

Charge Prediction by Constitutive Elements Matching of Crimes

Jie Zhao, Ziyu Guan*, Cai Xu, Wei Zhao and Enze Chen

School of Computer Science and Technology, Xidian University

{jzhao1992@stu., zyguan@, cxu@, ywzhao@mail., 21031211569@stu.}@xidian.edu.cn

Abstract

Charge prediction is to automatically predict the judgemental charges for legal cases. To convict a person/unit of a charge, the case description must contain matching instances of the constitutive elements (CEs) of that charge. This knowledge of CEs is a valuable guide for the judge in making final decisions. However, it is far from fully exploited for charge prediction in the literature. In this paper we propose a novel method named Constitutive Elements-guided Charge Prediction (CECP). CECP mimics human’s charge identification process to extract potential instances of CEs and generate predictions accordingly. It avoids laborious labeling of matching instances of CEs by a novel reinforcement learning module which progressively selects potentially matching sentences for CEs and evaluates their relevance. The final prediction is generated based on the selected sentences and their relevant CEs. Experiments on two real-world datasets show the superiority of CECP over competitive baselines.

1 Introduction

Charge prediction is to identify the final charges/crimes (e.g., fraud and bribery) on the basis of fact descriptions of legal cases. As a considerable application of legal artificial intelligence, charge prediction is pivotal for legal assistant systems. For legal professionals, it provides a handy reference to relieve the burdensome work and increase working efficiency. For ordinary people, a qualified charge prediction system can provide useful and inexpensive legal services.

In the past decades, charge prediction has been studied as a text classification problem. Different from the ordinary text classification, a large amount of legal knowledge (e.g., law articles in the Criminal Law and related judicial interpretations) could be exploited for predicting charges precisely. There are mainly two general ideas for exploiting the legal knowledge. The first one is to assess the correlations between fact descriptions and law articles and incorporate the results for charge prediction [Luo *et al.*, 2017;

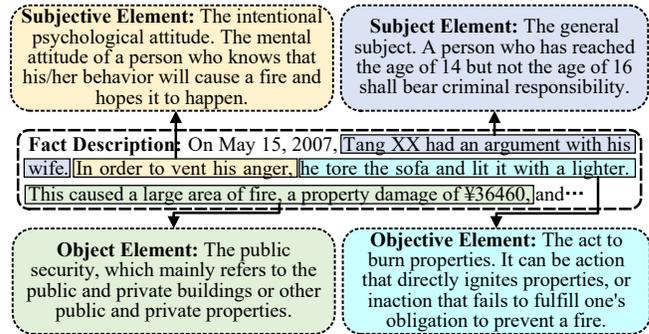


Figure 1: Four CEs of the charge “arson”. The instances are marked with the same colors as the corresponding CEs, respectively.

Shen *et al.*, 2018; Xu *et al.*, 2020]. However, leveraging legal knowledge at the article-level would be too coarse-grained since one article can involve multiple charges. Such articles fail to provide distinct support for distinguishing confusing charges. The second idea is to manually derive several legal attributes, such as whether the criminal has the act of violence, from law articles or judicial interpretations in advance [Ashley and Brüninghaus, 2009; Hu *et al.*, 2018; Zhong *et al.*, 2020a]. These legal attributes are then extracted from fact descriptions and used to predict charges. However, manually defining attributes requires significant expertise and effort, and is hard to be comprehensive.

For countries with a civil law system, constitutive elements (CEs) of a charge are important judicial interpretations (not included in law articles) and guides for convicting a person/unit of the charge in practice. That is, the fact of the case must match the CEs of the corresponding charge [Gao, 2009]. Consider the four-CEs system of China. Each charge is described by subject element, subjective element, object element and objective element, all of which are crucial for distinguishing different charges. Figure 1 shows an example where the matching instances and the corresponding CEs are shown in the same colors, respectively. Tang is convicted of *arson* because the fact matches the four CEs of arson. However, the knowledge of CEs has not been adequately exploited in charge prediction. Recently, Li *et al.* [2021] designed a double-layer criminal system for interpretable charge prediction, where objective and subjective information was ex-

*Corresponding Author

tracted. However, this work ignores the object and subject elements which are also important. For instance, when someone steals cables, he/she will be convicted of *theft* if the object is cables in a warehouse, and of *damage of communication equipment* if the object is cables used for communication.

