Generating Complete Court Judgments with Large Language Models: A Study on CMML

Zhongyi Guan[1], Zhitian Hou[1], Nanli Zeng[2], Kun Zeng[1]*
[1]School of Computer Science and Engineering, Sun Yat-sen University
[2]China Mobile Internet Co., Ltd.
guanzhy33@mail2.sysu.edu.cn
zengkun2@mail.sysu.edu.cn

Abstract

Given the criminal fact, a complete legal judgment usually contains two parts: Court View Generation (CVG) for the case and Legal Judgement Prediction (LJP) for each defendant. However, existing studies are still unable to support unified modeling and evaluation of complete legal judgments. First, at the data level, existing legal datasets mostly cover only single-defendant scenarios or multi-defendant scenarios, and the set of attributes predicted by LJP is not rich enough, which makes it hard for models to generalize to real-world complete legal judgments. Second, CVG and LJP are usually modeled independently, which ignores their intrinsic connection. Third, the role of auxiliary articles has also not been considered. To address this, we build CMML (**C**hinese **M**ixed-Defendant **M**ulti-Attribute **L**egal Judgment Dataset), a large-scale mixed single- and multi-defendant dataset with multi-attribute annotations. Based on CMML, we propose a new task, LJP-x, which includes more legal prediction attributes. We further systematically study the interaction between CVG and LJP-x, and explore how auxiliary articles influence both tasks. Our dataset is available at https://github.com/magicalpig2003/CMML.

1 Introduction

In recent years, large language models (Brown et al., 2020; Ouyang et al., 2022) have developed rapidly, and their applications in the legal domain have become increasingly widespread (Yue et al., 2023; Fei et al., 2025; He et al., 2023; Hou et al., 2025b). Two important applications are CVG and LJP, which constitute two inseparable parts of a complete legal judgment. CVG aims to draft the court view based on the ascertained criminal facts, so as to clarify the legal analysis and the reasons for conviction. LJP aims to generate the final legal judgment for each defendant, which usually involves multiple attributes such as charges, articles, term of penalty, fines, and probation. Figure 1 illustrates the process by which a judge produces a complete legal judgment in real-world judicial practice. Existing legal LLM work on CVG either decomposes criminal facts into a set of intermediate results to better guide LLM generation (Xu et al., 2024; Yue et al., 2024), or improves CVG performance through domain knowledge injection (Liu et al., 2024; Li et al., 2024). For LJP, some studies use chain-of-thought techniques to enhance interpretability and robustness (Zhang et al., 2024; Jiang and Yang, 2023), while others adopt methods such as RAG to improve prediction performance (Peng and Chen, 2024; Yang et al., 2026; Wu et al., 2023).

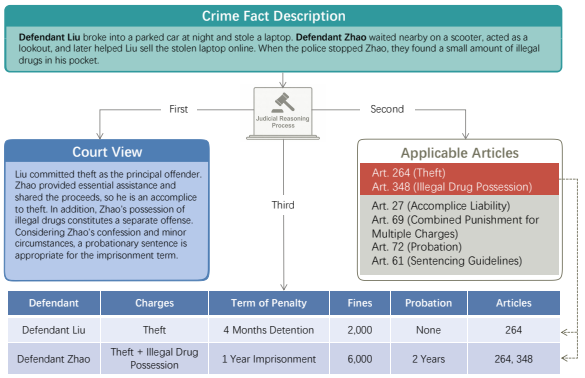


Figure 1: An illustration of a complete legal-judgment generation pipeline. Given a crime fact description, the judge is required to draft the court view, determine the applicable articles, and render a judgment for each defendant.

However, the generation of complete legal judgments remains insufficiently addressed in prior work. First, at the data level, existing legal datasets are largely confined to single-defendant-only (Xiao et al., 2018) or multi-defendant-only (Huang et al., 2024; Lyu et al., 2023) scenarios, whereas real judicial practice frequently involves the coexistence

*Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 22990–23001
October 24-29, 2026 ©2026 Association for Computational Linguistics

of single- and multi-defendant cases. This mismatch undermines the generalization of models for complete judgment generation. Meanwhile, most datasets lack sufficient attribute coverage to support the generation of complete legal judgments. In addition, the charge expressions and article annotations in existing datasets are often misaligned with the latest revisions of the legal system, which further limits their practical utility. Second, CVG and LJP are commonly modeled as two independent tasks, and their interdependence has not been systematically investigated. Furthermore, auxiliary articles constitute an indispensable component of complete legal judgments, yet prior studies have rarely examined their generation or how they influence LJP and CVG.

To address the above issues, the contributions of this paper are as follows:

First, we construct CMML (Chinese Mixed-Defendant Multi-Attribute Legal Judgment Dataset), a multi-attribute dataset tailored to the Chinese legal system that covers both single- and multi-defendant cases. CMML annotates diverse judgment attributes, including fines, probation, and auxiliary articles, and extends the term-of-penalty attributes to cover detention and surveillance. Moreover, it refines article annotations to the paragraph level. In addition, the charge and article annotations in CMML are aligned with the latest legal system. Experiments demonstrate the necessity of a mixed dataset, showing that models trained on mixed single- and multi-defendant data exhibit stronger generalization than those trained on single-defendant-only or multi-defendant-only data.

Second, based on CMML, we propose the extended task LJP-x and systematically study the interaction between LJP-x and CVG. On top of the traditional LJP prediction targets of charges, articles, and term of penalty, LJP-x adds fines and probation, refines article prediction to the paragraph level, and covers surveillance and detention in term-of-penalty prediction. We compare three settings: CVG-only, LJP-x-only, and joint modeling of the two tasks, and further examine the effects of different generation orders. The results show that both the generation order between CVG and LJP-x and whether joint modeling is adopted significantly affect the performance of the two tasks.

Third, based on CMML, we explore the impact of the general provisions in the Criminal Law in the auxiliary articles on CVG and LJP-x. The results show that using them as an auxiliary prediction task can improve the sentencing related sub-tasks in LJP-x as a whole after fine-tuning, such as term of penalty, fines, and probation. However, the improvement of CVG is limited, indicating that auxiliary articles have a more direct effect on the prediction of judgment results.

2 Related Work

2.1 Legal Dataset

LJP datasets: CAIL2018 (Xiao et al., 2018), the first large-scale LJP dataset for single-defendant, single-charge cases, provides criminal facts and basic targets such as charges, articles, and term of penalty. Its term-of-penalty categories are limited to fixed-term imprisonment, life imprisonment and death penalty. Dep-LJP (Huang et al., 2021) extends the setting to single-defendant multiple charges. Multi-LJP (Lyu et al., 2023) is the first multi-defendant legal judgment dataset. CMDL (Huang et al., 2024) constructs the first large-scale dataset dedicated to multi-defendant cases. It augments traditional LJP with fine and auxiliary article annotations, refines articles to the paragraph level, expands term-of-penalty categories to include detention and surveillance, and additionally provides the court view section.

