Less is More: Knowledge-Aware Compression for Long Legal Judgment Prediction

Fanghao Lou[1]*, Qiqi Wang[1]*†, Guanyu Chen[1], Senbo Zhang[1], Kaiqi Zhao[2]†, Qian Liu[3]†, 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

[3]School of Computer Science, University of Auckland, New Zealand

{fanghao.lou, guanyu.chen, zhangbd}@mail.nankai.edu.cn, {qiqi.wang, hjli}@nankai.edu.cn, zhaokaiqi@hit.edu.cn, liu.qian@auckland.ac.nz

Abstract

Legal case facts are often lengthy, complex, and difficult to process, posing challenges for legal judgment prediction. Although recent advances leverage large language models (LLMs) for legal reasoning, they face high computational costs and information degradation when handling long cases. Previous approaches, such as architectural modifications and text compression methods, reduce computational complexity to some extent but still struggle to effectively capture legally salient information in complex cases. We propose a legal knowledge–adaptive compression framework for long legal judgment prediction that integrates domain-specific legal knowledge to guide adaptive context compression. Our approach selectively retains legally relevant information while reducing redundant or less informative content, enabling efficient and accurate long-context reasoning. We evaluate the proposed framework on four real-world datasets spanning multiple jurisdictions and languages. Experimental results demonstrate that our method outperforms existing approaches in both prediction performance and computational efficiency[1].

1 Introduction

As a key task in LegalAI, legal judgment prediction aims to infer judicial outcomes from case facts (Nigam et al., 2025; Xu et al., 2024; Liu et al., 2025b). While current LLM-based methods have improved prediction quality, they still face challenges in efficiency and in handling complex cases (Guha et al., 2023). The attention mechanism in LLMs scales quadratically with input length, leading to rapidly increasing computational costs for long legal documents (Wang et al., 2024a; Li et al., 2025). Moreover, legally salient information may be obscured or forgotten over long contexts, directly impairing judgment prediction accuracy (Klem and Moubayed, 2025). As a result, achieving efficient and accurate long-context reasoning has become an important problem for legal judgment prediction.

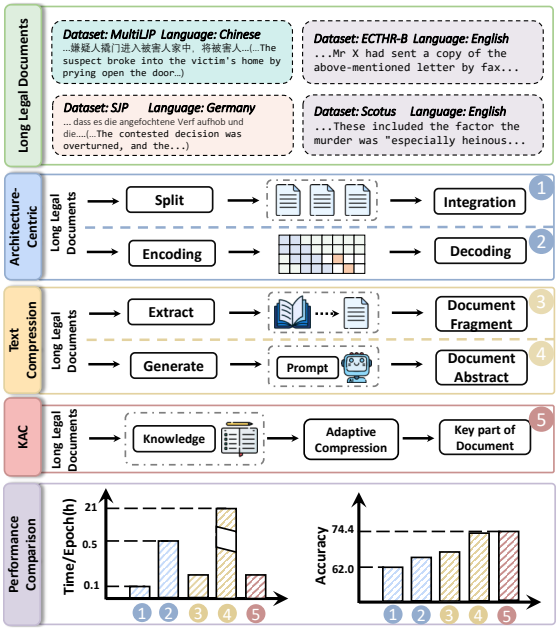


Figure 1: Comparison on Accuracy and Time cost between KAC and other models

Recent research on long context compression can be roughly divided into two technical directions. As shown in Figure 1, the first focuses on efficient model architectures, such as sparse attention series (Child et al., 2019; Zaheer et al., 2020; Yuan et al., 2025) and hierarchical frameworks (Amizadeh et al., 2025). The second focuses on text compression methods, including extractive and abstractive summarization (Li et al., 2024; Lou et al., 2026; Li et al., 2026). However, these approaches exhibit significant limitations when applied to long legal texts. Specifically, sparse atten-

*Equal Contribution

†Corresponding authors.

[1]The used code can be found at: https://github.com/Statistical-NLP-Lab/KAC.

Findings of the Association for Computational Linguistics: ACL 2026, pages 29015–29031

tion mechanisms struggle to capture the long-range dependencies necessary to connect key facts scattered throughout lengthy documents (Zhang et al.; Yang et al., 2024). Moreover, hierarchical classification and extractive summarization fail to account for logic coherence in legal texts, due to the lack of domain-specific understanding (Chen et al., 2025; Lee et al., 2025), resulting in compressed outputs that are legally inaccurate and inconsistent. Meanwhile, abstractive summarization faces two challenges: high computational complexity $O(n^2+m^2)$ and the tendency to generate hallucinations, where n represents the length of the input text and m denotes the length of the generated summary (Zhou et al., 2025).

To address these limitations, we propose a novel framework named ***K**nowledge-**A**daptive **C**ompression for Long Legal Documents Prediction*, *KAC*. KAC is specifically designed to overcome the shortcomings identified in existing approaches. First, by leveraging structured legal domain knowledge to guide the compression process, it ensures that scattered yet critical information is systematically identified and retained, preserving logical coherence while addressing long-range dependencies. Second, through its adaptive and knowledge-aware condensation of semantic units, KAC achieves high compression rates with $O(n + l^2)$ time complexity where l denotes the compression texts, effectively bypassing the prohibitive computation cost and hallucination risks associated with abstractive summarization. From an information-theoretic perspective, we show that KAC maximizes the mutual information between the compressed document and the underlying legal knowledge. Experiments on four multilingual, cross-jurisdictional legal judgment prediction datasets demonstrate that KAC delivers superior predictive performance while significantly improving computational efficiency.

Our contributions are summarized as follows:

- We propose a novel legal knowledge-aware compression framework for long legal judgment prediction, enabled by an efficient segmentation algorithm.
- We provide a theoretical analysis showing that the proposed method maximizes the mutual information between legal knowledge and document representations while minimizing document length.
- We evaluate the proposed framework in cross-lingual and cross-jurisdictional settings, demonstrating its effectiveness and efficiency in real-world LegalAI scenarios.

2 Related Work

Early legal judgment prediction relied on statistical or traditional machine learning methods (Segal, 1984; Katz et al., 2017; Șulea et al., 2017). The introduction of Transformer (Vaswani et al., 2017) brought significant progress, but its self-attention mechanism has $O(n^2)$ computational complexity, which limits its ability to process long documents.

2.1 Architecture Optimization Strategies

To overcome this limitation, research has mainly followed two directions. The first is to improve attention mechanisms for long sequences. For example, sparse attention methods like BigBird reduce computational cost by selecting tokens randomly or in a structured way (Child et al., 2019; Zaheer et al., 2020; Ainslie et al., 2020). The second direction is to design hierarchical processing frameworks. These methods split long documents into shorter semantic units, and then use a hierarchical structure to aggregate them into a global document representation (Zhang et al., 2019; Yang et al., 2016; Chalkidis et al., 2022a; Pappagari et al., 2019).

Recently, LLMs have been increasingly introduced to advance legal tasks (Xiao et al., 2021; Ho et al., 2023). This includes embedding LLMs as feature extractors within hierarchical frameworks (Skean et al., 2025). When document length far exceeds the model's processing limit, hierarchical frameworks often outperform simply fine-tuning very large LLMs (Prasad et al., 2024). In addition, integrating multiple deep representations from pretrained LLMs, combined with techniques like unsupervised clustering to approximate the internal structure of documents, can further improve the prediction performance of hierarchical models on unstructured legal texts (Prasad et al., 2024; Jawahar et al., 2019; Song et al., 2020; Yang and Zhao, 2019).

2.2 Text Compression Strategies

Text compression strategies for LLMs can be broadly categorized into extractive and abstractive methods. Extractive methods select important fragments from source texts using statistical features, graph-based algorithms (Zhong et al., 2024), or

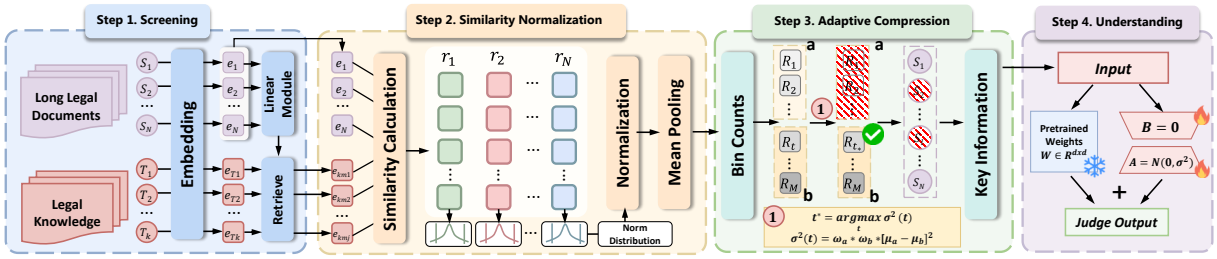


Figure 2: Overview of our framework. KAC includes four steps to achieve long legal judgment prediction efficiently.

sentence clustering (Rodrigues et al., 2025). These approaches maintain high fidelity and avoid hallucinations, but often miss implicit logical connections between sentences (Chuang et al., 2024; Xu et al., 2025). Abstractive methods leverage language models to produce shorter, more concise text (Liu et al., 2025a; Chang et al., 2024). However, these models need large amounts of training data and computational resources, and often produce factual errors when handling domain-specific or long texts (Wang et al., 2024b; Nagar et al., 2025).

Recent work has explored domain-adaptive prompt compression techniques, such as LLMLingua series (Jiang et al., 2023, 2024; Pan et al., 2024), which speed up inference while keeping task-relevant tokens. Unsupervised extractive summarization models like REDIRE (Rodrigues et al., 2025) use dimension reduction and clustering for efficient sentence extraction, but are still limited in using domain knowledge for content selection.