Furthermore, different CEs follow a logical order. A typical charge identification process is: the judicial officials first discover a certain detriment fact (**objective**), then find out its cause (**subject**) and whether/how the perpetrator should be responsible for it (**subjective**), and finally determine which kind of social relationship (**object**) is violated and whether the crime is established [Zhao, 2003]. Take “arson” as an example. First, the occurrence of a fire (objective) is discovered. Then it is investigated whether the fire is caused by human behavior (subject), because no one will be convicted if it is caused by non-human behavior (e.g., a naturally occurring forest fire). If so, the next focus is the psychological attitude (subjective) of the involved person. This is the key to distinguishing *arson* from *fire by imprudence*. Last, based on the above identified substance, whether the public safety (object) has been violated and whether anyone should be convicted are determined. We can see in this process the subsequent tasks are based on the previously identified substance. Therefore, the logical order, *objective element* \rightarrow *subject element* \rightarrow *subjective element* \rightarrow *object element*, could be useful and important knowledge for charge prediction.

In order to better exploit the knowledge of CEs, we propose a novel method for charge prediction named Constitutive Elements-guided Charge Prediction (CECP). CECP mimics human’s charge identification process to extract potential instances of CEs and generate predictions accordingly. To avoid laborious labeling of matching instances of CEs, we design a reinforcement learning module which defines a *legal agent* to mine instances of CEs automatically. The agent is formulated in the actor-critic framework [Mnih *et al.*, 2016] and takes the embeddings of fact descriptions and CEs from an *encoder network* as observations. It groups CEs of different charges by types (i.e., object, objective, subject, subjective) and follows the logical order of CEs circularly to iteratively select the most crucial sentences (i.e., instances) for each type. In a step concerning CE type p , it first performs weighted aggregations of the previously selected sentences for type p and the type p CEs respectively, where the weights are calculated by relevance estimation between them. This provides summarized representations of selected sentences and CEs focusing on the most relevant instances and charges respectively. Then the agent selects an un-selected sentence by considering both the summarized representation of CEs and a history embedding which encodes the previously selected sentences for all CE types (emphasizing sentences with high relevance weights). This history embedding represents the “previously identified substance” and is updated by the summarized representation of selected sentences for type p . Finally, we feed the summarized representations of selected sentences and CEs for all CE types into a prediction network to generate charge prediction. We design a reward function for the agent by the prediction result and the duplication degree of selected sentences. In summary, the contributions of this work are: (1) we provide the first solution for charge

prediction which comprehensively exploits the knowledge of constitutive elements of charges; (2) a novel reinforcement learning model is proposed to avoid laborious labeling of matching instances of CEs; (3) we conduct plentiful experiments on two real-world datasets to verify the effectiveness of our model and show its significant improvements over competitive baselines.

2 Related Work

Charge Prediction. Convicting someone or a unit of a certain charge is a key step of legal judgment. Due to the relatively small scale of data and limited computing power, inchoate works for charge prediction mainly relied on mathematical methods [Kort, 1957] or shallow textual features [Lin *et al.*, 2012]. Since these works are severely limited by manual efforts, they are difficult to be applied to diverse scenarios. Recently, researchers began to take advantage of deep learning methods and proposed a series of neural-based models which utilized the legal knowledge to enhance prediction. Luo *et al.* [2017] proposed an attention-based neural network, where law articles were incorporated and a two-stack attention mechanism was adopted to jointly model charge prediction and relevant article extraction. Xu *et al.* [2020] proposed a novel graph neural network to automatically learn subtle differences among confusing law articles, which were taken as the legal basis to attentively extract discriminative features from fact descriptions. Hu *et al.* [2018] introduced several discriminative attributes of charges as the internal mapping between fact descriptions and charges, and proposed an attribute-attentive charge prediction model. Nevertheless, most existing works either leverage the correlations to law articles or manually define attributes for incorporating legal knowledge. As aforementioned, (1) correlations at the article-level cannot well deal with multiple charges in the same article; (2) manually defining attributes is hard to be comprehensive. Although recent work [Li *et al.*, 2021] exploited objective and subjective elements of charges, the knowledge of CEs was far from being fully exploited.

