JurisNexus🤖👨‍⚖️: Enhancing Legal Judgment Prediction via Cross-Reasoning-Chain Representation Learning Mechanism

Pengjie Liu[1], Xiaoqing Zhang[2], Yulong Ding[1], and Shuang-Hua Yang[1,3(✉)]

[1] School of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China
{liupj2020, dingyl, yangsh}@mail.sustech.edu.cn
[2] Center for High Performance Computing and Shenzhen Key Laboratory of Intelligent Bioinformatics, Shenzhen institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China
xq.zhang2@siat.ac.cn
[3] Department of Computer Science, University of Reading, UK

Abstract. Legal Judgment Prediction (LJP) methods seek to generate judgments automatically by analyzing criminal cases. Current LJP techniques typically design and learn explainable handcrafted features hidden in the fact description. However, identifying semantic legal indicators in confusing crimes, e.g., fraud and robbery criminal charges, remains challenging. Inspired by the jury systems, an equitable judicial process should integrate the jury's non-legal-domain factual inference among similar criminal cases, with the veteran judges' professional legal knowledge for grounded reasoning. Thus, we propose a legal representation learning framework (JurisNexus), which designs a novel cross-reasoning-chain mechanism to establish a specialized semantic space for better performance. Our cross-reasoning-chain mechanism involves: 1) Similar Case Reasoning Chain, which makes judgments based on analogies with similar precedents; 2) Legal Judgment Reasoning Chain, which can enhance the judicial association between criminal cases and legal judgment generation. Our pre-training strategy boosts JurisNexus's accuracy by 4.6% on average, reflecting notable advances across various criminal scenarios. Experimental results indicate that JurisNexus effectively learns uniform and discriminative fact description representations, resulting in more precise predictions for confusing crimes and a significant minimize in uncertainty. The implementation details will be open on GitHub.

Keywords: Legal Judgment Prediction · Representation Learning · Contrastive Learning · Continuous Training · Data Mining.

1 Introduction

Legal Judgment Prediction (LJP) is an important research topic in LegalAI [13], serving as a pivotal role in assisting judgical system and promoting legal awareness [26,17]. LJP is designed to forecast the verdicts of criminal cases, thereby

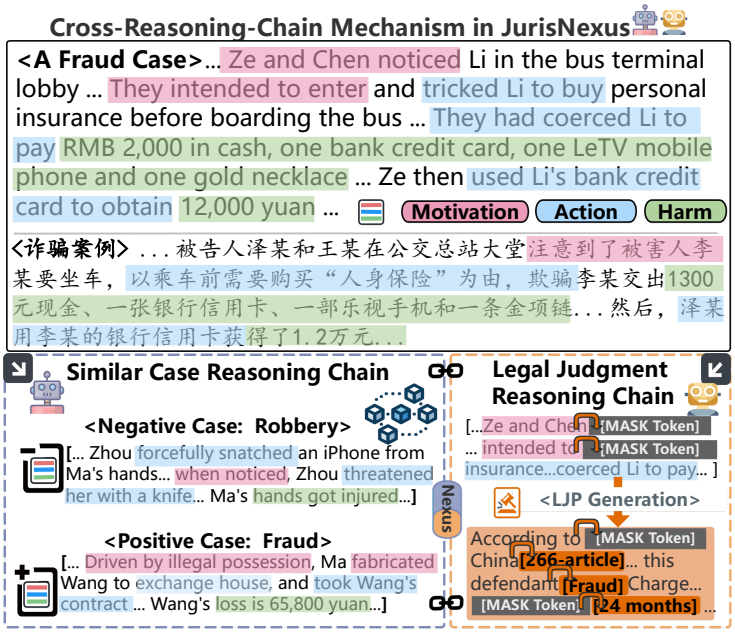


Fig. 1: An example of Cross-Reasoning-Chain (CRC) mechanism in a fraud case.

enhancing the efficiency and fairness of judicial processes. To promote fairness in judicial processing and achieve higher prediction accuracy, modern LJP approaches increasingly focus on minimizing bias during criminal fact representation learning [4]. They typically acquire effective representations of criminal fact description by utilizing task-specific crafted words or legal attributes [22,19,9] or by leveraging related LJP sub-task prediction [10,38,31,37]. In recent years, researchers have increasingly focused on Pre-training Language Models (PLM) using legal-domain corpora, thereby enabling the automatic extraction of semantic features from legal clues rather than relying on manual trigger detection [16,27,23,39]. They typically employ mask language modeling [5,37] to learn fine-grained semantic knowledge, rather than developing more comprehensive representations of the long document. Nevertheless, existing LJP methods tend to focus on token-level legal clues while overlooking the associations and nuanced differences among similar cases, as well as the in-depth logical analysis and judicial processing.

In this paper, we draw inspiration from the systematic referencing of similar case judgments by juries as a basis for judicial processing, combined with the legal knowledge endorsement of professional judges through cross-reasoning chains. Thus we introduce JurisNexus to continuously pre-train language models for better legal representation learning of legal document and criminal cases. Figure 1 illustrates that our model employs a cross-reasoning-chain mechanism to pre-train language models for developing a more customized semantic space using *Similar*

Case Reasoning (SCR) module and *Legal Judgement Reasoning* (LJR) module. Specifically, our SCR module emulates the jury by retrieving and analyzing similar precedents. It can contrastively train language models to differentiate criminal facts between various judicial outcomes and separate confusing criminal charges by recognizing legal semantic features in the criminal action and harm, such as employing the "forcefully snatched" and "fabricated" to distinguish the criminal charge of robbery and fraud. Then the LJR module serves as a qualified judge to further mine the criminal characteristics hidden in the fact description and applies the legal knowledge to generate the potential judgment results. This judgment generation module initiates its process by focusing on the given crime case. It prioritizes identifying and extracting legal clues from the detailed description of the crime facts. Subsequently, it generates legal judgments based on these identified legal clues. Through this process, the language model is trained to pinpoint critical legal elements within the crime facts, match the semantic context between the verdict and the crime facts, and establish a logical connection between the legal provisions and the criminal charge definition.

