Prompt4LJP: Prompt Learning for Legal Judgement Prediction

Anonymous ACL submission

Abstract

The task of Legal Judgment Prediction (LJP) involves predicting court decisions based on the facts of the case, including identifying the applicable law article, the charge, and the term of penalty. While neural methods have made significant strides in this area, they often fail to fully harness the rich semantic potential of language models (LMs). *Prompt learning*, a novel approach that reformulates downstream tasks as cloze-style or prefix-style prediction challenges for Masked Language Models using specialized prompt templates, has shown considerable promise across various Natural Language Processing (NLP) domains. However, the dynamic word lengths typical in LJP labels present a challenge to the standard prompt templates designed for single-word [MASK] tokens commonly used in many NLP tasks. To address this gap, we introduce the *Prompt4LJP* framework, a pioneering method tailored to incorporate the knowledge of LMs into the LJP task by effectively accommodating dynamic word lengths in labels. This framework leverages a dual-slot prompt template and correlation scoring to maximize the utility of LMs without requiring additional resources or complex tokenization schemes. Our method significantly outperforms current state-of-the-art techniques on the CAIL-2018 dataset, thereby enhancing the accuracy and reliability of LJP. This contribution not only advances the field of LJP but also demonstrates a novel application of prompt learning to complex tasks involving dynamic word lengths.

1 Introduction

Legal Judgment Prediction (LJP) aims to predict court decisions based on case facts, encompassing tasks such as law article prediction, charge prediction, and term of penalty prediction, as detailed in Figure 1. Substantial advancements in LJP have been achieved using sophisticated neural networks and text representation models (Luo et al., 2017; Feng et al., 2022; Zhang et al., 2023). For example, Luo et al. (2017) improved the integration of factual descriptions with law articles using attention mechanisms. However, these neural methodologies primarily focus on extracting in-domain information from LJP datasets, often neglecting the rich semantic and linguistic information available in language models (LMs). Current approaches typically treat LJP tasks as straightforward text classification problems (Fei et al., 2023), which limits their effectiveness in leveraging the comprehensive legal knowledge embedded in LMs.

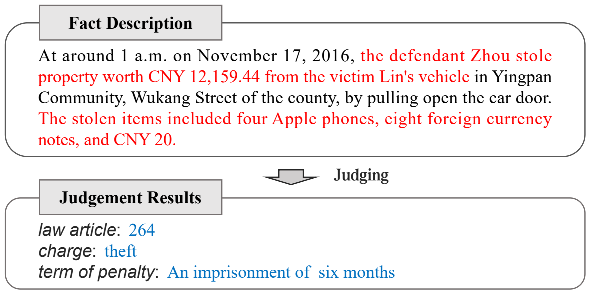

Figure 1: An illustration of legal judgment prediction. The text highlighted in red signifies key details extracted from the factual description, while the content highlighted in blue denotes the relevant law article, charge, and term of penalty applicable to the fact description.

Prompt learning, a transformative approach that reformulates downstream tasks as cloze-style or prefix-style prediction challenges for LMs, including Masked Language Models (MLMs) using specialized prompt templates, has shown considerable promise across various Natural Language Processing (NLP) domains (Schick and Schütze, 2020; Zhu et al., 2023; Wang et al., 2021; Zhang and Wang, 2023; Ding et al., 2021; Zhu et al., 2022; Xiang et al., 2022). One of the primary advantages of prompt learning is that it enables models to better understand downstream tasks, thereby stimulating the recall of relevant knowledge embedded in

LMs (Sahoo et al., 2024; Sabbatella et al., 2024). However, current prompt learning templates are not well-suited for LJP tasks due to the nature of LJP labels. The dynamic word lengths typical in LJP labels present a challenge to the standard prompt templates designed for single-word [MASK] token commonly used in many NLP tasks. LJP labels often consist of complex, multi-word expressions, such as legal charges like "intentional injury" or legislative references like "Article 256," which cannot be adequately captured by single [MASK] token. This misalignment hinders the effective use of LMs' extensive pre-trained knowledge in LJP tasks.

To address this gap, we introduce the *Prompt4LJP* framework, a pioneering method that effectively utilizes LMs knowledge tailored for the complex and dynamic nature of LJP labels. Our contributions are centered on harnessing the extensive pre-trained knowledge of LMs and enhancing their ability to recall and apply relevant legal information through two main technical innovations:

First, we developed a Dual-Slot Prompt Template that directly incorporates the given fact description and a potential label into the prompt, respectively, transforming LJP tasks into masked language model challenges. This design engages the language model to apply its learned semantic knowledge and intuitively grasp the task through structured template guidance, accommodating the complex, multi-word labels typical in LJP.

Second, we introduced a novel Correlation Scores Ranking system to assess candidate labels generated for each fact scenario. Although LJP tasks typically involve a single label per case, in real judicial applications, charges or law articles can be very similar. The ranking mechanism generates candidate labels that can assist in practical judicial decision-making by providing closely related alternatives and identifying the most accurate ground-truth label. This system significantly enhances the LM's capacity to leverage its extensive pre-trained legal knowledge, thereby boosting both accuracy and reliability in LJP tasks.

To evaluate the Prompt4LJP method, we conducted rigorous testing on the CAIL-2018 dataset, a recognized benchmark in the LJP field. The results are highly promising, showing that Prompt4LJP not only meets but often exceeds the performance of existing SOTA neural models in predicting charges and terms of penalty. These outcomes highlight the efficacy of our tailored prompt template in adeptly managing multi-word labels and markedly improving the accuracy and reliability of LJP systems.

Our findings demonstrate that the Prompt4LJP framework effectively utilizes the rich, pre-trained knowledge embedded in LMs, optimizing it specifically for LJP tasks without the necessity for additional external datas. This framework stimulates LMs to recall pre-trained legal knowledge relevant to LJP tasks, significantly enhancing their performance. Furthermore, this significant advancement in applying LMs knowledge directly addresses the unique demands of LJP, setting a new standard in the field.

