JudgeGen: Judgment Generation according to case fact

Yanqing An
School of Artificial Intelligence and Data Science, University of Science and Technology of China
Hefei, China
anyq@mail.ustc.edu.cn

Abstract

CCS Concepts

• **Do Not Use This Code → Generate the Correct Terms for Your Paper**; *Generate the Correct Terms for Your Paper*; Generate the Correct Terms for Your Paper; Generate the Correct Terms for Your Paper.

Keywords

Legal intelligence, Proficiency assessment, Lawyer

1 Introduction

With the rapid development of technology with large public data in legal domain and booming graphic computing resources, Legal Intelligence (LegalAI) is widely used in research and the legal tasks are deeply stacking with each other. As the wave of artificial intelligence sweeps through the legal profession, AI-based tools are flooding into the field[23]. According to reports, over 90% of legal professionals now utilize at least one AI tool in their daily work[1]. As a key segment of the legal profession, the quality of cases handled by lawyers significantly impacts their livelihood and work. The lawyer proficiency assessment is designed to quantify lawyers' proficiency and facilitate personalized career recommendations. As an emerging trend, more and more legal platforms [2] come to provide online services to meet the increasing legislative requirements, where people can seek legal assistance from lawyers conveniently. However, assessing a lawyer's proficiency is a quite difficult problem. In the Internet age, existing methods for evaluating a lawyer's competence include analyzing historical case records [1] and courtroom dialogue [2]. A toy example illustrates the process by which a lawyer handles a case, including the courtroom proceedings, as shown in Figure 1.

Tracing back the career of one lawyer, we can find that she has taken charge of some cases, where each case is corresponding to a textual description and is related to several fields of crimes (i.e., divorce dispute). Generally speaking, the lawyer first takes on the case after it occurs. At this point, the facts of the case already exist. After accepting the case, the lawyer goes through the trial process and reaches a verdict on the case. During this period, there are trial dialogues, including the plaintiff's demands and the defendant's responses. The judgment document summarizes this process.

Cognitive diagnostic tasks hold a significant position in educational intelligence [26]. Similar to GANs and VAEs, cognitive diagnosis generates students' abilities through latent features. Previous work by LawyerPAN [1] attempted this in the legal field. Large language models (LLMs) have accelerated the development of generative models. By combining the text generation capabilities of LLMs, simulated judgment results can be generated, thereby helping to assess the abilities of lawyers in various fields.

Due to its latent attributes and it is hard to quantify directly, The Proficiency Assessment task has several challenges. First, Proficiency need to be assessed indirectly according to handled records of the lawyer. Second, public legal data provides convenient, and brings known information. For instance, when the cost of committing a crime is known, people may still commit the crime in order to obtain greater benefits. Finally, how to verify the interpretability and effectiveness of proposed Proficiency is a question.

To address these challenges, we proposed LawyerGen, a judgment generation and generative proficiency assessment framework. Our motivation is first to simulate a judgment based on the facts of the case, and then to predict the actual judgment based on those facts and the simulated outcome. Generally, the more accurate the simulated judgment, the more accurate the actual judgment will be. Specifically, we first proposed a generated framework with proficiency assessment network. Then, a judgment generation module generating judgment result relies on facts and dialogues is fused to improve reliability. Finally, experiments on real-world dataset verifies the interpretability and effectiveness of proposed framework.[3]



Figure 1: A toy example of lawyers handling cases, court proceedings and judgments. Overall, it is divided into three parts: the case facts, the court dialogue, and the judgment. The lawyer takes on the case after the incident occurs, and the judge makes a judgment based on the facts of the case and the court proceedings.

2 Relate Works

In this paper, we focus on two main related works, the judgment generation and generative proficiency assessment. Both of them are generated research.

[1]https://www.wolterskluwer.com/en/know/future-ready-lawyer-2026
[2]https://www.findlaw.com, https://www.51djl.com/

[3]Our code and datasets are released at https://anonymous.4open.science/r/LawyerGen-F818/.

2.1 Legal Intelligence

Legal Intelligence (LegalAI) aims to help people make better use of the law to make their lives easier[36]. LegalAI is involved in a range of applications, including legal search engines, automated judgment systems, legal document generation, legalAI recommendations, legal text translation, legal Q&A, legal risk early warning, legal text mining, and legal compliance reviews. The task of legal judgment prediction aims to predict relevant legal provisions, charges, and sentences based on case facts[32, 35]. Among these, the task of predicting civil legal judgments has also been widely studied, such as in sales contract disputes[38] and simulations of real judicial processes[34]. Case retrieval involves using pre-trained models or large language models to retrieve similar cases, including methods that involve user participation [21], methods that focus on model pre-training [19], and methods that involve agents [16]. Both COLIEE 2023 and CAIL 2021 organized competitions that included case similarity retrieval tasks. Legal question-answering involves using models to answer questions on Chinese law exams[37]; CAIL 2020 organized a legal question-answering competition[4]. Court Argument Generation aims to generate court arguments based on legal facts[27, 33] Contract Review assists in classifying and refining contracts[10, 22] Legal Entity Extraction aims to extract legal entities from legal texts[13, 14]. Some studies in the case law retrieval task utilize legal elements[15]. Legal Summarization generates summaries extracted from case law retrieval[3, 24]. Legal translation includes research on legal-to-legal translation for minority languages. Existing legal risk early warning efforts rely on experts reviewing surveillance videos to achieve risk early warning. In recent years, evaluation benchmarks have emerged for the development of LLMs, namely LegalBench [9] and LawBench [8].

With the public of China Judgments Online[5], a vast amount of text data from the legal field has begun to be mined. Judicial documents are semi-structured data, comprising sections such as Party Information, Procedural Posture, Facts, Holdings, Decision, and Tails [21]. Research on predicting legal judgments involving multiple defendants includes modeling of legal users [11, 31].

2.2 Generative Models

Generative methods include GAN-based, diffusion model-based, autoencoding-based (e.g., VAEs, BERT), and autoregressive-based models (e.g., BAT-Fill). GPT-based LLMs belong to the autoregressive model category. The principles of cognitive diagnosis in educational psychology are similar, generating surface-level ability parameters based on students' test responses.