3 Methodology

3.1 KAC Framework

To achieve efficient and legally accurate compression for legal judgment prediction, we propose knowledge-aware compression (KAC), a two-stage framework based on legal knowledge selection and knowledge-aware compression. Figure 2 shows the overall framework. First, KAC quickly screens long legal documents by applying a lightweight classifier to identify core legal categories and computing the similarity between each text semantic units and the corresponding legal knowledge. Next, the adaptive compression module automatically selects key semantic units with high legal information density. Finally, the resulting high-quality compressed texts are used as input for legal judgment prediction.

3.2 Legal Knowledge Selection

Following prior work (Calleja et al., 2024; Liu et al., 2023), we decompose long case facts into a sequence of semantic units $D = \{s_1, s_2, ..., s_N\}$, where each s_i represents a sentence-level semantic unit. We first apply an LLM $\mathcal{M}_{\text{LLM}}$ to obtain a global semantic representation of the each case:

$$\mathbf{e}_{s_i} = \mathcal{M}_{LLM}(s_i), \tag{1}$$

where, d denotes the hidden dimension and $\mathbf{e}_D$ captures the overall case semantics.

Given the global case representation $\mathbf{e}_D$, we design a lightweight multi-layer perceptron (MLP) to perform legal knowledge selection. Specifically, the MLP predicts the most Top-M relevant statutory provisions $\mathcal{T} = \{t_1, t_2, ..., t_K\}$. This screening step restricts subsequent compression to a small subset of statutory provisions, thereby reducing both computational cost and noise. In legal judgment prediction tasks, statutory provisions, e.g., theft, fraud, and so on, are usually accompanied by official legal interpretations. We search for the corresponding professional legal interpretations from official sources such as the United States Code and the Criminal Law of the People's Republic of China, and then combine each statutory provision with its legal interpretation to form the legal knowledge $\mathcal{K}_m = \{k_{m1}, k_{m2}, ...\}$. We embed this knowledge using the same LLM.

$$\mathbf{e}_{k_{mj}} = \mathcal{M}_{LLM}(k_{mj}), \tag{2}$$

3.3 Knowledge-aware Compression

To quantify the relevance between each semantic unit s_i and a selected legal knowledge $\mathcal{K}_m$, we compute the maximum cosine similarity between s_i and all knowledge units in $\mathcal{K}_m$:

$$r_{im} = \max_{k_{mj} \in \mathcal{K}_m} \left(\frac{\mathbf{e}_{s_i} \cdot \mathbf{e}_{k_{mj}}}{\|\mathbf{e}_{s_i}\| \|\mathbf{e}_{k_{mj}}\|} \right), \tag{3}$$

where, $\mathbf{e}_{s_i} = \mathcal{M}_{LLM}(s_i)$.

However, raw cosine similarity scores are sensitive to embedding distribution bias and vary across documents and knowledge categories, making fixed thresholds ineffective. To obtain a statistically comparable relevance signal, we apply a normalization function that maps similarity scores into a bounded and approximately normalized space:

$$\hat{r}_{im} = \frac{\Phi_{\mu=0.5,\sigma=1}(r_{im}) - \Phi_{\mu=0.5,\sigma=1}(0)}{\Phi_{\mu=0.5,\sigma=1}(1) - \Phi_{\mu=0.5,\sigma=1}(0)} \quad (4)$$

where $\Phi_{\mu=0.5,\sigma}(\cdot)$ denotes the cumulative distribution function of a normal distribution with mean 0.5 and standard deviation 1. This normalization yields relevance scores in the range $[0, 1]$ and enables adaptive thresholding across different cases.

To obtain the most relevant compressed text, we assume that semantic units can be partitioned into high-relevance and low-relevance groups with respect to legal knowledge, resulting in a bimodal relevance score distribution. Motivated by previous work on automatic thresholding (Otsu, 1979; Kapur et al., 1985), we determine an optimal threshold by maximizing the separability between these two groups, which aligns with our goal of preserving information that is most informative for legal judgment prediction while discarding redundant content:

$$\sigma_B^2(\tau_m) = \omega_0(\tau_m)\omega_1(\tau_m)[\mu_0(\tau_m) - \mu_1(\tau_m)]^2, \quad (5)$$

where, ω_0, μ_0 and ω_1, μ_1 denote the proportions and mean relevance scores of the low- and high-relevance groups separated by threshold τ_m. The optimal threshold is given by:

$$\tau_m^* = \arg\max_{\tau_m \in [0,1]} \sigma_B^2(\tau_m), \quad (6)$$

All semantic units s_i that satisfy $\hat{r}_{im} \geq \tau_m^*$ are retained to form the compressed text D'_m for the target t_m. This process is repeated for all M targets, and their union yields the final refined text for generation, $D' = \bigcup_{m=1}^{M} D'_m$.

3.4 Legal Judgment Prediction

The compressed text D' is then fed into an LLM-based predictor $\mathcal{G}$. Guided by task-specific prompts (e.g., "In summary, this legal document belongs to the category of [Category]."), the generator produces the legal judgment prediction Y.

3.5 Theory Proof of KAC

We provide a theoretical justification for the proposed knowledge-aware compression framework from the perspective of mutual information theory. Specifically, we show that increasing the mutual information between the compressed text and legal knowledge enhances the performance of legal judgment prediction.

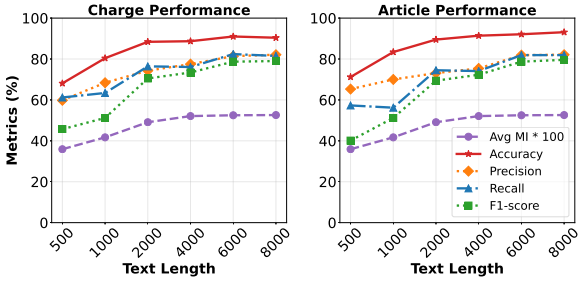


Figure 3: The correlation between mutual information (MI) and model performance highlights the importance of maximizing MI to improve the model's ability to understand legal long documents effectively.

Theorem 1. *Let the objective function for the legal text generation task be $\mathcal{L}(y, y^*)$, where y^* is the ground truth label. If the mutual information $I(D'; T)$ between the input text D' and the legal knowledge topic increases, the expected loss between the model prediction y and y^* decreases.*

Proof. According to the information bottleneck theory (Tishby et al., 2000), the optimal representation should maximize the mutual information with the target T while compressing the input information. We define:

$$\mathcal{R} = I(D'; T) - \beta I(D; D'), \quad (7)$$

where, β is a lagrange multiplier. By the data processing inequality, we have:

$$I(D'; T) \leq I(D; T), \quad (8)$$

when, D' contains more information relevant to T, $I(D'; T)$ increases. According to the variational bound for generative models:

$$\log P(y|D') \geq E_{q(z|D')}[\log P(y|z)] - \mathrm{KL}(q(z|D')\|p(z)), \quad (9)$$

As $I(D'; T)$ increases, the latent representation z contains more task-relevant information, which increases the conditional likelihood $P(y|z)$ and reduces the loss $\mathcal{L}$. □

We provide a concrete formulation for estimating the mutual information between the compressed text and legal knowledge in Appendix 3.

Table 1: Overall performance comparison on legal judgment prediction Datasets. The best results are in **Bold**. **Green** types denote improvements of KAC (blue background) relative to LLMs fine-tuned methods.

