Information Processing and Management 59 (2022) 102780

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/ipm

Improving legal judgment prediction through reinforced criminal element extraction

Yougang Lyu [a], Zihan Wang [a], Zhaochun Ren [a,*], Pengjie Ren [a], Zhumin Chen [a], Xiaozhong Liu [b], Yujun Li [a], Hongsong Li [c], Hongye Song [c]

[a] *Shandong University, Qingdao, China*
[b] *Worcester Polytechnic Institute, Worcester, MA, USA*
[c] *Alibaba Group, Hangzhou, China*

ARTICLE INFO

Keywords:
Legal text mining
Legal judgment prediction
Criminal element extraction
Reinforcement learning
Neural networks
Attention mechanism

ABSTRACT

Legal text mining is targeted at automatically analyzing the texts in the legal domain by employing various natural language processing techniques and has attracted enormous attention from the NLP community. As one of the most crucial tasks of legal text mining, Legal Judgment Prediction (LJP) aims to automatically predict judgment results (e.g., applicable law articles, charges, and terms of penalty) according to fact descriptions on law cases and becomes a promising application of artificial intelligence techniques.

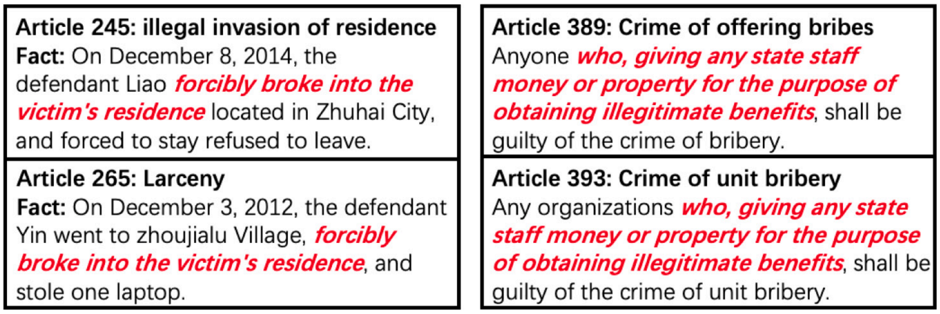
Unfortunately, ambiguous fact descriptions and law articles often appear due to a great number of shared words and legal concepts. Prior works are proposed to partially address these problems, focusing on introducing additional attributes to distinguish similar fact descriptions, or differentiating confusing law articles by grouping and distilling law articles. However, existing works still face two severe challenges: (1) **indistinguishable fact descriptions with different criminals and targets** and (2) **misleading law articles with highly similar TF–IDF representations**, both of which lead to serious misjudgments for the LJP task. In this paper, we present a novel reinforcement learning (RL) based framework, named Criminal Element Extraction Network (CEEN), to handle above challenges simultaneously. In CEEN, we propose four types of discriminative criminal elements, including the *criminal*, *target*, *intentionality*, and *criminal behavior*. To discriminate ambiguous fact descriptions, an reinforcement learning based extractor is designed to accurately locate elements for different cases. To enhance law article predictions, distinctive element representations are constructed for each type of criminal element. Finally, with the input of element representations, a multi-task predictor is adopted for the judgment predictions. Experimental results on real-world datasets show that extracting criminal elements is highly useful for predicting the judgment results.

1. Introduction

In recent years, with the access of massive legal texts, various NLP techniques have been applied to legal text mining field (Giacalone et al., 2018; Ji et al., 2020; K. & Thilagam, 2019; Qazi & Wong, 2019), and legal text mining has become one popular research topic. As one of the most important tasks of legal text mining, Legal Judgment Prediction (LJP) aims to predict

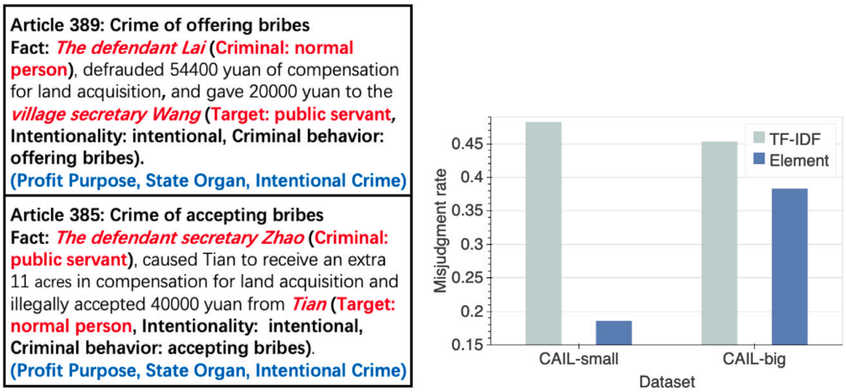
* Corresponding author.
E-mail addresses: youganglyu@gmail.com (Y. Lyu), zihanwang.sdu@gmail.com (Z. Wang), zhaochun.ren@sdu.edu.cn (Z. Ren), jay.ren@outlook.com (P. Ren), chenzhumin@sdu.edu.cn (Z. Chen), xliu14@wpi.edu (X. Liu), liyujun@sdu.edu.cn (Y. Li), hongsong.lhs@alibaba-inc.com (H. Li), hongye.shy@alibaba-inc.com (H. Song).

https://doi.org/10.1016/j.ipm.2021.102780
Received 22 April 2021; Received in revised form 23 September 2021; Accepted 24 September 2021
Available online 3 November 2021
0306-4573/



(a) Confusing fact descriptions (b) Similar law articles

Fig. 1. Examples for the problems of ambiguous fact descriptions and law articles. (left): an example for the confusing fact descriptions. The misleading part "Forcibly broke into ..." appears in both cases, while "stole .." only appears in the larceny case. (right): two similar law articles with shared words and legal concepts. All shown examples are translated from Chinese AI and Law challenge datasets (Xiao et al., 2018) for illustration.



(a) Indistinguishable fact descriptions with different criminals and targets. (b) Misjudgment rates of law articles.

Fig. 2. Intuitive explanations on two problems of existing works. (left): two similar fact descriptions with different criminals and targets. Two cases share many legal attributes (Hu et al., 2018a), including Profit Purpose, State Organ, and Intentional Crime. (right): misjudgment rates of law articles for TF–IDF features and criminal elements (see Section 3.2) respectively.

the judgment result (e.g., law articles, applicable charges, etc.) based on the fact of a case and has received an increasing amount of attention for decades (Aletras et al., 2016; Hu et al., 2018a; Keown, 1980; Kort, 1957; Nagel, 1963; Xu et al., 2020; Zhong et al., 2018). Early studies on LJP focus on statistical solutions (Keown, 1980; Kort, 1957; Lauderdale & Clark, 2012; Nagel, 1963), whereas most recent studies on LJP address the task as a specific classification problem. Various text classification approaches have been applied to address the LJP task (Lin et al., 2012; Liu et al., 2004; Sulea et al., 2017).

However, two challenges still exist: (1) lots of fact descriptions in LJP are ambiguous, and (2) a massive number of charges and their applicable law articles are similar in semantics. Fig. 1 shows examples from a real-world dataset. The left example shows confusing fact descriptions resulting in different applicable law articles and charges, whereas the right example demonstrates two similar law articles, most of which are shared words. Because of the above challenges, existing LJP approaches easily misjudge the law cases (Hu et al., 2018a; Xu et al., 2020).

Few recent works have been proposed to partially address the above problems. Hu et al. (2018a) attempt to extract ten empirical attributes from fact descriptions to distinguish confusing fact descriptions for more accurate prediction of charges. However, they do not consider the criminal and the target of the crime mentioned in fact descriptions, which may lead to different judgment results. To distinguish similar law articles, LADAN (Xu et al., 2020) first groups law articles into multiple communities using their TF–IDF representations, then calculates discriminative features for law articles. After that LADAN aggregates discriminative features of law articles in the same community into single community feature, and finally re-encodes fact descriptions with the community features attentively. However, in LADAN, law articles with similar TF–IDF still share the same community feature instead of having specific discriminative features. Thus it is difficult to distinguish those law articles in LADAN. That is, previous LJP models still suffer from the following two severe problems:

- **Indistinguishable fact descriptions with different criminals and targets.** Hu et al. (2018a) introduces ten additional attributes to discriminate similar fact descriptions to enhance charge predictions. However, criminals and targets in cases still remain unconsidered, which are highly important for law articles and charge predictions. For example, as shown in Fig. 2(a), two similar cases share many same crime attributes (Profit Purpose, State Organ, and Intentional Crime). Nevertheless, different criminals and targets lead to totally distinctive law articles and charges for the two cases in Fig. 2(a). To make legal judgments more accurate, in addition to existing legal attributes, it is of utmost significance to extract criminal and target relevant information from fact descriptions.
- **Misleading law articles with highly similar TF–IDF representations.** As mentioned above, LADAN clusters law articles using their TF–IDF vectors into different communities, learns discriminative features for each law article, and aggregates law article discriminative features in the same community into the overall feature vector. On this basis, LADAN re-encodes the fact descriptions attentively with the overall community features. Since LADAN relys on the overall community features to make judgment predictions, it is difficult for LADAN to distinguish law articles in the same community. In that case, law articles with highly similar TF–IDF representations are easily miscalculated by LADAN. In Fig. 2(b), on two real-world datasets (CAIL-small and CAIL-large) (Xiao et al., 2018; Xu et al., 2020), we calculate misjudgment rates using the same TF–IDF representations as LADAN. The applicable law article of a case is misjudged when this law article is clustered to the identical community with other law articles. We can observe that the misjudgment rates based on TF–IDF features are exceedingly high and close to 0.5 on both datasets. Therefore, more distinctive features are supposed to be employed to facilitate the LJP tasks.