2 Related Work

2.1 Legal Judgement Prediction

Traditional methods for LJP primarily utilized rule-based or mathematical models (Kort, 1957; Segal, 1984; Ulmer, 1963), which, despite their accuracy, are difficult to generalize due to the extensive cost of feature engineering. With the rise of neural network techniques in NLP, there's been a shift towards applying these methods to LJP, leading to numerous studies (Luo et al., 2017; Zhong et al., 2018; Yang et al., 2019; Xu et al., 2020; Yue et al., 2021; Feng et al., 2022; Zhang et al., 2023; Liu et al., 2023).

These studies often employ a multi-task learning (MTL) framework, modeling the subtasks of LJP to capture dependencies among them and enhance prediction accuracy. Notable examples include Zhong et al. (2018)'s topological framework, Yue et al. (2021)'s segmented factual analysis, Feng et al. (2022)'s use of key event information with consistency constraints, and Zhang et al. (2023)'s contrastive learning approach to differentiate between similar legal terms. MTL introduces complexities such as the need for optimal weight allocation and specific loss function design, which complicate hyperparameter tuning. Given the success of prompt learning in various domains(Xiang et al., 2022; Zhang and Wang, 2023), we are inspired to explore its potential in improving LJP.

2.2 Prompt Learning in LJP

Prompt learning, facilitated by pre-trained language models such as BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019), has revolutionized various NLP tasks by framing them as cloze-style or prefix-style prediction challenges. Although successful in domains like news recommendation (Zhang and Wang, 2023), implicit discourse rela-

TYPE	TASK	TEMPLATE
Discrete	Law Articles	According to the following fact, whether the defendant violates Article $< law >$ of the Criminal Law: [MASK]. $< fact >$
	Charges	According to the following fact, whether the defendant is guilty of $< crime >$: [MASK]. $< fact >$
	Terms of Penalty	According to the following fact, whether it is reasonable to impose a $< term >$ punishment on the defendant: [MASK]. $< fact >$
Continuous	All	$[P_1]...[P_{l_1}] < label > [Q_1]...[Q_{l_2}]$ [MASK] $[M_1]...[M_{l_3}] < fact >$

Table 1: Prompt templates designed for the three subtasks in this paper, including discrete and continuous templates.

tionship recognition (Xiang et al., 2022), and text classification (Schick and Schütze, 2020; Zhu et al., 2023; Wang et al., 2021), its application in LJP faces unique challenges due to the complexity of legal terminology and the often complex, multi-word expressions that LJP labels entail.

Sun et al. (2024) tackled the challenge of representing intricate legal charges by employing a fixed template with ten [MASK] tokens. This approach, while innovative, resulted in significant data sparsity from the numerous possible combinations of the [MASK] tokens. To mitigate this issue, they incorporated external knowledge bases to enrich contextual understanding, though their method depended on calculating the similarity between predicted outputs and actual legal terms, which can be problematic due to nuanced differences between similar terms.

Prompt4LJP diverges significantly by simplifying the integration of labels and facts. Our model uses a dual-slot prompt template that directly incorporates the given fact and a potential label into the prompt. This method efficiently evaluates correlation scores between facts and labels, converting the traditional multi-class classification challenge into a more straightforward binary prediction task. The Prompt4LJP framework guides LMs to leverage their extensive pre-trained knowledge more effectively, enhancing their ability to recall and apply relevant legal information for LJP tasks. This enhancement not only eliminates the need for external data sources and complex tokenization strategies but also increases the accuracy and applicability of prompt learning for LJP.

3 Our proposed method

In this section, we first give the essential definitions of LJP task. Then, we explain the details of our prompt templates, verbalizer and answer words. Finally, we present a comprehensive overview of our Prompt4LJP framework and detail the training strategy employed.

3.1 Task Definition

The three subtasks in LJP are denoted as t_a, t_c, and t_b, representing law article prediction, charge prediction, and term of penalty prediction, respectively. Each subtask is a multi-label classification task. To maintain consistency with previous studies (Zhang et al., 2023; Feng et al., 2022; Xu et al., 2020), we consider only samples in which each subtask has a single label in the dataset. Given the factual description x of a legal case, the LJP task, denoted as $T = \{t_a, t_c, t_b\}$, aims to predict the result labels for the three subtasks. Formally, y_i^t represents the i-th label of a subtask-specific label set Y^t, where $y_i^t \in Y^t$ and $i = 1, 2, ..., |Y^t|$ for $t \in T$. For instance, in the charge prediction subtask t_c, the label set $Y^c = \{Theft, Robbery, ..., Arson\}$ includes $y_1^c = Theft$, $y_2^c = Robbery$, and $y_{|Y^c|}^c = Arson$.

3.2 Dual-Slot Prompt Template for LJP

Within our research, each subtask—t_a, t_c, and t_b—is treated as an independent task, each utilizing a consistent prompt template format. Unlike conventional multi-class classification templates, which typically feature only one input slot for a single-word label or its expanded word from the LM's vocabulary, our templates include two slots: $< fact >$ and $< label >$. Specifically, $< fact >$ represents the fact statement x, while $< label >$ corresponds to the label y_i^t. Here, $< label >$ serves as a unified representation of $< crime >$, $< law >$, and $< term >$, representing charges, law articles, and terms of penalty, respectively. For instance, in the charges prediction task t_c, we construct a prompt template denoted as $f_{prompt}^c(< fact >, < crime >)$ as follows:

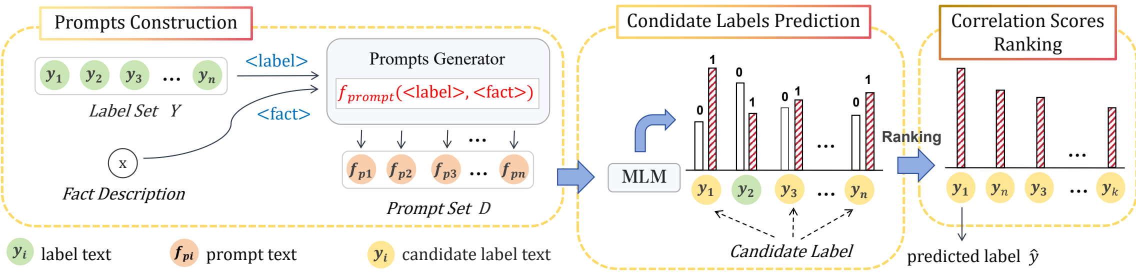

Figure 2: The framework of Prompt4LJP. The three areas represent the three steps: Prompts Construction, Candidate Labels Prediction, and Correlation Scores Ranking.

According to the following fact, whether the defendant is guilty of <crime> : [MASK]. <fact>

Our method innovatively converts the multi-class classification task into a cloze-style mask-prediction task within the template. Prompt-guided MLMs aim to predict whether a given label y_i^t is a plausible result label for the factual description x. Additionally, we explore the impact of employing the continuous prompt on prediction performance for LJP. In our continuous template, we maintain the structure of the discrete prompts but replace discrete tokens with custom pseudo tokens: $[P_{1:l_1}]$, $[Q_{1:l_2}]$, and $[M_{1:l_3}]$. These pseudo tokens, serving as learnable parameters, are strategically positioned before $< label >$, [MASK], and $< fact >$ respectively, where l_1, l_2, l_3 represent the numbers of pseudo tokens. For further clarity, Table 1 provides an overview of our designed prompt templates for the three subtasks.

3.3 Answer Words and Verbalizer

Based on our provided prompt template $f_{prompt}^t(< fact >, < label >)$, we simply select two opposite words from the vocabulary V of the MLMs as our answer words, specifically *yes* and *no*. These two words constitute our answer space V_a, where $V_a = \{no, yes\} \subset V$. In MLMs M, the probability of filling each word v from V_a into the [MASK] can be calculated as follows:

$$P(v \in V_a | x, y_i^t) = P_M(w | f_{prompt}^t(x, y_i^t)) \quad (1)$$

where w represents filling the $[MASK]$ with the answer word $v \in V_a$, i.e., $[MASK] = v$. As we don't directly use labels as answer words, the primary emphasis of our work doesn't focus on the construction of the verbalizer. Nonetheless, within the prompt learning paradigm, the verbalizer holds significance. Thus, we provide a simplified formulation for our work:

$$f_{verbalizer}(v) = \begin{cases} x \Rightarrow y_i^t & v = yes \\ x \nRightarrow y_i^t & v = no \end{cases} \quad (2)$$

In this formulation, $x \Rightarrow y_i^t$ indicates that y_i^t is a possible result label for x, while $x \nRightarrow y_i^t$ indicates the absence of y_i^t as a potential result label for x.

3.4 Our Prompt4LJP Framework

For each given factual description x_i, we aim to predict the ground-truth label $\hat{y}_i^t$ for the three subtasks—t_a, t_c, and t_b—respectively. Figure 2 illustrates our Prompt4LJP framework, which encompasses three steps for legal judgments: prompts construction, candidate labels prediction, and correlation scores ranking.

Step1:Prompts Construction. In subtask t, we generate $|Y^t|$ prompts for each given fact x_i. Formally, we denote $f_{prompt,i,j}^t(x_i, y_j^t)$ as the j-th prompt associated with the fact x_i, where x_i and y_j^t are inserted into the prompt template's $< fact >$ and $< label >$ slots, respectively. We represent these $|Y^t|$ prompts for x_i as a prompt set $D_i^t = \{f_{prompt,i,j}^t(x_i, y_j^t) \mid y_j^t \in Y^t\}$ in the subtask t. The prompt whose $< label >$ corresponds to the target label $\hat{y}_i^t$ is called the *positive prompt*, while the remaining $|Y^t| - 1$ prompts, which contain labels from Y^t excluding the label corresponding to $\hat{y}_i^t$, are termed *negative prompts*.

Step2:Candidate Labels Prediction. Denoted $s_{i,j}^t$ as the correlation score between x_i and y_j^t. The correlation score can be served as the confidence whether the label y_j^t is a result label for x_i. The higher correlation score, the greater probability that y_j^t is the ground-truth label for x_i. For the k-th prompt $f^t_{prompt,i,k}(x_i, y^t_k)$ of x_i, if the probability of the answer word $v = yes$ is greater than that of $v = no$, i.e., $P(v = yes|x_i, y^t_k) > P(v = no|x_i, y^t_k)$, we consider x_i and y^t_k to be correlated, with a correlation score of $s^t_{i,k} = P(v = yes|x_i, y^t_k)$. Then, we create a pair consisting of the corresponding label and the correlation score, denoted as $< y^t_k, s^t_{i,k} >$, and add it into the candidate label set C^t_i. When the same evaluation is applied to all prompts in the prompt set D^t_i, we will obtain a candidate label set C^t_i for the fact x_i in the subtask t, where $0 \leq |C^t_i| \leq |Y^t|$.

Step3:Correlation Scores Ranking. We will sort the pair $< y^t_j, s^t_{i,j} >$ in the candidate label set C^t_i by the correlation score $s^t_{i,j}$ in order of largest to smallest. The label y^t_k corresponding to $< y^t_k, s^t_{i,k} >$ with the highest correlation score $s^t_{i,k}$ will be served as the final prediction label $\hat{y}^t_i$ for the fact x_i in the subtask t, which can be formalized as follow:

$$\hat{y}^t_i = g(max\{s^t_{i,j}| < y^t_j, s^t_{i,j} > \in C^t_i\}), t \in T \quad (3)$$

where g is a function finding the corresponding label y^t_k of the highest correlation score $s^t_{i,k}$ from the candidate label set C^t_i.