Generative Proficiency Assessment. We emphasize the concept inspired by educational psychology [4, 18] and measurement psychology. Proficiency assessment is a common task in user modeling, aimed at identifying users' proficiency levels across various domains. Item Response Theory (IRT) [7] is a concept from measurement psychology and educational psychology that is widely used in proficiency assessment tasks. IRT uses the sigmoid function to map the difference between a user's ability and an item's difficulty to a probability space. MIRT [20] extends IRT to multiple domains. DINA [5] is a discrete characterization of user capabilities, while FuzzyCDF [17] is a modeling of users' subjective capabilities. NeuralCD combines IRT with neural networks, applying it to the field of education and extending it to high-dimensional spaces to predict the probability that a student will answer a given test item correctly. Subsequently, LawyerPAN applied this approach to the legal field and made adjustments to the text representation component. Research on ability assessment in the legal field remains limited and awaits further exploration.

Generative models are also widely used in fields other than education. For example, in recommendation systems, latent feedback is used to represent users' latent preferences during the e-commerce purchasing process. However, in the legal field, research is predominantly focused on text-based representation learning—such as the generation of courtroom arguments and legal summaries—with relatively few user-related generative models. There are two known effective methods for assessing a lawyer's proficiency: one is reviewing the lawyer's case handling record [1], and the other is evaluating their proficiency through conversation[2].

3 Preliminaries

3.1 Problem Definition

In this section, we define the input data and the modeling problem in this paper. We summarize the notations in Table 1. Specifically, we assume that there are N Lawyers, M Cases and K Fields at a legal system, which can be represented as $L = \{l_1, l_2, \ldots, l_N\}$, $C = \{c_1, c_2, \ldots, c_M\}$ and $F = \{f_1, f_2, \ldots, f_K\}$ respectively.

Lawyer records. Each lawyer handles some cases, and the judgment records are represented by R, which $R_{ij} = (l_i, c_j, y_{ij})$ denotes the lawyer l_i obtains judgment result y_{ij} on case c_j. The judgment result $y_{ij} = 1$ if lawyer l_i has positive effectiveness (e.g., compensation or reducing penalty) on case c_j and $y_{ij} = 0$ otherwise.

Case logs. Each case includes lawyers in plaintiff and defendant teams denoted as $c_j = \left\{G_j^{pla}, G_j^{def}, D_j^w\right\}$, $G_j^* = \{l_1, l_2, \ldots, l_U\}$, which U represents the number of lawyers in corresponding team. Further, case facts are expressed as word sequences $D_j^f = \left\{d_{j,1}^f, d_{j,2}^f, \ldots, d_{j,n_f}^f\right\}$, where n_f denotes the length of the sequence. In addition, we have Q-matrix $\boldsymbol{Q} \in \mathbb{R}^{M \times K}$ to represent related fields.

Definition 3.1 (*Proficiency Assessment*). *Given a set of lawyers L, a set of cases C, and a set of domains F, the goal is to identify the proficiencies of lawyers in each domain.*

3.2 Flowchart

Since proficiency is an implicit concept that cannot be directly quantified, we use the wining rate prediction task to assess proficiency.

Definition 3.2 (*Wining Rate Prediction*). *Given the lawyer l and the fact description d^f, the goal is to predict the outcome $Y \in \{0, 1\}$ of the lawyer's handling of the case, where 0 indicates a victory and 1 indicates a defeat.*

Our solution process is shown in Figure 2. The solution consists of a two-stage framework. Firstly, the model LawyerGen is trained through the win rate prediction task. Secondly, the capability representation is obtained.

In order to predict the winning probability more accurately, we incorporated trial simulation. By generating simulated judgment results and integrating the representation process, we achieved this.

[4] http://cail.cipsc.org.cn/
[5] https://wenshu.court.gov.cn/

Table 1: Several important mathematical notations.

Notation	Description
L	The set of lawyers.
C	The set of cases.
F	The set of fields.
R	The set of judgment records.
G^*	The set of lawyers in plaintiff or defendant team.
Q_c	The case-field mappings of cases.
h^l	The proficiency of lawyers.
h_*^{diff}	The difficulty of cases when the lawyer agent for plaintiff or defendant.
h_*^{disc}	The discrimination of cases.
x	The output of assessor module.
s^f	The representation of case facts.
s^d	The representation of case dialogue.
s^r	The representation of judgment results.
s^c	The representation of the case.
y	The output of prediction result.

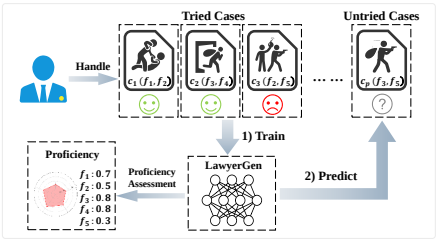


Figure 2: The example of the court and judgment process.

Definition 3.3 (*Judgment Generation*). *Given the description of the case facts and the court dialogue records, the goal is to generate the judgment result* s^r.

3.3 Proficiency Assessment

Proficiency Assessment is a common task in data mining, the aim of which is to assess the proficiency of specific users through data mining. In the Lawyer Proficiency Assessment task, the hidden layer parameters of a neural network are used to measure a lawyer's proficiency by predicting the outcomes of cases they handle. We use IRT to characterize a lawyer's proficiency, while also representing the difficulty of the cases they handle. Generally, the higher a lawyer's proficiency, the easier it is to win the cases they represent. The more difficult a case is, the harder it is for a lawyer to win it. IRT can be expressed in the following form:

$$p(\theta) = Sigmoid(\alpha(\theta - \beta)), \tag{1}$$

here α represents the item discrimination, θ represents the student's proficiency, β represents the item difficulty, and $p(\theta)$ represents the probability that the student will answer the item correctly in education system. NeuralCD extends IRT to multidimensional and multidisciplinary contexts, as illustrated in the legal field as follows [1, 26]:

$$x = Q(h^l - h^{diff}) \times h^{disc}, \tag{2}$$

where Q denotes the relevant field of the case, h^l denotes the lawyer's proficiency, h^{diff} denotes the difficulty of the case, h^{disc} denotes the distinctiveness of the case, and x denotes the probability of the lawyer winning the case.