In this paper, we propose a new reinforcement learning (RL) based framework, namely Criminal Element Extraction Network (CEEN), to handle the problems of confusing fact descriptions and law articles in LJP simultaneously. CEEN consists of four components: (1) a fact description encoder, (2) an RL-based element extractor, (3) a criminal element discriminator, and (4) a multi-task judgment predictor. Specifically, the fact description encoder first projects sentences of fact descriptions into latent spaces with the hierarchical Bi-LSTM (Yang et al., 2016). Then, for each case, we explicitly introduce four types of criminal elements, including the *criminal*, *target*, *intentionality* and *criminal behavior* (see Section 3.2 for more details), as bridges between the fact description and applicable law article. To distinguish fairly similar fact descriptions, an RL-based element extractor is employed to elaborately identify sentences, which contain the above criminal elements, for different fact descriptions. To enhance law article predictions, for each law-relevant criminal element, extracted sentences are classified and discriminative representations are carefully learned based on a specially designed attention mechanism. To this end, as demonstrated in Fig. 2(b), comparing to TF–IDF features, the misjudgment rates for criminal elements dramatically drop. Note that, a law article is considered misjudged when having the same criminal elements as other law articles. That is, for law article predictions, criminal elements are much more discriminative than TF–IDF features. Finally, for the inference of legal judgments, a multi-task predictor is established with the input of element representations. Conducted on the real-world datasets, our experimental results verify the effectiveness of our proposed framework. We find that extracting criminal elements is tremendously useful for legal judgment prediction.

To sum up, our contributions are as follows:

- We focus on jointly distinguishing similar law articles and confusing fact descriptions in the legal judgment prediction task. To the best of our knowledge, we are the first to handle the issues of confusing fact descriptions and law articles concurrently.
- We propose a novel reinforced criminal element extraction network (CEEN). In CEEN, a reinforced criminal extractor is applied to differentiate fact descriptions by uncovering distinctive criminal elements (including criminals and targets), whereas an element discriminator is designed to distinguish law articles with similar TF–IDF representations to promote overall law article predictions.
- Extensive experiments on the benchmark dataset verify the effectiveness of our proposed method for LJP. Extracting criminal elements is tremendously useful for predicting the judgment results.
- Ablation studies on model components and criminal elements further demonstrate the superiority of our proposed CEEN. Both RL-based extractor and law-relevant element discriminator enhance the performances, and every criminal element is indispensable for CEEN.

2. Related work

In this section, we survey relevant works along two directions: deep reinforcement learning and legal judgment prediction.

2.1. Legal judgment prediction

Legal text mining aims to automatically analyze the texts in the legal domain by applying various NLP techniques (Giacalone et al., 2018; Ji et al., 2020; K. & Thilagam, 2019; Qazi & Wong, 2019). As Legal Judgment Prediction (LJP) is one of the most important tasks of legal text mining, LJP receives increasing attention in recent years. Early studies on LJP focus on statistical algorithms (Boreham & Niblett, 1976; Keown, 1980; Kort, 1957; Lauderdale & Clark, 2012; Moens & Uyttendaele, 1997; Nagel, 1963). With the development of machine learning technologies, researchers start to formalize LJP as a text classification task (Lin et al., 2012; Liu et al., 2004, 2015; Sulea et al., 2017). However, these traditional methods are limited to specific scenarios with manually crafted features, suffering from serious generalization issues. Recently, neural network-based methods have been proposed to address this problem for LJP. Luo et al. (2017) employ an attention-based neural network to capture interactions between fact descriptions and applicable laws. Zhong et al. (2018) propose a topological multi-task learning framework for LJP, which formalizes the explicit dependencies over subtasks as a directed acyclic graph. To utilize result dependencies among multiple subtasks, Yang et al. (2019) employ a multi-perspective framework with forward predictions and backward verifications. To distinguish confusing fact descriptions, in Few-Shot (Hu et al., 2018a), an attribute-attentive charge prediction model is adopted to infer the attributes and charges concurrently. To discriminates confusing law articles, LADAN (Xu et al., 2020) uses a graph neural network to learn the difference between confusing law articles using their TF–IDF representations. To better apply the pre-trained model to the legal tasks, Xiao et al. (2021) pre-train Lawformer with the masked language modeling (MLM) objective on the legal text.

The most related works are Few-Shot (Hu et al., 2018a) and LADAN (Xu et al., 2020). Few-Shot identifies ten additional legal attributes to strengthen the representations of fact descriptions, while LADAN clusters law articles with TF–IDF representations, and computes distinguished vectors for each community. However, Few-Shot neglects knowledge about the criminal and target of a crime and easily misjudges fact descriptions with different criminals and targets. In the meantime, LADAN may be simply misled by the law articles with fairly similar TF–IDF representations. To this end, our proposed CEEN tackles the problems of above confusing fact descriptions and law articles at the same time. To differentiate indistinguishable fact descriptions, we explicitly define and extract four types of criminal elements, including criminal, target, intentionality, and criminal behavior, for each case. To discriminate confusing law articles, discriminative vectors are elaborately learned for every extracted element and then inputted into the multi-task judgment predictor.

2.2. Deep reinforcement learning

Deep Reinforcement Learning (DRL) concerns with how software agents ought to take actions in an environment to maximize cumulative reward based on a deep neural network. There are several typical methods in the DRL framework including Deep Q-Network (Mnih et al., 2015), Policy Networks (Silver et al., 2016), and actor-critic (Haarnoja et al., 2018). In addition, DRL-based methods have been successfully applied in various types of NLP applications (Chali et al., 2015; Feng et al., 2018a, 2018b; Hu et al., 2018b; Li et al., 2019; Miao et al., 2020; Narasimhan et al., 2015; Wu et al., 2018; Xiong & Ji, 2016). These previous works demonstrate the effectiveness of deep reinforcement learning on NLP tasks and support our proposed CEEN on the LJP tasks. Similar to the instance selection in (Feng et al., 2018b), our element extraction process has two properties of trial-and-error search and delayed feedback, which stimulates us to utilize reinforcement learning techniques.

Some recent works adopt reinforcement learning to enhance interpretability for charge prediction and law prediction. Jiang et al. (2018) employ deep reinforcement learning to extract phrases to form rationales, and then make interpretable charge predictions based on the rationales. Zhong et al. (2020) present QAjudge based on reinforcement learning to model the prediction process and further enhance interpretability. However, the previous RL-based LJP methods are mainly designed to improve the model interpretability by extracting phrase-level rationale or modeling the judgment process, while still suffering from ambiguous fact descriptions and law articles. In contrast, our proposed CEEN is designed to extract the four criminal elements to distinguish confusing fact descriptions and similar law articles.

3. Problem formulation

In this section, we first describe the formulation of the legal judgment prediction (LJP) task and then introduce definitions of criminal elements.

3.1. Legal judgment prediction

Similar to the previous works (Xu et al., 2020; Yang et al., 2019; Zhong et al., 2018), for each law case, a fact description and some judgment results are taken into consideration. The fact description is represented as a text document, which is denoted by f. For each case, we consider three types of judgment results: (1) *applicable law articles*, (2) *charges*, (3) *terms of penalty*. The ith type of judgment result can be represented as a category label y^i, where $y^i \in Y^i$, $i = 1, 2, 3$ and Y^i is the label set for ith judgment result. Given a training dataset $D \triangleq \{(f, y^1, y^2, y^3)_m\}_{m=1}^{n_D}$ of size n_D, our goal is to train a model $\mathbf{F}(\cdot)$ that is able to predict legal judgment results according to the fact description f, i.e., $\mathbf{F}(f) = (\hat{y}^1, \hat{y}^2, \hat{y}^3)$, where $\hat{y}^i \in Y^i$, and $i = 1, 2, 3$.

3.2. Criminal elements

Inspired by Elemental Trial (Cohen, 1982; Quintard-Morénas, 2010; Tadros & Tierney, 2004), we introduce four types of criminal elements: (1) The *criminal* is the person or organization who violates the law and should bear criminal responsibility. (2) The *target* of crime represents a person or thing to which the infringement is directed. (3) *Intentionality* refers to whether a crime is committed intentionally or negligently. (4) The *criminal behavior* is the act that a criminal violates the provisions of the criminal law. For example, the criminal of larceny is a person, who reaches the age of criminal responsibility and has the ability to constitute criminal responsibility. The target of larceny is public or private property. Obviously, larceny is committed intentionally, and the criminal behavior is to steal a large amount of public and private property or repeatedly steal public and private property. In the following sections, we denote element sets for the criminal, target, intentionality and criminal behavior as $\mathcal{E}^1$, $\mathcal{E}^2$, $\mathcal{E}^3$, and $\mathcal{E}^4$, respectively. For each law case, we aim to predict kth type of criminal element category label $\hat{z}^k$, where $\hat{z}^k \in \mathcal{E}^k$ and $k = 1, 2, 3, 4$, based on the fact description.

