# ZipLJP: Zipped Information Processor for Legal Judgment Prediction

**Fanghao Lou[1*], Qiqi Wang[1*†], Guanyu Chen[1], Kaiqi Zhao[2†], Huijia Li[1†]**

[1]School of Statistics and Data Science, AAIS, LPMC and KLMDASR, Nankai University, China
[2]Shenzhen Key Laboratory of Internet Information Collaboration, Harbin Institute of Technology (Shenzhen), China
{fanghao.lou, guanyu.chen}@mail.nankai.edu.cn, {qiqi.wang, hjli}@nankai.edu.cn
zhaokaiqi@hit.edu.cn

## Abstract

Large Language Models (LLMs) are widely used in legal judgment prediction tasks, which aim to enhance judicial efficiency. However, the length of legal fact descriptions poses a significant challenge to the application of LLMs. Long inputs not only introduce noise, affecting output quality, but also increase processing time. While existing text compression methods, such as generating summaries or training models to implicitly reduce text dimensionality, can shorten input length, they often face the slow generation speeds and limited interpretability issues. To address these issues and inspired by information bottleneck-based text compression, we propose the Zipped Information Processor for Legal Judgment Prediction method, ZipLJP. By effectively integrating legal knowledge into the compression process, ZipLJP not only reduces input length but also improves processing efficiency and prediction quality. Experiments show that our approach achieves better performance compared to the previous methods on two widely used open-source and real-world datasets.

**Code** — https://github.com/77-qiqi-wang/ZipLJP

## Introduction

Legal Judgment Prediction (LJP) is a critical research area in LegalAI (Luo et al. 2017). Its goal is to suggest judicial results, thus significantly reducing human workload in analyzing legal cases (Wu et al. 2023), and to enhance judicial efficiency (Deng et al. 2023). In criminal cases, the LJP task typically includes three key components: identifying applicable statutory provisions, classifying charges, and predicting sentencing results (Deng, Mao, and Dou 2024).

Previous approaches to LJP can be broadly categorized into three types: statistical methods (Katz, Bommarito, and Blackman 2017; Sulea et al. 2017), deep learning methods (Xu et al. 2020; Yue et al. 2021; Zhao et al. 2022; Zhang and Dou 2023; Wu et al. 2023), and language model-based methods (Xiao et al. 2021). Furthermore, previous research has increasingly applied legal knowledge (Wang et al. 2025b) to help models better understand complex factual scenarios. With the rapid advancement of LLMs, leveraging them for LJP tasks has become common. However, legal texts, particularly criminal fact descriptions, are often exceedingly long. For instance, the widely used MultiLJP dataset, which includes multiple defendants, contains criminal fact descriptions that can reach up to 98,688 characters (Lyu et al. 2023). This length poses significant challenges for LLMs in terms of comprehension, prediction accuracy, and computational cost (Bai et al. 2024).

Reducing the input text length is thus essential for enabling LLMs to process and understand effectively. Common text reduction methods are data-driven, utilizing data to

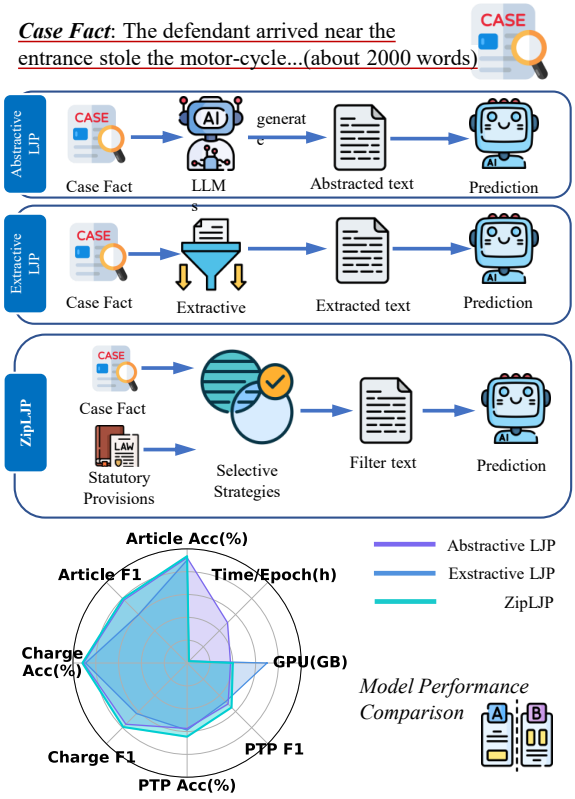



Figure 1: Through comprehensive multi-dimensional comparisons among traditional methods, and our ZipLJP model.

*These authors contributed equally.
†Corresponding authors

train model compression, fall into two main categories: extractive summarization(Gu, Ash, and Hahnloser 2022) and abstractive summarization(Lebanoff et al. 2019). Extractive methods aim to select important words or sentences to highlight the core facts while reducing overall length. Common extractive techniques include sparse attention(Zaheer et al. 2020) and text selection(Liu and Lapata 2019). Sparse attention techniques randomly or strategically select tokens during the attention mechanism in LLMs, whereas text selection models identify and extract important sentences to serve as summaries. Extractive methods, though efficient, often struggle to capture the implicit reasoning and inter-sentence dependencies in legal texts. Abstractive summarization methods, on the other hand, generate a short paragraph of the text based on semantic understanding, often using specially trained summarizers or LLMs. However, these models are costly to train and often fail to generalize across diverse legal domains due to the variability in document types and structures.

To achieve efficient and accurate information compression, reducing redundant information, we propose ***Z**ipped **I**nformation **P**rocessor for **L**egal **J**udgment **P**rediction*, ZipLJP. This method enables fast and effective text compression for processing lengthy fact descriptions in LLMs. Unlike data-driven text compression techniques, such as abstractive or extractive summarization and sparse attention, ZipLJP is a knowledge-driven text compression method. Specifically, it leverages statutory provisions to identify similar content in the facts and selects the most relevant parts. As a result, the compressed content is both legally grounded and interpretable. Figure 1 shows the comparison between our methods and the previous text compression. Experiments conducted on two real-world, open-source benchmark datasets for legal judgment prediction demonstrate that ZipLJP achieves better predictive performance.

Our contributions are summarized as follows:

- We propose the Zipped Information Processor for Legal Judgment Prediction, ZipLJP, to compress legal fact descriptions for using LLMs to predict the article, charge and prison terms.
- The proposed ZipLJP introduces a compression approach that leverages statutory provisions to identify and retain legally relevant information in fact descriptions. This makes the compressed content more interpretable and legally grounded, distinguishing it from traditional extractive or abstractive summarization methods.
- We conduct experiments on two open-source LJP datasets to demonstrate that ZipLJP achieves better predictive performance compared to existing methods, highlighting its practical effectiveness.

## Related Work

### Legal Judgment Prediction

Legal judgment prediction (LJP) has changed from early rule-based systems (Segal 1984) to statistical machine learning approaches (Katz, Bommarito, and Blackman 2017; Sulea et al. 2017), and more recently to deep learning models (Xu et al. 2020; Yue et al. 2021; Zhang and Dou 2023). Current research has been paid attention to the integration of domain-specific legal knowledge (Zhao et al. 2022) and the use of precedents (Zhang and Dou 2023; Wu et al. 2023), both of which contribute to more accurate and context-aware predictions. The scope of LJP has also broadened, moving beyond simple multi-class classification to address the complexities of real-world legal cases, such as those involving multiple defendants (Lyu et al. 2023) or overlapping legal provisions (Liu et al. 2023).

