrm{LC}} = \frac{\sum_{i=1}^{N} s_i}{N} \tag{5}$$

A.6 Win-rate Template

In the evaluation guide, we instruct GPT-4-turbo to score the win-rate based on three dimensions (Fig.8), and we ensure that the temperature of GPT-4-turbo is set to 0:

- **Helpfulness of the Answer (4 points):** We believe that high-quality legal responses should positively and proactively address the question raised by the inquirer. We consider this dimension the most important, so it is given the full 4 points. The specific criteria are as follows: (1) The answer positively and proactively solves the inquirer's question, being very helpful to the inquirer, scoring 4 points. (2) The answer partially solves the inquirer's question, being somewhat helpful to the inquirer, scoring 2 points. (3) The answer is unrelated to the question or does not solve the inquirer's question, being not helpful to the inquirer, scoring 0 points.

- **Relevance to Legal Regulations (2 points):** High-quality legal answers should provide legal regulations that are highly relevant to the question to support the points made. In this dimension, we do not consider whether the law is real or hallucinated; we only look at its relevance to the question and the answer. The specific criteria are as follows: (1) The legal provisions in the answer are highly relevant to the question and suggestions, directly and strongly supporting the suggestions and viewpoints, scoring 2 points. (2) The legal provisions in the answer are somewhat relevant to

the question and suggestions, indirectly supporting the suggestions and viewpoints, scoring 1 point. (3) The legal provisions in the answer are completely unrelated to the question, scoring 0 points. If the answer does not contain any legal provisions, it scores 1 point.

- **Completeness of the Answer (2 points):** High-quality legal responses should comprehensively cover all aspects of the question and address all issues raised by the user. The specific criteria are as follows: (1) The answer is comprehensive and addresses all of the user's questions, scoring 2 points. (2) The answer covers some aspects of the question or addresses some of the user's questions, scoring 1 point. (3) The answer is scattered and does not cover all the aspects that the question aims to solve, scoring 0 points.

We require GPT-4-turbo to provide its evaluation in a step-by-step reasoning format. According to our experiments, step-by-step reasoning is more objective and accurate than directly giving scores. The specific reasoning process is as follows:

(1) We require GPT-4-turbo to analyze each dimension of the two candidate answers.

(2) Based on the analysis, GPT-4-turbo provides the scores.

(3) The scores for each evaluation dimension are summarized to obtain the total scores for Answer A and Answer B.

(4) Finally, based on the total scores of the candidate answers, the preference is determined.

A.7 Generating Legal Questions Using GPT-4-turbo

In Sec. 4.1, to enable the model to learn about a specific legal provision L, we need to construct a set of questions related to the specific legal provision L.

(1) Provide the legal provision and set the task: Provide the legal provision L to GPT-4-turbo and instruct it to assume the role of a law professor, designing multiple questions that can be answered based on this legal provision. For this task, we set GPT-4-turbo's temperature parameter to 0.7 to ensure that the generated questions are both creative and reasonable.

(2) Impose constraints on the questions: Impose constraints on the questions to ensure they are reasonable and answerable within the framework of the provided legal provision. Each question should be contextualized rather than being a simple query. Additionally, the phrasing should align with the vocabulary and style commonly used in modern Chinese.

(3) Design diverse question formats: Design a range of question formats to enhance the diversity of the generated questions. Examples include using a "first-person perspective with subjective emotion," a "first-person perspective maintaining objective neutrality," and a "colloquial style," among others, to create a variety of question forms and styles.

A.8 Generating High-Quality Responses Based on Given Legal Materials

In Sec. 4.1, we provide GPT-4-turbo with a legal question Q and the relevant legal provision L, asking it to generate a response. The process is as follows: First, we provide the reference legal provision L to GPT-4-turbo and instruct it to act as a legal expert, using the provision to answer the question. We set GPT-4-turbo's temperature to 0.7.

To ensure the response's quality, we impose specific constraints on GPT-4-turbo's output through prompt guidelines. These guidelines require GPT-4-turbo to first summarize and clarify the question, identifying the key points that need addressing. It should then use the provided legal provision L to answer the question. If the provision does not fully address the issue, GPT-4-turbo should supplement the answer with additional relevant information to improve completeness and usefulness.

A.9 Detailed Introduction to Baseline

- **Llama 3.1-70B:** Llama 3.1 70B shows comprehensive improvements over its predecessor, natively supporting 8 languages with a maximum context window of 128k.

- **Llama 3.1-405B:** Llama 3.1 405B is trained on 150 trillion tokens (equivalent to 750 billion words), with fine-tuning on 25 million synthetic data. Llama 3.1 405B is fully competitive with the most advanced proprietary models in various tasks.

- **GPT4o-mini:** GPT-4o mini is the mini version of OpenAI's GPT-4o, featuring low cost and rapid response capabilities, suitable for a variety of application scenarios.

- **GPT4o:** GPT-4o is the latest commercial model developed by OpenAI, with perfor-

mance in English text and code comparable to GPT-4-turbo.

- **wisdomInterrogatory:** Based on the Baichuan 7B architecture, this model undergoes secondary pre-training using 40GB of legal data. During the instruction fine-tuning phase, 100k of instruction fine-tuning training data is used.

- **LexiLaw:** This model, based on the ChatGLM-6B architecture, is fine-tuned with a variety of legal and general domain datasets. It includes BELLE 1.5M general domain data, 144k legal QA from LawGPT_zh and Baidu Zhidao, law exam and directive data from Lawyer LLaMA, and 20k high-quality QA data from Hualaw, supplemented with laws, regulations, and legal reference texts.