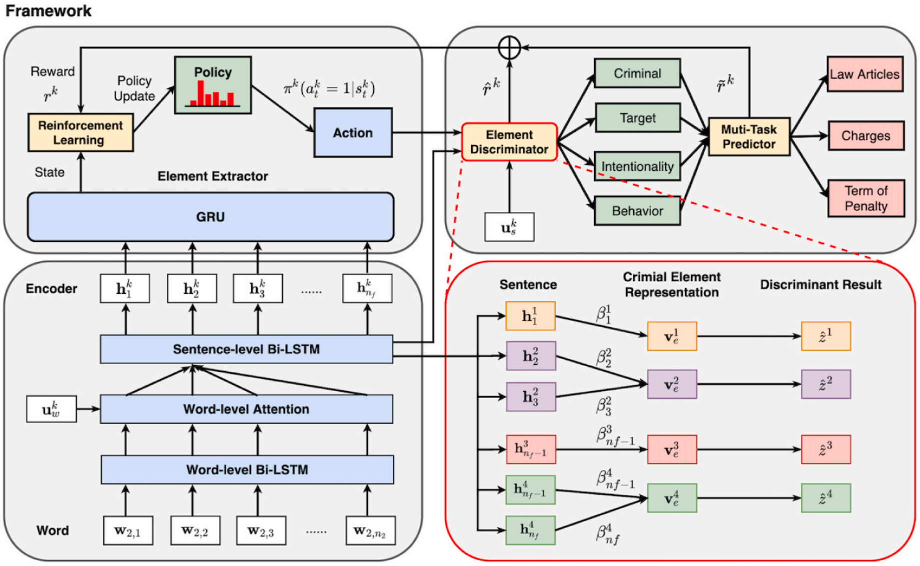


Fig. 3. Overview of our proposed CEEN. CEEN consists of four components: (1) a fact description encoder (2) an RL-based element extractor (3) a criminal element discriminator (4) a multi-task judgment predictor.

4. Method

In this section, we describe our proposed method, CEEN, in detail. First, we give an overview of CEEN in Section 4.1. Then, we describe the fact description encoder (Section 4.2), RL-based element extractor (Section 4.3), and criminal element discriminator (Section 4.4). Next, judgment results are predicted by a multi-Task judgment predictor (Section 4.5). Finally, a two-stage training process for CEEN is comprehensively explained (Section 4.6).

4.1. Overview

As Fig. 3 shows, our proposed CEEN contains four components: (1) a fact description encoder, (2) an RL-based element extractor, (3) a criminal element discriminator, and (4) a multi-task judgment predictor. First, we utilize the hierarchical Bi-LSTM (Yang et al., 2016) as the fact description encoder to generate contextual sentence representations. Next, to distinguish confusing fact description, we use an RL-based element extractor to elaborately select sentences that contain criminal elements. Then, to enhance the law article prediction, we employ a criminal element discriminator to generate the kth discriminative criminal element representation v_e^k by fusing the contextual representations of selected sentences, where $k = 1, 2, 3, 4$. After that, the concatenation of four discriminative criminal element representations $[v_e^1, v_e^2, v_e^3, v_e^4]$ is fed into the multi-task judgment predictor to precisely predict judgment results. At last, we describe the two-stage training process of CEEN in detail.

4.2. Fact description encoder

To model contextual sentence representations in the fact description, we employ four hierarchical Bi-LSTMs. Each hierarchical Bi-LSTM has two levels of Bi-LSTM, the word-level Bi-LSTM outputs contextual word representations, then the basic sentence representations are calculated by attention mechanism, and finally the sentence-level Bi-LSTM outputs the contextual sentence representations. Each fact description has multiple sentences, and each sentence has multiple words. More specifically, for each input sentence $S_i = [\mathbf{w}_{i,1}, \dots, \mathbf{w}_{i,n_i}]$ in the fact description f, where n_i refers to the number of words in the ith sentence, the word-level Bi-LSTM outputs a contextual word representations sequence, that is,

$$\mathbf{h}_{i,j}^k = [\overrightarrow{\text{LSTM}}\left(\mathbf{w}_{i,j}, \Theta_{LSTM_w}^k\right), \overleftarrow{\text{LSTM}}\left(\mathbf{w}_{i,j}, \Theta_{LSTM_w}^k\right)], \tag{1}$$

where $\mathbf{h}_{i,j}^k \in \mathbb{R}^{d_w}$, $\mathbf{w}_{i,j}$ is the jth basic word embedding in S_i and initialized by the Skip–Gram model (Mikolov et al., 2013). $\Theta_{LSTM_w}^k$ denotes learnable parameters of the kth word-level Bi-LSTM, and $k = 1, 2, 3, 4$. Based on the contextual word representations sequence and the word-level context vector u_w^k, we compute a word-level attentive vector $[\beta_{i,1}^k, \dots, \beta_{i,n_i}^k]$ with the word-level attention mechanism as:

$$\beta_{i,j}^k = \frac{exp(tanh(\mathbf{W}_w^k \mathbf{h}_{i,j}^k)^T u_w^k)}{\sum_{j'} exp(tanh(\mathbf{W}_w^k \mathbf{h}_{i,j'}^k)^T u_w^k)}, \tag{2}$$

where $\mathbf{W}_w^k$ is a trainable weight matrix for word-level attention mechanism. Then, we obtain a basic representation of the sentence S_i, that is,

$$\mathbf{v}_{S_i}^k = \sum_{j=1}^{n_i} \beta_{i,j}^k \mathbf{h}_{i,j}^k, \tag{3}$$

Based on the word-level attention mechanism, we obtain basic sentence representations sequence $[\mathbf{v}_{S_1}^k, \mathbf{v}_{S_2}^k, \dots, \mathbf{v}_{S_{n_f}}^k]$, where n_f denotes the sentence number in the fact description f. Then, we apply the sentence-level Bi-LSTM to compute the contextual sentence representations sequence, that is,

$$\mathbf{h}_i^k = [\overrightarrow{\text{LSTM}}\left(\mathbf{v}_{S_i}^k, \Theta_{LSTM_s}^k\right), \overleftarrow{\text{LSTM}}\left(\mathbf{v}_{S_i}^k, \Theta_{LSTM_s}^k\right)], \tag{4}$$

where $\mathbf{h}_i^k \in \mathbb{R}^{d_s}$, $\Theta_{LSTM_s}^k$ denotes learnable parameters of the kth sentence-level Bi-LSTM, and $k = 1, 2, 3, 4$.

4.3. RL-based element extractor

To distinguish confusing fact descriptions, we employ four agents to respectively select sentences containing different types of the criminal element. Each agent adopts a stochastic policy π^k to sample action at each state. According to the sampled action, the kth agent determines if the sentence contains the kth type of criminal element, where $k = 1, 2, 3, 4$. We demonstrate action, state, policy, and reward as follows:

Action: The action a_t^k is selected from action space $\{positive = 1, negative = 0\}$, where *positive* means the sentence contains the kth type of criminal element, and *negative* indicates the sentence that it does not contain the kth type of criminal element.

State: We use a recurrent neural network to encode the state representation:

$$\mathbf{s}_t^k = \text{GRU}(\mathbf{s}_{t-1}^k, \mathbf{h}_t^k, \Theta_{GRU}^k), \tag{5}$$

where $\mathbf{h}_t^k$ is the contextual representation of the tth sentence S_t, $t \in [1, n_f]$, Θ_{GRU}^k denotes learnable parameters of the kth GRU, and $k = 1, 2, 3, 4$. Note that at the beginning ($t = 1$), the state s_0^k is a trainable vector.

Policy: The stochastic policy for the kth type of criminal element extraction which specifies a probability distribution over actions:

$$\pi^k(a_t^k | s_t^k, \Theta_{policy}^k) = sigmoid(\mathbf{W}_s^k \mathbf{s}_t^k + b_s^k), \tag{6}$$

where $\Theta_{policy}^k = \{\mathbf{W}_s^k, b_s^k\}$ denotes learnable parameters of the kth type of criminal element extraction policy.

Reward: As there is no annotation on which sentences contain the criminal elements, we design two kinds of delayed rewards to measure the correctness of the kth type of criminal element extraction once all the sentences in fact description are scanned, $k = 1, 2, 3, 4$. Specifically, the criminal element discriminator provides the first delayed reward r_d^k for the kth agent by simply measuring the kth type of criminal element discrimination error over gold-standard annotations:

$$r_d^k = \begin{cases} 0, if\ \ argmax_m(\hat{z}_m^k) \neq argmax_m(z_m^k) \\ 1, if\ \ argmax_m(\hat{z}_m^k) = argmax_m(z_m^k), \end{cases} \tag{7}$$

where $\mathbf{z}^k = [z_1^k, z_2^k, \dots, z_{|\mathcal{E}^k|}^k]$ denotes the ground truth vector of the kth type of criminal element, $\hat{\mathbf{z}}^k$ refers to the discrimination result of the kth type of criminal element, $m \in [1, |\mathcal{E}^k|]$ and $|\mathcal{E}^k|$ refers to the number of different classes for the kth type of criminal element discrimination. The correctness of the four law-relevant criminal elements can affect the result of the law article prediction task. Therefore, the law article predictor provides the second delayed reward r_p for all agents by simply measuring the law article prediction error over gold-standard annotation:

$$r_p = \begin{cases} 0, if\ \ argmax_m(\hat{y}_m^1) \neq argmax_m(y_m^1) \\ 1, if\ \ argmax_m(\hat{y}_m^1) = argmax_m(y_m^1), \end{cases} \tag{8}$$