CVG datasets: Existing CVG datasets, such as C3VG (Yue et al., 2021), LCVG (Xu et al., 2024), and CCVG (Hou and Zeng, 2025), mainly contain court views and charge annotations.

2.2 LJP and CVG

For LJP, existing work mainly relies on deep learning methods to separately investigate the relationships among LJP sub-tasks (Zhong et al., 2018; Yang et al., 2019) and to address practical issues such as confusable charges and articles (Xu et al., 2020; Hu et al., 2018; Lyu et al., 2022). With the development of large language models, unified text-to-text frameworks and techniques such as retrieval-augmented generation are used to unify the modeling of charge, article, and term-of-penalty prediction, significantly improving the reasoning and prediction ability of LJP with no or only light fine-tuning (Huang et al., 2021; Peng and Chen, 2024; Liu et al., 2024). More recently, knowledge-driven frameworks further incorporate case-specific keyword extraction and precedent-based legal knowledge retrieval to guide LLMs toward more structured legal judgment prediction without additional

Dataset	Defendant		Court View	Charges	Law Articles		Term of Penalty			Fines	Probation	Auxiliary Articles	
	single	multi			A-level	P-level	tra	sur	det			criminal	other
C3VG	62939	✗	✓	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
LCVG	101291	✗	✓	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
CCVG	110k	✗	✓	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
CAIL2018	2676075	✗	✗	202	183	✗	✓	✗	✗	✓	✗	✗	✗
Dep-LJP*	25444	✗	✓	200	183	✗	✓	✗	✗	✗	✗	✗	✗
Multi-LJP	✗	23717	✓	23	22	✗	✓	✗	✓	✗	✗	✗	✗
CMDL	✗	393945	✓	321	275	✓	✓	✓	✓	✓	✗	✓	✗
CMML	133050	33600	✓	257	221	✓	✓	✓	✓	✓	✓	✓	✓

Table 1: Statistics and Task Coverage of Representative Legal Datasets. A-level denotes article-level and P-level denotes paragraph-level. For term of penalty, tra refers to traditional LJP terms (fixed-term imprisonment, life imprisonment, and death penalty), sur refers to surveillance, and det refers to detention. * indicates that the dataset name is taken from the paper where it was introduced (not an official dataset name provided by the dataset authors). For Dep-LJP*, we only obtained its test and valid splits.

fine-tuning (Hou et al., 2025a) .

For CVG, research has evolved from LSTM- and attention-based generation models to frameworks that integrate pre-trained language models, so as to better capture conviction and sentencing reasoning (Ye et al., 2018; Yue et al., 2021). More recent work further leverages large language models for event extraction and court view generation, or fine-tunes general-purpose models on self-constructed data, achieving significant performance gains on CVG (Xu et al., 2024; Yue et al., 2024; Hou and Zeng, 2025).

However, prior work focuses only on single-defendant-only or multi-defendant-only cases, covers too few attributes, and models CVG and LJP separately, which fails to meet the needs of complete legal judgment generation. In this paper, we propose a more comprehensive dataset, CMML, and study complete legal judgment generation based on it. As shown in Table 1, CMML offers the most comprehensive attribute coverage while preserving substantial data scale and broad coverage of charges and articles.

3 CMML

In this section, we describe how we construct CMML and how we split the dataset to support the subsequent experiments. Additional details on dataset construction and maintenance are also provided in Appendix A. More analyses of our data can be found in Appendix B.

3.1 Data Construction

Figure 2 illustrates the overall pipeline for constructing our dataset. We retrieved and downloaded approximately 300,000 criminal judgments from https://www.pkulaw.com spanning 2018–2025, and performed structured extraction of four components: fact, court view, applicable articles, and the result of the judgment. The applicable articles are categorized into two types: (i) conviction-basis articles that directly correspond to specific charges (special provisions of the Criminal Law from Article 102 onward) and (ii) auxiliary articles, including the general provisions of the Criminal Law (Articles 1–101) as well as relevant statutes and regulations beyond the Criminal Law. From the result of the judgment, we extract each defendant's charges, term of penalty, fines, and probation, where the term of penalty covers detention, surveillance, fixed-term imprisonment, death penalty, and life imprisonment. For defendants with multiple charges, we use the final summary in the verdicts as the annotations for the term of penalty and fines.

For sentencing, we label death penalty and life imprisonment as boolean variables, and convert fixed-term imprisonment, detention, surveillance, and probation into a unified month-based scale. During preprocessing, we anonymize defendants' names and explicit charge mentions in the fact section.

To align with current legal norms, following CMDL, we construct a "charge–article" mapping dictionary based on the 2025 revision of the Criminal Law, and use n-gram similarity to align raw charge expressions with standardized charge labels. Subsequently, after subjecting the matched standardized charges to manual review and programmatic correction, we consult the dictionary again to obtain paragraph-level article annotations.

3.2 Data Division

Table 2 presents the basic statistical information of our constructed dataset. To compare the dif-