Our experimental results reveal that JurisNexus substantially outperforms existing LJP methods, yielding an average gain of more than 4. 6% on the CAIL-2018 dataset [28]. Meanwhile, JurisNexus enhances the representation of criminal facts through the cross-reasoning-chain mechanism, which constructs and reasons within a legal domain semantic space to systematically encode case fact and legal knowledge. Our model achieves outstanding performance in handling both low-frequency and ambiguous criminal charges, highlighting its effectiveness in few-shot learning and fine-grained reasoning. Besides, JurisNexus can significantly reduce uncertainty in judicial processing by providing more distinct and uniform semantic representation of fact description and their corresponding verdicts.

2 Related Work

Earlier LegalAI models are typically designed to identify critical legal attributes within case materials pertaining to criminal offenses, with the objective of assisting judicial assessments. However, the performance of feature-based Legal Judgment Prediction (LJP) methods is constrained by the quality of legal-specific trigger words and the paradigms of relevant judgment documents. As a result, these statistical LJP models have consistently struggled to accurately identify infrequent or ambiguous criminal charges [2,17]. Unlike these feature-based LJP methods, recent LJP research utilizes various neural network architectures to automatically extract legal clues from criminal description. These architectures include LSTM [38,9,31,30], CNN [14,11], and Transformer models [5,39,33,15]. By leveraging the collected legal clues, these models predict violations of corresponding legal provisions, criminal charges, and terms of penalty respectively. Moreover, Inspired by the interactions observed in real-world legal trials, some knowledge graph based LJP models establish dependencies between subtasks to further improve performances [38,32,31,17]. The representation learning of legal knowledge plays a pivotal role in bridging the gap between textual descriptions

and semantic space embeddings in LJP models. This requires models to effectively understand and represent legal documents, including criminal fact descriptions, law article provisions, and legal judgment documents. In this context, more legal representation learning based models are leveraging Pre-trained Language Models (PLM) to encode legal texts for judgment prediction. These models typically employ masked language modeling to continue training the model, aiming to capture legal clues, understand criminal fact semantics, and provide more effective representations of legal documents [17,15]. Recent works have addressed this issue by introducing noise and constructing text pairs, using contrastive training to align these pairs [35,7]. To capture finer-grained semantics and represent legal texts for distinguishing differences between criminal cases, several contrastive learning based models offer promising approaches to learning sentence representations by introducing dropout noise directly into sentence embeddings for contrastive training. This enhances the sensitivity of PLM-based LJP methods in representation learning [6,17,34].

3 Methodology

This section illustrates the details of Cross-Reasoning-Chain (CRC) mechanism in our JurisNexus model. We first introduce the Legal Judgment Prediction (LJP) implementation (Section. 3.1). Subsequently, we present our cross-reasoning-chain framework (Section. 3.2), which continuously pre-trains language models to enhance the criminal representation for different LJP sub-tasks.

3.1 Preliminary of Legal Judgment Prediction

The LJP model focuses on generating verdicts for the query criminal case based on the provided criminal fact (F). As illustrated in Figure 2, JurisNexus addresses three LJP sub-tasks: Law **A**rticle prediction, **C**riminal charge prediction, and **I**mprisonment prediction. Specificlly, we first derive legal clues from criminal fact representation (H^F) by encoding criminal cases using the continuously trained language model. Then our model is optimized to predict each judgment result ($L_{I/C/A}$) utilising the multi-task loss function $\mathcal{L}_{\text{LJP}}$ as:

$$\mathcal{L}_{\text{LJP}} = \sum_{i \in I,C,A} \text{CrossEntropy}(L_i^*, P(L_i|H^F)), \tag{1}$$

where $L_{I/C/A}^*$ contains each ground truth of law articles, charges, or term of Penalty for the query case. Additionally, the $P(L_{I/C/A}|H^F)$ represents the probability of predicting the authentic judgment result for LJP verification, respectively.

$$P(L_{I/C/A}|H^F) = \text{softmax}(H_F), \tag{2}$$

$$H_F = \text{Linear}(\textbf{JurisNexus}(F)), \tag{3}$$

where **JurisNexus** model stands for the checkpoint saved after continuous pre-training on the legal corpus, and we add a dimensionality reduction layer to transform high-dimensional embeddings into low-dimensional space.

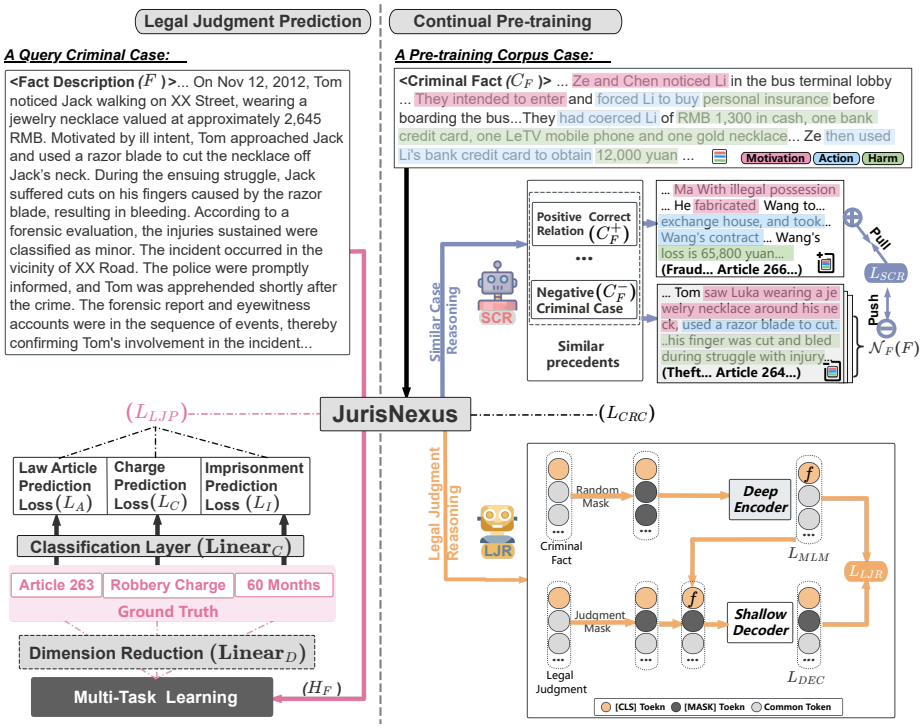


Fig. 2: Illustration of Legal Judgment Prediction (LJP) implementation and Cross-Reasoning-Chain (CRC) representation learning mechanism in the JurisNexus.