where $\mathbf{y}^1 = [y_1^1, y_2^1, \dots, y_{|Y^1|}^1]$ refers to the ground truth vector of law article prediction, $\hat{\mathbf{y}}^1$ denotes the prediction result of law article, $m \in [1, |Y^1|]$, and $|Y^1|$ is the number of different classes for law article prediction. The final reward for the kth agent is computed by the sum of two delayed rewards, that is:

$$r_{fin}^k = r_d^k + r_p. \tag{9}$$

4.4. Criminal element discriminator

To enhance law article prediction, we fuse the contextual representations of sentences which contain the criminal elements to generate the kth discriminative criminal element representation v_e^k, where $k = 1, 2, 3, 4$ and $\mathbf{v}_e^k \in \mathbb{R}^{d_e}$. Based on the actions sequence $\mathbf{a}^k = [a_1^k, a_2^k, \dots, a_{n_f}^k]$, contextual sentence representations and the sentence-level context vector u_s^k, we compute a sentence-level attentive vector $[\beta_1^k, \beta_2^k, \dots, \beta_{n_f}^k]$ with sentence-level attention mechanism as:

$$\beta_i^k = \frac{a_i^k exp(tanh(\mathbf{W}_s^k \mathbf{h}_i^k)^T u_s^k)}{\sum_{i'} a_{i'}^k exp(tanh(\mathbf{W}_s^k \mathbf{h}_{i'}^k)^T u_s^k)}, \tag{10}$$

where $\mathbf{W}_s^k$ is a trainable weight metric for sentence-level attention mechanism. Note that if all actions in sequence $\mathbf{a}^k$ are 0, we set $\beta_i^k = 0$. Then, the representation of the kth type of criminal element is computed as:

$$\mathbf{v}_e^k = \sum_{i=1}^{n_f} \beta_i^k \mathbf{h}_i^k, \tag{11}$$

where $v_e^k \in \mathbb{R}^{d_s}$ and n_f refers to the number of sentences in the fact description. To obtain the kth criminal element discrimination result, we use a multi-class classifier.

$$\hat{z}^k = softmax(\mathbf{W}_d^k \mathbf{v}_e^k + \mathbf{b}_d^k), \tag{12}$$

where $\mathbf{W}_d^k$ and $\mathbf{b}_d^k$ are parameters specific to the kth type of criminal element discrimination.

4.5. Multi-task judgment predictor

To accurately predict the judgment results, we concatenate four element representations as the representation of fact description f, i.e., $\mathbf{v}_f = [\mathbf{v}_e^1, \mathbf{v}_e^2, \mathbf{v}_e^3, \mathbf{v}_e^4]$. Based on $\mathbf{v}_f$, we generate a specific fact representation $\mathbf{v}_f^i$ for ith legal judgment prediction, where $\mathbf{v}_f^i \in \mathbb{R}^{d_f}$ and $i = 1, 2, 3$.

$$v_f^i = \mathrm{ReLU}(\mathbf{W}_f^i \mathbf{v}_f + \mathbf{b}_f^i), \tag{13}$$

where $\mathbf{W}_f^i$ and $\mathbf{b}_f^i$ are learnable parameters specific to the ith legal judgment prediction. To obtain the legal judgment prediction result, we employ a linear classifier:

$$\hat{y}^i = softmax(\mathbf{W}_p^i \mathbf{v}_f^i + \mathbf{b}_p^i), \tag{14}$$

where $\mathbf{W}_p^i$ and $\mathbf{b}_p^i$ are parameters specific to the ith legal judgment prediction.

4.6. Training

In this section, we introduce two-stage training processes of the model, the first stage is the training process for the RL-based element extractor, and the second stage is the training process for the criminal element discriminator and the multi-task predictor. In the overall training process, the two training processes iteratively optimize different modules of the model.

Training Process for Extraction To optimize the policy of the criminal element extractor, we aim to maximize the expected cumulative rewards from the kth type of criminal element extraction at each time step t as the agent samples trajectories following the policy π^k, which can be computed as follows:

$$\mathcal{J}(\Theta_{\pi^k}) = \mathbb{E}_{a^k \sim \pi^k(a^k|s^k)}[\sum_{m=t}^{n_f} \gamma^{m-t} r_{fin}^k], \tag{15}$$

where $\Theta_{\pi^k} = \{\Theta_{GRU}^k, \Theta_{policy}^k\}$, γ is a discount factor in RL, n_f denotes the number of sentences in the fact description, and the whole sampling process π^k takes n_f time steps before it terminates.

By decomposing the cumulative rewards into a Bellman equation, we obtain:

$$\mathcal{R}^{\pi^k}(s_t^k, a_t^k) = \mathbb{E}[r_{final}^k + \gamma \mathcal{R}^{\pi^k}(s_{t+1}^k, a_{t+1}^k) | s_t^k, a_t^k]. \tag{16}$$

According to the policy gradient method (Sutton et al., 1999) and the REINFORCE algorithm (Williams, 1992), we compute the gradient for the kth type of criminal element extraction policy π^k as:

$$\begin{aligned} \nabla_{\Theta_{\pi^k}} \mathcal{J}(\Theta_{\pi^k}) =& \mathbb{E}_{a^k \sim \pi^k(a^k|s^k)}[\mathcal{R}^{\pi^k}(s_t^k, a_t^k) \\ & \nabla_{\Theta_{\pi^k}} \log \pi^k(a^k|s_t^k)]. \end{aligned} \tag{17}$$

Training Process for Discrimination and Prediction In order to train criminal element discriminator and multi-task judgment predictor, we compute the cross-entropy loss function for criminal element discrimination and legal judgment prediction. The loss of the criminal element discrimination is formally computed as:

$$\mathcal{L}_d = -\sum_{k=1}^{4} \sum_{m=1}^{|\mathcal{E}^k|} z_m^k \log(\hat{z}_m^k), \tag{18}$$

where $|\mathcal{E}^k|$ denotes the number of different classes for kth type of criminal element, z^k is the ground truth of prediction for kth criminal element and $k = 1, 2, 3, 4$. Meanwhile, the loss of legal judgment prediction is formally calculated as:

$$\mathcal{L}_p = -\sum_{i=1}^{3} \sum_{m=1}^{|Y^i|} y_m^i \log(\hat{y}_m^i), \tag{19}$$

Algorithm 1: Training Process of CEEN

1 **Input**:CEEN Model M
2 Initialize M with parameters $\{\Theta_{ext}, \Theta_{enc+dis+pre}\}$ randomly
3 **while** *not convergence* **do**
4   **Training Process for Extraction:**
5   Freeze $\Theta_{enc+dis+pre}$;
6   **for** $k \leftarrow 1$ ***to*** 4 **do**
7   **for** $t \leftarrow 1$ ***to*** n_f **do**
8   Calculate s_t^k by Eq. (5);
9   Sample a_t^k from $\pi^k(a_t^k|s_t^k;\Theta_{policy}^k)$ by Eq. (6);
10   **end**
11   **end**
12   Calculate $\{\hat{z}^k\}_{k=1}^4$ and $\{\hat{y}^i\}_{i=1}^3$ by Eq. (10) (14) respectively;
13   Obtain final reward by Eq. (7) (8) (9);
14   Update Θ_{ext} via policy gradient Eq. (17);
15   **Training Process for Discrimination and Prediction:**
16   Freeze Θ_{ext};
17   **for** $k \leftarrow 1$ ***to*** 4 **do**
18   **for** $t \leftarrow 1$ ***to*** n_f **do**
19   Calculate s_t^k by Eq. (5);
20   Sample a_t^k from $\pi^k(a_t^k|s_t^k;\Theta_{policy}^k)$ by Eq. (6);
21   **end**
22   **end**
23   Calculate $\{\hat{z}^k\}_{k=1}^4$ and $\{\hat{y}^i\}_{i=1}^3$ by Eq. (10) (14) respectively;
24   Update $\Theta_{enc+dis+pre}$ by Eq. (18) (19) (20);
25 **end**

where $|Y^i|$ denotes the number of different classes for ith legal judgment prediction, y^i is the ground truth of prediction for ith legal judgment and $i = 1, 2, 3$. Finally, the final loss is defined as:

$$\mathcal{L} = \mathcal{L}_d + \mathcal{L}_p. \tag{20}$$

Overall Training Process. The first training process for extraction aims to train the RL-based element extractor, and the second training process for discrimination and prediction aims to train the criminal element discriminator and the multi-task judgment predictor. In the training process for extraction, we only optimize the parameters Θ_{ext} of the criminal element discriminator and multi-task judgment predictor, where $\Theta_{ext} = \{\Theta_{\pi^k}\}_{k=1}^4$. Specifically, as described in lines 4 to 14, we first freeze the parameters of the fact description encoder, the criminal element discriminator, and the multi-task judgment predictor $\Theta_{enc+dis+pre} = \{\Theta_{enc}, \Theta_{dis}, \Theta_{pre}\}$; and then optimize the parameters of RL-based element extractor by Eq. (17). In the training process for discrimination and prediction, we only optimize the parameters $\Theta_{enc+dis+pre}$ of the criminal element discriminator and multi-task judgment predictor. As described in lines 15–24 of Algorithm 1, we first freeze the parameters Θ_{ext}, and then we optimize the parameters $\Theta_{enc+dis+pre}$ by Eq. (18)(20). The overall training process is described in Algorithm 1.