LLMs become powerful tools for legal reasoning, based on its reasoning capabilities (Ho, Schmid, and Yun 2023; Mukherjee et al. 2023; Xu et al. 2024). Recent studies have shown that prompting techniques can effectively guide LLMs toward more structured and logical decision-making (Wei et al. 2022; Press et al. 2023; Deng, Mao, and Dou 2024; Gao et al. 2025a). In particular, incorporating legal norms has been shown to enhance performance by aligning model reasoning with established judicial logic (Jiang and Yang 2023; Zhang, Wei, and Yu 2024).

### Text Compression Strategies

Lengthy input to language models has become an important issue that affects performance. There are two main types of methods to address this: sparse attention and text summarization. Sparse attention reduces the length of the attention process through random (Child et al. 2019) or rule-based selection strategies (Ainslie et al. 2020). In contrast, text summarization is a more effective approach that condenses long texts into shorter paragraphs (Lee et al. 2025). Text summarization can be categorized into two types: extractive and abstractive. Extractive methods aim to reduce input length by selecting salient sentences from the source text, but they often fail to capture implicit reasoning across sentences (Rodrigues et al. 2025; Chuang et al. 2024). Abstractive summarization, on the other hand, uses language models to generate novel summaries that may not contain any exact sentences from the original text (Wang et al. 2024, 2025a; Nagar et al. 2025). However, abstractive methods typically involve high computational costs and limited generalizability, as they require additional models or LLMs along with reference labels. Despite these advances, most existing methods overlook the use of legal knowledge to compress text while preserving law-relevant information (Gao et al. 2025b; Mondshine, Paz-Argaman, and Tsarfaty 2025; Li et al. 2025).

## Methodology

To overcome the aforementioned difficulties, we propose ZipLJP, a plug-in module for LLMs that allows efficient prediction of legal judgment by extracting only essential information from lengthy fact descriptions. Meanwhile, we provided a mathematical proof based on the information bottleneck theory. The mathematical symbols and their meanings are summarized in Table 1, where $\mathcal{Y}$ is the representation of triples: charge, law article, prison term.

### ZipLJP Process

Figure 2 illustrates the overall framework of the proposed method. ZipLJP consists of two stages: in the information

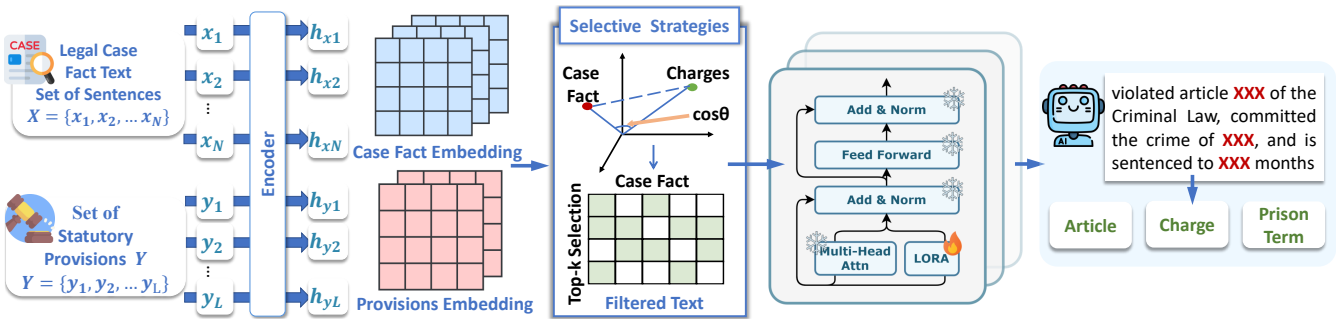



Figure 2: Overview of our framework. ZipLJP includes the text compression part and fine-tuning part.

compression stage, we apply embedding models and leverage legal knowledge to identify and select critical information; in the judgment prediction stage, we utilize LLMs with a structured output design to generate accurate predictions based on the compressed input.

**Information Compression** We first divide the criminal facts $\mathcal{X}$ into sentences $\{S_1, S_2, \ldots, S_n\}$, then input both the sentences $S_i$ and all charge statutory provisions $\tau = \{\tau_1, \tau_2 \ldots \tau_m\}$ into the embedding model. The embedding model subsequently outputs vectors $E_{\text{acu}}$ and $E_{\text{text}}$, where,

$$\begin{aligned} E_{\text{text}} &= Emb(S_i), i = \{1, 2 \ldots n\}, \\ E_{\text{acu}} &= Emb(\tau_j), j = \{1, 2..m\}. \end{aligned} \tag{1}$$

The similarity calculation is performed between the vectors using the function: sim($E_{\text{acu}}$, $E_{\text{text}}$). Based on the similarity scores, we select the top $k$ sentences as a summary. Guided by the legal knowledge derived from statutory provisions, these $k$ sentences are identified as the most legally relevant. We therefore define the compressed representation as $T = TopK(sim(E_{\text{law}}, E_{\text{text}}))$, with a compression rate of $k$.

**Judgment Prediction** After compressing the information, we input it into the LLMs. Inspired by prior work on structured generation with LLMs (Lu et al. 2025), we design the output format as a complete contextual template with blanks, such as: *"In conclusion, the defendant violated article **[Specific Provision]** of the Criminal Law of the People's Republic of China, committed the crime of **[Specific Charge]**, and is sentenced to **[Specific Duration]** months of imprisonment."* We compute the loss on the structured output segment. Mathematically, this is implemented using a masked cross-entropy loss function,

$$\mathcal{L} = -\frac{1}{N-m} \sum_{i=m+1}^{N} \log P\left(w_i \mid T, w_1, \ldots, w_{i-1}\right), \tag{2}$$

where, $N$ and $m$ are the total lengths of the input sequence and factual part, respectively. $w_i$ represents the $i$-th token. The summation range only covers the judgment prefix part $i > m$.

## Theory Proof for ZipLJP

Inspired by the Information Bottleneck (IB) theory (Tishby, Pereira, and Bialek 2000), we prove our design is effective through:(1) Once $k$ reaches a certain threshold, further increasing it does not yield additional gains in effective information. (2) The more informative the input, the better the model performs across all prediction metrics.

| Notation | Definition |
|---|---|
| $\mathcal{X}$ | Space of raw factual texts |
| $\mathcal{Y}$ | Space of legal judgments |
| $X \in \mathcal{X}$ | Input text of length |
| $T$ | Compressed text representation |
| $L$ | Knowledge of Law |

Table 1: Notations and Definitions

**Theorem 1.** *There exists a threshold $k^*$ such that for all $k \geq k^*$, the improvement in $I(T;L)$ becomes negligible. This point approximately corresponds to the optimal solution of the objective function* $\min\ I(X;T) - I(T;L)$.