3.2 Legal Representation Continual Pre-training through Cross-Reasoning-Chain Mechanism

Our cross-reasoning-chain mechanism simulates the judicial process from two chains: Similar Case Reasoning (SCR) module and Legal Judgment Reasoning (LJR) module. As shown in Figure 2, we formulate the loss function $\mathcal{L}_{\text{CRC}}$ to pre-train language models for legal representation learning:

$$\mathcal{L}_{\text{CRC}} = \mathcal{L}_{\text{SCR}} + \mathcal{L}_{\text{LJR}}, \tag{4}$$

where $\mathcal{L}_{\text{SCR}}$ represents the loss of contrastive learning, which is derived from distinguishing different semantics in similar criminal cases. And $\mathcal{L}_{\text{LJR}}$ confusing cases criminal cases, and $\mathcal{L}_{\text{LJR}}$ is a professional process that can generate the legal judgments based on the given criminal cases, denotes a systematic process for making legal judgments by identifying and analyzing the legal clues.

Similar Case Reasoning. We simulate the jury system, which relies on similar case reasoning to identify the subtle difference among legal clues [1,36]. JurisNexus performs similar criminal case reasoning chain by referencing similar precedents,

as described in the criminal motivation, actions, and harm within the given case. Specifically, for a given criminal fact (C_F), SCR module contrastive training of pre-trained language models (PLM) using the loss function $\mathcal{L}_{\text{SCR}}$:

$$\mathcal{L}_{\text{SCR}} = -\log \frac{e^{sim(C_F, {C_F}^+)/\tau}}{e^{sim(C_F, {C_F}^+)/\tau} + \sum_{C_F^- \in \mathcal{N}_F(F)} e^{sim(C_F, {C_F}^-)/\tau}}, \tag{5}$$

where we adopt BM25 as the ranking function to select positive fact ${C_F}^+$ and negative facts ${C_F}^-$ with higher ranking scores. Specifically, ${C_F}^+$ refers to a criminal case with the same charges and law article results as C_F, while $\mathcal{N}_F(F)$ denotes the collection of negative facts ${C_F}^-$ assigned different charging or legal item labels. We set the temperature hyperparameter τ to adjust the diffusion level in the softmax function, and the similarity between two crime facts C_F and ${C_F}^{+/-}$ is computed using cosine similarity score as:

$$sim(C_F, C_F^{+/-}) = \cos(C_F, {C_F}^{+/-}). \tag{6}$$

Legal Judgment Reasoning. Unlike juries, professional judges usually concentrate on extracting critical legal elements from the case itself to form their rulings [3,18,36,24]. In this case, JurisNexus builds a legal judgment reasoning chain for finding the judgment triggers behind the fact description and the association with their legal judgments:

$$\mathcal{L}_{\text{LJR}} = \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{DEC}}, \tag{7}$$

where $\mathcal{L}_{\text{MLM}}$ and $\mathcal{L}_{\text{DEC}}$ are the loss functions from fact's mask language language modeling loss and judgment decoder loss. Firstly, we randomly replace some tokens of the fact with a special token [MASK]. Following the previous work [5,15], we feed the masked criminal fact into the encoder to recover masked fact tokens $m(F)$ through the Masked Language Modeling (MLM) loss:

$$L_{MLM} = -\sum_{x' \in m(F)} \log p\left(x' \mid H_F \backslash m(F)\right). \tag{8}$$

In Chinese legal system, the legal judgment content typically consists of law articles, charges, and penalty terms. Formally, for a given legal judgment $D = [d_1, d_2, d_3, \ldots, d_n]$, its structure can be expressed as: "*According to the Criminal Law of China (PRC, People's Republic of China)* $[S_L]$*, this defendant committed* $[S_C]$ *Charge, and defendant was sentenced to* $[S_I]$". The S_L, S_C, and S_I correspond to the specific provision content, definitions of criminal charges, and descriptions of imprisonment terms, respectively. To facilitate training, specific tokens within this structured legal judgment are masked, collectively denoted as S.

We leverage the dense vector f of the criminal fact [CLS] token to recover the original legal judgment. Specifically, the input D_{input} of the legal judgment decoder is:

$$D_{input} = [f, e_{d_1} + p_{d_1}, \ldots, e_{d_n} + p_{d_n}], \tag{9}$$

Table 1: Data Statistics of CAIL-2018. Our experimental datasets include the legal judgment prediction phase and the pre-train phase.

Phases	**Legal Judgment Prediction**							**Pre-train**
Scenarios	**High Frequency Criminal Scenario**		**Low Frequency Criminal Scenario**		**Confusing Criminal Scenario**			-
Datasets	**CAIL-small**	**CAIL-big**	**Low-small**	**Low-big**	**Confuse-small**	**Confuse-medium**	**Confuse-big**	**-**
Raw Dataset								
Case	142,634	1,773,099	2,451	2,314	50,000	75,000	100,000	10,000,000
Law Articles	103	118	57	44	12	12	12	183
Criminal Charges	119	130	65	57	10	10	10	202
Term of Penalty	11	11	11	11	11	11	11	11
Data Split								
Train	101,619	1,587,979	1,769	1,644	40,000	60,000	80,000	8,000,000
Valid	13,768	-	251	-	5,000	7,500	10,000	1,000,000
Test	26,749	185,120	431	670	5,000	7,500	10,000	1,000,000

where e_{d_n} represents the embedding and p_{d_n} denotes the extra positional embedding of d_n. Then the decoder can be trained through the loss function:

$$L_{DEC} = -\sum_{x' \in S} \log p\left(x' \mid D_{input} \backslash S\right). \tag{10}$$

The legal judgment decoder proficiently simulates the legal judicial processing, facilitating the generation of highly discriminative vectors.

4 Experiment

This section describes the experimental datasets, baseline models, evaluation metrics, and implementation details of JurisNexus and baseline models.