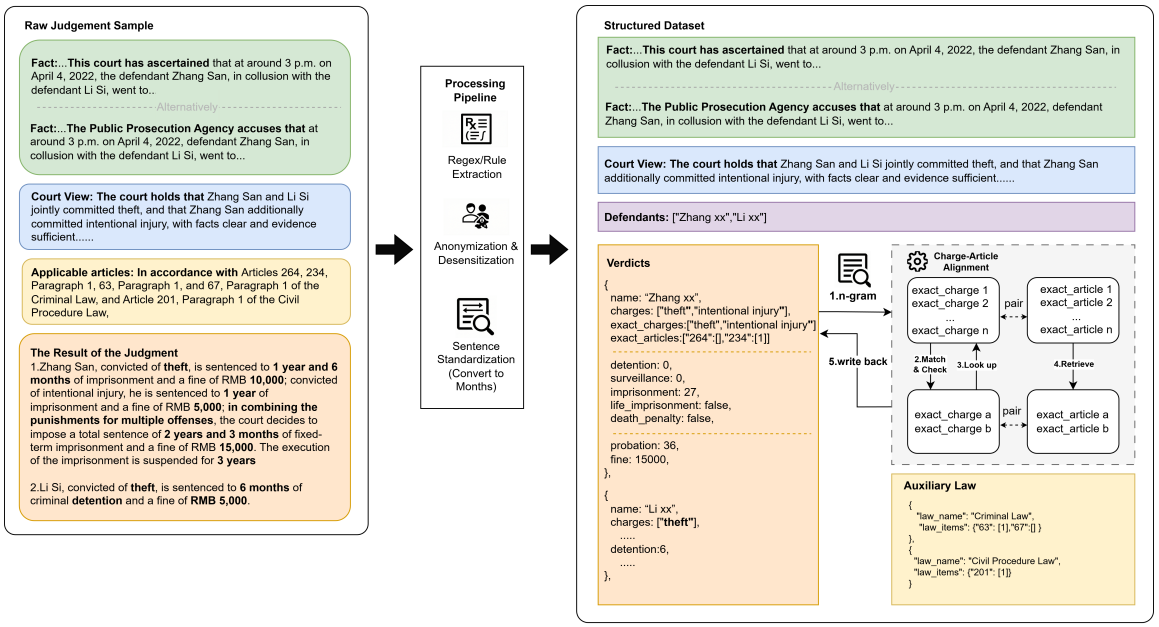


Figure 2: Data Construction Pipeline

ferences and advantages of single-defendant-only, multi-defendant-only, and mixed single- and multi-defendant data in model training, we construct four subsets from the full dataset: a single-defendant-only training set, a multi-defendant-only training set, a mixed single- and multi-defendant training set, and an independent test set. To keep the training scale consistent, we set the number of samples in each training set to 29,600. Specifically, the mixed single- and multi-defendant training set consists of 14,800 single-defendant samples and 14,800 multi-defendant samples; the single-defendant-only training set contains the 14,800 single-defendant samples used in the mixed set plus an additional 14,800 single-defendant samples; the multi-defendant-only training set is constructed in the same way, consisting of the 14,800 multi-defendant samples from the mixed set together with another 14,800 multi-defendant samples.

Dataset	CMML-test	CMML-train			CMML
		single	multi	mixed	
# Single-defendant cases	4000	29600	0	14800	133050
# Multi-defendant cases	4000	0	29600	14800	33600
# Total	8000	29600	29600	29600	166650
# Charges	216	239	233	242	257
# Law articles (article-level)	183	199	196	204	221
# Law articles (paragraph-level)	411	453	463	489	632
Minimum Court View Length	129	127	130	129	118
Maximum Court View Length	3987	3744	4602	4602	4602
Average Court View Length	408.83	277.11	423.33	368.14	298.42
Max Charges per Defendant	6	5	5	5	6
Average Charges per Defendant	1.30	1.25	1.07	1.12	1.10
Max Charges per Case	7	5	6	6	7
Average Charges per Case	1.65	1.25	1.23	1.39	1.14

Table 2: Statistics of CMML-test, CMML-train subsets, and CMML.

4 Experiments

4.1 Experimental Setup

Input: The model input consists of the criminal *fact* and the defendant set $Defendants = \{d_1, \ldots, d_k\}$, where d_j denotes the name of the j-th defendant.

Tasks and Outputs: We jointly consider two tasks in a single model, namely CVG and LJP-x.

- **CVG** aims to generate the *court_view* text conditioned on *fact*.
- **LJP-x** aims to predict structured verdict attributes for each defendant, including *charges*, *articles*, *term_of_penalty*, *fine*, and *probation*.

For LJP-x, we denote the structured verdicts of all defendants as $Verdicts = \{v_1, \ldots, v_k\}$. For the j-th defendant, the predicted verdict is

$$v_j = \{\text{charges}, \text{articles}, \text{term of penalty}, \text{fines}, \text{probation}\},$$

where $\text{charges} = \{c_1, c_2, \ldots\}$ and c_i is the i-th predicted charge for this defendant, and $\text{articles} = \{a_1, a_2, \ldots\}$ and a_i is the i-th predicted article applicable to this defendant. In addition, we predict a case-level set of auxiliary articles $\text{Aux_law} = \{al_1, al_2, \ldots\}$, where al_i is the i-th auxiliary article applicable to the case.

To systematically analyze the relationship between CVG and LJP-x and the influence of

Dataset		CVG			Charges		Law Articles				Term of Penalty		Fines	Probation
		ROUGE	BLEU	LLMS	Acc.	F1	A-Level		P-Level		Acc.	F1	Dis↓	Dis↓
							Acc.	F1	Acc.	F1				
test(single)	train(multi)	53.88	54.90	2.657	73.74	84.08	76.58	85.86	53.16	68.36	43.05	33.07	5.398	0.858
	train(mixed)	**54.41**	**56.05**	**2.725**	**75.28**	**85.72**	**78.06**	**87.11**	**54.63**	**70.13**	**45.46**	**37.12**	**5.354**	**0.781**
test(multi)	train(single)	50.77	51.62	1.868	60.13	72.38	64.89	75.08	44.26	58.15	30.03	22.62	5.886	**0.963**
	train(mixed)	**51.22**	**52.45**	**1.927**	**66.73**	**75.31**	**71.33**	**78.45**	**50.27**	**61.54**	**34.62**	**29.10**	**5.773**	1.117
test(single)	train(single)	54.13	56.08	2.710	75.23	85.88	78.04	86.76	54.43	70.17	46.09	39.36	5.421	0.785
	train(multi)	53.88	54.90	2.657	73.74	84.08	76.58	85.86	53.16	68.36	43.05	33.07	5.398	0.858
		(**0.25**↓)	(1.18↓)	(**0.053**↓)	(**1.49**↓)	(**1.8**↓)	(**1.46**↓)	(**0.9**↓)	(**1.27**↓)	(**1.81**↓)	(**3.04**↓)	(**6.29**↓)	(**0.023**↓)	(0.073↑)
test(multi)	train(multi)	51.44	52.20	1.980	69.16	77.10	73.36	79.81	51.84	63.02	35.34	30.47	5.682	1.125
	train(single)	50.77	51.62	1.868	60.13	72.38	64.89	75.08	44.26	58.15	30.03	22.62	5.886	0.963
		(0.67↓)	(**0.58**↓)	(0.112↓)	(9.03↓)	(4.72↓)	(8.47↓)	(4.73↓)	(7.58↓)	(4.87↓)	(5.31↓)	(7.85↓)	(0.204↑)	(**0.162**↓)

Table 3: Test results of models fine-tuned on the single-defendant-only dataset, the multi-defendant-only dataset, and the mixed single- and multi-defendant dataset, respectively. In the last two blocks, the value in parentheses is the difference between the upper and lower rows (upper−lower). Charges, law articles, term of penalty, fines, and probation are the evaluation attributes for LJP-x.

Aux_law, we vary the generation order of CVG, LJP-x, and Aux_law. We use the notation `A_B_C` to denote producing A, B, and C in the order $A \rightarrow B \rightarrow C$. For example, `Aux-law_LJP-x_CVG` means predicting Aux-law first, then predicting LJP-x, and finally generating CVG. Single-task settings are denoted as `only_A`, such as `only_CVG`.

Metrics: The evaluation metrics are provided in Appendix C. LLMS in CVG denotes evaluation scores assigned by a large language model. For metrics specific to multi-defendant cases, we follow the protocol adopted in CMDL.

Model and Parameters: Both the base model and our fine-tuned model are Qwen3-8B. We fine-tune the model using the `ms-swift` framework with LoRA-based SFT. We set the maximum input length to 10,240 and the maximum number of generated tokens (`max_new_tokens`) to 2,048. For LoRA, we use `lora_rank`=8 and `lora_alpha`=32, and set `warmup_ratio`=0.05. We use fixed hyperparameters throughout all experiments.