*Proof.* To characterize how $I(T;L)$ accumulates with the number of selected elements, we define an information accumulation function:

$$F(k) = I(T;L) = \sum_{i=1}^{k} I(S_i;L) - \sum_{m<n<k} I(S_m; S_n \mid L), \tag{3}$$

where $S$ is the output of sim($E_{\text{law}}$, $E_{\text{text}}$), sorted in descending order of similarity. Here, $F(k)$ is a heuristic approximation of the mutual information between the compressed representation $T$ and the label $L$.

Since more similar samples (i.e., higher-ranked $S_i$) are likely to contain more information about $L$, we assume $I(S_i;L)$ decreases as $i$ increases. Meanwhile, larger indices $m, n$ correspond to more similar and redundant pairs, increasing $I(S_m; S_n \mid L)$, and reducing the marginal gain of information. As a result, the function $F(k)$ increases with $k$, but exhibits diminishing returns; that is, the incremental change $\Delta F(k) = F(k) - F(k-1)$ decreases as $k$ grows. Therefore, $F(k)$ is a convex Function. Now consider the objective:

$$\min I(X;T) - I(T;L), \tag{4}$$

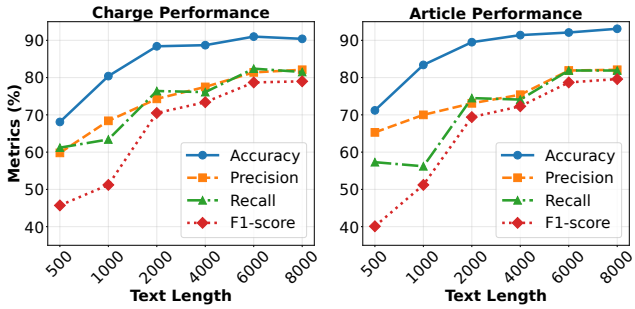



Figure 3: The Impact of Input Text Length on Model Predictive Performance

where both $I(X;T)$ and $I(T;L)$ increase with $k$, but at different rates. Although $I(T;L)$ increases rapidly at first, it eventually saturates and approaches its upper bound $I(X;L)$ because of the data processing inequality. In contrast, $I(X;T)$ continues to grow as more features from $X$ are retained in $T$, introducing redundancy and harming the objective.

Therefore, there exists a value $k^*$ such that:

$$\forall \epsilon > 0, \exists k^* : \frac{F(k^*)}{I(X;L)} \geq 1 - \epsilon, \tag{5}$$

indicating that most of the relevant information in $X$ about $L$ has been captured. Beyond this point, further increasing $k$ yields negligible improvement in $I(T;L)$ while continuing to increase $I(X;T)$, thereby worsening the objective.

Thus, a finite threshold $k^*$ exists which effectively balances informativeness and redundancy, serving as an approximate solution to the optimization problem. □

**Theorem 2.** *As the number of selected sentences in the set $X$ increases, the mutual information between the text and the label, $I(X;L)$, is non-decreasing, and the prediction performance metric $A$ improves accordingly. However, due to the upper bound exists, proved by Theorem 1, both the mutual information and prediction performance have a maximum limit, and the marginal gains diminish as more sentences are added.*

*Proof.* Let $X$ be the current set of selected sentences. When a new sentence $S_{m+1}$ is added, if it is relevant to the label $L$, i.e., $I(S_{m+1};L) > 0$, then by the chain rule of mutual information:

$$\begin{aligned} &I(X \cup \{S_{m+1}\}; L) \\ &= I(X;L) + I(S_{m+1};L) - I(X \cap S_{m+1};L) \\ &\geq I(X;L), \end{aligned} \tag{6}$$

where, the overlap term $I(X \cap S_{m+1};L) \leq I(S_{m+1};L)$, and is strictly less when the new sentence is not fully redundant with the existing set.

Therefore, the mutual information $I(X;L)$ is non-decreasing as more sentences are selected. On the other hand, the mutual information is upper bounded by the entropy of the label, i.e., $I(X;L) \leq H(L)$, which implies the existence of an upper limit. According to the corollary of Fano's inequality (Cover and Thomas 2005), the prediction accuracy $A$ is a monotonically increasing function of $I(X;L)$. Hence,

$$I(X;L) \leq I(X';L) \implies A(X) \leq A(X'). \tag{7}$$

We conduct a simple experiment by incrementally adding input tokens to observe changes in prediction accuracy on an LJP dataset. The results are shown in Figure 3. We observe that as more tokens are used, prediction accuracy generally increases. However, beyond 6000 tokens, the improvement becomes marginal or even slightly decreases.

In summary, selecting more sentences leads to better prediction performance. However, as established in Theorem 1, the improvement exhibits diminishing returns and eventually saturates. □

We further validate these theoretical findings and the effectiveness of our proposed method through experiments.

# Experiments

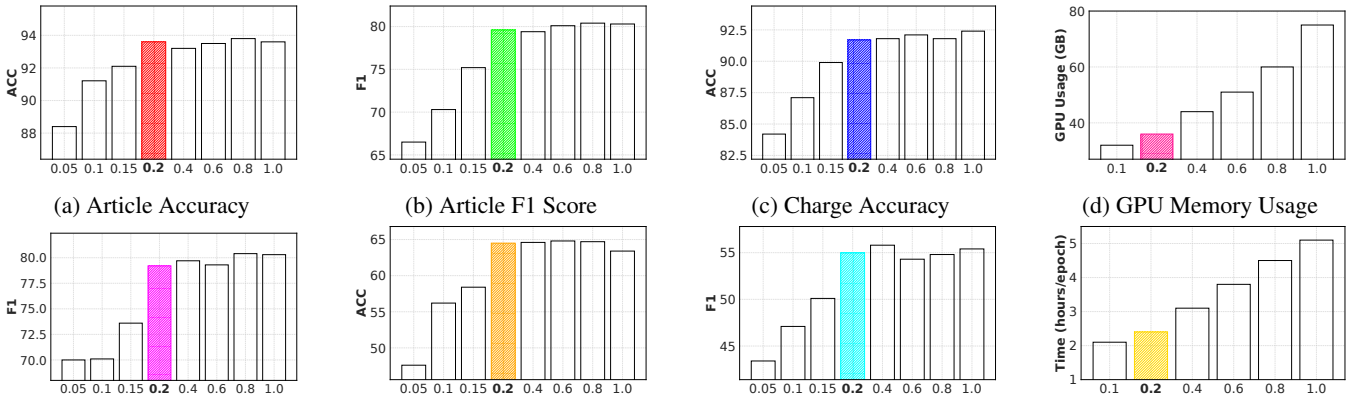
## Datasets and Evaluation Metrics

Following previous works (Deng et al. 2024), we conducted experiments on two open-source and real-world datasets, including both single-defendant and multi-defendant cases. For single-defendant scenarios, CAIL2018 (Xiao et al. 2018) serves as the foundation benchmark in the legal judgment prediction task. For multi-defendant cases, we utilize the MultiLJP dataset (Lyu et al. 2023). The dataset contains lengthy and complex criminal facts, which are significantly difficult than CAIL2018. The average text length is 441 and 3041 for the CAIL2018 and MultiLJP datasets. To evaluate our proposed model and baselines, we also use multiple metrics, including Accuracy (Acc.), Macro Precision (Ma-P), Macro Recall (Ma-R), and Macro F1 score (Ma-F).