Experimental datasets and pre-processing. As shown in Table 1, we adopt a standard Chinese legal judgement dataset, CAIL-2018 [28], which aligns with most existing research in the LegalAI [38,9,31,30,33,17]. Following these studies, we maintain the same experimental configuration by filtering out cases containing fewer than 10 meaningful words, and crimes involving multiple law articles or criminal charges. During the pre-training phase, we aggregate tens of millions of case documents from China Judgment Online [4], leveraging this extensive corpus to enhance model learning and legal text representation.

Testing scenarios. We have established three kinds of testing scenarios, including high-frequency, low-frequency and confusing criminal scenarios, respectively. Specifically, our high-frequency criminal scenarios involve accusations that appear in at least 100 cases, along with associated law articles that each occur more than 100 times within the dataset. To validate that our cross-reasoning-chain has consistently exceptional legal representation learning ability in more complex crimes, we gather charges with frequencies ranging from 0 to 100 to build the low-frequency criminal charge scenario. Meanwhile, we define 10 high-frequency accusations that are easily misjudgment as the confusing criminal charges. Our confusing criminal charge scenario constructs three subsets: Confused-small

[4] https://wenshu.court.gov.cn/

(50,000 cases), Confused-medium (75,000 cases), and Confused-big (100,000 cases). And each subset contains the same 10 high-frequency criminal charges.

Baseline methods. In our experiments, we compare JurisNexus model with the following Legal Judgment Prediction (LJP) baseline methods:

TF-IDF+SVM LJP method: It employs the TF-IDF [21] methods for legal clues learning, and applies the SVM [25] for legal judgment classification.

CNN-based LJP methods: `TextCNN` [14] can improve text classification performance by utilizing various filter widths. `DPCNN` [11] employs region embedding, convolutional kernels, and residual connections to construct a deep model.

LSTM-based LJP methods: The LSTM architecture [8] is employed to develop a text classification model for predicting LJP verdicts. `TopJudge` [38]is a multi-task learning framework designed to prediction judgments. It incorporates the dependency in a Directed Acyclic Graph (DAG) structure to enhance the reasoning process and improve prediction accuracy. The `Few-Shot` [9] introduces an attribute-guided attention mechanism that leverages legal elements and attributes to refine the semantic representation of criminal facts. `LADAN` [30] employs a knowledge distillation approach to automatically discern nuanced distinctions between legal provisions, thereby improving the accuracy of charge prediction in ambiguous crimes. `NeurJudge` [33] integrates intermediate judicial sub-task outcomes, enabling refined semantic differentiation among similar criminal cases for enhanced prediction accuracy.

PLM-based LJP methods: `BERT-Chinese` [5] model leverages multiple layers of Transformer encoders to effectively capture bidirectional contextual information. `BERT-Crime` [39] is fully pre-trained on a Chinese large-scale legal corpus. `SEMDR-tt` [17] adopts a two-tower paradigm, using BERT-Chinese to encode criminal case and legal judgment embeddings separately. The dot-product similarity between the two embeddings serves as the core interaction metric for measuring semantic alignment. SAILER [15] designs an asymmetric encoder-decoder framework to capture the structural and logical dependencies within legal cases.

Evaluation Metrics. We evaluate legal judgment prediction methods using the official evaluation metrics from CAIL-2018 [28], which include Accuracy (Acc), Macro-F1 (F1), and Macro-Precision (MP). Each experiment conducted with the JurisNexus model is replicated three times, and we report the mean results.

Implementation Details. JurisNexus incorporates two core representation learning modules: Similar Case Reasoning (SCR) and Legal Judgment Reasoning (LJR). To conduct cross-reasoning-chain mechanism, we employ a pre-trained language model, Bert-Chinese as encoder to embed criminal fact description. For each given criminal, the SCR module ranks the entire legal corpus using BM25 retrieval settings [12,29], selecting 1 positive criminal fact and 15 negative criminal facts with higher rankings. Based on the experimental validation, we determine 0.05 as the optimal temperature hyperparameter to properly calibrate the similarity metric. During LJR module, the decoders are randomly initialized transformer layers. The default mask ratio is 0.15 for the encoder and 0.45 for the decoder. We pre-train the model for up to 10 epochs using the AdamW optimizer, with a learning rate of 1e-5, a batch size of 72, and a linear learning rate schedule

Table 2: Overall performance of legal judgment prediction models in the high-frequency criminal scenario. The highest F1 score are highlighted in **bold**, and the second-best results are underlined.

Methods	High Frequency Criminal Scenario					
	CAIL-small			CAIL-big		
	Law Articles	Charges	Term of Penalty	Law Articles	Charges	Term of Penalty
SEMDR-tt	62.15	63.74	17.03	68.42	75.17	35.65
TF-IDF+SVM	73.89	75.16	28.63	73.88	73.58	41.54
LSTM	77.43	73.41	30.90	74.66	79.47	39.33
DPCNN	70.74	71.74	28.83	75.40	75.77	39.10
TextCNN	69.70	70.93	31.28	76.02	78.66	32.97
MPBFN	75.64	80.14	29.83	78.35	83.37	41.36
TopJudge	73.60	78.71	29.35	77.85	81.33	44.05
CTM	76.35	76.29	28.06	79.28	83.99	37.92
NeurJudge	77.67	76.80	39.25	78.73	86.65	47.88
Few-Shot	76.58	80.97	27.14	81.59	83.61	43.40
BERT-Chinese	77.64	81.09	34.15	92.94	93.32	48.25
LADAN	76.80	82.85	34.20	81.85	83.85	50.17
BERT-Crime	77.32	82.20	25.53	93.11	94.68	49.49
SAILER	78.66	82.05	34.80	94.88	95.37	50.06
JurisNexus	**83.57**	**85.97**	**44.20**	**96.65**	**97.92**	**53.29**

with a warmup ratio of 0.1. We mask the law article content, criminal charges definition, and term of penalty in the decision decoder phase.

Legal Judgment Prediction. We tokenize and encode the description of the criminal fact into a sequence of up to 256 tokens. Subsequently, JurisNexus fine-tunes all parameters of the pre-trained BERT model, including the classifier layer, for predicting law articles, criminal charges, and terms of penalty. Additionally, we utilize the AdamW optimizer with a learning rate of 0.01.