4.2 Demonstrating the Advantage of Mixed Single- and Multi-defendant Data

In this section, we demonstrate through experiments the advantage of the mixed single- and multi-defendant dataset in terms of generalization ability for generating complete legal judgments. Following the typical structure of legal judgments, we adopt the CVG_LJP-x generation order in this subsection. We evaluate from two perspectives: (1) on the multi-defendant test set, whether the model fine-tuned on the mixed single- and multi-defendant data outperforms the model fine-tuned on the single-defendant-only data; (2) on the single-defendant test set, whether the model fine-tuned on the mixed single- and multi-defendant training data outperforms the model fine-tuned on the multi-defendant-only training data. Table 3 presents the test results on CVG and LJP-x of models fine-tuned on different datasets.

From the overall results, across both tasks CVG and LJP-x and under both Perspective 1 and Perspective 2 (Groups 1 and 2 in Table 3), the model trained on the mixed single- and multi-defendant data consistently achieves the best or near-best overall performance, which indicates that covering both types of case structures during training helps the model learn more generalizable representations.

Further, we analyze zero-shot transfer performance (Groups 3 and 4 in Table 3). For LJP-x, the performance drop under zero-shot transfer from multi-defendant training to single-defendant testing is relatively small, whereas the drop from single-defendant training to multi-defendant testing is more pronounced. This trend holds across most sub-tasks, suggesting that the structural complexity introduced by multi-defendant data benefits cross-scenario generalization. In contrast, for CVG, the two zero-shot transfer directions behave differently across metrics: under zero-shot transfer from multi-defendant to single-defendant, the drops in ROUGE and LLMS are smaller, indicating stronger content coverage; under zero-shot transfer from single-defendant to multi-defendant, the drop in BLEU is smaller, indicating an advantage in local wording and exact n-gram matching.

Dataset		CVG			Charges		Law Articles				Term of Penalty		Fines	Probation
		ROUGE	BLEU	LLMS	Acc.	F1	A-Level		P-Level		Acc.	F1	Dis↓	Dis↓
							Acc.	F1	Acc.	F1				
Fine-tuned models														
test(single)	CVG_LJP-x	54.41	**56.05**	2.725	75.28	85.72	78.06	87.11	54.63	70.13	45.46	37.12	5.354	0.781
	LJP-x_CVG	**54.57**	55.97	**2.830**	77.95	87.21	80.66	88.67	55.98	71.75	45.59	37.79	5.400	0.594
	only_CVG	53.58	55.32	2.664	–	–	–	–	–	–	–	–	–	–
	only_LJP-x	–	–	–	**81.43**	**89.69**	**83.43**	**90.53**	**58.10**	**73.06**	**48.04**	**40.75**	**5.27**	**0.563**
test(multi)	CVG_LJP-x	51.22	**52.45**	1.927	66.73	75.31	71.33	78.45	50.27	61.54	34.62	29.10	5.773	1.117
	LJP-x_CVG	**51.56**	52.24	**2.065**	73.08	79.87	76.95	82.64	53.60	64.60	34.54	29.80	5.663	**0.850**
	only_CVG	50.52	51.99	1.908	–	–	–	–	–	–	–	–	–	–
	only_LJP-x	–	–	–	**77.02**	**83.32**	**80.49**	**85.65**	**55.87**	**66.75**	**37.55**	**32.60**	**5.621**	0.861
Base models														
test(single)	CVG_LJP-x	**43.01**	34.63	2.586	**59.75**	**75.83**	**60.53**	**73.04**	**28.50**	**41.57**	**17.31**	**11.10**	7.071	2.092
	LJP-x_CVG	40.52	37.77	2.032	55.60	71.92	59.48	72.28	26.53	38.73	16.49	10.89	6.840	1.855
	only_CVG	42.42	**43.88**	**2.648**	–	–	–	–	–	–	–	–	–	–
	only_LJP-x	–	–	–	56.71	72.63	54.05	67.87	25.71	38.03	14.71	9.58	**6.741**	**1.676**
test(multi)	CVG_LJP-x	39.59	25.49	1.983	**42.30**	**59.39**	**48.61**	**58.45**	**22.34**	**28.90**	**14.83**	**9.99**	7.459	2.125
	LJP-x_CVG	39.16	29.18	1.721	37.35	54.75	43.74	56.29	20.18	26.97	14.43	9.40	**7.267**	2.011
	only_CVG	**40.61**	**34.03**	**2.051**	–	–	–	–	–	–	–	–	–	–
	only_LJP-x	–	–	–	38.55	55.02	40.44	51.69	20.16	26.42	12.53	8.89	7.292	**1.929**

Table 4: Test results of the model under different CVG and LJP-x generation settings. Missing entries are denoted by "–".

4.3 Exploring the Relationship between CVG and LJP-x

Based on the empirical results from the previous experiment, we adopt the mixed single- and multi-defendant dataset as the fine-tuning dataset for this and subsequent sections. We consider four generation schemes: `only_LJP-x` , `only_CVG`, `CVG_LJP-x`, and `LJP-x_CVG`. The test results are shown in Table 4.

From the perspective of CVG, (1) for the fine-tuned model, `only_CVG` is overall inferior to the settings that jointly incorporate LJP-x. This can be viewed as positive transfer from multi-task learning: the charges and sentencing factors appearing in the court view largely overlap with the prediction targets of LJP-x, and joint training provides stronger structured supervision, thereby improving the accuracy and completeness of court view generation. In addition, LJP-x_CVG achieves higher LLMS scores than CVG_LJP-x, indicating that the more accurate LJP-x predictions enabled by fine-tuning can positively transfer to CVG. (2) For the base model, `only_CVG` usually outperforms `LJP-x_CVG`. This is because the base model is not specifically trained for LJP-x, so the LJP-x outputs generated first are of low quality. These structured outputs are then used as conditional inputs and reused in the court view, and the accumulated errors consequently lead to a substantial degradation in CVG performance. (3) Interestingly, on the multi-defendant test set, the fine-tuned model achieves clearly better ROUGE and BLEU scores, but slightly lower LLMS scores than the base model in CVG_LJP-x and only_CVG. This suggests that fine-tuning may encourage the model to better match reference-style expressions, while not always improving the overall semantic quality preferred by the LLM-based judge.