5. Experiments

5.1. Research questions

We aim to answer the following research questions:

(RQ1) What is the learning tendency of CEEN during training? Is there any conflict among all three subtasks during the training process? (Section 6.1)

(RQ2) Does CEEN outperform state-of-the-art baselines on the legal judgment prediction task? (Section 6.2)

5.2. Datasets

Following prior works (Xu et al., 2020; Yang et al., 2019; Zhong et al., 2018), we conduct experiments on two datasets of the Chinese AI and Law challenge (CAIL2018) (Xiao et al., 2018), including CAIL-small (the exercise stage dataset) and CAIL-big (the first stage dataset). Each case in datasets contains one paragraph of fact descriptions, applicable law articles, charges, and the terms of penalty. Similar to Xu et al. (2020), we filter out cases with fewer than 10 meaningful words, multiple applicable law articles, or

Table 1
Statistics of CAIL-small and CAIL-big.

Dataset	CAIL-small	CAIL-big
Training Set Cases	101,619	1,587,979
Test Set Cases	26,749	185,120
Law Articles	103	118
Charges	119	130
Term of Penalty	11	11
Criminal	7	11
Target	75	76
Intentionality	2	2
Criminal behavior	63	71

multiple charges. In addition, infrequent charges and law articles that appear no more than 100 times are removed from datasets. And terms of penalty are divided into the same non-overlapping intervals as Zhong et al. (2018).

To distinguish confusing fact descriptions and enhance the prediction of law articles, we introduce four types of crime elements for all law articles in the datasets. For each (law article, criminal element) pair, it can be label as one-hot vector z^k, $k = 1, 2, 3, 4$. In our experiments, we only annotate the *criminal*, *target*, *intentionality*, and *criminal behavior* of 118 law articles in CAIL-big manually, since all the law articles in CAIL-small have appeared in CAIL-big. Then, we assign each case with the same criminal elements of its corresponding law article. The detailed statistics are demonstrated in Table 1.

5.3. Baselines

In this section, we compare CEEN and pre-trained CEEN$_{\text{BERT}}$ with various typical text classification models, state-of-the-art judgment prediction methods and a pre-trained model, including:

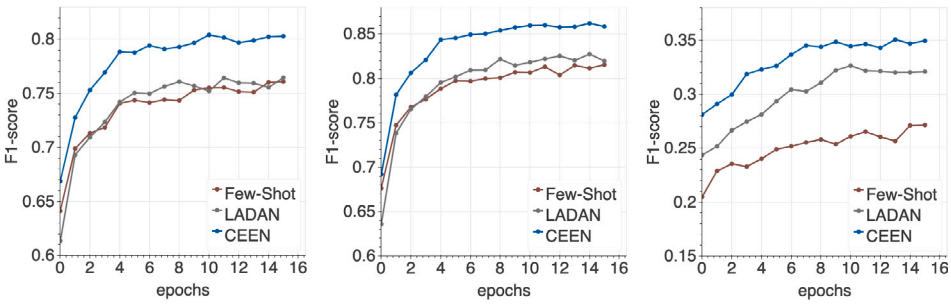
- **CNN** (Kim, 2014): a CNN-based text classifier with multiple filter window widths.
- **HARNN** (Yang et al., 2016): an RNN-based document classification method with a hierarchical attention mechanism.
- **FLA** (Luo et al., 2017): a rule-based charge prediction method that models correlations between fact descriptions and related law articles.
- **Few-Shot** (Hu et al., 2018a): an attribute-attentive prediction model that infers ten additional discriminative attributes and charges simultaneously.
- **TOPJUDGE** (Zhong et al., 2018): a topological multi-task learning framework for LJP, which formalizes the explicit dependencies over subtasks as a directed acyclic graph.
- **MPBFN** (Yang et al., 2019): a multi-perspective framework that utilizes result dependencies among multiple subtasks with forward predictions and backward verifications.
- **LADAN** (Xu et al., 2020): a graph neural network based method that automatically captures subtle differences among confusing law articles.
- **BERT** (Cui et al., 2019): a Transformer-based method which is pre-trained on Chinese wikipedia documents.

Following the existing works (Luo et al., 2017; Xu et al., 2020; Zhong et al., 2018), we train CNN, HARNN, FLA, and Few-Shot using the multi-task framework (MTL). For MPBFN, we employ the improved variant, MPBFN-WCA (Yang et al., 2019), which adds a number embedding method and a word collocation attention mechanism to enhance the performances. Note that, to exclude the influence of learning frameworks, we incorporate LADAN only with MTL in Section 6.2. For BERT, we individually fine-tune the model on above two datasets. More specifically, we insert one [CLS] token at the start of the fact description, then we select the representation of [CLS] token as the representation of the whole fact description, and finally the representation of [CLS] token is fed into the multi-task judgment predictor to predict judgment results. Different from BERT, for CEEN$_{\text{BERT}}$, in order to extract sentences containing criminal elements, we are inspired by Liu and Lapata (2019) to insert additional [CLS] tokens at the beginning of each sentence, then each [CLS] token gathers features for the sentence following it, and finally the representation of each [CLS] token is fed into the RL-based element extractor and the criminal element discriminator. In addition, we further explore the effects of different multi-task schemes in the Ablation Studies (Section 7.1).

5.4. Experimental settings

For models without Transformer-based encoder, we use the THU-LAC tool (Sun et al., 2016) for word segmentation. Word embeddings are initialized by the Skip-Gram model (Mikolov et al., 2013) with 200 embedding dimensions. In that case, the frequency threshold is set to 25. For models with Transformer-based encoder, we use the pre-trained weights and the hidden size is 768. We set the maximum document length as 512 words for CNN-based and Transformer-based models in baselines, and set the maximum sentence length to 100 words and maximum document length to 15 sentences for LSTM-based models. Meanwhile, following Xu et al. (2020), we set the dimensions of all latent states (i.e., d_w, d_s, d_e and d_f) to 256 for LSTM-based models.

In the overall training process, we adopt Adam (Kingma & Ba, 2015) as the optimizer. We set the learning rate of the training process for extraction to 10^{-3}, the learning rate of the training process for discrimination and prediction is set to 10^{-4}, the learning



(a) *Law Articles* (b) *Charges* (c) *Terms of Penalty*

Fig. 4. Learning curves on CAIL-small.

Table 2
Judgment prediction results on CAIL-small. * indicates statistical significance compared to LADAN with $p < 0.05$ by t-test. † indicates statistical significance compared to LADAN and BERT with $p < 0.05$ by t-test.

	Law articles				Charges				Term of penalty			
Method	Acc.	MP	MR	F1	Acc.	MP	MR	F1	Acc.	MP	MR	F1
FLA	77.74	75.32	74.36	72.93	80.90	79.25	77.61	76.94	36.48	30.94	28.40	28.00
CNN	78.71	76.02	74.87	73.79	82.41	81.51	79.34	79.61	35.40	33.07	29.26	29.86
HARNN	79.79	75.26	76.79	74.90	83.80	82.44	82.78	82.12	36.17	34.66	31.26	31.40
Few-Shot	79.30	77.80	77.59	76.09	83.65	80.84	82.01	81.55	36.52	35.07	26.88	27.14
TOPJUDGE	79.88	79.77	73.67	73.60	82.10	83.60	78.42	79.05	36.29	34.73	32.73	29.43
MPBFN	79.12	76.30	76.02	74.78	82.14	82.28	80.72	80.72	36.02	31.94	28.60	29.85
LADAN	81.20	78.24	77.38	76.47	85.07	83.42	82.52	82.74	38.29	36.16	32.49	32.65
BERT	83.85	81.85	82.61	81.13	88.17	88.10	87.76	87.49	43.11	39.64	38.74	38.47
CEEN	82.50*	81.38*	81.82*	80.41*	86.83*	86.61*	86.86*	86.22*	40.04*	37.11*	35.00*	35.07*
$\text{CEEN}_{\text{BERT}}$	**84.20**†	**83.21**†	**83.73**†	**82.32**†	**89.44**†	**88.88**†	**89.01**†	**88.64**†	**43.80**†	**41.15**†	**39.72**†	**40.03**†

rate of Transformer-based methods is set to 5×10^{-5}. Meanwhile, the dropout probability, discount factor γ, and batch size are set to 0.5, 1, and 128, respectively. All models are trained for 16 epochs and then evaluated on the test set.

We employ accuracy (Acc.), macro-precision (MP), macro-recall (MR) and macro-F1 (F1) metrics to evaluate the performance.

6. Experimental results

In this section, to answer the research questions (RQ1 and RQ2), development experiments and legal judgment predictions are conducted.

6.1. Development experiments

For RQ1, we study the training processes of CEEN and baselines. Notably, we investigate the improvement trends and potential conflicts among subtasks (predictions for law articles, charges, and terms of penalty). The learning curves against training epochs are demonstrated in Fig. 4. Few-shot and LADAN are both state-of-the-art models for LJP. We can draw the following conclusions that:

- F1-scores of CEEN and baselines improve continually as the number of training epochs increases from 1 to 10. All models converge to a stable level when the epochs number increases to 10.
- Three subtasks in the LJP task have the same improvement trend, indicating that there exist no potential conflicts among the inference process of law articles, charges, and terms of penalty.
- Our proposed CEEN significantly outperforms both of Few-Shot and LADAN at all times during the training process, which demonstrates the effectiveness of the proposed CEEN.