4 LawyerGen

4.1 Overview

In this section, we describe the details of LawyerGen, which is a generative proficiency assessment network. Figure 3 illustrates the structure of LawyerGen. The entire architecture could be split into three parts in a high-level discussion. First, the Encoder component embeds lawyers and cases, representing lawyer proficiency, case difficulty, and discrimination. Second, the Generator generates a judgment based on the facts of the case and the courtroom dialogue, and feeds it into the Encoder. Thirdly, Assessor uses IRT to model the relationships among the field, the lawyer's proficiency, the case's difficulty, and the case's discrimination, and integrates them. Finally, Predictor and Trainer are designed to learn the proficiency relations on different levels adaptively.

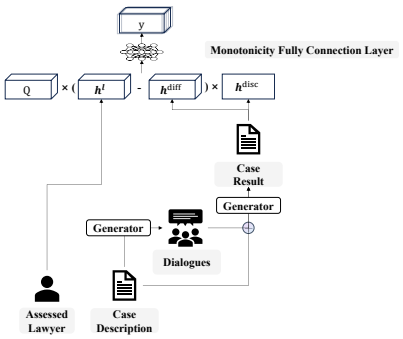


Figure 3: The illustration of the proposed LawyerGen model. In summary, it is a binary classification and generative model: 1) Encoder and Generator, which generate the judgment result and encode the lawyers and descriptions; 2) Assessor, which utilizes IRT to quantify the lawyer's proficiency, the difficulty of the case, and the discrimination of the case; 3) Predictor, which is designed for feature learning and final prediction.

4.2 Encoder and Generator

Here we introduce representations of lawyer factors, case factors and, team factors in LawyerGen.

lawyer factors. In LawyerGen, each lawyer is represented with a proficiency vector. The lawyer factor $\boldsymbol{h}^l$, is obtained by multiplying the lawyer's one-hot representation $\boldsymbol{x}^l$. That is,

$$\boldsymbol{h}^l = \text{sigmoid}\left(\boldsymbol{x}^l \times \boldsymbol{E}^l\right), \tag{3}$$

where $\boldsymbol{h}^l \in (0,1)^{1\times K}$, $x^l \in \{0,1\}^{1\times N}$, trainable matrix $\boldsymbol{E}^l \in \mathbb{R}^{N\times K}$.

case factors. In reality, each case itself has objective facts, involving relevant legal principles and legal articles. We regard them as case factors. To make a general representation and more precise assessment, we adopt three case factors: field vector, difficulty vector and discrimination vector.

Field vector. For each case, the case factor $\boldsymbol{Q}_c$, is directly from the pre-given Q-matrix:

$$\boldsymbol{Q}_c = x^c \times \boldsymbol{Q}, \tag{4}$$

where $\boldsymbol{Q}_c \in \{0,1\}^{1\times K}$, and $x^c \in \{0,1\}^{1\times M}$ is the one-hot representation vector of the case.

Generator. We simulate court proceedings based on the facts of a case to generate simulated court transcripts and case rulings. Our motivation is first to simulate a judgment based on the facts of the case, and then to predict the actual judgment based on those facts and the simulated outcome. Generally, the more accurate the simulated judgment, the more accurate the actual judgment will be. The generated judgement result is represented as follows:

$$s^r = Transformer(D^f), \tag{5}$$

where $D^f = [d_1^f, d_2^f, \ldots, d_{n_f}^f]$. We use a Transformer[25] architecture to generate.

Difficulty vector. First, given a word sequence of the case description $D^w = \left\{d_1^w, d_2^w, \ldots, d_{n_w}^w\right\}$, we map each word of D^w into word embedding by BERT [6], and get the case embedding sequence $s^d \in \mathbb{R}^{d_c}$ by applying mean-pooling over x^w, where d_c is the dimension of case embedding, n_w is the length of word sequence. Then, $\boldsymbol{h}^{diff} \in (0,1)^{1\times K}$ indicates the difficulty of each field. We further use $\boldsymbol{h}_{pla}^{diff}$ and $\boldsymbol{h}_{def}^{diff}$ to distinguish difficulties of defending for the plaintiff and defendant. The difficulty factors can be obtained by:

$$\boldsymbol{h}_{pla}^{diff} = \text{sigmoid}\left(\boldsymbol{W}_p^{diff}\boldsymbol{s}^c + \boldsymbol{b}_p^{diff}\right), \tag{6a}$$

$$\boldsymbol{h}_{def}^{diff} = \text{sigmoid}\left(\boldsymbol{W}_d^{diff}\boldsymbol{s}^c + \boldsymbol{b}_d^{diff}\right), \tag{6b}$$

$$s^f = BERT(D^f), \tag{6c}$$

$$s^c = W^d[s^f; s^r] + b^d, \tag{6d}$$

where $\boldsymbol{W}^*$ and $\boldsymbol{b}^*$ are learnable weight matrices and biases.

Discrimination vector. $\boldsymbol{h}^{disc} \in (0,1)$ indicates the capability of the case to differentiate between those lawyers whose field proficiency is high from those with low field proficiency. The discrimination factors can be obtained by:

$$h_{pla}^{disc} = \text{sigmoid}\left(\boldsymbol{W}_p^{disc}\boldsymbol{s}^c + \boldsymbol{b}_p^{disc}\right), \tag{7a}$$

$$h_{def}^{disc} = \text{sigmoid}\left(\boldsymbol{W}_d^{disc}\boldsymbol{s}^c + \boldsymbol{b}_d^{disc}\right). \tag{7b}$$

4.3 Assessor

For a more detailed assessment of the lawyer's proficiency in each field, we model the relationship between lawyers and cases.

To exploit relationships between assessed lawyers and case difficulties, following by formula 2, we formulate it as:

$$\boldsymbol{x} = \begin{cases} \boldsymbol{Q}_c \circ \left(\boldsymbol{h}^l - \boldsymbol{h}_{pla}^{diff}\right) \times \boldsymbol{h}_{pla}^{disc} & l \in G^{pla} \\ \boldsymbol{Q}_c \circ \left(\boldsymbol{h}^l - \boldsymbol{h}_{def}^{diff}\right) \times \boldsymbol{h}_{def}^{disc} & l \in G^{def} \end{cases}, \tag{8}$$

where $\circ$ is an element-wise product, l indicates the assessed lawyer, G^{pla} and G^{def} represents all of the lawyers which handled corresponding cases in plaintiff and defendant team.