## Baselines

We evaluate three categories of methods: LJP-specific approaches, traditional language models, and LLMs. For LJP-specific methods, we select several well-known and representative approaches. TopJudge (Zhong et al. 2018) models explicit topological relationships among the three constituent subtasks within its prediction framework. LADAN (Xu et al. 2020) introduces a specialized graph distillation mechanism to distinguish semantically similar legal provisions. NeurJudge (Yue et al. 2021) adopts a hierarchical representation learning strategy to process textual case facts across different subtasks.

In terms of language models, we choose a diverse set of architectures, including BERT (Devlin et al. 2019), Lawformer (Xiao et al. 2021), mT5 (Xue et al. 2021), and HRN (Lyu et al. 2023). For LLMs, we select models of varying sizes and training corpora, including LLaMA3.2-3B, LLaMA3.2-8B(Touvron et al. 2023), Qwen2.5-3B (Qwen et al. 2025), Qwen2-7B (Bai et al. 2023), and the legal domain-specific fine-tuned model LawyerLlama(Huang et al. 2023). For each LLM, we compare the LoRA fine-tuned version (Hu et al. 2022) with our proposed ZipLJP-enhanced variant. Then, we use the best-performing one to




(a) Article Accuracy (b) Article F1 Score (c) Charge Accuracy (d) GPU Memory Usage

(e) Charge F1 Score (f) PTP Accuracy (g) PTP F1 Score (h) Time Cost

Figure 4: Performance metrics across different top-k values. The bar at top-k=0.2 is highlighted with colored slashes.

| Methods | CAIL2018 | | | | | | MultiLJP | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Charge | | Law Article | | Prison Term | | Charge | | Law Article | | Prison Term | |
| | Acc. | Ma-F | Acc. | Ma-F | Acc. | Ma-F | Acc. | Ma-F | Acc. | Ma-F | Acc. | Ma-F |
| TopJudge | 65.5 | 74.1 | 68.2 | 74.3 | 32.1 | 32.4 | 67.6 | 55.7 | 73.9 | 54.1 | 36.1 | 33.1 |
| LADAN | 63.1 | 71.8 | 62.5 | 71.0 | 30.1 | 31.2 | 60.4 | 43.2 | 68.2 | 49.0 | 35.1 | 34.6 |
| NeurJudge | 65.7 | 71.4 | 67.4 | 70.9 | 29.6 | 33.2 | 64.8 | 51.2 | 71.8 | 55.7 | 33.9 | 32.0 |
| BERT | 64.6 | 74.6 | 68.3 | 73.5 | 31.7 | 33.5 | 66.3 | 54.2 | 73.6 | 54.0 | 35.6 | 32.9 |
| Lawformer | 66.2 | 73.1 | 67.5 | 74.4 | 30.4 | 30.7 | 68.1 | 53.8 | 76.2 | 53.8 | 36.1 | 34.7 |
| mT5 | 72.3 | 77.5 | 73.2 | 74.4 | 33.9 | 30.8 | 78.4 | 44.6 | 82.9 | 44.1 | 30.7 | 20.3 |
| HRN | - | - | - | - | - | - | 83.5 | 60.9 | 84.3 | 62.1 | 34.3 | 33.4 |
| *Base model: Qwen2-7B* | | | | | | | | | | | | |
| Finetune-LoRA | 74.1 | 78.6 | 74.0 | 75.5 | 32.0 | 31.3 | 85.4 | 65.2 | 87.7 | 63.5 | 32.0 | 31.3 |
| Finetune-CoT | 74.8 | 79.3 | 75.6 | 77.7 | 31.5 | 31.9 | 86.2 | 66.7 | 88.0 | 64.8 | 32.4 | 32.7 |
| ADAPT | 77.9 | 83.0 | 78.3 | 80.0 | 37.9 | 35.8 | 90.3 | 73.1 | 91.1 | 75.4 | 37.3 | 35.2 |
| ZipLJP (ours) | **87.7** | **85.7** | **88.5** | **85.9** | **63.9** | **36.1** | **91.7** | **79.2** | **93.6** | **79.6** | **64.5** | **55.0** |
| Improvements | 12.6% | 3.2% | 13.0% | 7.4% | 68.6% | 0.8% | 1.5% | 8.3% | 2.7% | 5.6% | 72.9% | 56.3% |
| *Base model: Other LLMs* | | | | | | | | | | | | |
| LLaMA3.2-3B Finetune | 77.4 | 65.4 | 74.3 | 68.9 | 55.6 | 31.2 | 85.3 | 63.4 | 81.3 | 72.3 | 51.2 | 33.1 |
| LLaMA3.2-3B ZipLJP | 78.3 | 67.0 | 73.5 | 71.5 | 54.3 | 30.5 | 87.2 | 62.4 | 82.4 | 73.8 | 53.4 | 36.9 |
| Qwen2.5-3B Finetune | 81.2 | 76.1 | 81.3 | 74.9 | 59.1 | 35.4 | 88.7 | 74.3 | 91.9 | 72.5 | 51.9 | 33.0 |
| Qwen2.5-3B ZipLJP | 83.1 | 75.9 | 83.9 | 76.7 | 61.0 | 34.3 | 90.1 | 76.7 | 92.1 | 74.4 | 54.8 | 33.4 |
| LLaMA3-8B Finetune | 83.0 | 79.2 | 85.1 | 79.0 | 57.8 | 33.7 | 90.5 | 63.0 | 91.9 | 65.3 | 57.6 | 38.7 |
| LLaMA3-8B ZipLJP | 85.0 | 83.4 | 84.0 | 82.0 | 55.1 | 30.9 | 90.9 | 60.0 | 92.9 | 62.6 | 58.8 | 41.8 |
| LawyerLLaMA-13B-V2 Finetune | 82.4 | 76.9 | 85.2 | 80.1 | 56.8 | 32.2 | 86.7 | 74.3 | 87.9 | 76.1 | 59.7 | 39.8 |
| LawyerLLaMA-13B-V2 ZipLJP | 86.4 | 81.9 | 86.7 | 83.1 | 59.2 | 34.4 | 91.3 | 78.2 | 92.1 | 75.5 | 57.4 | 47.1 |

Table 2: Experimental results on the baselines and our model. The best results are in bold.

further compare the Chain-of-Thought (CoT) reasoning(Ho, Schmid, and Yun 2023), which involves a two-phase process: generating preliminary reasoning paths and then optimizing the model using the synthesized chains. We also evaluate the best one in ADAPT (Deng et al. 2024) enhanced version, which introduces an Ask-Discriminate-Predict reasoning framework to improve the LLMs capabilities in legal judgment prediction.