Model	MultiLJP(CN)						EN				DE	
	Charge		Law Article		Prison Term		SCOTUS(US)		ECTHR-B(EU)		SJP(CH)	
	Acc	Ma-F1	Acc	Ma-F1	Acc	Ma-F1	Acc	Ma-F1	Acc	Ma-F1	Acc	Ma-F1
Baselines												
LADAN	63.1	71.8	62.5	71.0	30.1	31.2	-	-	-	-	-	-
NeurJudge	65.7	71.4	67.4	70.9	29.6	33.2	-	-	-	-	-	-
HRN	83.5	60.9	84.3	62.1	34.3	33.4	-	-	-	-	-	-
HNET	-	-	-	-	-	-	72.2	63.4	65.4	49.9	67.1	43.2
FNET	-	-	-	-	-	-	71.3	58.8	65.2	54.5	69.3	41.1
BERT	44.8	25.1	51.4	30.7	29.6	21.6	68.3	58.3	51.0	29.2	56.4	30.8
RoBERTa	67.5	66.4	71.5	56.9	31.2	30.7	71.6	62.0	58.6	37.5	60.1	29.5
DeBERTa	70.0	64.1	75.6	61.5	33.8	32.7	71.1	62.7	61.4	44.6	64.5	31.7
BigBird	73.6	54.5	78.1	56.2	48.3	39.1	72.8	62.0	62.0	40.1	67.7	41.3
Longformer	81.5	52.9	83.4	59.8	32.4	18.3	72.9	64.0	62.4	44.3	65.4	38.7
LLMs Base Method												
Qwen2-7B	90.4	72.1	91.1	73.5	54.8	31.3	71.5	65.1	63.0	50.9	71.2	42.1
w/ ADAPT	90.3	73.1	91.1	75.4	37.3	35.2	-	-	-	-	-	-
w/ KAC	91.3	**80.1**	**93.4**	**80.2**	**64.5**	**44.2**	75.0	67.6	73.3	61.0	74.7	49.1
ΔGain	**+0.9**	**+8.0**	**+2.3**	**+6.7**	**+9.7**	**+12.9**	**+3.5**	**+2.5**	**+10.3**	**+10.1**	**+3.5**	**+7.0**
LLaMA3.2-3B	85.3	63.4	81.3	72.3	51.2	33.1	69.4	57.4	57.3	49.6	68.6	48.0
w/ KAC	**91.9**	63.7	89.3	75.7	54.5	41.3	72.9	62.8	66.7	55.8	74.2	52.0
ΔGain	**+6.6**	**+0.3**	**+8.0**	**+3.4**	**+3.3**	**+8.2**	**+3.5**	**+5.4**	**+9.4**	**+6.2**	**+5.6**	**+4.0**
Qwen2.5-3B	88.7	74.3	91.9	72.5	51.9	33.0	66.9	51.60	63.0	49.0	68.7	49.5
w/ KAC	90.4	75.7	92.1	75.1	55.1	39.2	73.5	62.4	72.4	63.0	71.6	53.1
ΔGain	**+1.7**	**+1.4**	**+0.2**	**+2.6**	**+3.2**	**+6.2**	**+6.2**	**+10.9**	**+9.4**	**+14.0**	**+2.9**	**+3.6**
LLaMA3-8B	90.5	63.0	91.9	65.3	57.6	38.7	69.7	60.9	64.5	51.3	66.7	43.5
w/ KAC	90.9	64.1	92.2	66.4	60.1	41.3	**76.1**	**68.1**	**74.4**	**67.3**	76.5	53.4
ΔGain	**+0.4**	**+1.1**	**+0.3**	**+1.1**	**+2.5**	**+2.6**	**+6.4**	**+7.2**	**+9.9**	**+16.0**	**+9.8**	**+9.9**
Qwen3-14B	89.8	60.2	91.8	63.4	51.4	32.1	71.3	58.3	59.6	48.7	73.4	49.6
w/ KAC	90.8	68.5	92.8	72.4	60.1	40.2	72.9	63.4	72.6	59.0	76.3	**58.3**
ΔGain	**+1.0**	**+8.3**	**+1.0**	**+9.0**	**+8.7**	**+8.1**	**+1.6**	**+5.1**	**+13.0**	**+10.3**	**+2.9**	**+8.7**
LawyerLLaMA-13B	86.7	74.3	87.9	76.1	59.7	39.8	65.3	50.0	63.6	53.4	70.5	43.3
w/ KAC	88.7	76.8	92.3	77.1	61.2	41.5	72.4	61.4	67.4	57.2	**76.6**	51.2
ΔGain	**+2.0**	**+2.5**	**+4.4**	**+1.0**	**+1.5**	**+1.7**	**+7.1**	**+11.4**	**+3.8**	**+3.8**	**+6.1**	**+7.9**

In addition, we conduct a controlled experiment by progressively increasing the amount of legally salient case facts in the input. As shown in Figure 3, prediction performance consistently improves as more relevant information is retained, which empirically supports our theoretical analysis.

4 Experiments

4.1 Experiments Setting

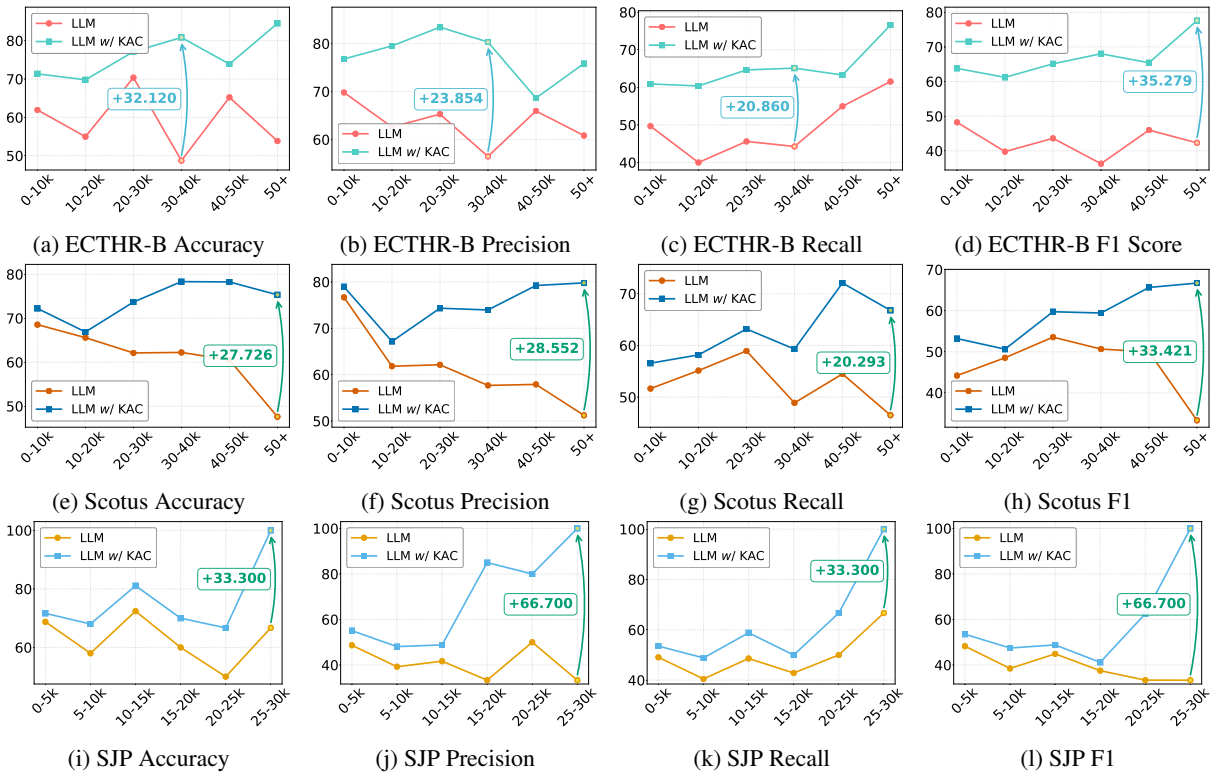
Datasets We conducted experiments on four long legal judgment prediction datasets across multiple languages and jurisdictions. For Chinese, we used MultiLJP (Lyu et al., 2023), which predicts charges, applicable laws, and sentences from criminal case facts. For English, we used SCOTUS (Chalkidis et al., 2022b) and ECTHR-B (Chalkidis et al., 2021), where the tasks are to predict issue areas from U.S. Supreme Court opinions and violated articles of the European Convention on Human Rights, respectively. We also used the German cases of Swiss-Judgment-Prediction (SJP) (Niklaus et al., 2021), a multilingual dataset annotated with binarized judgment outcomes from the Swiss Federal Supreme Court. To focus on long-document scenarios, we removed samples with text lengths shorter than 3,000 tokens.

Evaluation Metrics We evaluated all models using Accuracy (Acc.), Macro Precision (Ma-P), Macro Recall (Ma-R), and Macro F1-score (Ma-F).

Baselines We select three categories of baselines:

Traditional Legal Judgment Prediction Models Since prior work often focuses on a single jurisdiction, we selecte LADAN (Xu et al., 2020), HRN (Lyu et al., 2023), and NeurJudge (Yue et al., 2021) as representative for MultiLJP, HNet (Giofré and Ghantasala, 2023) and FNet (Lee-Thorp et al., 2022) for the remaining datasets.

Pre-trained Language Models We use



(a) ECTHR-B Accuracy (b) ECTHR-B Precision (c) ECTHR-B Recall (d) ECTHR-B F1 Score

(e) Scotus Accuracy (f) Scotus Precision (g) Scotus Recall (h) Scotus F1

(i) SJP Accuracy (j) SJP Precision (k) SJP Recall (l) SJP F1

Figure 4: Performance metrics across different text lengths. The dataset is grouped into categories with a 5K step size. Green arrows(→) indicate the categories with the largest improvements.

BERT (Devlin et al., 2019), RoBERTa (Zhuang et al., 2021), and DeBERTa (He et al., 2021). Additional, we also select efficient modification language models, BigBird (Zaheer et al., 2020) and Longformer (Beltagy et al., 2020).

Large Language Models We evaluated a range of LLMs with different model sizes and training corpora, including LLaMA3.2-3B, LLaMA3-8B (Touvron et al., 2023), Qwen2.5-3B (Qwen et al., 2025), Qwen2-7B (Bai et al., 2023), and Qwen3-14B (Yang et al., 2025). We also included LawyerLLaMA (Huang et al., 2023), a legal-domain fine-tuned model, and compared against ADAPT (Deng et al., 2024), a legal judgment prediction framework built on Qwen2-7B. Additionally, we evaluated the proposed KAC framework on larger LLMs, including DeepSeek-V3.2 (DeepSeek-AI, 2025) and GPT-4 (OpenAI et al., 2024).

Implementation Details Following prior work, we applied LoRA (Low-Rank Adaptation) (Hu et al., 2022) to efficiently fine-tune open-source LLMs. We set $M = 3$. We utilize Otsu (Otsu, 1979) as the automatic thresholding method. The learning rate was initialized at 1×10^{-5} and optimized using Adam with a dynamic learning rate schedule. All experiments were conducted on an NVIDIA A100-SXM4-80GB GPU.

Table 2: Comparison with larger LLMs on MultiLJP.