4.4 Predictor

We predict the actual outcome of a case based on the lawyers involved, the facts of the case, simulated courtroom dialogue, and simulated judgment outcomes.

To adaptively integrate proficiency information, we concatenate representations of anticipatory and adversarial module. Followings are full connection layers and a prediction layer:

$$\boldsymbol{y} = sigmoid\left(\boldsymbol{W}\boldsymbol{x}\right). \tag{9a}$$

To satisfy the positively correlation between the probability $\boldsymbol{x}$ and prediction result $\boldsymbol{y}$, we adopt a strategy which restricts each weights of fully connection layer (e.g., $\mathbf{W}$) to be positive(i.e.Monotonicity Assumption) [26]. It can be easily proved that $\frac{\partial y}{\partial h_i^l}$ is positive for each entry h_i^l in $\boldsymbol{h}^l$. Thus the positively correlation is always satisfied during training.

4.5 Trainer

The loss function of wining rate prediction in LawyerGen is cross entropy between the ground truth y and prediction probability of $\hat{y}$:

$$\mathcal{L}_{WRP} = -\sum_i \left(y_i \log \hat{y}_i + (1-y_i)\log\left(1-\hat{y}_i\right)\right). \tag{10}$$

For judgment generator, we adopt the common negative log-likelihood (NLL)[33] as the loss function:

$$\mathcal{L}_{JG} = \frac{1}{T}\sum_{t=1}^{T} -\log P\left(y_t^*\right), \tag{11}$$

where T is the length of vocab dict.

The joint loss function is designed as the follow:

$$\mathcal{L} = \mathcal{L}_{WRP} + \mathcal{L}_{JG}. \tag{12}$$

After training, the value of $\boldsymbol{h}^l \in (0,1)^{1\times K}$ is the assessment result, which denotes the lawyer's proficiency in K fields.

4.6 LawyerGenD

We use courtroom dialogue to train generative models. To make the process more efficient and straightforward, we have redesigned the LawyerGen framework.

We have incorporated the process of generating the judgement result into the courtroom dialogue, as shown below:

$$s^r = Transformer([D^f; D^d]), \tag{13}$$

where $D^d = [d_1^d, d_2^d, \ldots, d_{n_d}^d]$.

5 Experiments

In this section, we demonstrate the effectiveness of LawyerGen. First, we compare our LawyerGen with some baselines and its variants on real-world datasets. Then, we make some interpretation assessments of models. As we cannot obtain the true field proficiency of lawyers in the real world, the assessment results are

usually acquired through predicting lawyers' effectiveness, which can indirectly evaluate the model. Finally, case studies from individual and team perspectives are used to demonstrate the effectiveness and interpretability of lawyer assessments in the real world.

5.1 Dataset

LCHRGen, a *L*awyer *C*ases *H*andling *R*ecords dataset for *Gen*eration, comes from 51djl.com [6]. The case outcome labels were annotated by experts and are derived from the actual outcomes of legal documents and judgments. The labels are categorized into five levels: 0, 0.5, 0.7, 0.9, and 1. We processed these labels such that values less than 0.5 are treated as 0 (indicating a loss), while all others are treated as 1 (indicating a win).

We used regular expression techniques to extract the names of participating lawyers from legal documents and matched them with each lawyer's case history records. We selected cases involving at least two lawyers for inclusion in the dataset.

The expert-labeled data $Y \in [0, 1]$ is derived from case documents. We processed this data such that $Y \in 0, 1$ represents the outcome of a case handled by a lawyer and serves as the label, where Y less than 0.5 indicates a loss, and otherwise indicates a win.

Specifically, LCHRGen mainly contains lawyers, cases, legal fields and judgment records. Among them, case logs consist of fields, case descriptions, and participating lawyers. Many cases contain multiple plaintiffs and defendants. The cases are settled from 2014 to 2017 in a certain region of China, where all the dismissed cases and executive cases have been removed. Lawyer records are consist of judgment records (i.e., a set of case logs which the lawyer handled) and judgment results (i.e., labels of whether the lawyer has positive effectiveness on the case). Besides, the labels of the fields (i.e., Q-matrix) and judgment results are summarized from the judgment documents and labeled by legal experts who have no conflict of interest with those involved in these cases.

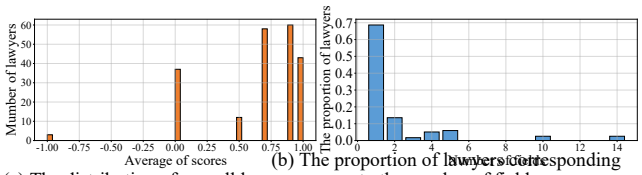
LCHRGenD has added courtroom dialogue data to LCHRGen. Dialogue Generation using deepseek-flash and deepseek-v4-pro [28].

Table 3 summarises the basic statistical information for the entire dataset. Furthermore, Figure 4 presents some fine-grained statistical information regarding the LCHRGen dataset. Specifically, the left-hand section is a histogram showing the distribution of lawyers' success rates, from which it is evident that there are significant variations in lawyers' professional competence. The right-hand section illustrates the relationship between the proportion of lawyers and the number of practice areas, revealing that whilst some lawyers specialise in one or two areas, others practise in as many as ten or more. The average success rate of lawyers and the number of lawyers allocated to each practice area indicate that individual lawyers may specialise in different areas or excel in multiple fields.

5.2 Parameter Settings

Meanwhile, we divide all datasets into training, validation, and test sets according to the ratio of 7:1:2. As the descriptions of facts are written by Chinese court clerk, word embeddings are obtained by the pre-trained "BERT-base, Chinese" model of BERT [6]. We adopt the pooler output layer of BERT to train our model. Meanwhile,

[6]https://www.51djl.com/



(a) The distribution of overall lawyer scores (b) The proportion of lawyers corresponding

Figure 4: The distribution statistics of average lawyer scores.