Extractive Summarization. The state-of-art neural extractive summarization (NES) models are mainly based on sentence (or smaller linguistic unit) scoring and selecting [Zhong *et al.*, 2020b]. These models aim to assess the representativeness scores of a sentence w.r.t contextual sentences, title sentences, or query sentences [Ren *et al.*, 2018]. In CECP, the legal agent also scores and selects sentences from fact descriptions. However, the existing NES techniques cannot well solve our problem. This is because in our case we need to assess multiple scores of a sentence w.r.t the four types of CEs of hundreds of charges, while NES only considers the representativeness of a sentence w.r.t some target sentences. Moreover, the logical order of the four CE types cannot be captured by NES.

Reinforcement Learning. Reinforcement learning is designed for sequential decision-making problems. Recently, deep reinforcement learning (DRL) which integrates the perception ability of deep learning with the decision-making ability of reinforcement learning has been proposed [Lange

et al., 2012]. There are two mainstreams of DRL: value-based models and policy-based models. Thanks to the great progress in this domain, many DRL based methods were proposed to solve various computer vision and natural language processing tasks and obtained desirable results [Rennie *et al.*, 2017]. In this work, we design a new DRL agent for mining matching instances of CEs and employ the A3C framework [Mnih *et al.*, 2016] for model training.

3 Method

3.1 Problem Formulation and Model Overview

A legal case contains a fact description F and the corresponding charge y . The fact description is comprised of n sentences ($F = \{s_i\}_{i=1}^n$) and the i -th sentence, s_i , contains m_i words, i.e., $s_i = \{w_{i,j}\}_{j=1}^{m_i}$. The corresponding charge y belongs to the charge set \mathcal{Y} with C charges in total. We introduce CEs of charges as legal knowledge. Assume that there are P CE types. We regard each CE as a word sequence and denote the p -th CE ($p \in \{1, \dots, P\}$) of the c -th charge ($c \in \{1, \dots, C\}$) by $e_{p,c} = \{w_{p,c,j}\}_{j=1}^{l_p}$, where l_p is the sequence length. In this paper, we use bold face lower/upper case letters to denote vectors/matrices respectively. Our goal is to develop a model that can accurately predict y for a given F and the auxiliary knowledge of CEs, $\{e_{p,c}\}$.

In CECP, we utilize the knowledge of CEs to extract potential instances of CEs from the fact description of a legal case, and predict the corresponding charge accordingly. As shown in Figure 2(a), the proposed CECP consists of encoder, reinforcement learning (RL) module and predictor. The encoder is responsible for learning sentence-level embeddings of the fact description and CEs. In the RL module, we develop a legal agent which observes these embeddings and mines instances of CEs from the fact description automatically. Specifically, it follows the logical order of CEs circularly to iteratively select the most crucial sentences for each CE type. After enough sentences are selected, the fact description can be represented by P parts, where the p -th part contains the instances (i.e., sentences) of type p CEs. We feed these P parts and the corresponding P groups of CEs (both after aggregation for each type according to the relevance estimation between selected sentences and CEs) to a predictor to get the prediction. In the following, we will elaborate on the three components of CECP.

3.2 Encoder Network

The encoder network is dedicated to learning embeddings of the fact description F and CEs $\{e_{p,c}\}$.

Fact Encoder. In CECP, the fact description is represented as sentence-level embeddings. We use the GRU [Cho *et al.*, 2014] to capture the dependencies among words of sentences and utilize the scaled dot-product attention [Vaswani *et al.*, 2017] to capture informative words and aggregate hidden states of GRU into a single embedding accordingly. A slight difference with the standard scaled dot-product attention is that we compute the query vector by a max-pooling operation over hidden states of GRU to better capture important sentence-level features. Full details can be found in Appendix

A.¹. Since different types of CEs often show language discrepancies, we use one GRU for detecting the instances for one CE type (with identical network architecture and separate parameters). Finally, for each sentence $s_i \in F$, we obtain P d -dimensional embeddings $\{\mathbf{f}_{p,i}\}$ ($p \in \{1, \dots, P\}$).