Model	Law Article		Charge		Persion Term	
	Acc.	Ma-F	Acc.	Ma-F	Acc.	Ma-F
MultiLJP						
DeepSeek	89.1	78.7	90.8	76.3	60.8	41.2
w/ KAC	91.4	80.1	93.6	80.7	65.1	44.3
ΔGain	**+2.3**	**+1.4**	**+2.8**	**+4.4**	**+4.3**	**+3.1**
GPT-4	87.3	76.1	89.7	77.5	58.6	37.2
w/ KAC	91.2	77.9	93.5	79.6	61.7	42.2
ΔGain	**+3.9**	**+1.8**	**+3.8**	**+2.1**	**+2.9**	**+5.0**

4.2 Overall Comparison

Table 1 reports the overall results.

Comparison with smaller models The results show that KAC consistently improves the performance of relatively smaller models. On MultiLJP, Qwen2-7B w/ KAC achieves an average improvement of 21%. On SCOTUS and ECTHR-B, LLaMA3-8B w/ KAC outperforms Longformer by approximately 20% on average. On SJP, LawyerLLaMA-13B w/ KAC achieves an average improvement of 18%.

Comparison with LLMs Compared with using original facts, LLMs with KAC achieve substan-

Table 3: Performance and time cost compare of the model through compressing texts in different ways. TI.(↑) represents the additional time overhead of other methods compared to KAC.

Model	Charge		Persion Term		Time Cost	
	Acc.	Ma-F	Acc.	Ma-F	Time	TI.(↑)
Qwen2-7B						
w/ Original Text	90.4	72.1	54.8	31.3	5.1	↑2.6
w/ LLMLingua	82.5	68.3	47.2	37.4	5.5	↑2.9
w/ LongLLMLingua	84.3	71.3	51.7	37.2	5.7	↑3.2
w/ LLMLingua-2	87.3	74.4	50.1	40.1	5.1	↑2.6
w/ ExSum	88.9	62.6	58,7	47.8	2.8	↑0.3
w/ Qwen-Plus	90.6	75.8	57.3	50.7	21.9	↑19.4
w/ KAC	**91.3**	**80.1**	**64.5**	**55.0**	2.5	**0.0**
ΔGain	**+1.2**	**+2.1**	**+0.7**	**+4.3**	**+7.2**	-

tial gains. On MultiLJP, Qwen2-7B w/ KAC improves Accuracy by 17% and Macro F1 by 41% on the Prison Term prediction task. On ECTHR-B, LLaMA3-8B w/ KAC achieves a 31% improvement in Macro F1. Similar improvements are observed across larger LLMs, as shown in Table 2, indicating that KAC generalizes well.

Overall, KAC achieves consistent and significant improvements across multilingual and cross-jurisdictional datasets. Compared with traditional legal judgment prediction models and uncompressed LLM baselines, KAC demonstrates superior effectiveness and scalability, highlighting the benefit of knowledge-guided compression for long legal document reasoning.

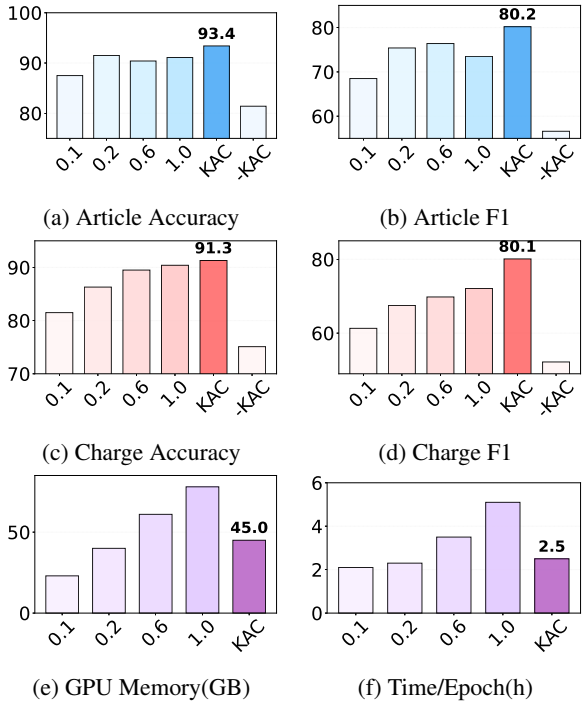
4.3 Length Analysis

To evaluate the model's performance on long legal judgment prediction, we group test samples by input length, with results shown in Figure 4. KAC performs similarly to the full text on shorter documents but shows a clear advantage as length increases. On the SCOTUS dataset, it achieves over 100% improvement in the 30K–35K length range, demonstrating its strength in processing extremely long legal texts.

4.4 Compression Strategy Comparison

We compare KAC with representative text compression methods, including LLMLingua series (Jiang et al., 2023, 2024; Pan et al., 2024) and generic text compression methods, ExSum (Zhou et al., 2022) and Qwen-Plus (Qwen et al., 2025). As shown in Table 3, KAC achieves the best overall performance on prediction tasks while incurring the lowest inference time.

These results indicate that KAC provides a more effective accuracy–efficiency trade-off than existing generic and LLM-based compression strategies for long legal judgment prediction.



(a) Article Accuracy (b) Article F1

(c) Charge Accuracy (d) Charge F1

(e) GPU Memory(GB) (f) Time/Epoch(h)

Figure 5: Fixed ratios and adaptive compression. We compare the text compressed with the top k% of the text retained based on similarity, as well as the text removed by KAC. The comparison includes the prediction performance, GPU memory, and time cost. The term **"-KAC"** refers to the text removed by KAC.

4.5 Ablation and Hyperparameter Analysis

Due to the task diversity, we use the MultiLJP as a benchmark to evaluate the hyperparameters and ablation.

Fixed Ratio and Adaptive Compression KAC employs an automatic thresholding strategy for adaptive compression. To assess its effectiveness, we compare automatic thresholding with fixed-ratio compression. As shown in Figure 5, we evaluate retaining the top-k semantic units versus the proposed approach. The results show that automatic compression achieves better model performance while improving GPU utilization and inference efficiency, avoiding the need for manual hyperparameter tuning.

Automatic Thresholding Methods Otsu (Otsu, 1979) and KSW (Kapur et al., 1985) are two common automatic thresholding methods. As shown in Appendix B, both maximize information-based criteria, but Otsu has lower computational com-

ECTHR-B: The applicants were born in 1981 and 1966 respectively and live in Ecemiş village near the town of Lice, located with the administrative jurisdiction of the province of Diyarbakır...(about 300 words)...Having regard to the particular brand of the glue which some people used for sniffing, as well as to the hunting rifle, seven soldiers concluded in their report that the applicant's relative had been a "terrorist posing as an ordinary citizen. **(The point of contention)** (about 500 words)... and asked for assistance in bringing the perpetrators of the killing of his nephew to justice. He also stated that since the killing, soldiers had been setting fire to the area where his nephew had been killed, and deliberately destroying the evidence.18. On 6 October 2008 the Lice prosecutor sent the investigation file to the Diyarbakır prosecutor who had jurisdiction to investigate incidents perpetrated by illegal organisations. In his letter accompanying the file the Lice prosecutor named the first applicant Fatma Güler as the "victim of a crime", **(Evidence interruption.)** and her deceased brother Murat Tekdal as the "deceased/suspect"...(about 500 words) The Assize Court stated that there was no on-going investigation into the killing of Murat Tekdal and that there had not been a decision not to prosecute anyone in respect of his killing. As there was no on-going investigation into the killing, the applicants had no standing to bring an objection against the Diyarbakır prosecutor's decision. **(Valid investigation missing.)**The Assize Court's decision was communicated to the applicants on 30 April 2010...

Analysis: The Compression Text contradicts the claim of a the official investigation shifts focus to blaming the deceased rather than the perpetrator → the judicial system ultimately concludes that the murder case itself was never investigated. This directly raises the fundamental legal question of whether the state fulfilled its obligation under Article 2 of the European Convention on Human Rights (right to life) to conduct an *effective investigation.*

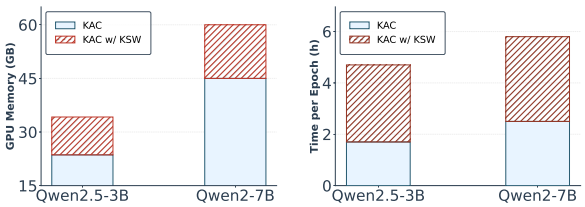
SCOTUS: Wyoming, a major coal-producing State, does not sell coal, but does impose a severance tax on those who extract it. From 1981 to 1986, Wyoming provided virtually 100% of the coal purchased by four Oklahoma electric utilities, including the Grand River Dam Authority , a state agency. However, after the Oklahoma Legislature passed an Act requiring coal-fired electric utilities to burn a mixture containing at least 10% Oklahoma-mined coal, the utilities reduced their purchases of Wyoming coal in favor of Oklahoma coal, and Wyoming's severance tax revenues declined. **(Case Cause)** ...(about 2000words)... would have been sold to Oklahoma utilities by a Wyoming producer would have been subject to the tax when extracted. Wyoming's loss of severance tax revenues' fairly can be traced' to the Act **(Crucial Sentence)** ...(about 1000 words) but also by considering what effect would arise if many States or every State adopted similar legislation. Healy v. Beer Inst., Inc., 491 U.S. 324, 336, 109 S.Ct. 2491, 2499, 105 L.Ed.2d 275. Pp. 450-454.\n3. The Act is invalid under the Commerce Clause because it discriminates against interstate commerce and Oklahoma has advanced no purposes to justify such discrimination. **(Crucial Sentence)** ...(About 5000 words)

***Analysis*:** These three statements form the backbone of the case: the cause lies in Oklahoma's protectionist legislation harming Wyoming's tax revenue → Wyoming has standing to sue, and the case is suitable for Supreme Court review → the legislation is indivisible and must be struck down in its entirety. Ultimately, the Supreme Court adopted most of the special master's recommendations, siding with Wyoming.