## Implementation Details

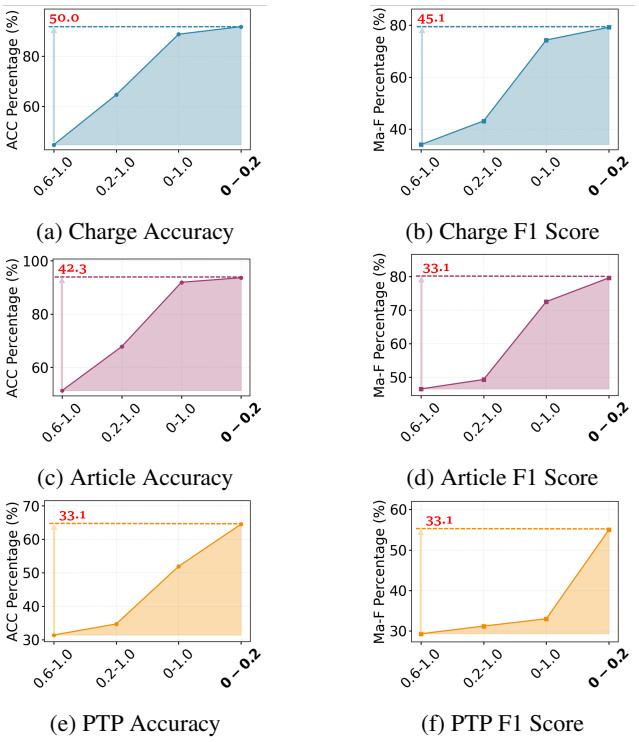
We apply the BAAI General Embedding (BGE)[1] as the embedding model to encode criminal charges and legal provisions. For efficient parameter-efficient fine-tuning, we follow the previous work settings (Deng et al. 2024) to implement LoRA (Low-Rank Adaptation) (Hu et al. 2022) across all linear modules. These LLMs are fine-tuned on five epochs with batch size of 1. The learning rate is initialized at 1e-5 and dynamically adjusted using the Adam optimizer to ensure optimal convergence. All experiments are conducted on a NVIDIA A100-SXM4-80GB GPU.

[1]https://huggingface.co/BAAI/bge-base-zh-v1.5

## Overall Comparison

Table 2 shows the comparison results.

**Compared to traditional methods** ZipLJP achieves better performance across all LJP-specific baselines and pre-




(a) Charge Accuracy (b) Charge F1 Score

(c) Article Accuracy (d) Article F1 Score

(e) PTP Accuracy (f) PTP F1 Score

Figure 5: Performance comparison across different proportions of retained text. For example, ***0.6–1.0*** means removing the top 60% most relevant portion of the text.

| Model | Params. | Law Article | | Charge | |
|---|---|---|---|---|---|
| | | Acc. | Ma-F | Acc. | Ma-F |
| Qwen2-7B | - | 87.7 | 63.5 | 85.4 | 65.2 |
| + ZipLJP w Legal-Bert | 110M | 79.7 | 55.4 | 75.5 | 52.1 |
| + ZipLJP w LawFormer | 149M | 89.2 | 63.9 | 85.6 | 60.3 |
| + ZipLJP w LaBSE | 110M | 90.1 | 68.8 | 87.0 | 70.1 |
| + ZipLJP w Erlangshen | 110M | 91.4 | 70.3 | 88.4 | 70.2 |
| + ZipLJP w M3E | 110M | 93.3 | 75.8 | 91.1 | 74.2 |
| + ZipLJP w BGE | 110M | **93.6** | **79.6** | **91.7** | **79.2** |

Table 3: The parameter size and predictive performance of different embedding models.

trained language models in all sub-tasks and datasets. On CAIL2018, ZipLJP achieves about 12.65% improvements in predicting charge, law article, and prison term. Similarly, on MultiLJP, there is 8.21% gain compared to traditional models by a Macro-F1.

**Compared to different LLMs** We compare ZipLJP-enhanced LLMs with their LoRA fine-tuned counterparts across various models, including LLaMA3.2-3B/8B, Qwen2.5-3B, Qwen2-7B and LawyerLLaMA-13B-V2. Notably, smaller models like Qwen2.5-3B with ZipLJP outperform larger fine-tuned models. Specifically, Qwen2.5-3B with ZipLJP achieves a 12.8% increase in prediction accuracy and a 3.2% improvement in Macro-F1 compared to the fine-tuned Qwen2-7B. On the domain-specific fine-tuned model, ZipLJP achieves a 4.7% improvement in accuracy.

| Model | Law Article | | Charge | | Persion Term | |
|---|---|---|---|---|---|---|
| | Acc. | Ma-F | Acc. | Ma-F | Acc. | Ma-F |
| Qwen2-7B | 87.7 | 63.5 | 85.4 | 65.2 | 32.0 | 31.3 |
| + BIGBIRDS | 78.1 | 56.2 | 73.6 | 54.5 | 48.3 | 39.1 |
| + LLMLingua | 79.1 | 69.1 | 82.5 | 68.3 | 47.2 | 37.4 |
| + LongLLMLingua | 78.2 | 71.2 | 84.3 | 71.3 | 51.7 | 37.2 |
| + LLMLingua-2 | 83.1 | 70.3 | 87.3 | 74.4 | 50.1 | 40.1 |
| + ExSum | 90.1 | 60.0 | 88.9 | 62.6 | 58,7 | 47.8 |
| + Qwen-Plus | 92.2 | 78.1 | 90.6 | 75.8 | 57.3 | 50.7 |
| + ZipLJP | **93.6** | **79.6** | **91.7** | **79.2** | **64.5** | **55.0** |

Table 4: The prediction effect of the model through compressing texts in different ways.

**Compared to different tuning methods** We also compare ZipLJP with several tuning strategies, including LoRA fine-tuning, Chain-of-Thought (CoT) prompting, and the ADAPT reasoning framework. ZipLJP consistently outperforms all these methods across all sub-tasks on both datasets. For example, on CAIL2018, ZipLJP improves the Macro-F1 score for law article prediction by 5.9% over ADAPT, which was previously a strong baseline.

These results demonstrate that ZipLJP is effective by highlighting legally essential information during compression, enabling downstream LLMs to make better predictions.

## Ablation Study

**Embedding Strategies Comparison** In the first step, ZipLJP requires encoding both case facts and legal knowledge. We compare several widely used legal context embedding strategies, and the results are shown in Table 3. These embedding methods include Legal-BERT (Chalkidis et al. 2020), LawFormer (Xiao et al. 2021), LaBSE (Feng et al. 2022), Erlangshen (Zhang et al. 2023), M3E (Wang, Sun, and He 2023), and BGE (Xiao et al. 2024).

According to the results, we select the BGE model due to its superior performance in Chinese legal tasks. This is because unlike traditional pre-trained models that primarily focus on language modeling, BGE is trained for the retrieval-enhanced objective, aiming to bring semantically similar sentences closer in the embedding space. This makes it particularly suitable for our scenarios, where semantic connection between fact descriptions and legal provisions.