CE Encoder. We treat each CE ($e_{p,c}$) as a long sentence, encode it in the same way as encoding the sentences in the fact description, and obtain a basic feature vector $\mathbf{e}_{\text{base}_{p,c}}$ for it. Motivated by the fact that different types of CEs could show different amounts of information, we design a Pivotal Feature Identification (PFI) layer to extract key features and ignore features showing low variances among different charges. Let $\text{var}_p[a]$ represent the variance of $[\mathbf{e}_{\text{base}_{p,1}}[a], \dots, \mathbf{e}_{\text{base}_{p,C}}[a]]$, where a in the square brackets represents the a -th dimension of a vector. We use \mathbf{u}_p which is obtained according to var_p to identify whether the a -th dimension of $\mathbf{e}_{\text{base}_{p,c}}$ should be ignored:

$$\mathbf{u}_p = 2 \times \text{sigmoid}(-\text{var}_p),$$

$$\mathbf{e}_{p,c} = \mathbf{e}_{\text{base}_{p,c}} - \frac{\sum_{c=1}^C \mathbf{e}_{\text{base}_{p,c}} \circ \mathbf{u}_p}{C}, \quad (1)$$

where \circ is the pointwise product. Due to the properties of sigmoid, when $\text{var}_p[a]$ is not close to 0, $\mathbf{u}_p[a]$ will be close to 0 and the impact of PFI on $\mathbf{e}_{p,c}[a]$ is very small (pivotal); when $\text{var}_p[a]$ is close to 0, $\mathbf{u}_p[a]$ will be close to 1. Then $\mathbf{e}_{p,c}[a]$ will approach 0 (eliminated) since the subtracted average-term is close to each $\mathbf{e}_{\text{base}_{p,c}}$ due to very small variance.

3.3 Reinforcement Learning Module

We model the selection of crucial sentences for each CE type as an RL problem. We formulate our task in the actor-critic framework and define a legal agent which takes the embeddings of the fact description and CEs, $\{\mathbf{f}_{p,i}\}$, $\{e_{p,c}\}$, $\forall p, i, c$, as its observations. The agent groups CEs of different charges by types (i.e., $\{e_{1,c}\}_{\forall c}, \dots, \{e_{P,c}\}_{\forall c}$) and iteratively selects the most crucial sentences for each CE type.

Figure 2(b) shows how the agent selects a sentence in step t concerning CE type $\tilde{p} = 1 + [(t-1) \bmod P]$. Here, we denote the set of embeddings of the selected sentences for type \tilde{p} as $\{\mathbf{f}_{\tilde{p},i_s}\}^t$ and that of the remaining un-selected sentences² for type \tilde{p} as $\{\mathbf{f}_{\tilde{p},i_u}\}^t$. The agent first performs weighted aggregations of the selected sentences $\{\mathbf{f}_{\tilde{p},i_s}\}^t$ and the set of type \tilde{p} CEs $\{e_{\tilde{p},c}\}$ respectively, where the weights are calculated by relevance estimation between them. This provides summarized representations, $\hat{\mathbf{f}}_{\tilde{p}}^t$ and $\hat{\mathbf{e}}_{\tilde{p}}^t$, of the selected sentences and CEs, focusing on the most relevant instances and charges respectively. The agent then updates a history embedding \mathbf{h}^t with $\hat{\mathbf{f}}_{\tilde{p}}^t$, i.e., recording the most relevant selected sentences for type \tilde{p} . Hence, \mathbf{h}^t memorizes salient sentences selected in all previous steps. Next, the agent computes a probability distribution on $\{\mathbf{f}_{\tilde{p},i_u}\}^t$ based on each $\mathbf{f}_{\tilde{p},i_u}$'s interactions with $\hat{\mathbf{e}}_{\tilde{p}}^t$ and \mathbf{h}^t , and selects one accordingly. It means we consider both the current type \tilde{p} CEs (emphasizing the most promising ones) and previously selected sentences

¹ Available at: <https://github.com/jiezhao6/CECP>

² Note that sometimes one sentence connects to multiple CE types. Hence, we need to maintain one un-selected set for each type.