From the perspective of LJP-x, (1) for the fine-tuned model, the `only_LJP-x` configuration overall outperforms the joint LJP-x+CVG setting. This is because LJP-x has a more structured output space and clearer supervision. When the model is required to simultaneously optimize the more difficult CVG task, its capacity is more easily allocated to fitting long-text generation, which in turn induces negative transfer on LJP-x. Within the joint setting, `LJP-x_CVG` usually performs better than `CVG_LJP-x`, indicating that placing CVG first may introduce inexact reasoning signals that interfere with the subsequent prediction of charges, articles, and sentencing-related attributes. (2) For the base model, on traditional LJP attributes (charges, articles, and term of penalty), `CVG_LJP-x` often outperforms `only_LJP-x` and `LJP-x_CVG`, suggesting that CVG can act as a "chain-of-thought / evidence aggregation" step and bring positive transfer. However, for fines and probation, `only_LJP-x` or placing LJP-x first tends to perform better. We conjecture that this is because these two attributes rely more on direct supervision, whereas the corresponding expressions in CVG are noisier. Using CVG as an intermediate step may therefore introduce bias and weaken the modeling of continuous sentencing signals.

Dataset		CVG			Charges		Law Articles				Term of Penalty		Fines	Probation	Auxiliary Articles			
		ROUGE	BLEU	LLMS	Acc.	F1	A-Level		P-Level		Acc.	F1	Dis↓	Dis↓	A-Level		P-Level	
							Acc.	F1	Acc.	F1					Acc.	F1	Acc.	F1
Fine-tuned models																		
test(single)	LJP-x_CVG	54.57	55.97	2.830	77.95	87.21	80.66	88.67	55.98	**71.75**	45.59	37.79	5.40022	0.59479	–	–	–	–
	Aux-Law_LJP-x_CVG	54.76	56.02	2.888	77.06	86.84	79.91	88.38	54.95	71.21	**46.54**	36.90	5.36106	0.66530	13.48	67.18	6.97	54.31
	LJP-x_Aux-Law_CVG	**55.04**	56.10	**2.933**	**78.53**	**87.70**	**80.84**	**88.84**	55.47	71.51	45.29	35.89	**5.29258**	**0.57688**	**14.68**	**68.34**	7.85	**55.75**
	LJP-x_CVG_Aux-Law	54.75	**56.15**	2.832	77.85	87.10	80.63	88.56	**56.09**	71.55	46.50	**38.43**	5.30276	0.59173	14.56	67.05	**7.95**	53.41
test(multi)	LJP-x_CVG	51.56	52.24	2.065	**73.08**	**79.87**	**76.95**	82.64	53.60	64.60	34.54	**29.80**	**5.66374**	**0.85068**	–	–	–	–
	Aux-Law_LJP-x_CVG	51.80	**52.68**	2.104	71.87	79.41	76.08	82.33	53.62	64.89	**35.40**	29.27	5.69570	0.98033	**5.61**	68.21	2.38	53.90
	LJP-x_Aux-Law_CVG	**51.93**	52.43	**2.123**	72.14	79.33	76.37	82.30	**54.81**	**65.38**	35.16	29.61	5.68197	0.87910	5.36	**68.94**	2.45	**54.91**
	LJP-x_CVG_Aux-Law	51.64	52.38	2.088	72.60	79.75	76.88	**82.67**	53.94	64.98	34.75	28.53	5.68755	0.86485	5.52	67.18	**2.68**	53.95
Base models																		
test(single)	LJP-x_CVG	40.52	37.77	2.032	55.60	71.92	**59.48**	**72.28**	**26.53**	**38.73**	16.49	10.89	6.84092	**1.85551**	–	–	–	–
	Aux-Law_LJP-x_CVG	42.33	38.87	2.138	**59.07**	**73.55**	42.66	53.01	19.82	29.24	**16.76**	**11.02**	**6.79877**	2.02690	0.26	**18.37**	0.24	**5.91**
	LJP-x_Aux-Law_CVG	41.80	**40.83**	2.137	57.11	72.88	45.34	61.23	22.95	34.55	16.08	10.36	7.00749	1.93982	**0.62**	1.00	**0.62**	0.69
	LJP-X_CVG_Aux-Law	**42.86**	37.46	**2.246**	57.32	72.87	45.44	61.43	22.83	34.26	16.05	10.58	6.95452	1.96993	0.02	0.91	0.02	0.29
test(multi)	LJP-x_CVG	39.16	29.18	1.721	37.35	54.75	**43.74**	**56.29**	**20.18**	**26.97**	**14.43**	**9.40**	**7.26782**	**2.01131**	–	–	–	–
	Aux-Law_LJP-x_CVG	40.61	28.64	1.732	**42.67**	**56.52**	35.45	42.32	14.90	19.50	14.11	8.62	7.29095	2.10782	**0.00**	**17.75**	**0.00**	**5.18**
	LJP-x_Aux-Law_CVG	39.31	**30.22**	1.761	39.97	56.30	36.90	47.37	17.82	23.69	13.70	9.05	7.36277	2.02657	0.00	0.44	0.00	0.13
	LJP-X_CVG_Aux-Law	**40.87**	27.89	**1.843**	39.72	56.21	36.79	47.40	17.44	23.59	13.68	9.07	7.32324	2.03486	0.00	1.22	0.00	0.33

Table 5: The test results of the model under different generation orders of Aux-law. Aux-law denotes the general-provision articles in the Criminal Law, namely Articles 1–101, within the auxiliary articles.

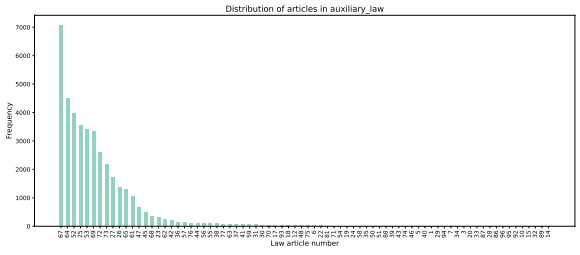


Figure 3: Distribution of auxiliary articles

4.4 Exploring the Effect of Auxiliary Articles

Based on the fine-tuning results in the previous section, `LJP-x_CVG` achieves better overall performance, so we adopt this order by default in this section. We focus on predicting the Criminal Law general-provision articles within the auxiliary-article annotations, which refer to Articles 1–101 of the Criminal Law, while leaving auxiliary article entries from laws and regulations other than the Criminal Law for future work. Figure 3 shows that the General Provisions clauses follow a long-tail distribution, and the high-frequency clauses are mostly related to sentencing circumstances such as voluntary surrender, meritorious service, fine, and probation. Therefore, we focus on their effects on three sentencing subtasks in LJP-x, namely `term of penalty`, `fine`, and `probation`. The test results are shown in Table 5.

From the perspective of CVG, (1) for the fine-tuned model, introducing Aux-law leads to only marginal changes in ROUGE and BLEU, but yields clearer gains in LLMS when Aux-law is placed before CVG, especially when it is positioned immediately before CVG. This suggests that placing Aux-law closer to CVG provides more useful legal-grounding signals for the overall quality of CVG. (2) For the base model, introducing Aux-law improves CVG more consistently across the three metrics, and placing Aux-law last performs best overall. This suggests that, in the autoregressive setting, treating Aux-law as a supplementary target can enhance legal grounding while reducing interference with the main CVG decoding process.