Figure 6: Case study on ECTHR-B and SCOTUS Dataset. The highlighted parts represent the compressed part generated by our proposed method.

Table 4: The performance of KSW and the Otsu algorithm is compared in terms of model effectiveness, GPU memory usage, and time cost.

Model	Charge		Law Article		Prison Term	
	Acc.	Ma-F	Acc.	Ma-F	Acc.	Ma-F
Qwen2.5-3B						
KAC *w/* Otsu	88.7	74.3	91.9	72.5	51.2	33.1
KAC *w/* KSW	87.2	75.6	92.1	71.5	50.7	32.7
Qwen2-7B						
KAC *w/* Otsu	91.3	80.1	93.4	80.2	64.5	44.5
KAC *w/* KSW	91.2	80.4	93.1	79.1	60.7	45.9



(a) GPU Memory(GB) (b) Time/Epoch(h)

Table 5: Performance comparison is conducted under different Top-M settings in the MLP. *Top-3* is the default in KAC, and $\triangle$DIF indicates the performance difference from *Top-3*.

Values	Acc	$\triangle DIF$	F1	$\triangle DIF$	Time	$\triangle DIF$
MultiLJP						
Top-1	64.3	**+9.2**	55.0	**+8.0**	0.60	**+0.07**
Top-2	70.3	**+3.2**	59.5	**+3.5**	0.64	**+0.03**
Top-3	73.5	-	63.0	-	0.67	-
Top-4	73.7	**-0.2**	62.7	**+0.3**	0.72	**-0.05**
Top-5	72.4	**+1.1**	63.4	**-0.4**	0.73	**-0.06**
Top-6	72.6	**+0.9**	63.5	**-0.5**	0.76	**-0.09**
Top-7	72.8	**+0.7**	63.1	**-0.1**	0.78	**-0.11**
Top-8	73.2	**+0.3**	62.1	**+0.9**	0.80	**-0.13**

plexity. Empirical results in Table 4 show that Otsu achieves similar performance while reducing GPU memory by 28% and speeding up processing by 60%, making it the more efficient choice.

Top-M Analysis We evaluate different Top-M settings in the MLP. As shown in Table 5, performance stabilizes when $M > 3$, while Top-1 and Top-2 are worse. Meanwhile, computational cost increases with larger M, supporting our choice.

4.6 Case Study

Figure 6 presents two case studies showing fact compression with KAC on the SCOTUS and ECTHR-B datasets. For ECTHR-B, KAC preserves the key facts: the deceased was labeled a terrorist, key evidence contradicted the alleged firefight, and authorities failed to investigate effectively, violating Article 2 of the European Convention on Human Rights. For SCOTUS, the compressed text highlights the case's core: Oklahoma's protectionist law harmed Wyoming's tax revenue, established standing, and violated the Commerce Clause by discriminating against interstate commerce. The law was deemed indivisible and struck down, with the Court largely following the special master's recommendations to rule in Wyoming's favor. These examples show that KAC effectively retains key legal content while removing redundancy.

5 Conclusion

We propose KAC, a knowledge-aware compression framework for legal judgment prediction. KAC leverages the relevance between case facts and legal knowledge, using adaptive thresholding to retain key content while reducing input length. Our analysis shows that maximizing mutual information between compressed text and legal knowledge reduces prediction loss. Experiments across jurisdictions and languages demonstrate that KAC improves prediction accuracy and efficiency.

Limitations

While the model demonstrates significant advantages in efficiency and performance, its practical application reveals that these advantages are primarily for long texts, with small benefits over existing models for short texts.

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities (Grant No. 63261068), the National Natural Sinence Foundation of China (Grant No. 72571150) and the Shenzhen Science and Technology Program No. SYSPG20241211173609009.

Ethical Statements

All datasets are sourced from open-access repositories and are used solely for research purposes. Due to current prediction accuracy and overall shortcomings, the model can serve as an research tool but is not recommended as a direct replacement for human experts in real courts.

References

Joshua Ainslie, Santiago Ontañón, Chris Alberti, Vaclav Cvicek, Zachary Fisher, Philip Pham, Anirudh Ravula, Sumit Sanghai, Qifan Wang, and Li Yang. 2020. ETC: Encoding long and structured inputs in transformers. In *EMNLP*, pages 268–284.

Saeed Amizadeh, Sara Abdali, Yinheng Li, and Kazuhito Koishida. 2025. Hierarchical self-attention: Generalizing neural attention mechanics to multi-scale problems. In *NIPS*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, and 29 others. 2023. Qwen technical report.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *Preprint*, arXiv:2004.05150.

Jesús Calleja, Thierry Etchegoyhen, and David Ponce. 2024. Automating easy read text segmentation. In *Findings of EMNLP 2024*, pages 11876–11894.

Ilias Chalkidis, Manos Fergadiotis, Dimitrios Tsarapatsanis, Nikolaos Aletras, Ion Androutsopoulos, and Prodromos Malakasiotis. 2021. Paragraph-level rationale extraction through regularization: A case study on European court of human rights cases. In *ACL 2021*, pages 226–241.

Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Katz, and Nikolaos Aletras. 2022a. LexGLUE: A benchmark dataset for legal language understanding in English. In *ACL 2022*, pages 4310–4330.

Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Martin Katz, and Nikolaos Aletras. 2022b. Lexglue: A benchmark dataset for legal language understanding in english. In *ACL 2022*.

Yapei Chang, Kyle Lo, Tanya Goyal, and Mohit Iyyer. 2024. Booookscore: A systematic exploration of book-length summarization in the era of llms. In *ICLR 2024*.

Shijing Chen, Mohamed Reda Bouadjenek, Usman Naseem, Basem Suleiman, Shoaib Jameel, Flora Salim, Hakim Hacid, and Imran Razzak. 2025. Leveraging taxonomy and LLMs for improved multimodal hierarchical classification. In *COLING 2025*, pages 6244–6254.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. *CoRR*, abs/1904.10509.

Yu-Neng Chuang, Tianwei Xing, Chia-Yuan Chang, Zirui Liu, Xun Chen, and Xia Hu. 2024. Learning to compress prompt in natural language formats. In *ACL 2024*, pages 7756–7767.

DeepSeek-AI. 2025. Deepseek-v3.2-exp: Boosting long-context efficiency with deepseek sparse attention.

Chenlong Deng, Kelong Mao, Yuyao Zhang, and Zhicheng Dou. 2024. Enabling discriminative reasoning in LLMs for legal judgment prediction. In *EMNLP*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186.

Daniele Giofré and Sneha Ghantasala. 2023. Legalhnet: Mixing legal long-context tokens with hartley transform. *Preprint*, arXiv:2311.05089.

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, Adam Chilton, K. Aditya, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel N. Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory M. Dickinson, Haggai Porat, Jason Hegland, and 21 others. 2023. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. In *NeurIPS*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. *Preprint*, arXiv:2006.03654.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2023. Large language models are reasoning teachers. In *ACL 2023*, pages 14852–14882.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *ICLR*.

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

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *ACL 2019*, pages 3651–3657.

Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2023. LLMLingua: Compressing prompts for accelerated inference of large language models. In *EMNLP 2023*, pages 13358–13376.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2024. LongLLMLingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. In *ACL 2024*, pages 1658–1677.

J.N. Kapur, P.K. Sahoo, and A.K.C. Wong. 1985. A new method for gray-level picture thresholding using the entropy of the histogram. *CVGIP*, 29(3):273–285.

Daniel Martin Katz, Michael J. Bommarito, II, and Josh Blackman. 2017. A general approach for predicting the behavior of the supreme court of the united states. *PLOS ONE*, 12(4):1–18.

Strahinja Klem and Noura Al Moubayed. 2025. Llms for llms: A structured prompting methodology for long legal documents. *Preprint*, arXiv:2509.02241.

Kangil Lee, Jinwoo Jang, Youngjin Lim, and Minsu Shin. 2025. Chain of knowledge graph: Information-preserving multi-document summarization for noisy documents. In *Proc. of NeusymBridge @ COLING 2025*, pages 1–5.

James Lee-Thorp, Joshua Ainslie, Ilya Eckstein, and Santiago Ontañón. 2022. FNet: Mixing tokens with Fourier transforms. In *NAACL-HLT 2022*, pages 4296–4313.

Jinpeng Li, Hang Yu, Ziqi Ma, and Peng Qi. 2026. Discovering latent facts from context to construct richer open knowledge graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(23):19233–19241.

Pengbo Li, Hang Yu, and Xiangfeng Luo. 2025. Context-aware graph neural network for graph-based fraud detection with extremely limited labels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 12112–12120.

Ziguang Li, Chao Huang, Xuliang Wang, Haibo Hu, Cole Wyeth, Dongbo Bu, Quan Yu, Wen Gao, Xingwu Liu, and Ming Li. 2024. Lossless data compression by large models. *NAT MACH INTELL*, 7:794 – 799.

Nayu Liu, Junnan Zhu, Yiming Ma, Zhicong Lu, Wenlei Xu, Yong Yang, Jiang Zhong, and Kaiwen Wei. 2025a. SARA: Salience-aware reinforced adaptive decoding for large language models in abstractive summarization. In *ACL 2025*.

Qian Liu, Rui Mao, Xiubo Geng, and Erik Cambria. 2023. Semantic matching in machine reading comprehension: An empirical study. *Information Processing & Management*, 60(2):103145.