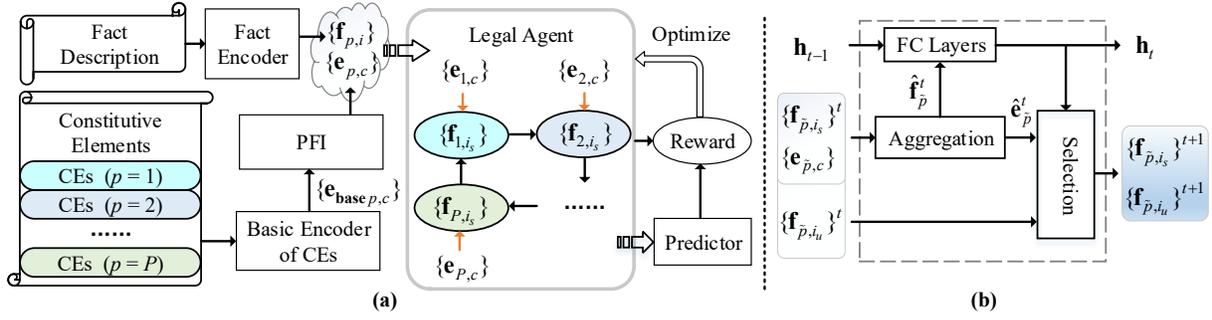


Figure 2: (a) The framework of CECP; (b) The workflow of the legal agent in step t : for CE type $\tilde{p} = 1 + [(t-1) \bmod P]$, it observes three sets, $\{\mathbf{f}_{\tilde{p},i_s}\}^t$, $\{\mathbf{f}_{\tilde{p},i_u}\}^t$ and $\{\mathbf{e}_{\tilde{p},c}\}$, representing selected/un-selected sentences and CE types for type \tilde{p} respectively. It takes an action to select a sentence according to the policy π defined in Eqs. (2) ~ (5).

(previously identified substance) for sentence selection. The agent continuously selects sentences cyclically for the P CE types until the total number of selected sentences meets a pre-set value. Finally, the selected sentences for all CE types are delivered to the predictor to generate a prediction. Formally, $\{\mathbf{f}_{p,i}\}$ and $\{\mathbf{e}_{p,c}\}$ can be considered as the **environment** in an ordinary RL problem. The parameters θ_a , which contain the parameters in Eqs. (3) ~ (5) that we describe later, define a **policy** π which results in an **action** of selecting a sentence. In the following, we present the details.

Aggregation. Let $n^t = |\{\mathbf{f}_{\tilde{p},i_s}\}^t|$. First, we take a max-pooling operation over these selected sentences:

$$\tilde{\mathbf{f}}_{\tilde{p}}^t[a] = \max_{1 \leq i_s \leq n^t} (\mathbf{f}_{\tilde{p},i_s}[a]), \quad a \in \{1, \dots, d\}. \quad (2)$$

Here $\tilde{\mathbf{f}}_{\tilde{p}}^t$ captures the most informative features of selected sentences in a type-view and could be regarded as a matching “instance” of type \tilde{p} . For the aggregation of CEs, we can just take the CE of a certain charge that is most relevant to $\tilde{\mathbf{f}}_{\tilde{p}}^t$. However, different charges may have very similar CE descriptions for type \tilde{p} , so we summarize the type \tilde{p} CEs of all charges in a weighted sum manner instead of risking missing important information by just taking the most relevant CE:

$$\alpha_{\tilde{p},c}^t = \frac{\exp((\tilde{\mathbf{f}}_{\tilde{p}}^t)^\top \mathbf{W}_{\tilde{p}} \mathbf{e}_{\tilde{p},c})}{\sum_{c=1}^C \exp((\tilde{\mathbf{f}}_{\tilde{p}}^t)^\top \mathbf{W}_{\tilde{p}} \mathbf{e}_{\tilde{p},c})}, \quad (3)$$

$$\hat{\mathbf{e}}_{\tilde{p}}^t = \sum_{c=1}^C \alpha_{\tilde{p},c}^t \mathbf{e}_{\tilde{p},c},$$

where $\alpha_{\tilde{p},c}^t$ is computed as the relevance degree between $\tilde{\mathbf{f}}_{\tilde{p}}^t$ and $\mathbf{e}_{\tilde{p},c}$, and $\mathbf{W}_{\tilde{p}} \in \mathbb{R}^{d \times d}$ is a bilinear term [Chen *et al.*, 2016] which allows estimating the relevance between $\tilde{\mathbf{f}}_{\tilde{p}}^t$ and $\mathbf{e}_{\tilde{p},c}$ more flexibly.