6.2. Judgment prediction results

Then, we turn to RQ2. Here, we evaluate the CEEN performance on three LJP subtasks, including the predictions of law articles, charges, and terms of penalty. As Tables 2 and 3 show, we conduct evaluations on both the CAIL-small and CAIL-big datasets. We can conclude that:

Table 3
Judgment prediction results on CAIL-big. * indicates statistical significance compared to LADAN with $p < 0.05$ by t-test. † indicates statistical significance compared to LADAN and BERT with $p < 0.05$ by t-test.

	Law articles				Charges				Term of penalty			
Method	Acc.	MP	MR	F1	Acc.	MP	MR	F1	Acc.	MP	MR	F1
FLA	93.23	72.78	64.30	66.56	92.76	76.35	68.48	70.74	57.63	48.93	45.00	46.54
CNN	95.84	83.20	75.31	77.47	95.74	86.49	79.00	81.37	55.43	45.13	38.85	39.89
HARNN	95.63	81.48	74.57	77.13	95.58	85.59	79.55	81.88	57.38	43.50	40.79	42.00
Few-Shot	96.12	85.43	80.07	81.49	96.04	88.30	80.46	83.88	57.84	47.27	42.55	43.44
TOPJUDGE	95.85	84.84	74.53	77.50	95.78	86.46	78.51	81.33	57.34	47.32	42.77	44.05
MPBFN	96.06	85.25	74.82	78.36	95.98	89.16	79.73	83.20	58.14	45.86	39.07	41.39
LADAN	96.57	86.22	80.78	82.36	96.45	88.51	83.73	85.35	59.66	51.78	45.34	46.93
BERT	96.92	86.90	83.84	84.55	96.80	89.46	86.45	87.35	61.57	51.97	50.37	50.86
CEEN	96.84*	88.26*	80.54	82.93*	96.74*	90.61*	83.79*	86.18*	60.05*	51.84*	45.62*	46.70
CEEN$_{BERT}$	**97.43**†	**89.14**†	**86.15**†	**86.88**†	**97.40**†	**91.34**†	**89.04**†	**89.86**†	**62.89**†	**54.85**†	**52.39**†	**53.20**†

- Compared with typical text classification models and state-of-the-art legal judgment prediction methods, CEEN and CEEN$_{BERT}$ significantly improve the performances on both of the CAIL-small and CAIL-big datasets. Furthermore, CEEN$_{BERT}$ significantly outperforms BERT on all subtasks and datasets. These experimental results indicate that extracting criminal elements is extremely helpful for legal judgment prediction.
- Comparing to Few-Shot, CEEN obtains remarkable improvements on all the tasks. It is because Few-Shot extracts limited charge attributes from fact descriptions and falls short in distinguishing confusing law articles. Likewise, comparing to LADAN, CEEN is capable of differentiating far more similar law articles and fact descriptions. Therefore, CEEN surpasses LADAN dramatically in most cases. The above two findings also prove the effectiveness of our proposed method.
- As shown in Tables 2 and 3, BERT is able to accurately predict the judgment results. Because of complex pre-training objectives and large model parameters, BERT effectively captures rich language knowledge from massive unlabeled text and achieves considerable performance improvements on LJP. Meanwhile, the combination of CEEN and BERT significantly outperforms all baselines including BERT, indicating that the language knowledge obtained from pre-training and the information of criminal elements are complementary to each other.
- Since the training data for CAIL-big is more sufficient than CAIL-small, all models perform better on all three subtasks on CAIL-big. Meanwhile, in contrast to law articles and charge predictions, all models are not able to predict terms of penalty effectively. Judgments on the terms of penalty may be influenced by many factors, and thus much more challenging to infer.

In summary, the RL-based crime element extractor and law-relevant element discriminator are able to promote the performances. That is, extracting law-relevant elements (criminal, target, intentionality, and criminal behavior) from fact descriptions and then distinguishing crucial elements for judgment predictions are vastly beneficial to the LJP task.

7. Analysis

In this section, we dive deep into the performances of CEEN. In Section 7.1, we investigate how the RL-based criminal element extractor and law-relevant discriminator contribute to the improvements, explore the influence of criminal elements, and analyze the impacts of multi-task frameworks. In addition, the influences of training data amount are further explored (Section 7.2), influences of pre-trained models are evaluated (Section 7.3), and a case study are also conducted for intuitive comparisons among CEEN and baselines (Section 7.4). Finally, we collect bad cases, and give an error analysis for CEEN (Section 7.5).

7.1. Ablation studies

As Table 4 shows, we conduct ablation studies on the CAIL-small dataset. To examine the contributions of the RL-based element extractor and law-relevant element discriminator to the performances, we consider the following three model variants:

- **-Ext**: to prove the effectiveness of the RL-based element extractor, we establish a CEEN model with the extractor removed. In that case, the actions of the extractor are set to 1.
- **-Dis**: to demonstrate the competence of the law-relevant element discriminator, a CEEN model without the discriminator is built. To this end, the RL-based element extractor only receives rewards from the multi-task judgment predictor, and the loss function $\mathcal{L}_d$ is removed.
- **-Ext-Dis**: to evaluate the significance of detecting law-relevant elements, we remove the extractor and discriminator from CEEN at the same time. In that case, CEEN is degraded to HARNN (Yang et al., 2016) and predicts the legal judgments with a hierarchical attention mechanism.

Ablation studies on model components. In Table 4, results show that both components (RL-based element extractor and law-relevant element discriminator) enhance the performances of CEEN. When removing extractor as well as discriminator, the accuracy of CEEN decreases significantly, which further points out the importance of extracting and discriminating law-relevant elements.

Table 4
Ablation studies on CAIL-small.

Models	Law		Charge		Penalty	
	Acc.	F1	Acc.	F1	Acc.	F1
CEEN	**82.50**	**80.41**	**86.83**	**86.22**	**40.04**	**35.07**
-Ext	81.39	78.60	85.43	85.01	39.95	34.50
-Dis	81.40	78.83	85.22	84.54	38.05	32.56
-Ext-Dis	79.71	75.01	83.75	82.15	36.04	31.52
-Cr	82.00	79.46	85.79	85.35	39.91	34.52
-Tar	81.03	79.13	85.89	85.04	39.77	34.64
-Int	81.24	79.06	85.88	85.07	39.84	33.30
-Beh	81.23	79.01	86.14	85.20	39.02	33.84
-Cr-Tar	81.11	78.65	85.10	84.82	39.15	33.38
-Cr-Tar-Int	80.80	78.22	85.00	84.71	38.43	32.95
-Cr-Tar-Int-Beh (-Ext-Dis)	79.71	75.01	83.75	82.15	36.04	31.52
CEEN (MTL)	**82.50**	**80.41**	**86.83**	**86.22**	**40.04**	**35.07**
CEEN+TOPJUDGE	81.37	78.37	84.89	84.74	39.99	34.10
LADAN (MTL)	81.20	76.47	85.07	82.74	38.29	32.65
LADAN+TOPJUDGE	81.53	77.10	85.12	83.14	38.34	33.53
LADAN+MPBFN	82.34	76.80	84.83	82.85	39.35	34.05

Ablation studies on criminal elements. To demonstrate the effectiveness of extracting criminal elements, we consider two settings for criminal element ablation study. One is removing the element extractor and element discriminator corresponding to each criminal element individually, the other is removing the element extractor and element discriminator corresponding to each criminal element progressively. In our experiments, we use **Cr** for *criminal*, **Tar** for *target*, **Int** for *intentionality*, and **Beh** for *criminal behavior*. Specifically, **-Cr** removes the element extractor and element discriminator corresponding to *criminal*; **-Tar** removes the element extractor and element discriminator corresponding to *target*; **-Int** removes the element extractor and element discriminator corresponding to *intentionality*; **-Beh** removes the element extractor and element discriminator corresponding to *criminal behavior*; **-Cr-Tar** removes the element extractor and element discriminator corresponding to *criminal* and *target*; **-Cr-Tar-Int** removes the element extractor and element discriminator corresponding to *criminal*, *target* and *intentionality*; **-Cr-Tar-Int-Beh** removes the element extractor and element discriminator corresponding to *criminal*, *target*, *intentionality* and *criminal behavior*.

In Table 4, results show that each criminal element (*criminal*, *target*, *intentionality*, and *criminal behavior*) enhance the performances of CEEN. In summary, in the case of two ablation experimental settings (individually and progressively), both accuracy and F1 of CEEN decrease, which proves the importance of each criminal element.

Ablation studies on multi-task frameworks. Following Xu et al. (2020), we incorporate CEEN with various multi-task frameworks, including MTL, TOPJUDGE (Zhong et al., 2018), and MPBFN (Yang et al., 2019). The original CEEN and LADAN are based on the multi-task framework (MTL). As shown in Table 4, CEEN based on MTL achieves the best performance overall three tasks. In addition, CEEN+MTL outmatches LADAN incorporated with any of the above three multi-task frameworks. Note that, CEEN+MPBFN overfits very easily since the model has a larger number of parameters. For this reason, we do not report the results of CEEN+MPBFN in our experiments.