Qian Liu, Hang Yu, Qiqi Wang, Qi Xu, Jinpeng Li, Zhuoqun Zou, Rui Mao, and Erik Cambria. 2025b. Legal knowledge infusion for large language models: A survey. *Information Fusion*, page 103426.

Fanghao Lou, Qiqi Wang, Guanyu Chen, Kaiqi Zhao, and Huijia Li. 2026. Zipljp: Zipped information processor for legal judgment prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 32302–32310.

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. In *EMNLP 2023*, pages 2198–2209.

Aishik Nagar, Yutong Liu, Andy T. Liu, Viktor Schlegel, Vijay Prakash Dwivedi, Arun-Kumar Kaliya-Perumal, Guna Pratheep Kalanchiam, Yili Tang, and Robby T. Tan. 2025. uMedSum: A unified framework for clinical abstractive summarization. In *ACL 2025*, pages 2653–2672.

Shubham Kumar Nigam, Deepak Patnaik Balaramamahanthi, Shivam Mishra, Noel Shallum, Kripabandhu Ghosh, and Arnab Bhattacharya. 2025. NyayaAnumana and INLegalLlama: The largest Indian legal judgment prediction dataset and specialized language model for enhanced decision analysis. In *COLING 2025*, pages 11135–11160.

Joel Niklaus, Ilias Chalkidis, and Matthias Stürmer. 2021. Swiss-judgment-prediction: A multilingual legal judgment prediction benchmark. In *NLLP 2021*.

OpenAI, Josh Achiam, and Steven Adler et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Nobuyuki Otsu. 1979. A threshold selection method from gray-level histograms. *IEEE SMC*, 9(1):62–66.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Victor Ruhle, Yuqing Yang, Chin-Yew Lin, H. Vicky Zhao, Lili Qiu, and Dongmei Zhang. 2024. LLMLingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. In *Findings of ACL 2024*, pages 963–981.

Raghavendra Pappagari, Piotr Zelasko, Jesús Villalba, Yishay Carmiel, and Najim Dehak. 2019. Hierarchical transformers for long document classification. *CoRR*, abs/1910.10781.

Nishchal Prasad, Mohand Boughanem, and Taoufiq Dkaki. 2024. Exploring large language models and hierarchical frameworks for classification of large unstructured legal documents. In *Adv Inf Retr*, pages 221–237.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Christophe Rodrigues, Marius Ortega, Aurélien Bossard, and Nédra Mellouli. 2025. Redire: Extreme reduction dimension for extractive summarization. *DATA KNOWL ENG*, 157:102407.

Jeffrey A. Segal. 1984. Predicting supreme court cases probabilistically: The search and seizure cases, 1962-1981. *Am. Polit. Sci. Rev.*, 78(4):891–900.

Oscar Skean, Md Rifat Arefin, Dan Zhao, Niket Nikul Patel, Jalal Naghiyev, Yann Lecun, and Ravid Shwartz-Ziv. 2025. Layer by layer: Uncovering hidden representations in language models. In *ICML 2025*, volume 267, pages 55854–55875.

Youwei Song, Jiahai Wang, Zhiwei Liang, Zhiyue Liu, and Tao Jiang. 2020. Utilizing BERT intermediate layers for aspect based sentiment analysis and natural language inference. *CoRR*, abs/2002.04815.

Naftali Tishby, Fernando C. Pereira, and William Bialek. 2000. The information bottleneck method.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Hulin Wang, Donglin Yang, Yaqi Xia, Zheng Zhang, Qigang Wang, Jianping Fan, Xiaobo Zhou, and Dazhao Cheng. 2024a. Raptor-t: A fused and memory-efficient sparse transformer for long and variable-length sequences. *IEEE Transactions on Computers*, 73(7):1852–1865.

Qiqi Wang, Ruofan Wang, Kaiqi Zhao, Robert Amor, Benjamin Liu, Jiamou Liu, Xianda Zheng, and Zijian Huang. 2024b. SKGSum: Structured knowledge-guided document summarization. In *Findings of ACL 2024*, pages 1857–1871.

Chaojun Xiao, Xueyu Hu, Zhiyuan Liu, Cunchao Tu, and Maosong Sun. 2021. Lawformer: A pre-trained language model for chinese legal long documents. *AI Open*, 2:79–84.

Nuo Xu, Pinghui Wang, Long Chen, Li Pan, Xiaoyan Wang, and Junzhou Zhao. 2020. Distinguish confusing law articles for legal judgment prediction. In *ACL*, pages 3086–3095.

Nuo Xu, Pinghui Wang, Junzhou Zhao, Feiyang Sun, Lin Lan, Jing Tao, Li Pan, and Xiaohong Guan. 2024. Distinguish confusion in legal judgment prediction via revised relation knowledge. 43(1).

Qi Xu, Qian Liu, Hao Fei, Hang Yu, Shuhao Guan, and Xiao Wei. 2025. Clear: A framework enabling large language models to discern confusing legal paragraphs. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 8937–8953.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Junjie Yang and Hai Zhao. 2019. Deepening hidden representations from pre-trained language models for natural language understanding. *CoRR*, abs/1911.01940.

Shuo Yang, Ying Sheng, Joseph E. Gonzalez, Ion Stoica, and Lianmin Zheng. 2024. Post-training sparse attention with double sparsity. *Preprint*, arXiv:2408.07092.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard H. Hovy. 2016. Hierarchical attention networks for document classification. In *NAACL*.

Jingyang Yuan, Huazuo Gao, Damai Dai, Junyu Luo, Liang Zhao, Zhengyan Zhang, Zhenda Xie, Yuxing Wei, Lean Wang, Zhiping Xiao, Yuqing Wang, Chong Ruan, Ming Zhang, Wenfeng Liang, and Wangding Zeng. 2025. Native sparse attention: Hardware-aligned and natively trainable sparse attention. In *ACL 2025*, pages 23078–23097.

Linan Yue, Qi Liu, Binbin Jin, Han Wu, Kai Zhang, Yanqing An, Mingyue Cheng, Biao Yin, and Dayong Wu. 2021. Neurjudge: A circumstance-aware neural framework for legal judgment prediction. In *SIGIR*, page 973–982.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. Big bird: Transformers for longer sequences. In *NeurIPS*, volume 33, pages 17283–17297.

Qingru Zhang, Dhananjay Ram, Cole Hawkins, Sheng Zha, and Tuo Zhao. Efficient long-range transformers: You need to attend more, but not necessarily at every layer. In *Findings of EMNLP 2023*, pages 2775–2786.

Xingxing Zhang, Furu Wei, and Ming Zhou. 2019. HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization. In *ACL 2019*, pages 5059–5069.

Yu Zhong, Bo Shen, and Tao Wang. 2024. Tgin: Document-level event extraction with two-phase graph inference network. *Neural Netw.*, 176:106343. Epub 2024 Apr 25. PMID: 38701598.

Yilun Zhou, Marco Túlio Ribeiro, and Julie Shah. 2022. Exsum: From local explanations to model understanding. In *NAACL-HLT*, pages 5359–5378.

Zhi Zhou, Kun-Yang Yu, Shi-Yu Tian, Xiao-Wen Yang, Jiang-Xin Shi, Pengxiao Song, Yi-Xuan Jin, Lan-Zhe Guo, and Yu-Feng Li. 2025. Lawgpt: Knowledge-guided data generation and its application to legal llm. *Preprint*, arXiv:2502.06572.

Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. A robustly optimized BERT pre-training approach with post-training. In *CCL 2021*, pages 1218–1227.

Octavia-Maria Șulea, Marcos Zampieri, Shervin Malmasi, Mihaela Vela, Liviu Dinu, and Josef Genabith. 2017. Exploring the use of text classification in the legal domain.

A Mutual information calculation

The mutual information between a semantic unit s_i and a topic t_j is:

$$I(s_i;t_j) = P(s_i,t_j)\log_2\frac{P(s_i,t_j)}{P(s_i)P(t_j)}, \quad (10)$$

where the logarithm is base 2, and the unit is bits. And we maximize mutual information for a single semantic unit:

$$I_{\max}(s_i;T) = \max_{j\in\{1,\dots,K\}} I(s_i;t_j), \quad (11)$$

For a text subset $D' = \{s_{i_1}, s_{i_2}, ..., s_{i_M}\} \subseteq D$, under the conditional independence assumption, the cumulative mutual information is:

$$I_{\text{cum}}(D';T) = \sum_{m=1}^{M} I_{\max}(s_{i_m};T). \quad (12)$$

More generally, it is the sum of the mutual information between each unit and its most relevant topic:

$$I_{\text{cum}}(D';T) = \sum_{s\in D'} \max_{t\in T} I(s;t). \quad (13)$$

B Automatic Thresholding Methods Comparisons

The KSW algorithm maximizes the information entropy. Therefore, we aim to prove the asymptotic equivalence between the Otsu algorithm and the KSW algorithm.

Theorem 1 (Asymptotic Equivalence of Otsu and KSW)**.** *Assume the normalized similarity scores $\{\hat{r}_i\}_{i=1}^N$ are i.i.d. samples from a normal distribution $\mathcal{N}(\mu,\sigma^2)$. Define:*

$$\tau^*_{KSW} = \arg\min_{\tau} H(T|D'_\tau), \quad (14)$$

$$\tau^*_{Otsu} = \arg\max_{\tau} \sigma^2_{between}(\tau), \quad (15)$$

where $\sigma^2_{between}(\tau) = p_1(\tau)p_2(\tau)[\mu_1(\tau)-\mu_2(\tau)]^2$ is Otsu's between-class variance.