Note that $\tilde{\mathbf{f}}_{\tilde{p}}^t$ is CE-independent, i.e., the max-pooling aggregation fails to explicitly utilize the knowledge of CEs. Therefore, we also use a weighted sum to summarize the selected sentences, where the weights are the relevance scores between $\mathbf{f}_{\tilde{p},i_s}$ and $\hat{\mathbf{e}}_{\tilde{p}}^t$:

$$\beta_{\tilde{p},i_s}^t = \frac{\exp((\mathbf{f}_{\tilde{p},i_s})^\top \mathbf{W}_{\tilde{p}} \hat{\mathbf{e}}_{\tilde{p}}^t)}{\sum_{i_s=1}^{n^t} \exp((\mathbf{f}_{\tilde{p},i_s})^\top \mathbf{W}_{\tilde{p}} \hat{\mathbf{e}}_{\tilde{p}}^t)}, \quad (4)$$

$$\hat{\mathbf{f}}_{\tilde{p}}^t = \sum_{i_s=1}^{n^t} \beta_{\tilde{p},i_s}^t \mathbf{f}_{\tilde{p},i_s}.$$

To a certain extent, the above computing process is similar to the bi-directional attention [Seo *et al.*, 2016]. However, the bi-directional attention is not suitable for our problem. This is because bi-directional attention computes pairwise relevance between two sets of items (in our case, sentences and CEs). To obtain aggregations of sentences and CEs, we need to first aggregate these pairwise relevance scores, either by max-pooling or average-pooling. However, (1) max-pooling is vulnerable to noises; (2) average-pooling would promote sentences with moderate scores to many CEs. Since the CEs of different charges general form clusters due to similar charges, a matching instance (sentence) of a CE in the fact description would show high relevance scores to a few CEs. Hence, our method, which first identifies a few relevant CEs according to the salient features in the selected sentences and then extracts relevant sentences to these CEs, is more suitable for mining matching instances.

Action. The summarized representations $\hat{\mathbf{f}}_{\tilde{p}}^t$ and $\hat{\mathbf{e}}_{\tilde{p}}^t$ focus on the most relevant sentences (instances) and charges respectively. $\hat{\mathbf{e}}_{\tilde{p}}^t$ can provide an external legal knowledge guide for selecting an un-selected sentence for type \tilde{p} . In step t , we first incorporate $\hat{\mathbf{f}}_{\tilde{p}}^t$ into \mathbf{h}^{t-1} , the previous history embedding, to add the latest most salient sentences for type \tilde{p} and get \mathbf{h}^t . Then, both $\hat{\mathbf{e}}_{\tilde{p}}^t$ and \mathbf{h}^t are considered for selecting an un-selected sentence. We design a learnable interpolation weight $g_{i_u}^t$ to tune the relative importance between them, since they may have different contributions in different stages (e.g., in the initial steps, the history embedding is weak, so the agent should pay more attention to the knowledge of CEs; as t increases, the history embedding is enhanced, and could be more important in evaluating sentences). Formally, the selecting probabilities of un-selected sentences $\{\mathbf{f}_{\tilde{p},i_u}\}^t$ for type \tilde{p} are computed as follows:

$$\mathbf{h}^t = \tanh(\mathbf{W}_{his}[\mathbf{h}^{t-1}; \hat{\mathbf{f}}_{\tilde{p}}^t] + \mathbf{b}_{his}),$$

$$g_{i_u}^t = \text{sigmoid}(\mathbf{W}_g[\mathbf{f}_{\tilde{p},i_u}; \mathbf{h}^t; \hat{\mathbf{e}}_{\tilde{p}}^t] + \mathbf{b}_g),$$

$$\tilde{\pi}_{i_u}^t = g_{i_u}^t \times (\mathbf{f}_{\tilde{p},i_u})^\top \mathbf{W}_e \hat{\mathbf{e}}_{\tilde{p}}^t + (1 - g_{i_u}^t) \times (\mathbf{f}_{\tilde{p},i_u})^\top \mathbf{W}_h \mathbf{h}^t, \quad (5)$$

$$\pi_{i_u}^t = \frac{\exp(\tilde{\pi}_{i_u}^t)}{\sum_{i_u=1}^{n-n^t} \exp(\tilde{\pi}_{i_u}^t)},$$

where \mathbf{W}_{his} , \mathbf{b}_{his} , \mathbf{W}_g , \mathbf{b}_g , \mathbf{W}_e and \mathbf{W}_h are trainable parameters. Based on the probability of each un-selected sen-

tence, the agent will take an action a^t that selects the sentence with max probability (when testing) or samples a sentence by probability (when training).