Moreover, for the CNN-based LJP methods, the maximum document length is set to 256 tokens, and the THULAC toolkit is utilized for sentence segmentation. For the LSTM-based LJP methods, we impose a limit of 150 words per sentence and a maximum of 15 sentences per document. In the case of PLM-based LJP methods, a token limit of 256 per document is enforced during BERT model embedding. All experiments are conducted on 4 NVIDIA RTX 4090 GPUs.

5 Evaluation Results

This section presents a comparison and analysis of the experimental results to validate the effectiveness of JurisNexus in Legal Judgment Prediction (LJP).

5.1 Overall Performance

We have compared our model with other baselines across all LJP sub-tasks, and observed that JurisNexus consistently outperforms other approaches.

Table 2 illustrates that JurisNexus outperforms baseline methods in the high-frequency criminal charge scenarios, and demonstrating greater performance

Table 3: Evaluation results on low-frequency and confusing criminal charge scenarios. The best prediction performance is highlighted in **bold**, and the second-best results are underlined.

Methods	Low Frequency Criminal Scenario						Confusing Criminal Scenario								
	Low-small			Low-big			Confuse-small			Confuse-medium			Confuse-big		
	Acc	MP	F1	Acc	MP	F1	Acc	MP	F1	Acc	MP	F1	Acc	MP	F1
SEMDR-tt	43.09	43.14	45.60	43.95	44.26	45.33	84.10	85.22	84.31	86.67	87.49	87.85	87.52	86.04	85.27
TF-IDF+SVM	42.67	42.84	40.21	45.16	45.95	43.10	87.21	87.65	87.36	89.25	88.56	88.54	90.21	83.21	86.65
DPCNN	47.71	48.26	47.94	59.34	60,03	58.03	87.20	87.89	87.27	89.69	88.63	88.21	93.56	93.69	93.54
LSTM	47.36	49.18	47.12	52.69	53.18	52.41	63.91	65.32	62.37	75.62	74.32	73.25	93.91	93.91	93.88
TextCNN	50.20	50.06	49.18	52.13	51.79	51.84	88.91	89.19	88.93	90.22	90.21	87.23	93.97	93.78	93.79
TopJudge	61.31	58.92	57.53	67.20	65.82	64.12	89.01	88.92	88.79	90.33	90.25	90.25	92.65	91.88	92.03
MPBFN	58.16	55.96	56.37	65.41	63.55	62.72	88.21	83.26	88.45	90.25	84.11	88.96	93.66	91.17	90.26
CTM	60.14	53.84	51.17	68.89	62.34	57.77	87.45	85.06	84.10	90.67	85.45	89.27	94.57	92.36	91.80
NeurJudge	59.98	52.56	50.74	69.06	62.11	58.20	86.81	87.44	87.37	88.63	88.05	85.39	92.13	90.22	90.68
Few-Shot	62.29	64.01	64.85	70.11	69.44	68.57	88.55	89.09	88.38	90.85	90.55	89.24	93.56	93.11	90.51
LADAN	59.04	51.79	55.32	68.43	61.20	57.93	88.89	88.76	88.72	90.48	89.95	89.51	93.05	92.47	91.33
BERT-Chinese	59.63	52.17	51.30	68.81	61.82	58.14	92.19	92.37	92.23	93.22	93.01	92.55	94.48	94.48	94.47
BERT-Crime	84.41	83.40	82.88	91.05	86.33	84.58	92.37	92.61	92.41	93.33	93.22	92.69	95.03	94.80	94.91
SAILER	86.31	**85.88**	84.92	93.72	91.36	90.14	92.31	91.92	90.89	93.45	92.07	91.84	95.10	94.67	94.19
JurisNexus	**89.61**	85.51	**87.94**	**96.79**	**92.11**	**90.40**	**95.74**	**96.07**	**95.58**	**96.49**	**94.69**	**93.44**	**98.82**	**95.33**	**96.92**

gains on the CAIL-small dataset. Specifically, JurisNexus has exceeded BERT-Crime in each sub-task on CAIL-small, with more than 3.12% F1 score gain in criminal charge prediction. Analysis on the CAIL-big dataset further indicates that JurisNexus maintains consistent gains over PLM-based models across all LJP sub-tasks. These improvement support our research that the effectiveness of its cross-inference chain mechanism in legal representation learning. Meanwhile, compared to task-specific baselines that rely on intricate, expert-designed architectures for extracting legal clues, JurisNexus enhances criminal fact representations by integrating the logical reasoning processes of juries and presiding judges. This approach yields broader applicability to other legal downstream tasks. In addition, PLM-based LJP methods generally outperform LSTM-based approaches, with this advantage growing on larger training sets. Such as, on the CAIL-small dataset, BERT-Crime improves law article prediction by 2.08% over Few-Shot model, and on the CAIL-big dataset, it achieves a 13.29% increase. Moreover, JurisNexus surpasses legal-domain PLM-based models, BERT-Crime, and SAILER, indicating that its superior performance derives from cross-reasoning-chain pre-training on criminal fact descriptions and legal judgment generation.

5.2 Effectiveness of JurisNexus on Low-Frequency and Confusing Criminal Charge Scenarios

We also construct low-frequency and confusing criminal charge scenarios to assess the effectiveness of LJP models in these more complex circumstances.

Table 3 reveals that JurisNexus showcases its superiority in dealing with low-frequency and confusing accusation. It attains LJP performance comparable to or even better than that of baseline models. In particular, JurisNexus can potentially render more precise judgments on low-frequency charges, a crucial factor in establishing a trustworthy LJP system. When it comes to ambiguous

Table 4: Prediction performance on ablation models. The BERT-Chinese is our backbone model for JurisNexus, and we only report the F1 score.