7.2. Influences of training data amount

In the further, we investigate influences of training data amount over three LJP subtasks on CAIL-small. We compare F1-scores of Few-Shot, LADAN, and CEEN in 16 training epochs. The results are shown in Fig. 5. We can observe that:

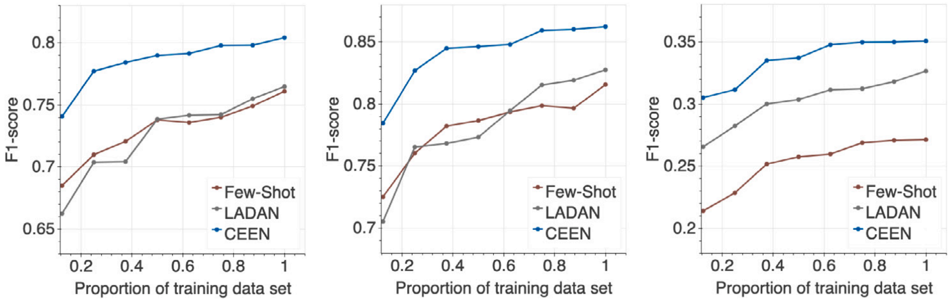
- With the increase of training data, performances of baselines and CEEN steadily improve.
- CEEN surpasses baselines to a great extent with different amounts of training data, which manifests the robustness and effectiveness of our proposed method.

7.3. Influences of pre-trained models

In this section, we compare BERT (Cui et al., 2019) with other pre-trained models, including:

- **RoBERTa-wwm-ext (RoBERTa)** (Cui et al., 2019): it is pre-trained with the whole word masking strategy, in which the tokens that belong to the same word will be masked simultaneously.
- **Lawformer** (Xiao et al., 2021): it is pre-trained on large-scale Chinese legal long case documents.

As shown in Table 5, $\text{CEEN}_{\text{BERT}}$ and $\text{CEEN}_{\text{RoBERTa}}$ outperform the pre-trained baselines in all tasks, indicating that the extraction of the four criminal elements facilitates the judgment prediction.



(a) *Law Articles* (b) *Charges* (c) *Term of Penalty*

Fig. 5. Influences of training data amount.

Table 5
Bert-based model Judgment prediction results on CAIL-small.

	Law articles				Charges				Term of penalty			
Method	Acc.	MP	MR	F1	Acc.	MP	MR	F1	Acc.	MP	MR	F1
BERT	83.85	81.85	82.61	81.13	88.17	88.10	87.76	87.49	43.11	39.64	38.74	38.47
RoBERTa	83.10	81.34	82.89	80.73	88.68	87.27	88.91	87.72	42.28	39.00	40.96	39.63
Lawformer	83.28	83.01	83.08	82.13	88.99	87.90	89.34	88.17	43.35	40.14	41.00	40.08
$\mathrm{CEEN}_{\mathrm{BERT}}$	**84.20**	**83.21**	**83.73**	**82.32**	89.44	88.88	89.01	88.64	**43.80**	**41.15**	39.72	40.03
$\mathrm{CEEN}_{\mathrm{RoBERTa}}$	83.28	82.08	83.41	81.38	**89.58**	**89.21**	**89.83**	**89.21**	43.12	39.80	**41.49**	**40.33**

Table 6
Examples for intuitive comparisons.

Models	Fact descriptions (Predictions of law articles and charges)
LADAN	"From the first half of 2015 to September 2016, the defendant Lianyungang Chemical Company, in order to seek illegal gains, the company's general manager Wan Mou bribes a total of RMB 100,000 to the national staff member Chen Mou in two installments." (Article 389: Offering bribes ×)
Few-Shot	"On January 28, 2014, the defendant *Liao Mou A*, in order to thank *Liao Mou B*, director of Shenhe community neighborhood committee, for his help in contracting the road and drainage project of Shenhe Neighborhood Committee, gave Liao Mou B RMB 70,000." (Article 385: Acceptance of bribe ×)
CEEN	"From the first half of 2015 to September 2016, the defendant Lianyungang Chemical Company (**Criminal: company**), in order to seek illegal gains (**Intentionality: intentional**), the company's general manager Wan Mou bribes a total of RMB 100,000 to the national staff member Chen Mou in two installments (**Target: public servant, Criminal behavior: offering bribes**)." (Article 393: Crime of unit bribery ✓)
	"On January 28, 2014, the defendant Liao Mou A (**Criminal: normal person**), in order to thank Liao Mou B, director of Shenhe community neighborhood committee (**Target: public servant**), for his help in contracting the road and drainage project of Shenhe Neighborhood Committee (**Intentionality: intentional**), gave Liao Mou B RMB 70,000 (**Criminal behavior: offering bribes**)." (Article 389: Offering bribes ✓)

7.4. Case studies

In order to substantiate that CEEN is able to locate and differentiate criminal elements, we collect examples for confusing law articles and fact descriptions. In Table 6, prediction results for two fact descriptions are selected. For LADAN, we found that it does not distinguish well between charges corresponding to the same community of law articles (Article 389: Offering bribes, Article 393: Crime of unit bribery). Similarly, the legal attributes of the Few-shot do not contain any knowledge about criminals and targets,

Table 7
Error analysis for CEEN.

Fact description (Prediction of LJP results)
Between 2003 and 2004, the defendant Yin repeatedly secretly stole others Santana car (**Criminal: normal person, Target: private property, Intentionality: intentional, Criminal behavior: stealing**), the total value of the items involved 603,460 yuan. (Article 264 ✓)(Charge: Larceny ✓)(Term of penalty: 3 to 5 years ×)
The defendant Tu in the Utopia County Road Administration Finance Director (**Criminal: public servant**), in October 2006 to June 2008 to use the convenience of his position (**Intentionality: intentional**), the misappropriation of public funds totaling 395,000 yuan for personal use (**Target: public property, Criminal behavior: Misappropriation**), has not been returned. Article 384 ✓)(Charge: Misappropriation of public funds ✓) (Term of penalty: 3 to 5 years ×)

leading to the misjudgment for the acceptance of bribes and offering bribes. In contrast, by effectively extracting and discriminating criminal elements, CEEN is capable of making precise judgments for ambiguous fact descriptions and law articles.

7.5. Error analysis

To conduct error analysis, we collect bad cases for CEEN. As shown in Table 7, our proposed CEEN can accurately predict the corresponding law articles and charges, while giving erroneous penalty terms. It is because the judgment for terms of penalty should not only accord with law articles and charges, but need to consider the severity of the criminal behavior to society as well. For the two cases in Table 7, despite predicting the correct charges and law articles, CEEN still cannot accurately estimate terms of penalty. In our future work, we tend to take elements related to the impact of the criminal behavior into consideration, so as to promote the term of penalty predictions.

8. Conclusions

In this paper, we have focused on the task of legal judgment prediction. To simultaneously disambiguate similar law articles and fact descriptions, we have proposed a criminal element extraction network (CEEN). To handle confusing fact descriptions with different criminals and targets, we define and identify four categories of criminal elements, including the criminal, target, intentionality, and criminal behavior, from each case. To tackle misleading law articles with highly similar TF–IDF representations, unique feature vectors for criminal elements are obtained and inputted into the multi-task predictor to enhance the judgment process. We have conducted extensive experiments on benchmark datasets. Experimental results have verified the effectiveness of our proposed method. Extracting criminal elements is highly effective for predicting the judgment results. Development experiment results show that our model converges faster than baselines and performs better in the test set.

As to our future work, we will explore evidence information extraction to provide interpretability for judgment results. We also plan to combine the fundamental task of legal text mining, such as coreference resolution, to further enhance the legal judgment prediction by clarifying the information of the criminal in the fact description.

9. Reproducibility

Our datasets and codes include: (1) CAIL-small dataset and CAIL-big dataset annotated with the four criminal elements, (2) code used to label the four criminal elements, (3) code for the CEEN model, (4) code for training CEEN. The above datasets and codes are released at https://github.com/lvyougang/CEEN/.

CRediT authorship contribution statement

Yougang Lyu: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Zihan Wang:** Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Zhaochun Ren:** Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Pengjie Ren:** Data curation. **Zhumin Chen:** Data curation. **Xiaozhong Liu:** Data curation, Methodology, Formal analysis, Writing – review & editing. **Yujun Li:** Data curation. **Hongsong Li:** Data curation. **Hongye Song:** Data curation.

Acknowledgments

We thank our reviewers for valuable feedback. This work was supported by the Natural Science Foundation of China (61902219, 61972234, 61672324, 61672322, 62072279, 62102234), the Key Scientific and Technological Innovation Program of Shandong Province (2019JZZY010129), the Natural Science Foundation of Shandong Province (ZR202102230192), Alibaba Group through Alibaba Research Intern Program, the Fundamental Research Funds of Shandong University, the National Key R&D Program of China with grant No. 2020YFB1406704. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

References

Aletras, N., Tsarapatsanis, D., Preotiuc-Pietro, D., & Lampos, V. (2016). Predicting judicial decisions of the European court of human rights: a natural language processing perspective. *PeerJ Computer Science*, *2*, Article e93.

Boreham, J., & Niblett, B. (1976). Classification of legal texts by computer. *Information Processing & Management*, *12*(2), 125–132.

Chali, Y., Hasan, S. A., & Mojahid, M. (2015). A reinforcement learning formulation to the complex question answering problem. *Information Processing & Management*, *51*(3), 252–272.

Cohen, J. A. (1982). The criminal procedure law of the people's Republic of China. *The Journal of Criminal Law and Criminology (1973-)*, *73*(1), 171–203.

Cui, Y., Che, W., Liu, T., Qin, B., Yang, Z., Wang, S., & Hu, G. (2019). Pre-training with whole word masking for Chinese BERT. CoRR, arXiv:1906.08101.