From the perspective of LJP-x, (1) in the multi-defendant setting, introducing Aux-law prediction leads to overall degradation in fine/probation distance and term-of-penalty-related metrics. This is mainly because the general-provision articles are annotated only at the case level, without fine-grained defendant–article alignment. When such a coarse-grained signal is incorporated into an autoregressive sequence, it injects noise into defendant-level sentencing modeling and causes interference. (2) In the single-defendant setting, for the fine-tuned model, placing Aux-law in the middle introduces a switch in decoding mode at the transition from discretionary decision-making to article matching and extraction. This makes attention and hidden states rely more on cue tokens and template-like patterns, thereby weakening holistic integration of factual factors and degrading term-of-penalty performance. By contrast, placing Aux-law first or last reduces the interference introduced by this switch, yielding more stable results and modest gains. (3) Also in the single-defendant setting, the fine-tuned model is more strongly constrained by conclusion–grounding consistency for fines and probation. Placing Aux-law in the middle forces the model to output the corresponding general-provision grounds immediately after producing discretionary outcomes. This effectively

	Dataset	CVG			Charges		Law Articles				Term of Penalty		Fine	Probation	Auxiliary Articles			
							A-Level		P-Level						A-Level		P-Level	
		ROUGE	BLEU	LLMS	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Dis↓	Dis↓	Acc.	F1	Acc.	F1
test(single)	Qwen3-8B	42.86	37.46	2.192	57.32	72.87	45.44	61.43	22.83	34.26	16.05	10.58	6.95452	1.96993	0.02	0.91	0.02	0.29
	InternLM3-8B	29.16	36.87	1.407	23.42	42.72	13.87	24.14	4.68	11.78	9.90	8.26	7.48672	1.27070	0.18	1.39	0.18	0.65
	GLM-4-9B-Chat	39.42	26.97	1.953	28.44	56.32	37.42	64.37	3.99	24.43	9.96	9.06	8.93355	2.72834	0.27	1.02	0.27	0.61
	Qwen3-32B	42.26	39.39	2.306	50.16	73.30	62.20	79.15	24.60	44.21	21.84	21.15	6.13355	1.36551	0.95	25.92	0.20	11.90
	DeepSeek-V3.2	52.09	49.47	**2.913**	67.55	83.27	80.69	89.84	45.79	62.55	29.54	24.25	6.04599	0.60232	3.78	47.33	1.82	25.31
	Dep-LJP	-	-	-	66.85	81.99	70.27	84.07	52.90	69.97	37.70	30.74	5.58164	0.66779	-	-	-	-
	DISC-LawLLM	35.10	40.88	1.867	58.50	76.88	67.88	80.25	29.25	43.33	11.74	9.70	6.81701	0.63036	-	-	-	-
	Hanfei	42.51	42.42	2.139	33.97	55.80	44.06	66.36	26.18	41.06	-	-	-	-	-	-	-	-
	Qwen3-8B-SFT	**54.75**	**56.15**	2.628	77.85	87.10	80.63	88.56	56.09	71.55	46.50	38.43	**5.30276**	**0.59173**	**14.56**	**67.05**	**7.95**	**53.41**
	GLM-4-9B-Chat-SFT	54.72	55.53	2.718	**81.22**	**89.28**	**83.57**	**90.35**	**59.45**	**73.98**	**47.07**	**39.19**	5.484	0.678	13.70	66.67	7.02	53.23
test(multi)	Qwen3-8B	40.87	27.89	1.953	39.72	56.21	36.79	47.40	17.44	23.59	13.68	9.07	7.32324	2.03486	0.00	1.22	0.00	0.33
	InternLM3-8B	27.46	28.84	1.305	16.71	30.00	12.99	21.58	3.30	7.32	8.92	8.15	8.12162	1.40578	0.00	1.06	0.00	0.38
	GLM-4-9B-Chat	32.88	12.26	1.727	22.97	44.71	28.56	49.07	4.51	17.42	5.87	5.55	9.22271	2.88574	0.02	0.51	0.02	0.25
	Qwen3-32B	39.30	29.40	2.014	35.01	57.90	51.74	69.03	18.86	33.14	14.53	12.58	6.96616	1.24645	0.15	22.59	0.00	11.15
	DeepSeek-V3.2	49.43	42.45	**2.449**	55.41	72.75	**81.47**	**87.29**	40.30	51.19	28.23	23.83	6.34396	**0.81169**	1.50	52.49	0.35	25.61
	Dep-LJP	-	-	-	62.41	72.45	68.61	77.86	50.81	63.31	24.27	20.75	6.14179	1.03479	-	-	-	-
	DISC-LawLLM	32.09	33.13	1.426	41.37	62.06	52.09	68.03	22.09	33.27	10.67	7.31	7.06430	0.84818	-	-	-	-
	Hanfei	40.16	41.94	1.454	28.10	40.16	42.91	53.86	23.09	30.58	-	-	-	-	-	-	-	-
	Qwen3-8B-SFT	51.64	**52.38**	2.111	72.60	79.75	76.88	82.67	53.94	64.98	34.75	28.53	**5.68755**	0.86485	**5.52**	**67.18**	**2.68**	**53.95**
	GLM-4-9B-Chat-SFT	**52.17**	52.31	2.193	**76.49**	**83.22**	79.93	85.41	**57.27**	**68.46**	**35.44**	**30.54**	5.96846	0.98587	4.76	66.63	1.67	52.99

Table 6: The baseline test results. "-" denotes missing metrics for contents that the model cannot generate or cannot generate stably.

imposes a local, sequence-level consistency regularization and stabilizes high-variance discretionary outputs. This is usually better than placing Auxlaw first, which may introduce biased priors, or last, which is largely post-hoc supplementation and provides limited constraints during generation. (4) For general-provision article prediction, the base model rarely produces valid outputs, so we do not analyze its metrics. After targeted fine-tuning, both accuracy and F1 increase substantially, indicating that this auxiliary task benefits from supervision. However, because accuracy uses exact set matching and each case often involves many applicable articles, the model typically predicts only a subset, resulting in relatively low accuracy.

5 Baselines

The baseline experimental setup can be found in Appendix D.