Table 2: Dataset Statistics.

Dataset	LCHR-Gen-s	LCHR-Gen	LCHR-Gen-d
# Lawyers	1	1	1
# Cases	1	1	1
# Fields	1	1	1
# Judgment records	1	1	1
# Lawyers per case	1	1	1
# Causes per case	1	1	1
Dialogues	-	-	√
Avg. Length of fact			

Table 3: Dataset Statistics.

Dataset	LCHRGenD	LCHRGen
# Lawyers	26,430	29,015
# Cases	193,444	772,920
# Fields	525	636
# Judgment records	440,614	1,020,090
# Lawyers per case	2.442	1.623
# Causes per case	1.005	1.004

we set the maximum document length as 510 words. If either party of the plaintiff or defendant's team has no participating lawyer, the lawyer with id 0 shall fill in. For training, the learning rate of Adam optimizer [12] is initialized as 10^{-4}. We train every model for 20 epochs with batch size 64 and choose the best model on the validation set for testing. To evaluate the performance of our model, all baseline models are implemented by Python, and all experiments are run on a Linux server with Intel Gold 6458Q @ 3.1-4.0GHz and GeForce RTX4090D GPU.

5.3 Evaluation Metrics

The performance of a proficiency assessment model is difficult to evaluate as we cannot obtain the true legal field proficiency of lawyers in the real world. As the assessment results are usually acquired through predicting lawyers' wining rate, performance on this prediction task can indirectly evaluate the model. Therefore, for predicting the wining rate of lawyers, we choose four evaluation metrics from both classification aspect and regression aspect, including Accuracy, RMSE (Root Mean Square Error), AUC (Area Under the Curve), and F1-score. Generally speaking, the better prediction has a high value among Accuracy, AUC, and F1-score. And the lower values of the RMSE values are, the better prediction is obtained. For generating judgment result, we use Rouge-1 and Rouge-L.

5.4 Baseline Approaches

To evaluate the performance of our model on lawyer effectiveness prediction task, we selecte a number of representative methods as baselines. Specifically, we first choose some general and basic approaches as follows:

- **IRT** [7]: a linear model with logistic loss for classification.
- **MIRT** [20]: a statistical-based approach by taking an average of the effectiveness value in all cases of each lawyer.
- **NeuralCD** [26]: a word encoder representation model using one-hot vectors of lawyers and case descriptions.
- **BART**[30]: a statistical-based approach by taking an average of the effectiveness value in all cases of each lawyer.
- **C3VG** [33]: a word encoder representation model using one-hot vectors of lawyers and case descriptions.

5.5 Overall Performance

To evaluate the overall performance of LawyerPAN on lawyer effectiveness prediction task, we first show the comparison results of LawyerPAN and its variants with baselines in Table 6. Then, we compare several uniformization functions to verify the rationality of team proficiency representation. Finally, the visualization of assessment parameters demonstrated the reasonable relationship between learned lawyer proficiency and case difficulty.

5.6 Ablation Study

To evaluate the overall performance of LawyerPAN on lawyer effectiveness prediction task, we first show the comparison results of LawyerPAN and its variants with baselines in Table 6. Then, we compare several uniformization functions to verify the rationality of team proficiency representation. Finally, the visualization of assessment parameters demonstrated the reasonable relationship between learned lawyer proficiency and case difficulty.

5.7 Difficulty Parameter Visualization

Cases in rare fields are more challenging / Criminal cases are more challenging than civil cases / Cases with a lower average win rate are more challenging

5.8 Discrimination Parameter Visualization

In cases with high distinction, the standard for a lawyer's competence is higher.

5.9 Monotonicity Assumption Visualization

A lawyer's competence is directly proportional to their average win rate.

5.10 Proficiency Interpretation

Inspired by the degree of agreement in NeuralCD In a given case, the more skilled the lawyer, the higher the win rate.

5.10.1 ***Accuracy of each category.*** We counted the prediction accuracy of various types of cases of the processed data, and the results are shown in Table 7. The results of intellectual property, civil disputes and contract disputes were preferred, while the prediction accuracy of administrative disputes was low, which we guessed

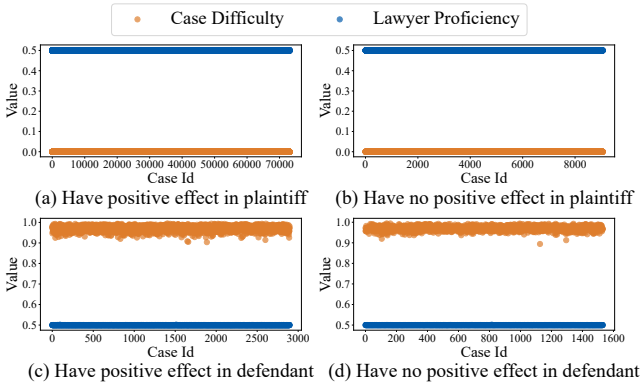


Figure 5: Visualization of assessment parameters.

was caused by the different number distribution of various types of cases in the data set.

5.10.2 ***Impact of IRT parameter.***

5.11 Proficiency Parameter Visualization

To compare with the lawyer proficiency, our LawyerPAN model evaluate the difficulty of the cases and Figure 5 shows the value of lawyer proficiency and case difficulty in the field "*housing presale contract dispute*". We assume that if the lawyer proficiency is higher than the case difficultly, the lawyer might have a positive effectiveness on this case mentioned in Section **??**. Through the observation, we find that, in most cases where lawyers have a positive effectiveness in the plaintiff and defendant, lawyer proficiency value (i.e., blue points) is greater than case difficulty value (i.e., yellow points). In the cases where lawyers have no positive effectiveness, the difficulty value of most cases is greater. The model evaluation results are consistent with our hypothesis, and this further suggests that there is a huge distinction between a lawyer agent for the plaintiff and defendant.

Based on neural network analysis, in cases with a high win rate, the lawyer's proficiency generally outweighs the difficulty of the case, and vice versa.