Methods	CAIL-small			CAIL-big		
	Law Articles	Charges	Term of Penalty	Law Articles	Charges	Term of Penalty
JurisNexus (Full model)	83.57	85.97	**44.20**	96.65	97.92	**53.29**
JurisNexus* (768 dims)	**85.73**	**86.65**	37.04	**96.83**	**97.93**	51.85
w/o Similar Case Reasoning	78.30	82.17	33.15	89.73	91.95	50.40
w/o Legal Judgment Reasoning	72.28	75.59	32.09	82.79	86.06	50.15
w/o Both (Backbone)	61.71	63.59	29.53	73.73	75.45	49.77

charges, JurisNexus consistently outperforms all LJP baselines, highlighting its proficiency in discerning subtle legal cues from convoluted criminal cases.

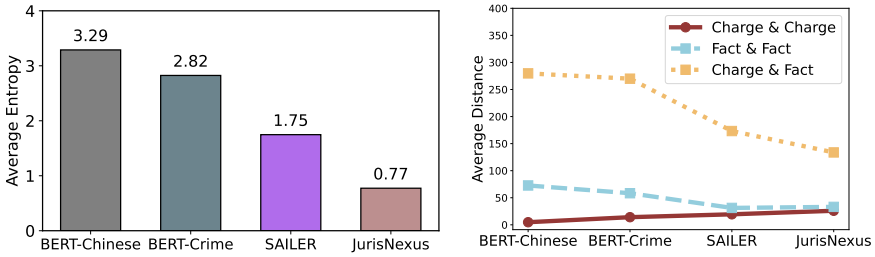
Moreover, JurisNexus has consistently exhibited improvement and maintained robust performance across all confusing conditions, including three sub-dataset of different scales: confuse-small, confuse-medium, and confuse-big. This suggests that JurisNexus can efficiently leverage a limited set of criminal supervision signals to generate reliable and convincing predictions of legal judgments.

5.3 Ablation Study

In our experiments, JurisNexus's cross-reasoning-chain mechanism consists of two main modules: Similar Case Reasoning (SCR) module and Legal Judgment Reasoning (LJR) module. As shown in Table 4, we perform ablation studies to assess the contribution of each module to the overall performance.

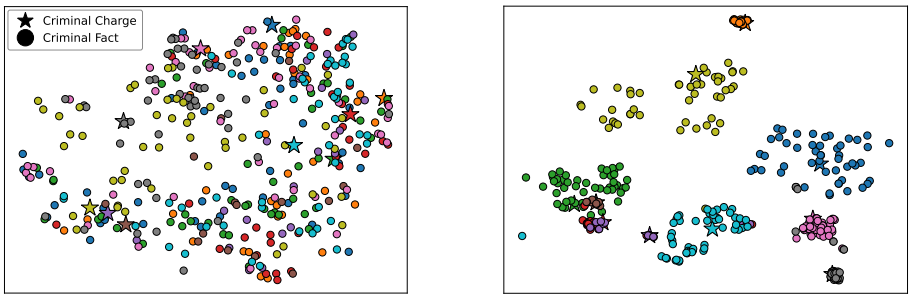
Generally, each perspective reasoning chain has improved the performance in law articles, charges, and term of penalty tasks, as evidenced by a significant performance drop when either the SCR or LJR representation learning module is removed. These experimental findings reveal that learning relevant crimes and legal knowledge is crucial for deeper reasoning. In addition, we can observe that the LJR module has made more contributions in the prediction of the accusation and legal provision, which further illustrates that the semantic association between the criminal case and potential trials actually helps to make better legal judgments. Notably, on the CAIL-small dataset, JurisNexus shows significant improvements than each ablation model, especially for the term of penalty prediction task. This also supports our research that JurisNexus has strong reasoning capability to address the few-shot accusation prediction problem.

Furthermore, our LJR module can guide language models to generate potential legal clues and judgments. Because it jointly leveraging semantic information from legal judgment documents and criminal fact during the continual pretraining phase. Meanwhile, we have investigated the influence of the dimension reduction layer on model performance. Specifically, JurisNexus*, a variant that omits dimensionality reduction during legal judgment prediction phase, exhibits a slight accuracy drop in law article and charge prediction tasks but demonstrates significant improvement in the imprisonment prediction task. This dimensionality reduction module thus serves as a critical regularizer for the semantic representa-



(a) Cross entropy. (b) Embedding distance.

Fig. 3: Characteristics of legal charge prediction behaviors in the confusing criminal charge scenarios. Figure 3(a) and Figure 3(b) compare the cross-entropy, and average distance between legal charges and criminal cases representation of BERT-Chinese, BERT-Crime, SAILER and JurisNexus respectively.



(a) BERT-Chinese. (b) JurisNexus.

Fig. 4: Embedding visualization (t-SNE) of confusing criminal cases and charge definition. The " ★ " symbols in various colors represent the semantic embeddings of criminal charges, including **Theft**, **Robbery**, **Intentional Injury**, **Guilt of Manslaughter**, **Negligence Causing Serious Injury**, **Fraud**, **Credit Card Fraud**, **Dangerous Driving**, **Causing Traffic Casualties**. The " ● " symbols denote the embeddings of the corresponding criminal facts.

tion space, substantially improving JurisNexus's generalization capabilities across diverse LJP sub-tasks.

5.4 Effectiveness of Legal Representations Learning by JurisNexus

This subsection visualizes the guidance provided by different JurisNexus modules regarding the representations of criminal fact description, and the definition content of confusing criminal charges.

According to the results in Figure 3(a), both BERT-Crime and SAILER have smaller entropy than using the BERT-Chinese embedding directly, which

indicates that legal pre-trained language models can effectively reduce the prediction uncertainty [20] of PLM-based models. The Figure 3(b) further shows the semantic distances between the representation of verdicts and corresponding crimes, demonstrating that JurisNexus enhances the distinctiveness by increasing the distances between different charge definitions. Conversely, it shortens the Fact-Charge distances, suggesting that it may learn a more consistent embedding space distribution of criminal cases and their corresponding predicted charges.

To conduct a further analysis of the semantic representation distributions of criminal facts and charge definition, we select 10 confusing criminal charges and randomly sample corresponding criminal cases for each accusation. As shown in Figure 4, the t-SNE is employed to visualize their embeddings, and we find that the Figure 4(a) illustrates a chaotic and collapsed semantic embedding distribution encoded by the BERT-Chinese model directly. Meanwhile, Our model 4(b) further calibrates the distribution of embeddings, creating a more distinct and uniform legal semantic space. Specifically, JurisNexus can effectively cluster cases alongside their corresponding criminal charges, thereby clarifying the boundaries of these ambiguous crimes. This enhancement facilitates more accurate predictions on confusing criminal cases.