As $N\to\infty$, the following hold:

(i). Threshold Convergence: $\tau^*_{Otsu} \xrightarrow{P} \tau^*_{KSW}$, i.e., both thresholds converge in probability to the same theoretical value.

(ii). Objective Equivalence: Minimizing conditional entropy $H(T|D'_\tau)$ is equivalent to maximizing between-class variance $\sigma^2_{between}(\tau)$ under the population distribution.

(iii). Computational Advantage: Otsu's algorithm computes the optimal threshold in $O(N)$, while KSW requires $O(N\log N)$ due to sorting.

Proof. **(i). Objective Equivalence**

For $\hat{r}\sim\mathcal{N}(\mu,\sigma^2)$, thresholding at τ splits the population into two classes:

$$C_1 : \hat{r} < \tau, \quad C_2 : \hat{r} \geq \tau. \quad (16)$$

Let the proportions of the two classes be $p_1 = \Phi\left(\frac{\tau-\mu}{\sigma}\right)$ and $p_2 = 1 - p_1$, where Φ is the standard normal CDF.

The conditional entropy $H(T|D'_\tau)$ reflects the uncertainty in predicting T using only samples with $\hat{r} \geq \tau$. For normal distributions, the entropy of each class is determined by its variance:

$$H(T|D'_\tau) = \frac{1}{2} \ln\left(p_1\sigma_1^2 + p_2\sigma_2^2\right) + \text{constant}. \tag{17}$$

Using the total variance decomposition:

$$\sigma^2 = p_1\sigma_1^2 + p_2\sigma_2^2 + p_1p_2(\mu_1 - \mu_2)^2, \tag{18}$$

minimizing $H(T|D'_\tau)$ is equivalent to maximizing $\sigma^2_{\text{between}}(\tau)$.

(ii).Threshold Convergence

Let the theoretical optimal threshold be:

$$\tau^* = \arg\min_\tau \sigma^2_{\text{within}}(\tau). \tag{19}$$

By the law of large numbers, the sample-based within-class variance $\hat{\sigma}^2_{\text{within}}(\tau)$ converges to $\sigma^2_{\text{within}}(\tau)$. Since the objective is strictly convex, the sample-based thresholds $\hat{\tau}^*_{\text{Otsu}}$ and $\hat{\tau}^*_{\text{KSW}}$ converge to the same limit.

(iii).Computational Complexity

Otsu's Algorithm The between-class variance $\sigma^2_{\text{between}}(\tau)$ can be computed in $O(N)$ using cumulative statistics (e.g., sums and squared sums).

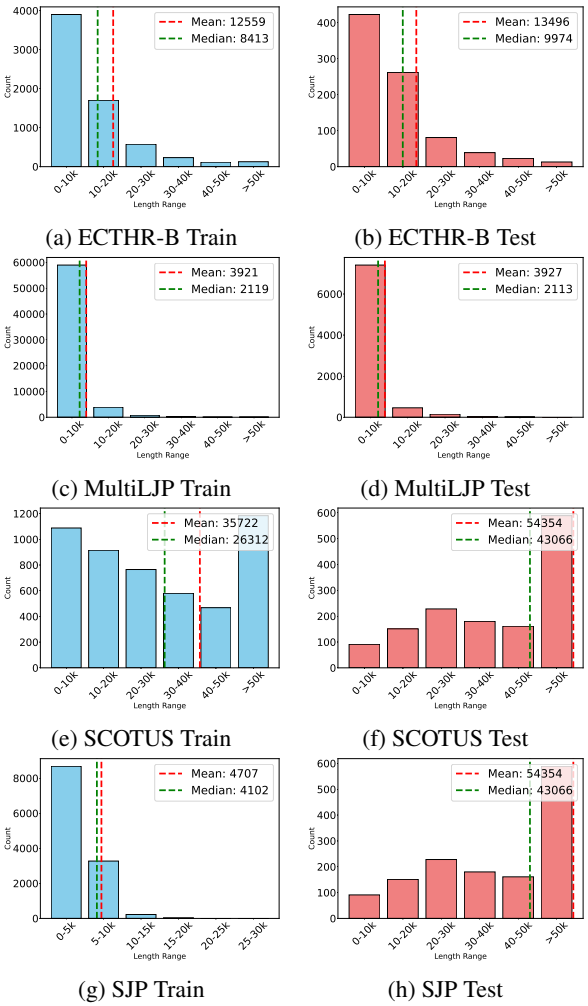
KSW Algorithm Computing $H(T|D'_\tau)$ involves sorting the scores ($O(N \log N)$) and estimating probabilities ($O(N)$). Thus, the total complexity is $O(N \log N)$.

This demonstrates Otsu's computational advantage. □

Table 4 analyzes the effects and time costs of using KSW and Otsu on the MultiLJP dataset, further validating Theorem 1.

C More Details of Experiments Setup

Datasets We conducted experiments on four long legal judgment prediction datasets across different languages and jurisdictions, including Chinese, English, and German. For the Chinese dataset, we used MultiLJP (Lyu et al., 2023), which involves predicting charges, laws, and sentences based on criminal facts. For the English datasets, we used SCOTUS (Chalkidis et al., 2022b) and ECTHR-B (Chalkidis et al., 2021). SCOTUS is derived from the U.S. Supreme Court, where the task is to predict the relevant issue areas given a court opinion document. ECTHR-B is derived from specific cases of the European Court of Human Rights (ECHR). The dataset provides a series of factual paragraphs from case descriptions. Each case is mapped to specific articles of the ECHR that are allegedly violated. Swiss-Judgment-Prediction (SJP) (Niklaus et al., 2021) is a multilingual, diachronic dataset from the Swiss Federal Supreme Court, annotated with binarized judgment outcomes, making it a challenging text classification task. We selected the German subset of this dataset. Furthermore, to evaluate the model's ability to handle long texts, we removed samples with text lengths shorter than 3000 from the ECTHR-B and SJP datasets. As shown in Fig 7, we grouped and counted the number of samples with different lengths across four datasets.



(a) ECTHR-B Train (b) ECTHR-B Test

(c) MultiLJP Train (d) MultiLJP Test

(e) SCOTUS Train (f) SCOTUS Test

(g) SJP Train (h) SJP Test

Figure 7: Length Statistics Across Different Datasets

Evaluation Metrics We used Accuracy (Acc.), Macro Precision (Ma-P), Macro Recall (Ma-R),

and Macro F1 score (Ma-F) as metrics to assess the performance of the baseline and the model.

Baselines We primarily evaluated four types of models: traditional legal judgment prediction models, transformer-based pre-trained language models and their improved versions, and LLMs.

Traditional Models For traditional legal judgment prediction models, we selected two representative methods. LADAN (Xu et al., 2020) introduces a specialized graph distillation mechanism to differentiate semantically similar legal provisions. NeurJudge (Yue et al., 2021) employs a hierarchical representation learning strategy to handle textual case facts across various subtasks. For hierarchical classification frameworks, we base our work on HRN (Lyu et al., 2023), HNet, and FNet (Giofré and Ghantasala, 2023). HRN is a reasoning framework designed to address specific legal tasks through logical hierarchies. In contrast, HNet and FNet construct general-purpose long-text encoders using Fourier and Hartley transforms.

Pre-trained Language Models BERT (Devlin et al., 2019) is a bidirectional encoder based on Transformer, pre-trained using masked language modeling. RoBERTa (Zhuang et al., 2021) improves upon BERT by leveraging larger datasets, longer training times, and a dynamic masking strategy, significantly enhancing performance. DeBERTa (He et al., 2021) further introduces a disentangled attention mechanism and an improved masked decoder, enabling more precise context modeling. Transformer variants, including BigBird (Zaheer et al., 2020) and Longformer (Beltagy et al., 2020), incorporate sparse attention mechanisms to reduce computational complexity, making them effective for processing long document-level inputs.

LLMs Without compressing the text, we fine-tuned LLMs with different parameters and versions. For LLMs, we selected models of varying sizes and training corpora, including LLaMA3.2-3B, LLaMA3-8B (Touvron et al., 2023), Qwen2.5-3B (Qwen et al., 2025), Qwen2-7B (Bai et al., 2023), Qwen3-14B (Yang et al., 2025), and the legal domain-specific fine-tuned model LawyerLlama (Huang et al., 2023). Previous work fine-tuned the ADAPT framework (Deng et al., 2024) on Qwen2-7B, which we also included for comparison.

D Generalization Ability Analysis

KAC demonstrates strong generalization capabilities across languages and legal jurisdictions, making it adaptable to various legal tasks in different jurisdictions and languages. Its knowledge-guided compression mechanism is highly transferable, making it flexible to various legal domains and suitable for diverse legal tasks in the future. We will demonstrate this property through theoretical analysis. We assume that the structure of legal knowledge across different jurisdictions or tasks exhibits a certain degree of implicit alignment. This means that, despite differences in surface expressions, the underlying legal logic, reasoning patterns, and key fact types share significant similarities. KAC achieves cross-domain generalization through the following mechanisms:

D.1 Unified Knowledge Representation

KAC leverages a single LLM to embed legal knowledge from different jurisdictions into a unified semantic space. Let the source jurisdiction knowledge set be $\mathcal{K}_S$ and the target jurisdiction knowledge set be $\mathcal{K}_T$. Their embeddings are represented as:

$$\mathbf{e}_{\mathcal{K}_S} = \mathcal{M}_{LLM}(\mathcal{K}_S), \quad \mathbf{e}_{\mathcal{K}_T} = \mathcal{M}_{LLM}(\mathcal{K}_T). \tag{20}$$

Since pre-trained LLMs possess multilingual and multi-domain understanding capabilities, knowledge embeddings from different jurisdictions often exhibit cluster consistency in the vector space. This means that similar legal concepts across jurisdictions tend to have close representations, forming the basis for cross-jurisdictional knowledge transfer.