3.5 Training

We fine-tune the parameters of a MLM using the public CAIL2018-small dataset (Xiao et al., 2018) with our custom prompt templates and answer space. For each subtask t, we use the cross-entropy loss function to train the corresponding model:

$$L^t = \frac{1}{K}\sum_{k=1}^{K}[z^t_k log p^t_k + (1 - z^t_k) log(1 - p^t_k)] \quad (4)$$

where z^t_k and p^t_k are the gold label and predicted probability of the k-th training instance in the subtask t, respectively. We use the AdamW optimizer (Loshchilov and Hutter, 2017) with L2 regularization for model training.

4 Experiments

In this section, we introduce our experimental settings and conduct a series of experiments to evaluate the proposed Prompt4LJP framework.

4.1 Experimental Settings

Dataset. We validate the effectiveness of our method using the publicly available dataset from the Chinese AI and Law challenge (Xiao et al., 2018): CAIL-small (the exercise stage dataset). Each sample in the dataset comprises a factual description of a legal case, along with applicable law articles, charges, and terms of penalty. To maintain consistency with the latest SOTA method like EPM (Feng et al., 2022) and CL4LJP (Zhang et al., 2023), we adhere to their data preprocessing pipelines, which involves filtering out samples with multiple labels. Statistical details about the dataset are presented in Table 2.

Dataset	CAIL-small
#Training Set Cases	96,540
#Validation Set Cases	12,903
#Testing Set Cases	24,848
#Law Articles	101
#Charges	117
#Term of Penalty	11

Table 2: Statistics on CAIL-small.

Implementation Details. We utilize the pre-trained language model bert-base-chinese (Devlin et al., 2018) provided by HuggingFace transformers (Wolf et al., 2020). For consistency across all subtasks of LJP, we employ the same training strategy. Specifically, the model is trained on 4 NVIDIA GeForce RTX 3090 GPUs simultaneously, with a batch size of 16 for each GPU. We use the AdamW optimizer (Loshchilov and Hutter, 2017) with a learning rate of 2e-5 and train the model for 8 epochs. The final epoch model is evaluated on the testing set. To assess the performance of our methods and baseline models, we employ four widely-used metrics for multi-class classification tasks: accuracy (Acc.), macro-precision (MP), macro-recall (MR), and macro-F1 (F1).

4.2 Baseline Methods

To fully demonstrate the effectiveness and superiority of our methodology in LJP tasks, emphasizing its capability to leverage knowledge acquired during the pre-training process, we conducted comparisons with three fundamental paradigms: state-of-the-art (SOTA) neural networks, LLM-specific techniques such as prompt-based in-context learning (ICL), and parameter-efficient fine-tuning (PEFT).

Neural Methods. We selected seven SOTA neural models in Chinese LJP as our baseline models, including: (1) ***MLAC*** (Luo et al., 2017), which is an attention-based neural network method for charge prediction that incorporates the k most relevant law articles. (2) ***TOPJUDGE*** (Zhong et al., 2018), which constructs a topological multi-task learn-

Tasks	Law Articles				Charges				Terms of Penalty			
Metrics	Acc.	MP	MR	F1	Acc.	MP	MR	F1	Acc.	MP	MR	F1
ICL Method												
Few-shot	64.01	59.81	53.65	51.94	65.06	69.70	60.52	59.72	08.13	17.84	13.37	07.68
PEFT Method												
PEFT-Qwen1.5	50.85	42.03	36.52	36.96	57.25	50.49	46.59	45.82	23.11	28.20	18.27	18.48
SOTA Neural Methods												
MLAC∗	73.02	69.27	66.14	64.23	74.73	72.65	69.56	68.36	36.45	34.50	29.95	29.64
TOPJUDGE∗	78.60	76.59	74.84	73.72	81.17	81.87	80.57	79.96	35.70	32.81	31.03	31.49
MPBFN∗	76.83	74.57	71.45	70.57	80.17	78.88	75.65	75.68	36.18	33.67	30.08	29.43
LADAN∗	78.70	74.95	75.61	73.83	82.86	81.69	80.40	80.05	36.14	31.85	29.67	29.28
NeuralJudge∗	79.02	75.69	75.23	74.87	81.22	77.51	78.17	77.99	36.84	34.80	32.22	32.48
EPM∗	**84.65**	**80.82**	77.55	**78.10**	84.10	84.55	80.22	81.43	36.69	35.60	32.70	32.99
CL4LJP	77.01	75.42	73.38	72.48	79.14	78.45	78.11	77.25	36.31	33.20	30.05	29.53
Prompt4LJP(ours)												
Discrete	79.24	78.28	77.27	76.25	84.66	85.19	84.12	83.68	**40.44**	39.40	**37.02**	**37.75**
Continuous	80.95	80.08	**78.42**	77.49	**86.01**	**85.82**	**84.68**	**84.67**	40.41	**40.92**	34.86	37.04

Table 3: Experimental results on CAIL2018-small dataset. Text in **bold** denotes the best result, while underline indicates the second best result across the entire table. Results marked with ∗ represent those from models reported in (Feng et al., 2022), which share the same data preprocessing pipeline with our approach. All other results were obtained from our own experiments.

ing framework to capture dependencies among the three subtasks of LJP. (3) ***MPBFN-WCA*** (Yang et al., 2019), which utilizes dependencies among the three subtasks and integrates word collocation features of fact descriptions into the network via an attention mechanism to distinguish similar cases. (4) ***LADAN*** (Xu et al., 2020), which proposes a graph neural network to capture discriminative features between confusing law articles. (5) ***NeurJudge*** (Yue et al., 2021), which separates the factual description into several parts, each making a judgment for other subtasks. (6) ***EPM*** (Feng et al., 2022), which leverages key event information of legal cases to predict the result and utilizes consistency constraints between the three subtasks. (7) ***CL4LJP*** (Zhang et al., 2023), which introduces a neural contrastive learning framework to capture the relationship between factual descriptions, similar law articles, and corresponding charges.