**Hyperparameter Comparison** To determine the appropriate compression rate, we conduct ablation experiments with $k = \{0.05, 0.1, 0.15, 0.2, 0.4, 0.6, 0.8, 1.0\}$. Figure 4 shows the results on the MultiLJP dataset. From the results, we observe that settings with $k < 0.2$ have low performance, while $k > 0.2$ leads to similar high performance levels. Since providing more content to LLMs increases computational cost and inference time, we choose $k = 0.2$ as it make a balance between performance and input length. As shown in Fig. 5, we also removed the top 20% and top 60% of the most similar text and fed the remaining context into the model. The results indicate that the removed text negatively affects the model's performance.

| Item | Details |
|---|---|
| Facts | On November 6, 2014, Defendant [Defendant A] ...(*About 200 words*)... Defendants [Defendant A] and [Defendant B] appeared in court to participate in the proceedings. The trial has now concluded. The [LOC] Procuratorate alleged that at approximately 9:00 PM on November 5, 2014, Defendants [Defendant A] and [Defendant B] went to [LOC] and sold a small bag containing 11.2 grams of ketamine (a.k.a "K powder") to a drug buyer for 500 RMB. After completing the drug transaction, the two defendants were immediately apprehended by public security officers. ...(*about 150 words*)...the drug seizure certificate, identification transcripts and photos, the list of seized items, the account of the arrest, and household registration documents. These pieces of evidence are sufficient to establish the facts. |
| Results | Label: {"Charge": *Crime of drug trafficking*, "Article": *347*, "Term": *5*}, ADAPT: {"Charge": *Crime of drug trafficking*, "Article": *348*, "Term": *12*}, ZipLJP: {"Charge": *Crime of drug trafficking*, "Article": *347*, "Term": *8*} |
| Facts | In October 2013, Y sought to fraudulently obtain bank acceptance bills. He approached L, the legal representative of Company A; K, the legal representative of Company B; and W1, the legal representative of Company C. After discussion, they agreed to apply to a certain bank for a total of 18 million yuan in bank acceptance bills—6 million yuan under the name of each of the three companies—using a "three-party joint guarantee" arrangement. [Defendant], along with K and W1, consented to the plan, agreeing that the acceptance bills would be used solely by Y, who would also be responsible for the required deposits and repayment obligations. On October 22, 2013, ...(*about 100 words*)... credit line of 3 million yuan. Between October 22, 2013, and October 23, 2014, under Y's actual control, the three companies submitted falsified purchase and sales contracts on three occasions to conceal the true purpose of the funds. They obtained 6 million yuan in bank acceptance bills respectively from the bank. All of these bills were actually controlled and used by Y. The first two sets of acceptance bills were repaid on time. ...(*about 200 words*)... Also submitted were witness testimonies from fifteen individuals, as well as the confessions and defenses of [Defendant]. The public prosecutor holds that since [Defendant] voluntarily confessed in court, a lenient punishment may be considered at discretion. However, as [Defendant] is a recidivist, a heavier punishment shall be imposed according to the law. The prosecutor requests this court to convict and sentence [Defendant] in accordance with the relevant provisions of the Criminal Law of the People's Republic of China. |
| Results | Label: {"Charge": *Crime of obtaining loans and negotiable instruments by fraud*, "Article": *175*, "Term": *8*}, ADAPT: {"Charge": *Crime of obtaining loans and negotiable instruments by fraud*, "Article": *175*, "Term": *18*}, ZipLJP: {"Charge": *Crime of negotiable instrument fraud*, "Article": *175*, "Term": *12*} |

Table 5: Case study on MultiLJP and CAIL2018. The underlined text represents the compressed part generated by our proposed method, ZipLJP.

| Model | Law Article | | Charge | | Persion Term | |
|---|---|---|---|---|---|---|
| | Acc. | Ma-F | Acc. | Ma-F | Acc. | Ma-F |
| DeepSeek | 89.1 | 78.7 | 90.8 | 76.3 | 60.8 | 41.2 |
| *w/* ZipLJP | 91.1 | 77.3 | 92.7 | 78.4 | 61.2 | 40.1 |
| GPT-4 | 87.3 | 76.1 | 89.7 | 77.5 | 58.6 | 37.2 |
| *w/* ZipLJP | 90.6 | 78.2 | 93.1 | 76.9 | 59.7 | 39.8 |

Table 6: Comparison of zero-shot prediction performance of large-scale LLMs with and without incorporating ZipLJP.

**Text Compression Comparison** To demonstrate the effectiveness of the proposed method in text compression, we compare it with several approaches, including sparse attention, BIGBIRD (Zaheer et al. 2020), extractive summarization, ExSum(Zhou, Ribeiro, and Shah 2022) and LLMLingua Series (Jiang et al. 2023, 2024; Pan et al. 2024), and abstractive summarization using the Qwen-Plus API[2]. The results are presented in Table 4.

**Zero-Shot LLMs Comparsion** To validate ZipLJP's plug-in capability, we conducted zero-shot experiments on two large-scale LLMs: Deepseek-V3.2-Exp (DeepSeek-AI 2025) and GPT-4 (OpenAI, Achiam, and et al. 2024).

The results show that the proposed method achieves higher prediction accuracy. Moreover, compared to extractive and abstractive summarization methods, ZipLJP also offers better computational efficiency, as it does not require training additional models or using extra resources.

## Case study

We select examples from both datasets to illustrate the compressed text and the corresponding prediction results. Table 5 presents these examples.

We observe the compressed part, is the crucial portion of the factual description. For instance, in the first example, the underlined text directly describes the criminal act and supporting evidence. This indicates that our proposed method, ZipLJP, can effectively extract key information.

## Conclusion

In this work, we present ZipLJP, an information compression framework for legal judgment prediction using LLMs. By integrating legal knowledge into the compression process, ZipLJP effectively shorts fact descriptions into concise and legally meaningful representations. Our structured output scheme further enhances performance and reduces hallucinations in model predictions. Experimental results on two real-world LJP datasets demonstrate that ZipLJP outperforms existing methods. In future work, we aim to automatically determine the optimal threshold $k$ or develop a stopping criterion for sentence selection.

## Ethical Statement

All data used is anonymized and sourced from publicly available datasets for research purposes. We do not suggest that these models be straightforwardly applied in real courts, due to their current limitations in predictive accuracy and overall shortcomings compared to human experts.


## Acknowledgments

We thank the anonymous reviewers for their valuable feedback. This work is supported by the National Natural Science Foundation of China (Grant No. 72571150) and the Shenzhen Science and Technology Program (Grant No. SYSPG20241211173609009).


[2]https://qwen.ai/apiplatform

## References


Ainslie, J.; Ontañón, S.; Alberti, C.; Cvicek, V.; Fisher, Z.; Pham, P.; Ravula, A.; Sanghai, S.; Wang, Q.; and Yang, L. 2020. ETC: Encoding Long and Structured Inputs in Transformers. In *EMNLP*, 268–284.

Bai, J.; Bai, S.; Chu, Y.; Cui, Z.; Dang, K.; Deng, X.; Fan, Y.; Ge, W.; Han, Y.; Huang, F.; Hui, B.; Ji, L.; Li, M.; Lin, J.; Lin, R.; Liu, D.; Liu, G.; Lu, C.; Lu, K.; Ma, J.; Men, R.; Ren, X.; Ren, X.; Tan, C.; Tan, S.; Tu, J.; Wang, P.; Wang, S.; Wang, W.; Wu, S.; Xu, B.; Xu, J.; Yang, A.; Yang, H.; Yang, J.; Yang, S.; Yao, Y.; Yu, B.; Yuan, H.; Yuan, Z.; Zhang, J.; Zhang, X.; Zhang, Y.; Zhang, Z.; Zhou, C.; Zhou, J.; Zhou, X.; and Zhu, T. 2023. Qwen Technical Report. arXiv:2309.16609.