5.11.1 ***Visualization of proficiency scope in LawyerPANE..*** We believe that the scope of the lawyer's competence depends on the difficulty of the case, that is, greater than the lowest difficulty of the failed case and less than the highest difficulty of the successful case. Because below the minimum difficulty of losing cases that all lawyers can win, we can assume that the lawyer has little ability to win any lawsuit, even lower than non-law practitioners. And above the highest difficulty of a successful case where none of the lawyers who take this case can win the lawsuit, we can say that the lawyer has basically achieved the highest level in the field. The red line of the graph shows the highest and lowest difficulty of the successful case, the blue dot shows shows the proficiency of this lawyer in Figure **??**.

5.11.2 General proficiency and Professional proficiency of Lawyers. Generally speaking, the lawyer's proficiency is divided into **G**eneral proficiency and **P**rofessional proficiency. General proficiency refers to the level of proficiency that most of lawyers can achieve, and professional proficiency refers to the level of proficiency that a

Table 4: Predictions of lawyers' effect on cases. The values in brackets of LawyerPAN indicate relative increase over NeuralCDM-BERT. The values in brackets of variants of LawyerPAN indicate relative decrease compared to LawyerPAN.

Methods	LCHR-small				LCHR			
	Accuracy	RMSE	AUC	F1	Accuracy	RMSE	AUC	F1
AVG	0.706	0.443	0.663	0.810	0.715	0.438	0.685	0.816
LR	0.709	0.459	0.619	0.829	0.694	0.458	0.652	0.819
BERT	0.723	0.416	0.763	0.805	0.731	0.412	0.779	0.804
PMF	0.504	0.459	0.500	0.588	0.498	0.462	0.494	0.579
IRT	0.701	0.484	0.523	0.823	0.693	0.506	0.498	0.818
TrueSkill	0.700	0.483	0.663	0.824	0.693	0.447	0.681	0.819
NeuralCDM	0.720	0.423	0.749	0.808	0.756	0.412	0.785	0.844
LawyerPAN	0.875 (+0.13)	0.304 (-0.10)	0.932 (+0.13)	0.909 (+0.08)	**0.889** (+0.13)	**0.286** (-0.11)	**0.946** (+0.13)	**0.920** (+0.10)
LawyerGen	**0.890** (+0.02)	**0.281** (-0.02)	**0.951** (+0.02)	**0.921** (+0.01)	0.889 (+0.00)	0.287 (+0.00)	0.944 (-0.00)	0.920 (+0.00)

Table 5: Evaluations are quoted from [29].

Methods	General Tasks				Legal Tasks			
	GPQA	MMLU	HumanEva	Multi-Exam	LCHR	LeCaRDv2	LegalQA	CAIL2024-drdz
AVG	0.706	0.443	0.663	0.810	0.715	0.438	0.685	0.816
LR	0.709	0.459	0.619	0.829	0.694	0.458	0.652	0.819
BERT	0.723	0.416	0.763	0.805	0.731	0.412	0.779	0.804
PMF	0.504	0.459	0.500	0.588	0.498	0.462	0.494	0.579
IRT	0.701	0.484	0.523	0.823	0.693	0.506	0.498	0.818
TrueSkill	0.700	0.483	0.663	0.824	0.693	0.447	0.681	0.819
Qwen2.5-7B-Instruct	0.720	0.423	0.749	0.808	0.756	0.412	0.785	0.844
DeepSeek-V4.1-Flash-1017	0.875 (+0.13)	0.304 (-0.10)	0.932 (+0.13)	0.909 (+0.08)	**0.889** (+0.13)	**0.286** (-0.11)	**0.946** (+0.13)	**0.920** (+0.10)
LegalGen	**0.890** (+0.02)	**0.281** (-0.02)	**0.951** (+0.02)	**0.921** (+0.01)	0.889 (+0.00)	0.287 (+0.00)	0.944 (-0.00)	0.920 (+0.00)

Table 6: Predictions of lawyers' winning rate and judgment generation on cases. The values in brackets of LawyerGen indicate relative increase over NeuralCDM-BERT. The values in brackets of variants of LawyerGen indicate relative decrease compared to LawyerPAN. The bold values represent the optimal results.

Methods	WRP		JG
	MSE↓	RMSE↓	Accuracy↑
IRT	0.723	0.416	0.763
MIRT	0.723	0.416	0.763
NeuralCD	0.723	0.416	0.763
BART	0.723	0.416	0.763
LawyerGen	0.723	0.416	0.763
w/o Judgment generation	0.867 (-0.01)	0.313 (+0.01)	0.918 (-0.01)
w/o dialogues	0.867 (-0.01)	0.313 (+0.01)	0.918 (-0.01)
w/o Judgment generation + dialogues	0.867 (-0.01)	0.313 (+0.01)	0.918 (-0.01)
w/o Encoder			
w/o positionEncoding			
w/ 1-layer Decoder			

Table 7: Accuracy of each category.

Category	Accuracy
civil dispute	0.903
contract dispute	0.819
company dispute	0.906
intellectual property right	0.960
administrative dispute	0.790
maritime and merchant shipping	0.813

Table 8: Experimental results on IRT parameter.

Methods	Accuracy	RMSE	AUC	F1
# 1-parameter	0.753	0.450	0.784	0.821
# 2-parameter	0.890	0.281	0.951	0.921
# 3-parameter	0.695	0.461	0.500	0.820

lawyer exceeds other lawyers in an area of expertise. We use the sum of the two to represent a lawyer's proficiency. Therefore, we define the lawyer proficiency as:

representation $\boldsymbol{x}^l$. That is,

$$\boldsymbol{h}^l = \text{sigmoid}\left(\left(W\boldsymbol{x}^l + q\right) \times \boldsymbol{E}^l\right), \tag{14}$$

where $x^l \in \{0,1\}^{1\times N}$ is one-hot representation of the lawyer, matrix q represents general proficiency of all lawyers, trainable matrix $\boldsymbol{E}^l \in \mathbb{R}^{N\times K}$.

Table **??** shows the experiments on the representation of lawyers. The results demonstrate the validity of dividing lawyer's proficiency into general and professional proficiency.