PEFT Method. Considering the limited computational resources, we chose the newly released Qwen1.5-1.8B-Chat with small parameters (Bai et al., 2023), which is tailored for Chinese, as our backbone for LoRA fine-tuning (Hu et al., 2021) on the CAIL2018-Small training set. The fine-tuned model is denoted as ***PEFT-Qwen1.5***. More details about the fine-tuning settings can be found in Appendix A.

ICL Method. Evaluation results from (Fei et al., 2023) indicate that the Qwen7B-Chat model (Bai et al., 2023) exhibits the best performance on legal tasks among Chinese-oriented LLMs. Therefore, we selected the latest known Qwen model, Qwen1.5-14B-Chat, as our backbone for zero-shot and few-shot experiments. For the given descriptions, we constructed suitable prompts to guide the model in providing the applicable law article, charge, and term of penalty. Due to the maximum input token limit of the model, we only conducted 0-shot, 2-shot, and 4-shot experiments. Experimental results show that the 4-shot setup yields the best results. Table 3 only presents the optimal results. We denote this method as ***Few-shot***. Specific prompts for LJP are shown in Appendix B.

4.3 Main Experimental Results

The main results of the three subtasks are summarized in Table 3. Our Prompt4LJP method, whether using discrete or continuous templates, outperforms baseline methods, particularly in charges and terms of penalty prediction. Compared to the best baseline model EPM, Prompt4LJP shows F1-score improvements of 2.25% and 4.76% (charges and penalties) with discrete templates, and 3.24% and 4.05% with the continuous template, demonstrating prompt-learning's ability to leverage pre-trained knowledge and adapt flexibly to specific tasks.

However, our performance in law article prediction is inferior to EPM, likely because the $< law >$ slot is filled with simple numbers (e.g., "256"), while the $< crime >$ and $< term >$ slots contain contextually rich phrases like "robbery" and "imprisonment for less than one year," providing more linguistic and semantic information to the model. Future research will focus on handling this issue.

Continuous templates generally outperform discrete ones in law articles and charges prediction, with accuracy and F1-score differences exceeding 1%, due to the flexibility of learnable pseudo tokens in continuous prompts. However, for penalty prediction, continuous templates slightly underperform discrete ones, with gaps averaging less than 0.4%, possibly because penalty prediction benefits from the stability of discrete templates.

Overall, both direct fine-tuning and prompting of LLMs perform poorly on LJP tasks, indicating a failure to fully utilize the encyclopedic linguistic evidence embedded in the pre-training process. In contrast, our Prompt4LJP framework, with its dual-slot prompt template, effectively harnesses legal knowledge within the pre-trained LMs, significantly improving LJP performance by better capturing domain-specific information and enhancing the model's capability in complex legal reasoning, ultimately surpassing current SOTA methods in accuracy and reliability.

4.4 Hyperparameter Analysis

The number of pseudo tokens within continuous templates and the number of negative prompts for training represent our core hyperparameters. Experimental findings reveal that different parameter configurations exert distinct impacts across LJP subtasks.

The Influence of the Quantity of Pseudo Tokens. From Table 1, we note that the three independent subtasks of LJP share the same continuous template pattern with $[P_{1:l_1}]$, $[Q_{1:l_2}]$, and $[M_{1:l_3}]$. To explore the impact of varying the number of pseudo tokens (i.e., l_1, l_2, l_3) on prediction performance, we conduct three experiments for each subtask. We adopt a basic hyperparameter tuning strategy, setting l_1, l_2, l_3 to the same value, denoted as $l = l_1 = l_2 = l_3$. Given that in Chinese expressions, a complete word typically consists of two Chinese characters, we vary l in $\{0, 2, 6, 10, 16\}$, increasing in multiples of 2.

Figure 3 demonstrates varying optimal l values across subtasks according to F1-score, solely based on differences in label values and quantities within the same continuous template pattern. Additionally, for terms of penalty prediction at $l = 10$ and $l = 16$, F1-score increases while accuracy decreases, suggesting the introduction of ambiguities due to the absence of pre-training knowledge in randomly initialized pseudo tokens.

The Effect of Number of Negative Prompts. In our experiments, we noticed a significant impact on prediction performance based on the number of negative prompts (n) used during training. We varied n from 1 to 10 and focused on predicting law articles and terms of penalty. Our experiments solely employed the discrete template and analyzed F1-score, considering the imbalanced data distribution in the CAIL2018 dataset.

Results, as depicted in Figure 4, illustrate that increasing n enhances the model's ability to predict low-frequency labels, particularly evident in law articles prediction with its larger label set (101 labels). This improvement is attributed to the model's capacity to learn relevant characteristics between the fact and the target label, alongside irrelevant characteristics between the fact and non-target labels. Consequently, the model captures more discriminative information, benefiting overall prediction performance. Conversely, in terms of penalty prediction, which involves a smaller label set (11 labels), setting n to its maximum value (i.e., $n = 10$) leads to overfitting, thereby diminishing prediction performance.

4.5 Different Label Selection Strategies

The Prompt4LJP framework enables the prediction of a candidate labels set for each provided factual description, a crucial aspect in tasks with extensive label sets as it helps narrow down potential labels. In the charges prediction, our method significantly reduces the candidate charges set size (117 labels) to approximately $1/24$ of the original label set size on average. Additionally, our approach achieves a macro-recall exceeding 90%, indicating that the candidate label sets generated by our method almost entirely encompass the ground-truth labels.