- **Qwen2-Instruct-7B:** Qwen2-7B-Instruct supports context lengths of up to 131,072 tokens, based on the Transformer architecture.

- **GLM4-Chat-9B:** GLM4-9B is the open-source version in the latest generation of the GLM-4 series of pre-trained models.

The retrieval system we use is the BAAI General Embedding (BGE), specifically version bge-base-zh-v1.5. BGE is suitable for tasks such as text similarity, ranking, question answering, and retrieval across multiple languages.

A.10 Evaluating Model Performance on LawBench

In Tab. 8, our model demonstrates superior performance in the tasks of Case Analysis and Legal Consultation on the Lawbench dataset, outperforming its baseline model, GLM4-Chat-9B, and surpassing all other open-source models. The improvement in accuracy for single-choice questions can be attributed to the model's effective acquisition of a substantial amount of scenario, statute matching information pairs. This information plays a key role in analyzing and accurately answering single-choice questions, which contributes to the observed performance increase.

A.11 Statistical Distribution of LegalHalBench

Tab. 9 shows the Statistical Distribution of LegalHalBench.

Methods	**Case analysis** (Acc)	**Legal consultation** (ROUGE-L)
lawgpt-7b-beta1.1-hf	9.20	7.62
lexilaw-6b-hf	28.60	15.82
lawyer-llama-13b-hf	26.60	16.94
fuzi-mingcha-7b-hf	20.00	16.64
chatlaw-13b-hf	28.80	17.17
chatlaw-33b-hf	34.20	16.55
wisdomInterrogatory	25.40	18.26
GPT-3.5-turbo-0613	27.40	17.45
GPT4	48.60	19.65
GLM4-Chat-9B	54.40	15.11
Ours M_3	63.00	18.87

Table 8: Results on Lawbench.

	Civil Law	**Criminal Law**	**Total**
Legal Provisions	750	85	835
Questions	1,384	604	1,988
Avg. Length of Question	172.52	299.37	211.06
Avg. Length of Answer	471.43	295.20	417.89

Table 9: Statistical Distribution of LegalHalBench.

A.12 Implementation Details

In the first phase of training, based on Qwen2-Instruct-7B and GLM4-Chat-9B, we conduct two rounds of SFT with a learning rate of 5e-6. In the second phase, we perform three cycles of HIPO while maintaining the same initial learning rate of 5e-6. We use both SFT and HIPO under the LoRA framework (Hu et al., 2021). With a single A100 40G GPU, training one epoch of SFT takes approximately 1 hour, while training one full epoch of HIPO takes around 3.5 hours. For the traditional DPO, we adopt the same parameter settings as HIPO. For SimPO, we utilize the parameter settings described in the paper (Meng et al., 2024).

A.13 Cost of Constructing the LegalHalBench

We provide the needed cost details as follows: Initially, we use the GPT4-turbo API to answer 1,988 questions, costing approximately $111. In the second phase, four lawyers review and refine the responses, followed by a final review by two senior lawyers to ensure accuracy. Overall, this process requires over 200 hours of professional lawyer time.

```
Template = (
    "Task description: As a legal expert, you are required to evaluate which of the two different answers is superior based on the given question.\n\n"
    "Evaluation dimensions and grading criteria:\n"
    "1. Answer helpfulness: A high-quality answer should positively and proactively address the question posed by the inquirer.\n"
    "- 4 points: The answer positively and proactively resolves the inquirer’s question and is very helpful to the inquirer.\n"
    "- 2 points: The answer resolves some of the inquirer's issues and is helpful.\n"
    "- 0 points: The answer is irrelevant to the question, or does not resolve the inquirer’s question, and is not helpful to the inquirer.\n"
    "2. Relevance of legal provisions: A high-quality answer should provide legal provisions that are highly relevant to the question to support the presented views.\n"
    "- 2 points: The legal provisions in the answer are highly relevant to the question and the advice, and directly support the suggestions and viewpoints.\n"
    "- 1 point: The legal provisions in the answer are somewhat related to the question and advice, indirectly supporting the suggestions and viewpoints.\n"
    "- 0 points: The legal provisions in the answer are completely unrelated to the question.\n"
    "- If there are no legal provisions in the answer, this item scores 1 point.\n"
    "3. Completeness of the answer: A high-quality answer should comprehensively cover all aspects of the question and address all issues raised by the user.\n"
    "- 2 points: The answer is comprehensive and addresses all the user’s questions.\n"
    "- 1 point: The answer covers some aspects of the question or answers some of the user’s questions.\n"
    "- 0 points: The answer is fragmented and does not cover all the aspects intended to be addressed by the question.\n\n"
    "User's question:\n"
    f"{problem}\n\n"
    "Answer A:\n"
    f"{response1}\n\n"
    "Answer B:\n"
    f"{response2}\n\n"
    "First, please analyze Answers A and B based on the evaluation dimensions and scoring criteria. Based on the analysis, score Answers A and B in each evaluation dimension."
    "Secondly, sum up the total scores for Answers A and B based on the above evaluation dimensions."
    "Then, indicate your final preference with 'A', 'B', or 'Same'.\n\n"
    "Your response should use the following format:\n"
    "Answer helpfulness: <Analysis>, A:<score>/4, B:<score>/4\n\n"
    "Relevance of legal provisions: <Analysis>, A:<score>/2, B:<score>/2\n\n"
    "Completeness of the answer: <Analysis>, A:<score>/2, B:<score>/2\n\n"
    "Total score: A:<total score>, B:<total score>\n\n"
    "Final preference: <'A' or 'B' or 'Same'>\n\n"
)
```

Figure 8: Win-rate Template