5.11.3 cooperative and competitive relationship between lawyers. **C**ooperation and **C**ompetition exist in the process of lawyer-represented litigation. For example, lawyer A is aware of lawyer B's professional weakness, and lawyer A may argue with lawyer B about this during the trial. Or lawyer A and lawyer B often work together, and their cooperation can greatly improve the probability of the agent obtaining the rights. Therefore, the modeling of cooperation and competition is very important in the litigation process. In this report, we define the anticipatory module of the assessment layer as:

$$\boldsymbol{x} = \boldsymbol{Q}_c \circ \left(\boldsymbol{h}^l + \boldsymbol{h}^r - \boldsymbol{h}^{\text{diff}} \times \left(1 + \boldsymbol{h}^{fac}\right)\right) \times \boldsymbol{h}^{disc}, \tag{15}$$

where $\boldsymbol{h}^r$ denotes the cooperation and competition of lawyers.

Then, we explore the modeling of collaboration and competition using the following three approaches:

- LPAN-C&C-A:

$$\boldsymbol{h}^r = \boldsymbol{h}^s \circ \left(\boldsymbol{x}^l \times \mathbf{F}\right), \tag{16}$$

where matrix F represents the every correlation of cooperation and competition between two lawyers.

- LPAN-C&C-B:

$$\boldsymbol{h}^r = \boldsymbol{x}^l \times \mathbf{F} \times \boldsymbol{E}^l, \tag{17}$$

- LPAN-C&C-C: Considering that the matrix F is a symmetric matrix, we optimize the formula as the follow:

$$\boldsymbol{h}^r = \boldsymbol{x}^l \times \mathbf{F}^T\mathbf{F} \times \boldsymbol{E}^l, \tag{18}$$

Table ?? shows the experiments on cooperation and competition of lawyers. The results show that the factors of cooperation and competition are not considered as well as LPAN-base model, which may be because the representation is not sufficient for interaction. We will study the modeling of cooperation and competition in the future.

5.12 Dialogue-assisted generative training

5.13 Case Study

facts lawyer proficiency judgment generation judgment ground truth

To demonstrate the effectiveness and interpretability of the assessment in the real world, we provide a qualitative analysis of LawyerGen. Figure 6

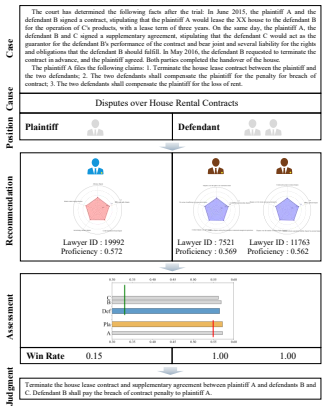


Figure 6: An assessment example of LawyerGen. The radar figure shows proficiency assessment on 5 common fields with similar average effectiveness.

5.14 Visual Analysis

To further explore the classification capabilities of the proposed method, we used each lawyer's most frequently handled practice area and the area with the highest win rate as classification labels, and employed t-SNE [?] to visualise the vector distribution of lawyers' capabilities. Specifically, we selected six types of offences that are easily confused, including intentional assault, provocation, affray, theft, robbery and fraud. For each offence, we randomly sampled a certain number of cases from the pre-training corpus and generated case vectors using off-the-shelf pre-trained models under a zero-shot setting. We used t-SNE to map lawyer representations across different practice areas onto a two-dimensional vector space, where clustered lawyer representations are more similar. We visualise these feature spaces using t-SNE in Figure 7. As can be seen from the figure, LawyerGen demonstrates strong discriminative capability. NeuralCDM consistently groups cases involving different offences together; whilst LawyerPAN, although capable of distinguishing between theft, robbery and fraud, confuses intentional assault, provocation and brawling. By reconstructing the original text, LRN is able to learn more information and generate uniformly distributed vectors. However, the boundaries between similar offences are too blurred, which is detrimental to legal case retrieval. In contrast, LawyerGen can distinguish vectors from different domains more accurately without any supervised data, indicating that it has uncovered a more authentic representation of capabilities through our training objectives, thereby validating the interpretability of our proposed method.

Figure 7: Visualization of lawyer proficiency.

6 Discussion

In this paper, we use case documents as a representation of the litigation context. In the future, we plan to incorporate courtroom video data as a representation of the litigation context.

There is a vast amount of publicly available courtroom video footage on the internet, and we plan to integrate this video data into the litigation context in the future to make the model more stable and robust.

7 Conclusion

In this paper, we presented a focused study on proficiency assessment for trial lawyers and proposed a LawyerGen model for mining proficiency of lawyers in various fields. To be specific, we first designed a Generator to generate simulated case outcomes based on case facts, in order to better predict actual case outcomes. Second, the Encoder encodes lawyers and cases and represents them in terms of lawyer competence, case difficulty, and case discriminative power. Next, the Assessor models the relationships among the case domain, lawyer competence, case difficulty, and case discriminative power. Finally, the Predictor is used to fuse these factors and

predict case outcomes. We hope that LawyerGen can assist in the development of the Legal Ethics Assistant in the future.

References

[1] Yanqing An, Qi Liu, Han Wu, Kai Zhang, Linan Yue, Mingyue Cheng, Hongke Zhao, and Enhong Chen. 2021. LawyerPAN: A proficiency assessment network for trial lawyers. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 5–13.

[2] Yanqing An, Linan Yue, Weibo Gao, Kai Zhang, and Qi Liu. 2024. Post-hoc Facts augmented Legal Judgment Prediction. In *International Conference on Database Systems for Advanced Applications*. Springer, 476–485.

[3] Paheli Bhattacharya, Kaustubh Hiware, Subham Rajgaria, Nilay Pochhi, Kripabandhu Ghosh, and Saptarshi Ghosh. 2019. A comparative study of summarization algorithms applied to legal case judgments. In *European conference on information retrieval*. Springer, 413–428.

[4] Hugh Burns, Carol A Luckhardt, James W Parlett, and Carol L Redfield. 2014. *Intelligent tutoring systems: Evolutions in design*. Psychology Press.

[5] Jimmy De La Torre. 2009. DINA model and parameter estimation: A didactic. *Journal of educational and behavioral statistics* 34, 1 (2009), 115–130.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. [n. d.]. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*. 4171–4186.