However, accurate label selection remains crucial even with the candidate label set at hand. In addition to our proposed ***Ranking*** strategy, which selects the final ground-truth label based on the

Tasks	Law Articles		Charges		Terms of Penalty	
Metrics	Acc.	F1	Acc.	F1	Acc.	F1
Best Baseline						
EPM	84.65	78.10	84.10	81.43	36.69	32.99
Continuous Template						
Train-10%	73.85	64.52	75.86	74.34	27.27	23.79
Train-30%	78.01 (+4.16)	71.87 (+7.35)	81.48 (+5.62)	80.66 (+6.32)	29.28 (+2.01)	29.35 (+5.56)
Train-50%	78.77 (+0.76)	72.45 (+0.58)	83.34 (+1.86)	**81.70** (+1.04)	**40.83** (+11.55)	**35.12** (+5.77)
Train-70%	79.06 (+0.29)	74.62 (+2.17)	**85.41** (+2.07)	**84.24** (+2.54)	**38.93** (-1.90)	**36.80** (+1.68)
Train-90%	79.98 (+0.92)	76.09 (+1.47)	84.99 (-0.42)	84.11 (-0.13)	41.52 (+2.59)	36.80 (+0.00)
Train-Full	80.95 (+0.97)	77.49 (+1.40)	86.01 (+1.02)	84.67 (+0.56)	40.41 (-1.11)	37.04 (+0.24)

Table 4: Experimental results with fewer training data under the continuous prompt template. The datas in **bold** indicates that our method outperformed the best baseline model EPM when using less than 70% of the training data.

highest correlation score, we also investigate the impact of alternative selection strategies on the experimental results. These strategies include random selection (***Random***) and further refinement using the LLM (***Qwen1.5-14B***). Detailed explanations of these strategies are provided in Appendix C.

Due to space limitations, our analysis primarily centers on charge prediction within the continuous template, with results for the other subtasks following a similar trend. Table 5 showcases the outcomes of the three distinct selection strategies. Particularly noteworthy is the comparable performance of the LLM-based refinement strategy in contrast to our ***Ranking*** approach. Enhancements in LLM selection efficacy could be pursued through the refinement of more suitable prompts.

Metrics	Acc.	MP	MR	F1
Random	79.44	76.89	77.04	76.42
Qwen1.5-14B	85.69	85.74	83.29	83.83
*Ranking(**ours**)*	86.01	85.82	84.68	84.67

Table 5: Experimental results with different selection strategies for the charge prediction on the continuous template.

4.6 Impact of Training Dataset Size

Several studies have highlighted the efficacy of prompt learning with smaller datasets in various NLP tasks (Wang et al., 2021; Xiang et al., 2022; Zhang and Wang, 2023). Our study examines the Prompt4LJP model's impact on prediction performance across three subtasks with limited data. Due to space constraints, we present only the continuous template results, with similar trends under the discrete template detailed in Appendix D.

We trained the three subtasks using 10%, 30%, 50%, 70%, and 90% of the available data. The results, as shown in Table 4, shows significant improvements in accuracy and F1-score from 10% to 30% of the training data for charges and law articles. However, gains diminish from 50% to 100%, indicating robust generalization capabilities by harnessing the knowledge embedded in pre-trained MLs to support the LJP task, irrespective of training data quantity. Furthermore, our approach surpasses the top-performing baseline EPM, trained on the entire dataset, utilizing only 50% of the training data for predicting penalty terms and 70% for predicting charges. These findings underscore the advanced capabilities of prompt learning methods in managing situations with limited training data, showcasing their superiority compared to shallow neural networks.

5 Conclusion

In our paper, we introduce Prompt4LJP, a novel prompt learning framework tailored for LJP tasks. To address the challenge posed by multi-word labels in LJP, we employ a dual-slot prompt template along with correlation scoring. Through extensive experiments conducted on the CAIL2018-small dataset, we demonstrate the efficacy of Prompt4LJP in enhancing the accuracy and reliability of LJP predictions. Our results indicate that Prompt4LJP outperforms existing SOTA approaches, particularly in the prediction of charges and penalty terms. Moreover, our model's performance underscores its adept utilization of the rich, pre-trained knowledge inherent in LMs, obviating the need for additional external data. This highlights the genuine application of LM knowledge in LJP tasks.

Limitations

In our endeavor to integrate the prompt learning paradigm into LJP, we acknowledge several limitations. Firstly, our investigation was limited to singular prompt templates and simple answer word pairs for each subtask. Secondly, our study has solely examined the performance of our method on the chinese-bert-base pre-trained model, neglecting its applicability to larger parameter LLMs. Future work will address this limitation. Thirdly, to maintain consistency with prior SOTA methodologies, our analysis concentrated exclusively on single-label instances within the CAIL2018-small dataset. Nevertheless, we recognize the potential applicability of our approach to scenarios involving multiple labels and plan to conduct more comprehensive investigations in future research endeavors.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ning Ding, Yulin Chen, Xu Han, Guangwei Xu, Pengjun Xie, Hai-Tao Zheng, Zhiyuan Liu, Juanzi Li, and Hong-Gee Kim. 2021. Prompt-learning for fine-grained entity typing. *arXiv preprint arXiv:2108.10604*.

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*.

Yi Feng, Chuanyi Li, and Vincent Ng. 2022. Legal judgment prediction via event extraction with constraints. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 648–664.

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*.

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.

Yifei Liu, Yiquan Wu, Yating Zhang, Changlong Sun, Weiming Lu, Fei Wu, and Kun Kuang. 2023. Ml-ljp: Multi-law aware legal judgment prediction. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1023–1034.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Bingfeng Luo, Yansong Feng, Jianbo Xu, Xiang Zhang, and Dongyan Zhao. 2017. Learning to predict charges for criminal cases with legal basis. *arXiv preprint arXiv:1707.09168*.