Feng, J., Huang, M., Zhao, L., Yang, Y., & Zhu, X. (2018). Reinforcement learning for relation classification from noisy data. In *Proceedings of the thirty-second aaai conference on artificial intelligence, (aaai-18), the 30th innovative applications of artificial intelligence (iaai-18), and the 8th aaai symposium on educational advances in artificial intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018* (pp. 5779–5786). AAAI Press.

Feng, J., Huang, M., Zhao, L., Yang, Y., & Zhu, X. (2018). Reinforcement learning for relation classification from noisy data. In *Proceedings of the thirty-second AAAI conference on artificial intelligence, (AAAI-18), the 30th innovative applications of artificial intelligence (AAAI-18), and the 8th AAAI symposium on educational advances in artificial intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018* (pp. 5779–5786). AAAI Press.

Giacalone, M., Cusatelli, C., Romano, A., Buondonno, A., & Santarcangelo, V. (2018). Big data and forensics: An innovative approach for a predictable jurisprudence. *Inform Sci*, *426*, 160–170.

Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of machine learning research*: *80, Proceedings of the 35th international conference on machine learning, ICML 2018, StockholmsmäSsan, Stockholm, Sweden, July 10-15, 2018* (pp. 1856–1865). PMLR.

Hu, Z., Li, X., Tu, C., Liu, Z., & Sun, M. (2018). Few-shot charge prediction with discriminative legal attributes. In *Proceedings of the 27th international conference on computational linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018* (pp. 487–498). Association for Computational Linguistics.

Hu, H., Wu, X., Luo, B., Tao, C., Xu, C., Wu, W., & Chen, Z. (2018). Playing 20 question game with policy-based reinforcement learning. In *Proceedings of the 2018 conference on empirical methods in natural language processing, Brussels, Belgium, October 31 - November 4, 2018* (pp. 3233–3242). Association for Computational Linguistics.

Ji, D., Tao, P., Fei, H., & Ren, Y. (2020). An end-to-end joint model for evidence information extraction from court record document. *Information Processing & Management*, *57*(6), Article 102305.

Jiang, X., Ye, H., Luo, Z., Chao, W., & Ma, W. (2018). Interpretable rationale augmented charge prediction system. In D. Zhao (Ed.), *COLING 2018, the 27th international conference on computational linguistics: system demonstrations, Santa Fe, New Mexico, August 20-26, 2018* (pp. 146–151). Association for Computational Linguistics.

K., S., & Thilagam, P. S. (2019). Crime base: Towards building a knowledge base for crime entities and their relationships from online news papers. *Information Processing & Management*, *56*(6).

Keown, R. (1980). Mathematical models for legal prediction. *Computer/Lj*, *2*, 829.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 conference on empirical methods in natural language processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, a meeting of SIGDAT, a special interest group of the ACL* (pp. 1746–1751). ACL.

Kingma, D. P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kort, F. (1957). Predicting supreme court decisions mathematically: A quantitative analysis of the "right to counsel" cases. *The American Political Science Review*, *51*(1), 1–12.

Lauderdale, B. E., & Clark, T. S. (2012). The supreme court's many median justices. *American Political Science Review*, 847–866.

Li, X., Yin, F., Sun, Z., Li, X., Yuan, A., Chai, D., Zhou, M., & Li, J. (2019). Entity-relation extraction as multi-turn question answering. In *Proceedings of the 57th conference of the association for computational linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers* (pp. 1340–1350). Association for Computational Linguistics.

Lin, W., Kuo, T., Chang, T., Yen, C., Chen, C., & Lin, S. (2012). Exploiting machine learning models for Chinese legal documents labeling, case classification, and sentencing prediction. *International Journal of Computational Linguistics & Chinese Language Process*, *17*(4).

Liu, C., Chang, C., & Ho, J. (2004). Case instance generation and refinement for case-based criminal summary judgments in Chinese. *Journal of Information Science and Engineering*, *20*(4), 783–800.

Liu, Y., Chen, Y., & Ho, W. (2015). Predicting associated statutes for legal problems. *Information Processing & Management*, *51*(1), 194–211.

Liu, Y., & Lapata, M. (2019). Text summarization with pretrained encoders. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019* (pp. 3728–3738). Association for Computational Linguistics.

Luo, B., Feng, Y., Xu, J., Zhang, X., & Zhao, D. (2017). Learning to predict charges for criminal cases with legal basis. In *Proceedings of the 2017 conference on empirical methods in natural language processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017* (pp. 2727–2736). Association for Computational Linguistics.

Miao, L., Cao, D., Li, J., & Guan, W. (2020). Multi-modal product title compression. *Information Processing & Management*, *57*(1).

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. CoRR, arXiv:1310.4546.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M. A., Fidjeland, A., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529–533.

Moens, M., & Uyttendaele, C. (1997). Automatic text structuring and categorization as a first step in summarizing legal cases. *Information Processing & Management*, *33*(6), 727–737.

Nagel, S. S. (1963). Applying correlation analysis to case prediction. *Texas Law Review*, *42*, 1006.

Narasimhan, K., Kulkarni, T. D., & Barzilay, R. (2015). Language understanding for text-based games using deep reinforcement learning. In *Proceedings of the 2015 conference on empirical methods in natural language processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015* (pp. 1–11). The Association for Computational Linguistics.

Qazi, N., & Wong, B. L. W. (2019). An interactive human centered data science approach towards crime pattern analysis. *Information Processing & Management*, *56*(6).

Quintard-Morénas, F. (2010). The presumption of innocence in the French and Anglo-American legal traditions. *The American Journal of Comparative Law*, *58*(1), 107–149.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T. P., Leach, M., Kavukcuoglu, K., Graepel, T., & Hassabis, D. (2016). Mastering the game of go with deep neural networks and tree search. *Nature*, *529*(7587), 484–489.

Sulea, O., Zampieri, M., Malmasi, S., Vela, M., Dinu, L. P., & van Genabith, J. (2017). Exploring the use of text classification in the legal domain. In *CEUR workshop proceedings*: *2143*, *Proceedings of the second workshop on automated semantic analysis of information in legal texts co-located with the 16th international conference on artificial intelligence and law (ICAIL 2017), London, UK, June 16, 2017*. CEUR-WS.org.

Sun, M., Chen, X., Zhang, K., Guo, Z., & Liu, Z. (2016). Thulac: An efficient lexical analyzer for chinese. 2017-01-17)[2019-04-02], https://GithubCom/Thunlp/THULACPython.

Sutton, R. S., McAllester, D. A., Singh, S. P., & Mansour, Y. (1999). Policy gradient methods for reinforcement learning with function approximation. In S. A. Solla, T. K. Leen, & K. Müller (Eds.), *Advances in neural information processing systems 12, [NIPS Conference, Denver, Colorado, USA, November 29 - December 4, 1999]* (pp. 1057–1063). The MIT Press.

Tadros, V., & Tierney, S. (2004). The presumption of innocence and the human rights act. *The Modern Law Review*, *67*(3), 402–434.

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, *8*, 229–256.

Wu, J., Li, L., & Wang, W. Y. (2018). Reinforced co-training. In *Proceedings of the 2018 conference of the north american chapter of the association for computational linguistics: Human language technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)* (pp. 1252–1262). Association for Computational Linguistics.

Xiao, C., Hu, X., Liu, Z., Tu, C., & Sun, M. (2021). Lawformer: A pre-trained language model for Chinese legal long documents. CoRR, arXiv:2105.03887.

Xiao, C., Zhong, H., Guo, Z., Tu, C., Liu, Z., Sun, M., Feng, Y., Han, X., Hu, Z., Wang, H., & Xu, J. (2018). CAIL2018: a large-scale legal dataset for judgment prediction. CoRR, arXiv:1807.02478.

Xiong, S., & Ji, D. (2016). Query-focused multi-document summarization using hypergraph-based ranking. *Information Processing & Management*, *52*(4), 670–681.

Xu, N., Wang, P., Chen, L., Pan, L., Wang, X., & Zhao, J. (2020). Distinguish confusing law articles for legal judgment prediction. In *Proceedings of the 58th annual meeting of the association for computational linguistics, ACL 2020, Online, July 5-10, 2020* (pp. 3086–3095). Association for Computational Linguistics.

Yang, W., Jia, W., Zhou, X., & Luo, Y. (2019). Legal judgment prediction via multi-perspective bi-feedback network. In *Proceedings of the twenty-eighth international joint conference on artificial intelligence, IJCAI 2019, Macao, China, August 10-16, 2019* (pp. 4085–4091). ijcai.org.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A. J., & Hovy, E. H. (2016). Hierarchical attention networks for document classification. In *NAACL HLT 2016, the 2016 conference of the north american chapter of the association for computational linguistics: Human language technologies, San Diego California, USA, June 12-17, 2016* (pp. 1480–1489). The Association for Computational Linguistics.

Zhong, H., Guo, Z., Tu, C., Xiao, C., Liu, Z., & Sun, M. (2018). Legal judgment prediction via topological learning. In *Proceedings of the 2018 conference on empirical methods in natural language processing, Brussels, Belgium, October 31 - November 4, 2018* (pp. 3540–3549). Association for Computational Linguistics.

Zhong, H., Wang, Y., Tu, C., Zhang, T., Liu, Z., & Sun, M. (2020). Iteratively questioning and answering for interpretable legal judgment prediction. In *The thirty-fourth AAAI conference on artificial intelligence, AAAI 2020, the thirty-second innovative applications of artificial intelligence conference, IAAI 2020, the Tenth AAAI Symposium on educational advances in artificial intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020* (pp. 1250–1257). AAAI Press.