[7] Susan E Embretson and Steven P Reise. 2025. *Item response theory: Foundations for psychologists and social scientists*. Taylor & Francis.

[8] Zhiwei Fei, Xiaoyu Shen, Dawei Zhu, Fengzhe Zhou, Zhuo Han, Alan Huang, Songyang Zhang, Kai Chen, Zhixin Yin, Zongwen Shen, et al. 2024. Lawbench: Benchmarking legal knowledge of large language models. In *Proceedings of the 2024 conference on empirical methods in natural language processing*. 7933–7962.

[9] Neel Guha, Julian Nyarko, Daniel Ho, Christopher Ré, Adam Chilton, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, et al. 2023. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Advances in neural information processing systems* 36 (2023), 44123–44279.

[10] Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. Cuad: An expert-annotated nlp dataset for legal contract review. *arXiv preprint arXiv:2103.06268* (2021).

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

[12] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. In *Proceedings of International Conference on Learning Representations*.

[13] Onur Kuru, Ozan Arkan Can, and Deniz Yuret. 2016. Charner: Character-level named entity recognition. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. 911–921.

[14] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*. 260–270.

[15] Qingquan Li, Yiran Hu, Feng Yao, Chaojun Xiao, Zhiyuan Liu, Maosong Sun, and Weixing Shen. 2023. Muser: A multi-view similar case retrieval dataset. In *Proceedings of the 32nd ACM International Conference on information and knowledge management*. 5336–5340.

[16] Bulou Liu, Yiran Hu, Qingyao Ai, Yiqun Liu, Yueyue Wu, Chenliang Li, and Weixing Shen. 2023. Leveraging event schema to ask clarifying questions for conversational legal case retrieval. In *Proceedings of the 32nd ACM international conference on information and knowledge management*. 1513–1522.

[17] Qi Liu, Runze Wu, Enhong Chen, Guandong Xu, Yu Su, Zhigang Chen, and Guoping Hu. 2018. Fuzzy cognitive diagnosis for modelling examinee performance. *ACM Transactions on Intelligent Systems and Technology (TIST)* 9, 4 (2018), 1–26.

[18] Zitao Liu, Guowei Xu, Tianqiao Liu, Weiping Fu, Yubi Qi, Wenbiao Ding, Yujia Song, Chaoyou Guo, Cong Kong, Songfan Yang, et al. 2020. Dolphin: A spoken language proficiency assessment system for elementary education. In *Proceedings of The Web Conference 2020*. 2641–2647.

[19] Yixiao Ma, Yueyue Wu, Qingyao Ai, Yiqun Liu, Yunqiu Shao, Min Zhang, and Shaoping Ma. 2023. Incorporating structural information into legal case retrieval. *ACM Transactions on Information Systems* 42, 2 (2023), 1–28.

[20] M.D. Reckase. 2009. *Multidimensional Item Response Theory*. Springer.

[21] Yunqiu Shao, Yueyue Wu, Yiqun Liu, Jiaxin Mao, Min Zhang, and Shaoping Ma. 2021. Investigating user behavior in legal case retrieval. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 962–972.

[22] Michele Soavi, Nicola Zeni, John Mylopoulos, and Luisa Mich. 2022. From legal contracts to formal specifications: A systematic literature review. *SN Computer Science* 3, 5 (2022), 345.

[23] Bruce Tonn, Dorian Stiefel, John M Scheb II, Colin Glennon, and Hemant Kumar Sharma. 2012. Future of the courts: Fixed, flexible, and improvisational frameworks. *Futures* 44, 9 (2012), 802–811.

[24] Vu Tran, Minh Le Nguyen, and Ken Satoh. 2019. Building legal case retrieval systems with lexical matching and summarization using a pre-trained phrase scoring model. In *Proceedings of the seventeenth international conference on artificial intelligence and law*. 275–282.

[25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[26] Fei Wang, Qi Liu, Enhong Chen, Zhenya Huang, Yuying Chen, Yu Yin, Zai Huang, and Shijin Wang. 2020. Neural Cognitive Diagnosis for Intelligent Education Systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[27] Yiquan Wu, Kun Kuang, Yating Zhang, Xiaozhong Liu, Changlong Sun, Jun Xiao, Yueting Zhuang, Luo Si, and Fei Wu. 2020. De-biased court's view generation with causality. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 763–780.

[28] Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. 2026. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348* (2026).

[29] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388* (2025).

[30] Hai Ye, Xin Jiang, Zhunchen Luo, and Wenhan Chao. 2018. Interpretable charge predictions for criminal cases: Learning to generate court views from fact descriptions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. 1854–1864.

[31] Weikang Yuan, Kaisong Song, Zhuoren Jiang, Junjie Cao, Yujie Zhang, Chengyuan Liu, Jun Lin, Ji Zhang, Kun Kuang, and Xiaozhong Liu. 2026. A multi-agent framework with legal event logic graph for multi-defendant legal judgment prediction. *Information Processing & Management* 63, 1 (2026), 104319.

[32] Linan Yue, Qi Liu, Binbin Jin, Han Wu, Kai Zhang, Yanqing An, Mingyue Cheng, Biao Yin, and Dayong Wu. 2021. Neurjudge: A circumstance-aware neural framework for legal judgment prediction. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 973–982.

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

[34] Lili Zhao, Linan Yue, Yanqing An, Yuren Zhang, Jun Yu, Qi Liu, and Enhong Chen. 2022. Cpee: C ivil case judgment p rediction centering on the trial mode of e ssential e lements. In *Proceedings of the 31st ACM international conference on information & knowledge management*. 2691–2700.

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

[36] Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. How does NLP benefit legal system: A summary of legal artificial intelligence. In *Proceedings of the 58th annual meeting of the association for computational linguistics*. 5218–5230.

[37] Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. JEC-QA: a legal-domain question answering dataset. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 9701–9708.

[38] Xin Zhou, Yating Zhang, Xiaozhong Liu, Changlong Sun, and Luo Si. 2019. Legal intelligence for e-commerce: multi-task learning by leveraging multiview dispute representation. In *Proceedings of the 42nd international acm sigir conference on research and development in information retrieval*. 315–324.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009