Antonio Sabbatella, Andrea Ponti, Ilaria Giordani, Antonio Candelieri, and Francesco Archetti. 2024. Prompt optimization in large language models. *Mathematics*, 12(6):929.

Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2024. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*.

Timo Schick and Hinrich Schütze. 2020. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*.

Jeffrey A Segal. 1984. Predicting supreme court cases probabilistically: The search and seizure cases, 1962-1981. *American Political Science Review*, 78(4):891–900.

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

S Sidney Ulmer. 1963. Quantitative analysis of judicial processes: Some practical and theoretical applications. *Law and Contemporary Problems*, 28(1):164–184.

Chengyu Wang, Jianing Wang, Minghui Qiu, Jun Huang, and Ming Gao. 2021. Transprompt: Towards an automatic transferable prompting framework for few-shot text classification. In *Proceedings of the*

2021 conference on empirical methods in natural language processing, pages 2792–2802.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Wei Xiang, Zhenglin Wang, Lu Dai, and Bang Wang. 2022. Connprompt: Connective-cloze prompt learning for implicit discourse relation recognition. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 902–911.

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

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

Wenmian Yang, Weijia Jia, Xiaojie Zhou, and Yutao Luo. 2019. Legal judgment prediction via multi-perspective bi-feedback network. *arXiv preprint arXiv:1905.03969*.

Linan Yue, Qi Liu, Binbin Jin, Han Wu, Kai Zhang, Yanqing An, Mingyue Cheng, Biao Yin, and Dayong Wu. 2021. Neurjudge: A circumstance-aware neural framework for legal judgment prediction. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 973–982.

Han Zhang, Zhicheng Dou, Yutao Zhu, and Ji-Rong Wen. 2023. Contrastive learning for legal judgment prediction. *ACM Transactions on Information Systems*, 41(4):1–25.

Zizhuo Zhang and Bang Wang. 2023. Prompt learning for news recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 227–237.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*.

Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Chaojun Xiao, Zhiyuan Liu, and Maosong Sun. 2018. Legal judgment prediction via topological learning. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 3540–3549.

Tiantian Zhu, Yang Qin, Qingcai Chen, Baotian Hu, and Yang Xiang. 2022. Enhancing entity representations with prompt learning for biomedical entity linking. In *IJCAI*, pages 4036–4042.

Yi Zhu, Ye Wang, Jipeng Qiang, and Xindong Wu. 2023. Prompt-learning for short text classification. *IEEE Transactions on Knowledge and Data Engineering*.

Appendix

A Fine-tuning Hyper-parameters

We utilized the LLama Factory (Zheng et al., 2024) for LoRA fine-tuning on Qwen1.5-1.8B-Chat, leveraging 4 NVIDIA GeForce RTX 3090 GPUs. Specifically, we transformed the original CAIL2018-small dataset into the "instruction-input-output" JSON format tailored for the model, as depicted in Table 10. The model hyperparameters were fine-tuned based on the training set of CAIL2018-small, with detailed settings provided in Table 6.

Hyper-parameter	**Value**
per-device-train-batch-size	4
gradient-accumulation-steps	2
learning-rate	5e-6
num-train-epochs	3.0
lr-scheduler-type	cosine
warmup-steps	0.1
bf16	true

Table 6: Hyper-parameter settings.

B Prompt for Legal Judgement Prediction

In alignment with the Qwen1.5-14B-Chat input format, we adhere to a structure comprising two message roles: "system" and "user." The "system" role is designated for task descriptions, while the "user" role is intended for text input. Additionally, we selected the sample with the most similar fact description to the given factual description as the input demonstrations. The specific prompt demonstration for Legal Judgment Prediction are shown in Table 11.

C Detailed Explanations for Two Strategies

Random. We randomly select one label from the predefined candidate label set six times. The label that appears most frequently is chosen as the final predicted label. To ensure robustness, we repeat this process six times and take the average of the predicted results as our final result.

Qwen1.5-14B. We use Qwen1.5-14B-Chat for our experiments. We create specific prompts for each

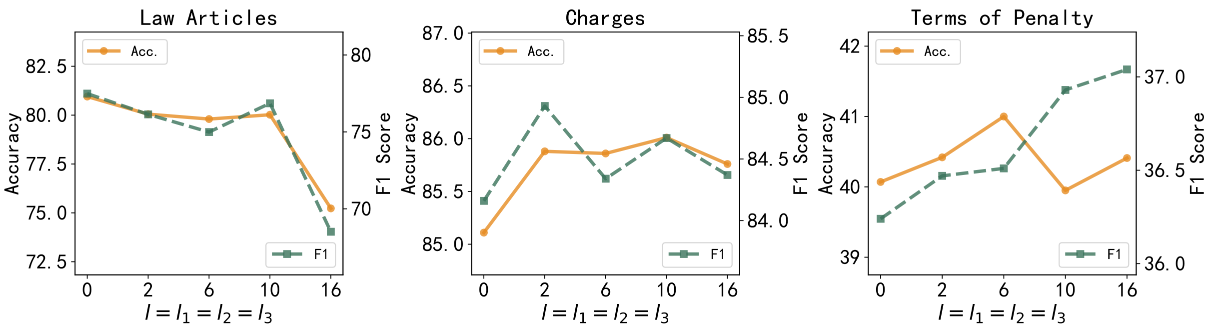

Figure 3: Impact of the number of pseudo tokens in the continuous templates for three subtasks.