Bai, Y.; Lv, X.; Zhang, J.; Lyu, H.; Tang, J.; Huang, Z.; Du, Z.; Liu, X.; Zeng, A.; Hou, L.; Dong, Y.; Tang, J.; and Li, J. 2024. LongBench: A Bilingual, Multitask Benchmark for Long Context Understanding. In *ACL*, 3119–3137.

Chalkidis, I.; Fergadiotis, M.; Malakasiotis, P.; Aletras, N.; and Androutsopoulos, I. 2020. LEGAL-BERT: The Muppets straight out of Law School. In *EMNLP*, 2898–2904.

Child, R.; Gray, S.; Radford, A.; and Sutskever, I. 2019. Generating Long Sequences with Sparse Transformers. arXiv:1904.10509.

Chuang, Y.-N.; Xing, T.; Chang, C.-Y.; Liu, Z.; Chen, X.; and Hu, X. 2024. Learning to Compress Prompt in Natural Language Formats. In *NAACL-HLT*, 7756–7767.

Cover, T. M.; and Thomas, J. A. 2005. Elements of Information Theory.

DeepSeek-AI. 2025. DeepSeek-V3.2-Exp: Boosting Long-Context Efficiency with DeepSeek Sparse Attention.

Deng, C.; Mao, K.; and Dou, Z. 2024. Learning Interpretable Legal Case Retrieval via Knowledge-Guided Case Reformulation. In *EMNLP*, 1253–1265.

Deng, C.; Mao, K.; Zhang, Y.; and Dou, Z. 2024. Enabling Discriminative Reasoning in LLMs for Legal Judgment Prediction. In *EMNLP*, 784–796.

Deng, W.; Pei, J.; Kong, K.; Chen, Z.; Wei, F.; Li, Y.; Ren, Z.; Chen, Z.; and Ren, P. 2023. Syllogistic Reasoning for Legal Judgment Analysis. In *EMNLP*, 13997–14009.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*, 4171–4186.

Feng, F.; Yang, Y.; Cer, D.; Arivazhagan, N.; and Wang, W. 2022. Language-agnostic BERT Sentence Embedding. In *ACL*, 878–891.

Gao, J.; Cao, J.; Bu, R.; Zhu, N.; Guan, W.; and Yu, H. 2025a. Promoting knowledge base question answering by directing LLMs to generate task-relevant logical forms. In *AAAI*, 9.

Gao, J.; Wu, H.; Cheung, Y.-m.; Cao, J.; Yu, H.; and Zhang, Y. 2025b. Mitigating Forgetting in Adapting Pre-trained Language Models to Text Processing Tasks via Consistency Alignment. In *WWW*, 3492–3504.

Gu, N.; Ash, E.; and Hahnloser, R. 2022. MemSum: Extractive Summarization of Long Documents Using Multi-Step Episodic Markov Decision Processes. In *ACL*, 6507–6522.

Ho, N.; Schmid, L.; and Yun, S.-Y. 2023. Large Language Models Are Reasoning Teachers. In *ACL*, 14852–14882.

Hu, E. J.; yelong shen; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *ICLR*.

Huang, Q.; Tao, M.; Zhang, C.; An, Z.; Jiang, C.; Chen, Z.; Wu, Z.; and Feng, Y. 2023. Lawyer LLaMA Technical Report. arXiv:2305.15062.

Jiang, C.; and Yang, X. 2023. Legal Syllogism Prompting: Teaching Large Language Models for Legal Judgment Prediction. In *ICAIL*, 417–421.

Jiang, H.; Wu, Q.; ; Luo, X.; Li, D.; Lin, C.-Y.; Yang, Y.; and Qiu, L. 2024. LongLLMLingua: Accelerating and Enhancing LLMs in Long Context Scenarios via Prompt Compression. In *ACL*, 1658–1677.

Jiang, H.; Wu, Q.; Lin, C.-Y.; Yang, Y.; and Qiu, L. 2023. LLMLingua: Compressing Prompts for Accelerated Inference of Large Language Models. In *EMNLP*, 13358–13376.

Katz, D. M.; Bommarito, M. J.; and Blackman, J. 2017. A general approach for predicting the behavior of the Supreme Court of the United States. *PLoS ONE*, 12(4): e0174698.

Lebanoff, L.; Song, K.; Dernoncourt, F.; Kim, D. S.; Kim, S.; Chang, W.; and Liu, F. 2019. Scoring Sentence Singletons and Pairs for Abstractive Summarization. In *ACL*, 2175–2189.

Lee, S. Y.; Bahukhandi, A.; Liu, D.; and Ma, K. 2025. Towards Dataset-Scale and Feature-Oriented Evaluation of Text Summarization in Large Language Model Prompts. *IEEE Trans. Vis. Comput. Graph.*, 31(1): 481–491.

Li, H.; Lou, F.; Wang, Q.; Li, G.; and Perc, M. 2025. Multiresolution Clustering on Massive Attributed Graphs by means of Optimal Aggregated Markov Chains. *IEEE Trans. Netw. Sci. Eng*, 1–16.

Liu, Y.; and Lapata, M. 2019. Text Summarization with Pretrained Encoders. In *EMNLP-IJCNLP*, 3730–3740.

Liu, Y.; Wu, Y.; Zhang, Y.; Sun, C.; Lu, W.; Wu, F.; and Kuang, K. 2023. ML-LJP: Multi-Law Aware Legal Judgment Prediction. In *SIGIR*, 1023–1034.

Lu, Y.; Li, H.; Cong, X.; Zhang, Z.; Wu, Y.; Lin, Y.; Liu, Z.; Liu, F.; and Sun, M. 2025. Learning to Generate Structured Output with Schema Reinforcement Learning. In *ACL*, 4905–4918.

Luo, B.; Feng, Y.; Xu, J.; Zhang, X.; and Zhao, D. 2017. Learning to Predict Charges for Criminal Cases with Legal Basis. In *EMNLP*, 2727–2736.

Lyu, Y.; Hao, J.; Wang, Z.; Zhao, K.; Gao, S.; Ren, P.; Chen, Z.; Wang, F.; and Ren, Z. 2023. Multi-Defendant Legal Judgment Prediction via Hierarchical Reasoning. In *EMNLP*, 2198–2209.

Mondshine, I.; Paz-Argaman, T.; and Tsarfaty, R. 2025. Beyond N-Grams: Rethinking Evaluation Metrics and Strategies for Multilingual Abstractive Summarization. In *ACL*, 19019–19035.

Mukherjee, S.; Mitra, A.; Jawahar, G.; Agarwal, S.; Palangi, H.; and Awadallah, A. 2023. Orca: Progressive Learning from Complex Explanation Traces of GPT-4. arXiv:2306.02707.

Nagar, A.; Liu, Y.; Liu, A. T.; Schlegel, V.; Dwivedi, V. P.; Kaliya-Perumal, A.; Kalanchiam, G. P.; Tang, Y.; and Tan, R. T. 2025. uMedSum: A Unified Framework for Clinical Abstractive Summarization. In *ACL*, 2653–2672.

OpenAI; Achiam, J.; and et al., S. A. 2024. GPT-4 Technical Report. arXiv:2303.08774.