D.2 Adaptive Threshold Mechanisms

KAC uses an adaptive thresholding method that does not rely on task-specific similarity distribution assumptions. Instead, it automatically determines the boundary between retention and discard by unsupervised analysis of the bimodal distribution of relevance scores. This process is task-agnostic, making it effective in distinguishing legally relevant from legally irrelevant content across different legal tasks without the need to adjust the threshold for each task.

D.3 Maximizing Mutual Information

We further formalize the goal of KAC as a cross-domain mutual information maximization problem:

Let $D^{(i)}$ represent the documents from the i-th legal domain or task, and $T^{(i)}$ denote the corresponding legal knowledge topics. KAC compresses $D^{(i)}$ into $D'^{(i)}$, with the objective:

$$\max I\left(D'^{(i)}; T^{(i)}\right) - \beta \cdot \mathcal{D}\left(P_{D'^{(i)}} \| Q_{D'}\right). \quad (21)$$

The second term is a domain discrepancy penalty, which encourages the distribution $P_{D'^{(i)}}$ of the compressed representation $D'^{(i)}$ to align with a shared cross-domain distribution $Q_{D'}$. This ensures that compressed representations from different legal domains remain consistent in the representation space, facilitating the transfer of reasoning abilities across tasks.

E Validation of Distributional Assumptions

As shown in Table 6, to verify the bimodal nature of similarity scores, we randomly sampled 1,000 instances from the test set of each dataset and calculated the **Bimodality Coefficient (BC)** for each sample. The BC metric measures the degree of bimodality in a distribution, calculated as follows:

$$BC = \frac{m_3^2 + 1}{m_4 + 3 \times \frac{(n-1)^2}{(n-2)(n-3)}} \quad (22)$$

where m_3 represents **skewness**, which measures the asymmetry of the distribution; m_4 represents **kurtosis**, which measures the tail thickness; is the sample size.

The interpretation is as follows:

- **BC** $> 5/9$ (≈ 0.555): indicates significant bimodality;
- **BC** $= 1$: indicates a perfect bimodal distribution;
- **BC** ≈ 0: indicates a uniform distribution.

Based on the BC coefficient, we set the threshold at **0.556**. Values above this threshold indicate a bimodal distribution, while values below do not. The results are shown below.

As shown in Table 7, of all datasets, MultiLJP has the lowest proportion of bimodal distributions. We observed that this is because many of its prediction targets are semantically similar, resulting in more closely clustered relevance scores. As the dataset with the lowest bimodality, we selected MultiLJP to further analyze our model's performance in such challenging scenarios.

Specifically, we selected the subset of test cases from MultiLJP where the Bimodality Coefficient (BC) is less than 0.56. We then tested various Top-K% truncation baselines against our KAC model on this subset.

Table 6: Statistics of data proportions with BC > 0.56 versus BC < 0.56 across datasets (sample size $n = 1000$). A BC > 0.56 indicates that the distribution is bimodal.

Dataset	BC > 0.55	BC < 0.55
MultiLJP	0.67	0.23
SCOTUS	0.81	0.19
ECTHR-B	0.78	0.22
SJP	0.87	0.13

Table 7: Performance on MultiLJP test cases with BC < 0.56, using Qwen2.5-3B as the backbone. We compare our KAC model with different Top-K% baselines, where Top-K% denotes selecting tokens with the top K percent of relevance scores.

Model	Charge		Persion Term	
	Acc.	Ma-F	Acc.	Ma-F
Qwen2.5-3B				
Top 10%	75.3	56.1	44.3	27.9
Top 20%	81.6	64.2	47.2	28.8
Top 60%	84.7	69.9	51.2	30.1
Top 100%	86.7	70.5	50.7	31.1
KAC (Ours)	**88.7**	**71.2**	**52.1**	**33.2**

F Compressed Text Evaluation

F.1 From a Reference-free Perspective

We evaluate the quality of our compressed text from a reference-free perspective. As shown in table 8, we use two metrics, Trustworthiness (Tru.) and Continuity (Con.), to measure the quality of the compressed text generated by the model. Specifically, we use the SentenceTransformer model to encode the original text and the compressed text into vectors. These metrics are calculated by comparing the k-nearest neighbor relationships in the vector spaces of the original and compressed texts. Trustworthiness checks whether the neighbors in the compressed space are truly neighbors in the original space, while Continuity checks whether

Table 8: Trustworthiness (Tru.) and Continuity (Con.) scores, prediction accuracy, and time consumption of different models. KAC is tested on MultiLJP using Qwen2-7B as the base model.

Model	Tru.	Con.	Charge		Prison Term		Time Cost
			Acc.	Ma-F	Acc.	Ma-F	
Qwen2-7B							
w/ Original Text	–	–	90.4	72.1	54.8	31.3	5.1
w/ LLMLingua	0.81	0.79	82.5	68.3	47.2	37.4	5.5
w/ LongLLMLingua	0.80	0.81	84.3	71.3	51.7	37.2	5.7
w/ LLMLingua-2	0.83	0.82	87.3	74.4	50.1	40.1	5.1
w/ ExSum	0.88	0.85	88.9	62.6	58.7	47.8	2.8
w/ Qwen-Plus	0.91	**0.98**	90.6	75.8	57.3	50.7	21.9
w/ KAC (Ours)	**0.97**	0.91	**91.3**	**80.1**	**64.5**	**55.0**	**2.5**

Table 9: Comparison of evaluation scores by the LLMs.

Model	Consistency	Coherence
Qwen2-7B	–	–
w/ LLMLingua	0.77	0.84
w/ LongLLMLingua	0.76	0.82
w/ LLMLingua-2	0.81	0.79
w/ ExSum	0.85	0.85
w/ Qwen-Plus	0.92	**0.98**
w/ KAC (Ours)	**0.92**	0.94

the neighbors in the original space are preserved in the compressed space by swapping the reference space.

Both metrics range from $[0, 1]$, and higher values indicate better quality of the compressed text. For Trustworthiness, a higher value means the compressed text is more similar to the original text. For Continuity, a higher value indicates better continuity in the compressed text. We retain the original comparison table of compression methods and add these two metrics to better demonstrate the advantages of our model.

F.2 Evaluation by LLMs

As shown in Table 9, to evaluate the quality of compressed text more accurately, we use the large language model DeepSeek-V3.2 via its API to assess the quality of compressed text generated by different models. The evaluation measures the consistency between the compressed text and the original text, as well as the coherence of the compressed text, which reflects its quality. The method involves inputting the original text and the compressed text into the model, which assigns a score between $[0, 1]$. Higher scores indicate better performance.

Table 10: The prediction accuracy of the MLP module and the downstream task performance under different Top-M settings, as well as the downstream task performance with disrupted knowledge.

Values	MLP		KAC	
	Acc.	F1	Acc.	F1
Top-1	55.1	42.7	64.3	55.0
Top-2	72.5	58.8	70.3	59.5
Top-4	96.3	88.1	73.7	62.7
Top-5	98.2	95.6	72.4	63.4
Top-6	99.1	97.3	72.6	63.5
Top-7	99.8	98.4	72.8	63.1
Top-8	**99.8**	**99.5**	73.2	62.1
KAC(Top-3)	95.4	85.7	**73.5**	**63.0**

Our prompt is: "Please score the compressed text based on the original text using two metrics. First, Consistency, which measures how similar the compressed text is to the original. The score ranges from 0 to 1, and a higher score means greater similarity. Second, Coherence, which measures the semantic fluency of the compressed text. The score ranges from 0 to 1, and a higher score means better fluency."

The final result is obtained by averaging the scores of all samples, as shown below:

G Top-K Analysis

We conducted a preliminary analysis of the selection of Top-M, exploring how the number of candidate legal articles affects the final performance. We further provide the prediction accuracy of the MLP module under different Top-M settings. For

the subset of the dataset where the MLP still fails after selecting the Top-3, we test both KAC and the fine-tuned model.

In Table 10, the first two columns show the probability that the correct label is among the Top-M candidate labels predicted by the MLP. In other words, if the correct label is included in the Top-M candidates, we consider the prediction correct and calculate Acc. and F1. The last two columns of experiments show the model performance of LLMs in legal judgment prediction, where the Top-M candidate labels are used as compressed knowledge text.

H Effectiveness of Legal Knowledge

Our knowledge is obtained through the relevant legal provisions. We treat the labels in the training set as keywords and find the official legal interpretations for them. To demonstrate that our constructed knowledge significantly outperforms using keywords alone, we conduct the experiments in Table 11:

- **Title-Only**: Uses only labels for similarity calculation.
- **No Keywords**: Uses only legal interpretations for similarity calculation.
- **Shuffled**: Labels and legal interpretations are randomly paired.

Table 11: Ablation studies on the effect of legal knowledge design on MultiLJP dataset.

Setting	Ratio (%)↓	Charge		Prison Term	
		Acc.	F1	Acc.	F1
KAC (Ours)	**15.8**	**91.3**	**80.1**	**64.5**	**44.2**
A: Title-Only	20.9	89.1	76.5	58.3	38.7
B: No Keywords	15.9	91.2	79.9	64.1	42.5
C: Shuffled	26.8	82.4	65.3	48.2	30.1

It can be seen that adding labels and their legal interpretations significantly improves model performance. This shows that legal interpretations, as knowledge, effectively help filter relevant information.