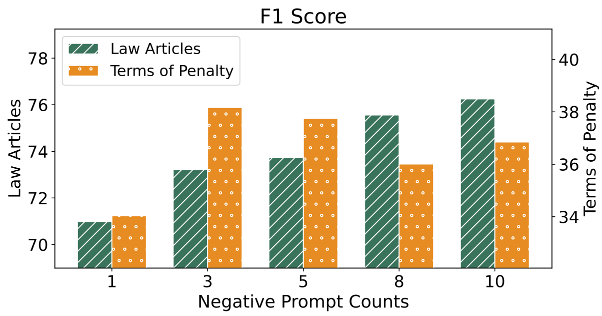

Figure 4: Impact of number of negative prompts for model training in the discrete templates for the prediction of law articles and terms of penalty.

sub-task, including predicting laws, charges, and penalty terms, to help the model choose the correct labels from the candidate set. For brevity, we only show the prompt example for charge prediction here, but the prompts for the other sub-tasks follow a similar pattern. The exact prompt examples are shown in Table 8.

D Results with Fewer Training Data under the Discrete Prompt Template

From Table 7, it can be observed that the trend of results under the discrete template aligns with the analysis findings under the continuous template.

Tasks	Law Articles		Charges		Terms of Penalty	
Metrics	Acc.	F1	Acc.	F1	Acc.	F1
			Best Baseline			
EPM	84.65	78.10	84.10	81.43	36.69	32.99
			Discrete Template			
Train-10%	71.35	62.67	75.39	73.16	27.33	23.38
Train-30%	77.93 (+6.58)	73.05 (+10.38)	79.08 (+3.69)	79.56 (+6.40)	27.38 (+0.05)	28.49 (+5.11)
Train-50%	78.74 (+0.81)	73.68 (+0.63)	83.64 (+4.56)	**82.49** (+2.93)	**40.43** (+13.05)	**35.46** (+7.08)
Train-70%	78.13 (-0.61)	74.74 (+1.06)	**84.30** (+0.66)	**83.18** (+0.69)	34.61 (-5.82)	**33.01** (-2.45)
Train-90%	79.85 (+1.72)	75.95 (+1.21)	85.13 (+0.83)	84.06 (+0.88)	40.71 (+6.10)	37.00 (+3.99)
Train-Full	79.24 (-0.61)	76.25 (+0.30)	84.66 (-0.47)	83.68 (-0.38)	40.44 (-0.27)	37.75 (+0.75)

Table 7: Experimental results with fewer training data under the discrete prompt template. The datas in **bold** indicates that our method outperformed the best baseline model EPM when using less than 70% of the training data.

Role	**Message**
system	You will participate in a legal judgment prediction task. Given a set of case facts, you are required to select the correct charge from a given list of candidate charges (only one charge). Please apply legal knowledge and logical reasoning based on the provided case facts. Output format requirement: Output the charge you select, without any other analysis or explanation.
user	fact description: *<fact>* candidate charge set: *<candidate>*

Table 8: Demonstration of prompt for selecting the final label from candidate label set: Here, "<fact>" stands for the factual statement of a legal case, while "<candidate>" represents the candidate charge set.

Role	**Message**
system	You will participate in a legal judgment prediction task. Given a description of the facts of a case, you need to predict the applicable law, the charge, and the possible sentence. Each charge and applicable law should be singular. Based on the provided case facts, use your legal knowledge and logical reasoning to make predictions. Output format requirements: output the predicted charge, law, and sentence separated by commas, without including any additional analysis or explanation. For example: theft,264,up to three years imprisonment.
user	**Example 1** input: fact description: *<fact 1>* output: *[charge 1],[law 1],[penalty 1]* **Example 2** input: fact description: *<fact 2>* output: *[charge 2],[law 2],[penalty 2]* **...** fact description: *<fact>* output:

Table 9: Demonstration of prompt for LJP: Here, "<fact>" represents the factual statement of a legal case, while "[charge]", "[law]", and "[penalty]" denote the charge, law, and penalty term, respectively.

Role	Message
system	You will participate in a legal judgment prediction task. Given a set of case facts, you need to predict the applicable laws, charges, and possible terms of penalty. Please use your legal knowledge and logical reasoning to make predictions.
user	**Fact description**: On the evening of October 4, 2012, the defendant, Luo Moujia, during a dinner with Luo Mouyi, Yu Mou, and others, learned that Yu Mou had a conflict with Xiang Moujia. The defendant, Luo Moujia, who was familiar with Xiang Moujia, called Xiang Moujia to mediate the conflict between Yu Mou and Xiang Moujia. Later, Luo Moujia and Xiang Moujia had an argument over the phone, and Luo Moujia said he would go to Xiang Moujia's residence to "talk about things." At about 10 p.m. that night, Luo Moujia, along with Luo Mouyi and Yu Mou, drove to the vicinity of Xiang Moujia's residence and called Xiang Moujia to come out...
assistant	**Law**: Article 234 of the Criminal Law of the People's Republic of China **Charge**: Intentional Injury **Penalty terms**: Imprisonment for less than six months

Table 10: The JSON format for fine-tuning, labeled as "instruction-input-output," assigns roles as follows: "system" describes the LJP task, "user" pertains to data input, and "assistant" denotes model output.

Role	Message
system	You will participate in a legal judgment prediction task. Given a description of the facts of a case, you need to predict the applicable law, the charge, and the possible sentence. Each charge and applicable law should be singular. Based on the provided case facts, use your legal knowledge and logical reasoning to make predictions. Output format requirements: output the predicted charge, law, and sentence separated by commas, without including any additional analysis or explanation. For example: theft,264,up to three years imprisonment.
user	**Example 1** input: fact description: <*fact 1*> output: *[charge 1],[law 1],[penalty 1]* **Example 2** input: fact description: <*fact 2*> output: *[charge 2],[law 2],[penalty 2]* **...** fact description: <*fact*> output:

Table 11: Demonstration of prompt for LJP: Here, "<fact>" represents the factual statement of a legal case, while "[charge]", "[law]", and "[penalty]" denote the charge, law, and penalty term, respectively.