Pan, Z.; Wu, Q.; Jiang, H.; Xia, M.; Luo, X.; Zhang, J.; Lin, Q.; Ruhle, V.; Yang, Y.; Lin, C.-Y.; Zhao, H. V.; Qiu, L.; and Zhang, D. 2024. LLMLingua-2: Data Distillation for Efficient and Faithful Task-Agnostic Prompt Compression. In *ACL*, 963–981.

Press, O.; Zhang, M.; Min, S.; Schmidt, L.; Smith, N.; and Lewis, M. 2023. Measuring and Narrowing the Compositionality Gap in Language Models. In *EMNLP*, 5687–5711.

Qwen; :; Yang, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Li, C.; Liu, D.; Huang, F.; Wei, H.; Lin, H.; Yang, J.; Tu, J.; Zhang, J.; Yang, J.; Yang, J.; Zhou, J.; Lin, J.; Dang, K.; Lu, K.; Bao, K.; Yang, K.; Yu, L.; Li, M.; Xue, M.; Zhang, P.; Zhu, Q.; Men, R.; Lin, R.; Li, T.; Tang, T.; Xia, T.; Ren, X.; Ren, X.; Fan, Y.; Su, Y.; Zhang, Y.; Wan, Y.; Liu, Y.; Cui, Z.; Zhang, Z.; and Qiu, Z. 2025. Qwen2.5 Technical Report. arXiv:2412.15115.

Rodrigues, C.; Ortega, M.; Bossard, A.; and Mellouli, N. 2025. REDIRE: Extreme REduction DImension for extRactivE Summarization. *Data Knowl. Eng.*, 157: 102407.

Segal, J. A. 1984. Predicting Supreme Court Cases Probabilistically: The Search and Seizure Cases, 1962-1981. *Am. Polit. Sci. Rev.*, 78(4): 891–900.

Sulea, O.; Zampieri, M.; Malmasi, S.; Vela, M.; Dinu, L. P.; and van Genabith, J. 2017. Exploring the Use of Text Classification in the Legal Domain. In *ASAIL*, volume 2143.

Tishby, N.; Pereira, F. C.; and Bialek, W. 2000. The information bottleneck method. arXiv:physics/0004057.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; Rodriguez, A.; Joulin, A.; Grave, E.; and Lample, G. 2023. LLaMA: Open and Efficient Foundation Language Models. arXiv:2302.13971.

Wang, L.; Wu, L.; Song, S.; Wang, Y.; Gao, C.; and Wang, K. 2025a. Distilling Structured Rationale from Large Language Models to Small Language Models for Abstractive Summarization. In *AAAI*, 25389–25397.

Wang, Q.; Wang, R.; Zhao, K.; Amor, R.; Liu, B.; Liu, J.; Zheng, X.; and Huang, Z. 2024. SKGSum: Structured Knowledge-Guided Document Summarization. In *ACL*, 1857–1871.

Wang, Y.; Sun, Q.; and He, S. 2023. M3E: Moka Massive Mixed Embedding Model.

Wang, Z.; Zhao, S.; Wang, Y.; Huang, H.; Xie, S.; Zhang, Y.; Shi, J.; Wang, Z.; Li, H.; and Yan, J. 2025b. Re-TASK: Revisiting LLM Tasks from Capability, Skill, and Knowledge Perspectives. In *ACL*, 4925–4936.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; ichter, b.; Xia, F.; Chi, E.; Le, Q. V.; and Zhou, D. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *NIPS*, volume 35, 24824–24837.

Wu, Y.; Zhou, S.; Liu, Y.; Lu, W.; Liu, X.; Zhang, Y.; Sun, C.; Wu, F.; and Kuang, K. 2023. Precedent-Enhanced Legal Judgment Prediction with LLM and Domain-Model Collaboration. In *EMNLP*, 12060–12075.

Xiao, C.; Hu, X.; Liu, Z.; Tu, C.; and Sun, M. 2021. Lawformer: A Pre-trained Language Model for Chinese Legal Long Documents. arXiv:2105.03887.

Xiao, C.; Zhong, H.; Guo, Z.; Tu, C.; Liu, Z.; Sun, M.; Feng, Y.; Han, X.; Hu, Z.; Wang, H.; and Xu, J. 2018. CAIL2018: A Large-Scale Legal Dataset for Judgment Prediction. arXiv:1807.02478.

Xiao, S.; Liu, Z.; Zhang, P.; Muennighoff, N.; Lian, D.; and Nie, J.-Y. 2024. C-Pack: Packed Resources For General Chinese Embeddings. In *SIGIR*, 641–649.

Xu, N.; Wang, P.; Chen, L.; Pan, L.; Wang, X.; and Zhao, J. 2020. Distinguish Confusing Law Articles for Legal Judgment Prediction. In *ACL*, 3086–3095.

Xu, Q.; Wei, X.; Yu, H.; Liu, Q.; and Fei, H. 2024. Divide and Conquer: Legal Concept-guided Criminal Court View Generation. In *ACL*, 3395–3410.

Xue, L.; Constant, N.; Roberts, A.; Kale, M.; Al-Rfou, R.; Siddhant, A.; Barua, A.; and Raffel, C. 2021. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In *NAACL-HLT*, 483–498.

Yue, L.; Liu, Q.; Jin, B.; Wu, H.; Zhang, K.; An, Y.; Cheng, M.; Yin, B.; and Wu, D. 2021. NeurJudge: A Circumstance-aware Neural Framework for Legal Judgment Prediction. In *SIGIR*, 973–982.

Zaheer, M.; Guruganesh, G.; Dubey, K. A.; Ainslie, J.; Alberti, C.; Ontanon, S.; Pham, P.; Ravula, A.; Wang, Q.; Yang, L.; and Ahmed, A. 2020. Big Bird: Transformers for Longer Sequences. In *NIPS*, volume 33, 17283–17297.

Zhang, H.; and Dou, Z. 2023. Case Retrieval for Legal Judgment Prediction in Legal Artificial Intelligence. In *CCL*, 434–448.

Zhang, J.; Gan, R.; Wang, J.; Zhang, Y.; Zhang, L.; Yang, P.; Gao, X.; Wu, Z.; Dong, X.; He, J.; Zhuo, J.; Yang, Q.; Huang, Y.; Li, X.; Wu, Y.; Lu, J.; Zhu, X.; Chen, W.; Han, T.; Pan, K.; Wang, R.; Wang, H.; Wu, X.; Zeng, Z.; and Chen, C. 2023. Fengshenbang 1.0: Being the Foundation of Chinese Cognitive Intelligence. arXiv:2209.02970.

Zhang, Y.; Wei, X.; and Yu, H. 2024. HD-LJP: A Hierarchical Dependency-based Legal Judgment Prediction Framework for Multi-task Learning. *KBS*, 299: 112033.

Zhao, J.; Guan, Z.; Xu, C.; Zhao, W.; and Chen, E. 2022. Charge Prediction by Constitutive Elements Matching of Crimes. In *IJCAI*, 4517–4523.

Zhong, H.; Guo, Z.; Tu, C.; Xiao, C.; Liu, Z.; and Sun, M. 2018. Legal Judgment Prediction via Topological Learning. In *EMNLP*, 3540–3549.

Zhou, Y.; Ribeiro, M. T.; and Shah, J. 2022. ExSum: From Local Explanations to Model Understanding. In *NAACL-HLT*, 5359–5378.