5.1 Baseline Models

(1) Qwen3-8B (2) InternLM3-8B-Instruct (3) GLM-4-9B-Chat (4) Qwen3-32B (5) DeepSeek-V3.2 (6) Dep-LJP (Huang et al., 2021): a SOTA method for LJP (7) DISC-LawLLM (Yue et al., 2023): a Chinese legal-domain LLM for intelligent legal services (8) Hanfei (He et al., 2023): a fully parameterized 7B Chinese legal LLM (9) Qwen3-8B-SFT (10) GLM-4-9B-Chat-SFT

5.2 Result Analysis

From the results in Table 6, we observe that the models fine-tuned on CMML outperform all specialized legal methods or models, and also outperform all general-purpose LLMs except DeepSeek-V3.2. Compared with the strong baseline DeepSeek-V3.2, Qwen3-8B-SFT and GLM-4-9B-Chat-SFT achieve better overall performance, with particularly clear advantages on extended attributes beyond traditional LJP settings, including paragraph-level article prediction, fine prediction, and auxiliary article prediction. For probation prediction, the gap between these models and DeepSeek-V3.2 remains within an acceptable range. These results indicate that the coverage of both single- and multi-defendant cases and the multi-attribute annotations provided by CMML play a critical role in improving model capability, and they further validate the effectiveness and practical potential of the CMML-fine-tuned model for generating complete court judgments.

Our experiments further show that Hanfei struggles with fine-grained term-of-penalty, fine, and probation prediction, while both DISC-LawLLM and Hanfei fail to produce reasonable auxiliary articles. This suggests that current legal LLMs may still lack sufficient training data for these fine-grained judgment attributes.

6 Conclusion

This paper focuses on generating complete legal judgments. We construct the CMML dataset, propose the LJP-x task that extends verdict attributes, and systematically analyze the interactions between CVG and LJP-x, as well as their interplay with auxiliary articles and the corresponding optimal generation order. Experimental results show that, under a unified prompting and generation framework, mixed single- and multi-defendant data together with the reasoning pipeline of first perform-

ing structured LJP-x prediction, then generating CVG, and finally supplementing auxiliary articles can achieve a favorable balance between judgment accuracy and interpretability.

Ethical Considerations

All of our raw data were obtained from lawful and publicly accessible websites, and the names of the defendants involved have been anonymized. Our original intention is to promote research on automatic generation of complete legal judgments by artificial intelligence, so as to better assist judges in making judgments, rather than to replace them.

Limitations

First, the charges and articles involved still do not cover all possible charges and articles. Second, the number of multi-defendant samples is still relatively small compared with single-defendant samples. Finally, for multi-defendant samples, we lack fine-grained annotation of the auxiliary laws that are applicable to each defendant. In our future work, we plan to supplement the insufficient data and provide more fine-grained annotations.

Acknowledgements

This work was supported by General Project of Guangdong Provincial Philosophy and Social Sciences Planning Program (2023): GD23CYJ08.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Zhiwei Fei, Songyang Zhang, Xiaoyu Shen, Dawei Zhu, Xiao Wang, Jidong Ge, and Vincent Ng. 2025. Internlm-law: An open-sourced chinese legal large language model. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 9376–9392.

Wanwei He, Jiabao Wen, Lei Zhang, Hao Cheng, Bowen Qin, Yunshui Li, Feng Jiang, Junying Chen, Benyou Wang, and Min Yang. 2023. Hanfei-1.0. https://github.com/siat-nlp/HanFei.

Zhitian Hou, Jinlin Li, Ge Lin, and Kun Zeng. 2025a. Knowjudge: A knowledge-driven framework for legal judgment prediction. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 47.

Zhitian Hou, Zihan Ye, Nanli Zeng, Tianyong Hao, and Kun Zeng. 2025b. Large language models meet legal artificial intelligence: A survey. *Preprint*, arXiv:2509.09969.

Zhitian Hou and Kun Zeng. 2025. Shizhi: A chinese lightweight large language model for court view generation. *arXiv preprint arXiv:2510.09297*.

Zikun Hu, Xiang Li, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. 2018. Few-shot charge prediction with discriminative legal attributes. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 487–498, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Wanhong Huang, Yi Feng, Chuanyi Li, Honghan Wu, Jidong Ge, and Vincent Ng. 2024. Cmdl: A large-scale chinese multi-defendant legal judgment prediction dataset. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 5895–5906.

Yunyun Huang, Xiaoyu Shen, Chuanyi Li, Jidong Ge, and Bin Luo. 2021. Dependency learning for legal judgment prediction with a unified text-to-text transformer. *arXiv preprint arXiv:2112.06370*.

Cong Jiang and Xiaolei Yang. 2023. Legal syllogism prompting: Teaching large language models for legal judgment prediction. In *Proceedings of the nineteenth international conference on artificial intelligence and law*, pages 417–421.

Ang Li, Yiquan Wu, Yifei Liu, Kun Kuang, Fei Wu, and Ming Cai. 2024. Enhancing court view generation with knowledge injection and guidance. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5896–5906.

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.

Yougang Lyu, Jitai Hao, Zihan Wang, Kai Zhao, Shen Gao, Pengjie Ren, Zhumin Chen, Fang Wang, and Zhaochun Ren. 2023. Multi-defendant legal judgment prediction via hierarchical reasoning. *arXiv preprint arXiv:2312.05762*.

Yougang Lyu, Zihan Wang, Zhaochun Ren, Pengjie Ren, Zhumin Chen, Xiaozhong Liu, Yujun Li, Hongsong Li, and Hongye Song. 2022. Improving legal judgment prediction through reinforced criminal element extraction. *Information Processing & Management*, 59(1):102780.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang,

Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Xiao Peng and Liang Chen. 2024. Athena: Retrieval-augmented legal judgment prediction with large language models. *arXiv preprint arXiv:2410.11195*.

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

Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, and 1 others. 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. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3086–3095, Online. Association for Computational Linguistics.

Qi Xu, Xiao Wei, Hang Yu, Qian Liu, and Hao Fei. 2024. Divide and conquer: Legal concept-guided criminal court view generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 3395–3410.

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

Xinyu Yang, Chenlong Deng, and Zhicheng Dou. 2026. GLARE: Agentic reasoning for legal judgment prediction. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13153–13170, San Diego, California, United States. Association for Computational Linguistics.

Hai Ye, Xin Jiang, Zhunchen Luo, and Wenhan Chao. 2018. Interpretable charge predictions for criminal cases: Learning to generate court views from fact descriptions. *arXiv preprint arXiv:1802.08504*.

Linan Yue, Qi Liu, Han Wu, Yanqing An, Li Wang, Senchao Yuan, and Dayong Wu. 2021. Circumstances enhanced criminal court view generation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 1855–1859.

Linan Yue, Qi Liu, Lili Zhao, Li Wang, Weibo Gao, and Yanqing An. 2024. Event grounded criminal court view generation with cooperative (large) language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2221–2230.

Shengbin Yue, Wei Chen, Siyuan Wang, Bingxuan Li, Chenchen Shen, Shujun Liu, Yuxuan Zhou, Yao Xiao, Song Yun, Xuanjing Huang, and 1 others. 2023. Disc-lawllm: Fine-tuning large language models for intelligent legal services. *arXiv preprint arXiv:2309.11325*.

Kepu Zhang, Haoyue Yang, Xu Tang, Weijie Yu, and Jun Xu. 2024. Beyond guilt: Legal judgment prediction with trichotomous reasoning. *arXiv preprint arXiv:2412.14588*.