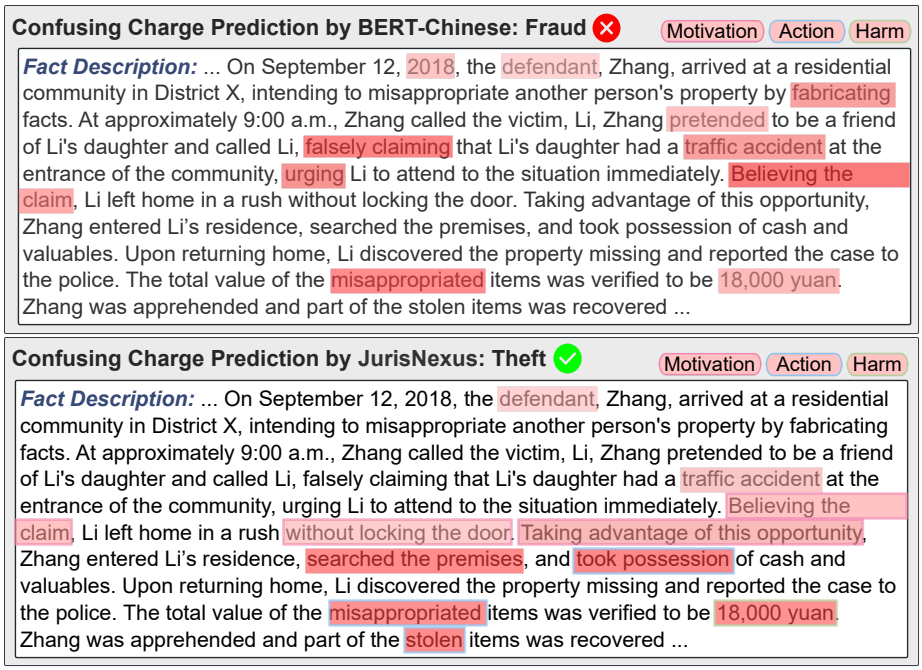


Fig. 5: An easily misjudged confusing charge case Predicted by BERT-Chinese and JurisNexus. The correct charge label of the given criminal case is theft. We highlight several key tokens (motivation, action, harm) in the fact description to evaluate the effectiveness of our cross-reasoning-chain mechanism.

5.5 Contributions of Cross-reasoning-chain Mechanism

This subsection illustrates a theft-charge crime to demonstrate the criminal clues learned by BERT-Chines and JurisNexus. As visualized in Figure 5, we segment the fact description into phrases and the darker color indicates higher attention.

Analyzing the criminal fact description, our model effectively captures legal clues and allocates higher attention to criminal motivation and harm content, including "Believing the claim", "misappropriated", and "18,000 yuan". These tokens are necessary for facilitating accurate judgments. Moreover, JurisNexus identifies pertinent criminal action "without locking the door" and "stolen items", whereas the property was taken without the victim's awareness, rather than being transferred through her consent or a direct misunderstanding. In contrast, the BERT-Chines model erroneously predicts fraud due to its excessive emphasis on misleading phrases such as "falsely claiming", "fabricating", and "pretended". The comparison of attention distributions between BERT-Chines and JurisNexus reveals that our cross-reasoning-chain mechanism successfully resolves ambiguities among similar charges, thereby reducing the misclassification probability and highlighting the significance of our research in legal representation learning.

6 Conclusion

This study introduces JurisNexus, an innovative pre-training framework for improving the ability of language models to better represent criminal facts. JurisNexus mimics the reasoning processes of judges through a cross-reasoning-chain approach to trace critical legal clues. It includes Similar Case Reasoning (SCR) module and Legal Judgment Reasoning chain (LJR) module, which replicate the jury-based and judge-based processes of practical judicial decision-making scenarios. Our experimental results reveal that JurisNexus achieves SoTA prediction performance in Legal Judgment Prediction (LJP) tasks and demonstrates exceptional performance in more complex criminal scenarios.

Acknowledgement. This research is supported in part by the National Natural Science Foundation of China (Grant No. 92067109), in part by Shenzhen Science and Technology Program (Grant No. ZDSYS20210623092007023)

References

1. Atkinson, K., Bench-Capon, T.: Legal case-based reasoning as practical reasoning. Artificial Intelligence and Law (2005), https://doi.org/10.1007/s10506-006-9003-3
2. Bi, S., Ali, Z., Wu, T., Qi, G.: Knowledge-enhanced model with dual-graph interaction for confusing legal charge prediction. Expert Systems with Applications **249**, 123626 (2024). https://doi.org/https://doi.org/10.1016/j.eswa.2024.123626, https://www.sciencedirect.com/science/article/pii/S0957417424004913