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.

A Additional Details on Dataset Construction and Maintenance

A.1 Additional Information on Data Construction

First, this dataset includes only first-instance criminal judgments and excludes second-instance judgments, retrials, and revised decisions. Second, the source website provides a unique identifier for each judgment document, which we use for deduplication. Third, this dataset retains the category of exemption from criminal punishment, where the offender is not sentenced to any of the following penalties: criminal detention, public surveillance, fixed-term imprisonment, life imprisonment, or death penalty.

A.2 Maintenance Mechanism

Since the construction pipeline of this dataset mainly relies on code-based processing with limited manual intervention, it is generally scalable. If future changes occur in legal provisions or charge formulations, the pipeline can support corresponding updates to the charge–article mapping and related annotations. If the dataset is expanded, supplemented, or revised in the future, we will adopt versioned management and document the major changes.

B Data Analysis

B.1 Case Type Categorization

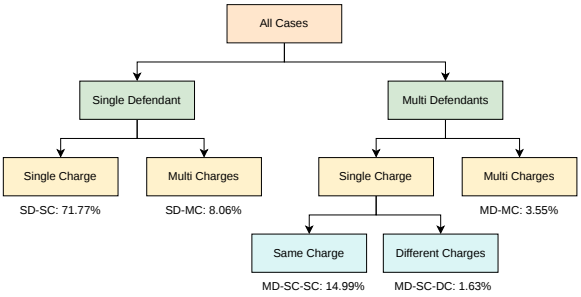


Figure 4: Case Type Categorization Tree

To systematically analyze the data, we hierarchically categorize all cases. First, we divide cases into two major groups by the number of defendants: single-defendant cases (single defendant, SD) and multi-defendant cases (multi defendants, MD). Then, within each group, we further distinguish single-charge (single charge, SC) and multiple-charge (multiple charges, MC) cases according to the number of charges involved for each defendant: if every defendant in a case corresponds to only one charge, the case is categorized as SC; otherwise, it is categorized as MC. For multi-defendant single-charge cases, we further split them into same-charge (same charge, SC) and different-charge (different charges, DC) depending on whether the charges across different defendants are exactly the same. Finally, all cases are summarized into five types: single-defendant single-charge (SD-SC), single-defendant multiple-charge (SD-MC), multi-defendant single-charge with the same charge (MD-SC-SC), multi-defendant single-charge with different charges (MD-SC-DC), and multi-defendant multiple-charge (MD-MC), as shown in Figure 4. The statistics show clear differences in the number of cases across the five types. Among them, SD-SC is the most common type, while cases involving multiple charges or multiple defendants are relatively fewer, which is broadly consistent with judicial practice where simple cases dominate and complex cases are less frequent.

B.2 Charge Distribution Analysis

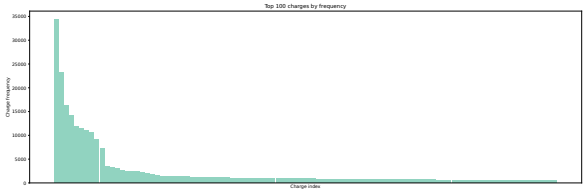


Figure 5: Charge Distribution

We further compute the frequency distribution of charge categories. Figure 5 shows the occurrence counts of the top 100 most frequent charges in the dataset, where the x-axis is the charge index ranked in descending order of frequency and the y-axis is the corresponding count. We observe that a small number of high-frequency charges account for a substantial proportion of the dataset, while the frequencies of a large number of remaining charges drop rapidly, forming a typical long-tail distribution. This distribution, on the one hand, reflects the strong concentration of criminal cases in a few common charges. On the other hand, it introduces pronounced class imbalance and few-shot issues, making it more challenging for models to learn low-frequency charges and their corresponding legal-application patterns.

C Metrics

C.1 Metrics for LJP-x

For the term of penalty attribute, we follow CMDL and partition it into several disjoint intervals, and model it as a multi-label task. For multi-class or multi-label attributes (such as charges), we use accuracy, precision, recall, and F1 for evaluation. Taking charge prediction as an example, let the gold label set contain m charges, the predicted set contain m_1 charges, and the overlap size be m_2. Then

$$\text{precision} = \frac{m_2}{m_1}, \quad \text{recall} = \frac{m_2}{m}.$$

A prediction is counted as correct only when the predicted set is exactly the same as the gold set (i.e., there are no extra or missing predictions).

For multi-defendant cases, we follow the case-level evaluation scheme proposed in CMDL. When a case contains n defendants, we first compute accuracy, precision, recall, and F1 for each defendant separately, then take the arithmetic mean of each metric over the n defendants in that case, and finally multiply it by $\log n$ to obtain the weighted

metric of that case. The final result on the multi-defendant dataset is the average of these weighted metrics over all cases.

For regression-type attributes (such as fines and probation), we use the logarithmic distance as the evaluation metric in order to reduce the impact of extreme values on the overall error. Smaller values indicate better performance, and the metric is defined as follows:

$$\text{log-distance} = \log\left(|\text{pred} - \text{gold}| + 1\right).$$

C.2 Metrics for CVG

For CVG, we report ROUGE, BLEU, and LLMS scores. Specifically, we compute ROUGE-1, ROUGE-2, and ROUGE-L, as well as BLEU-1, BLEU-2, and BLEU-4, and use their macro-averages as the final ROUGE and BLEU metrics.

For LLMS, we employ Qwen3.5-plus as an LLM-based judge to evaluate the generated court views. The prompt used for LLMS evaluation is provided in Figure 6.

Given the reference court view text: [Reference Court View]

Given the court view text generated by the model: [Generated Court View]

Please act as an evaluator and assign an overall quality score to the generated text on a scale from 1 to 5, where 1 indicates very poor quality and 5 indicates very high quality.

Please evaluate the generated text comprehensively based on the following factors:

1. **Semantic Consistency**: Whether the generated text is consistent with the core meaning expressed in the reference text;
2. **Content Completeness**: Whether it covers the key facts, main reasoning for the judgment, and important conclusions;
3. **Expression Quality**: Whether the language is clear, coherent, and consistent with the formal style of judicial documents.

Please output only the final score (1, 2, 3, 4, or 5), and do not output any explanation or other content.

Figure 6: The prompt used for LLMS evaluation

D Baseline Experimental Setup

Based on the experimental results and conclusions in Section 4.4, we adopt `LJP-x_CVG_Aux-law` as the unified generation order for the general-purpose LLMs and our fine-tuned models in the subsequent experiments, so as to balance verdict-item prediction performance and overall output stability.

For specialized legal LLMs, however, their prior task-specific fine-tuning makes it difficult for them to follow our prompt and generate the complete output in a single pass. Therefore, for evaluation, we decompose the task and prompt the model to generate only one sub-task at a time, and evaluate only the outputs that can be generated in a sufficiently stable manner.