3. Chalkidis, I., Jana, A., Hartung, D., Bommarito, M., Androutsopoulos, I., Katz, D., Aletras, N.: LexGLUE: A benchmark dataset for legal language understanding in English. In: Proceedings of ACL. Association for Computational Linguistics (2022), https://aclanthology.org/2022.acl-long.297/
4. Deng, W., Ren, P.: Syllogistic reasoning for legal judgment analysis. In: Proceedings of EMNLP (2023), https://aclanthology.org/2023.emnlp-main.864
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of NAACL-HLT. pp. 4171–4186 (2019), https://aclanthology.org/N19-1423
6. Gao, T., Yao, X., Chen, D.: SimCSE: Simple contrastive learning of sentence embeddings. In: Proceedings of EMNLP. pp. 6894–6910 (2021), https://aclanthology.org/2021.emnlp-main.552
7. Ge, J., Hu, W.: Learning fine-grained fact-article correspondence in legal cases. TASLP (2021), https://doi.org/10.1109/TASLP.2021.3130992
8. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation (8), 1735–1780 (1997)
9. Hu, Z., Li, X., Tu, C., Liu, Z., Sun, M.: Few-shot charge prediction with discriminative legal attributes. In: Proceedings of COLING. pp. 487–498 (2018)
10. Hwang, W., Lee, D., Cho, K., Lee, H., Seo, M.: A multi-task benchmark for korean legal language understanding and judgement prediction. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) Proceedings of NIPS. vol. 35, pp. 32537–32551 (2022), https://proceedings.neurips.cc/paper_files/paper/2022/file/d15abd14d5894eebd185b756541d420e-Paper-Datasets_and_Benchmarks.pdf
11. Johnson, R., Zhang, T.: Deep pyramid convolutional neural networks for text categorization. In: Proceedings of ACL (2017)
12. Karpukhin, V., Yih, W.t.: Dense passage retrieval for open-domain question answering. In: Proceedings of EMNLP. pp. 6769–6781 (2020), https://aclanthology.org/2020.emnlp-main.550
13. Katz, D.M., Hartung, D., Gerlach, L., Jana, A., Bommarito II, M.J.: Natural language processing in the legal domain. arXiv preprint arXiv:2302.12039 (2023), https://arxiv.org/pdf/2302.12039
14. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of EMNLP. pp. 1746–1751 (2014)
15. Li, H., Tian, Q.: Sailer: Structure-aware pre-trained language model for legal case retrieval. In: Proceedings of SIGIR (2023)
16. Liu, C.L., Chang, C.T., Ho, J.H.: Case instance generation and refinement for case-based criminal summary judgments in chinese. Journal of Information Science and Engineering **20**(4), 783–800 (2004), https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=d839f40a66b96c0742bd63f650867a584859022d
17. Liu, P., Zhang, W., Ding, Y., Zhang, X., Yang, S.H.: Semdr: A semantic-aware dual encoder model for legal judgment prediction with legal clue tracing. In: Proceedings of IEEE SMC. pp. 3447–3453 (2024), https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10830950
18. Liu, Y., Wu, Y., Zhang, Y., Sun, C., Lu, W., Wu, F., Kuang, K.: Ml-ljp: Multi-law aware legal judgment prediction. In: Proceedings of SIGIR (2023). https://doi.org/10.1145/3539618.3591731, https://doi.org/10.1145/3539618.3591731
19. Liu, Z., Chen, H.: A predictive performance comparison of machine learning models for judicial cases. In: IEEE SSCI (2017)
20. Pinkard, H., Waller, L.: A visual introduction to information theory. ArXiv preprint (2022), https://arxiv.org/abs/2206.07867

21. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Information Processing & Management pp. 513–523 (1988), https://www.sciencedirect.com/science/article/pii/0306457388900210
22. Saravanan, M., Ravindran, B., Raman, S.: Improving legal information retrieval using an ontological framework. Artificial Intelligence and Law (2009), https://doi.org/10.1007/s10506-009-9075-y
23. Shao, Y., Ma, S.: BERT-PLI: modeling paragraph-level interactions for legal case retrieval. In: proceedings of IJCAI. https://doi.org/10.24963/ijcai.2020/484
24. Sun, Z.: Law article-enhanced legal case matching: A causal learning approach. In: Proceedings of SIGIR (2023), https://doi.org/10.1145/3539618.3591709
25. Suykens, J., Vandewalle, J.: Least squares support vector machine classifiers. Neural Processing Letters pp. 293–300 (1999)
26. Xiao, C., Liu, Z., Lin, Y., Sun, M.: Legal Knowledge Representation Learning, pp. 401–432 (2023). https://doi.org/10.1007/978-981-99-1600-9_11
27. Xiao, C., Sun, M.: Lawformer: A pre-trained language model for chinese legal long documents. AI Open pp. 79–84, https://arxiv.org/pdf/2105.03887
28. Xiao, C., Xu, J.: Cail2018: A large-scale legal dataset for judgment prediction. ArXiv preprint (2018), https://arxiv.org/abs/1807.02478
29. Xiong, L., Overwijk, A.: Approximate nearest neighbor negative contrastive learning for dense text retrieval. In: Proceedings of ICLR (2021)
30. Xu, N., Wang, P., Chen, L., Pan, L., Wang, X., Zhao, J.: Distinguish confusing law articles for legal judgment prediction. In: Proceedings of ACL (2020)
31. Yang, W., Luo, Y.: Legal judgment prediction via multi-perspective bi-feedback network. In: Proceedings of IJCAI. pp. 4085–4091 (2019)
32. Ye, H., Chao, W.: Interpretable charge predictions for criminal cases: Learning to generate court views from fact descriptions. In: Proceedings of NAACL-HLT (2018)
33. Yue, L., Wu, D.: Neurjudge: A circumstance-aware neural framework for legal judgment prediction. In: Proceedings of SIGIR (2021)
34. Zhang, H., Dou, Z., Zhu, Y., Wen, J.R.: Contrastive learning for legal judgment prediction. ACM Trans. Inf. Syst. **41**(4) (2023). https://doi.org/10.1145/3580489, https://doi.org/10.1145/3580489
35. Zhang, H., Wen, J.R.: Contrastive learning for legal judgment prediction. TOIS (2023), https://doi.org/10.1145/3580489
36. Zhang, Y., Huang, W., Feng, Y., Li, C., Fei, Z., Ge, J., Luo, B., Ng, V.: LJPCheck: Functional tests for legal judgment prediction. In: Findings of the Association for Computational Linguistics: ACL 2024. Association for Computational Linguistics (2024), https://aclanthology.org/2024.findings-acl.350/
37. Zhang, Y., Wei, X., Yu, H.: Hd-ljp: A hierarchical dependency-based legal judgment prediction framework for multi-task learning. Knowledge-Based Systems **299**, 112033 (2024), https://www.sciencedirect.com/science/article/pii/S0950705124006671
38. Zhong, H., Guo, Z., Tu, C., Xiao, C., Liu, Z., Sun, M.: Legal judgment prediction via topological learning. In: Proceedings of EMNLP. pp. 3540–3549 (2018)
39. Zhong, H., Zhang, Z., Liu, Z., Sun, M.: Open chinese language pre-trained model zoo. Tech. rep. (2019), https://github.com/thunlp/openclap