The critic in our model, i.e., the value function, learns to evaluate how “good” the selected sentences are under the current policy. Hence, we feed the summarized representations of selected sentences to the critic. It is defined as a network containing two fully connected (FC) layers (parameters: θ_v) with a tanh nonlinearity following the first FC layer:

$$v^t = \text{FC}(\tanh(\text{FC}([\hat{\mathbf{f}}_1^t; \hat{\mathbf{f}}_2^t; \dots; \hat{\mathbf{f}}_p^t]]))). \quad (6)$$

Reward Function. Naturally, the prediction results shall be a part of the reward signal. Besides, to encourage the diversity of selected sentences and enhance the exploration ability of the agent when training, we design a penalty term to penalize actions which select previously selected sentences. The reward function is defined as:

$$r^t = \text{pred}^t - \lambda \times \text{dup}^t. \quad (7)$$

The penalty term $\text{dup}^t = n_{\text{dup}}^t / P$ measures the duplication degree of the currently selected sentence, where n_{dup}^t is the number of times the sentence selected in the current step has been selected in previous steps. This term will make the agent pay attention to the selection of diverse information. What’s more, it is also a form of exploration to discourage premature convergence, which is different from the entropy regularization [Mnih *et al.*, 2016] in the actor-critic framework. The entropy regularization term encourages exploration in one action, whereas this penalty term encourages exploration in multiple actions since it concerns previous steps. λ is a weight hyperparameter. pred^t is defined as:

$$\text{pred}^t = \begin{cases} \hat{\mathbf{y}}_{c^*}^t & \text{if } \underset{c'}{\text{argmax}} \hat{\mathbf{y}}_{c'}^t = c^* \\ \hat{\mathbf{y}}_{c^*}^t - 1 & \text{if } \underset{c'}{\text{argmax}} \hat{\mathbf{y}}_{c'}^t \neq c^*, \end{cases}$$

where $\hat{\mathbf{y}}_{c^*}^t$ is the prediction softmax output for c^* , the class index of the ground truth. When the predictor returns a correct prediction, the agent will be rewarded the corresponding probability score, to encourage it to generate a high score for the correct class. On the contrary, it will be punished with a negative reward if the prediction is wrong (still encourage it to generate high scores for ground truth charges).

3.4 Prediction Network

The prediction network outputs the final prediction according to the sentences selected by the legal agent. It in turn provides feedback that partly reflects the quality of the agent’s action. Supposing that n_p sentences have been selected for type p , we apply Eqs. (3) ~ (4) on the final set of selected sentences for type p to obtain the summarized representations of the selected sentences and type p CEs, respectively. Then we use a concatenation operation to represent them as $\mathbf{fe}_p = [\hat{\mathbf{f}}_p; \hat{\mathbf{e}}_p; \hat{\mathbf{f}}_p \circ \hat{\mathbf{e}}_p]$, where $\hat{\mathbf{f}}_p \circ \hat{\mathbf{e}}_p$ further enhances the interaction between the aggregated sentences and CEs. Finally, we calculate the probability distribution of charges by a linear classifier with parameters \mathbf{W} and \mathbf{b} :

$$\hat{\mathbf{y}} = \text{softmax}(\mathbf{W}[\mathbf{fe}_1; \mathbf{fe}_2; \dots; \mathbf{fe}_P] + \mathbf{b}). \quad (8)$$

Statistics	Training Cases	Test Cases	Charges
Criminal-S	61,589 (24.4)	7,702 (24.1)	149
Criminal-M	153,521 (24.4)	19,189 (24.4)	149
Criminal-L	306,900 (24.4)	38,368 (24.5)	149
CAIL	101,275 (22.3)	26,661 (22.2)	119

Table 1: Statistics of the Criminal and CAIL datasets. The real numbers represent the average number of sentences in a sample.

To train the encoder and predictor, the standard cross-entropy loss function is used. For training the RL module, we employ the A3C [Mnih *et al.*, 2016] framework. We alternately optimize them, and the details and the pseudocode of the training process are described in Appendix A.2.

4 Experiments

We evaluate the performance of CECP on **Criminal** [Hu *et al.*, 2018] and **CAIL** [Xiao *et al.*, 2018], which are both collected from the China Judgments Online³. Important statistics of them are summarized in Table 1. Criminal consists of roughly 500,000 legal cases of 149 charges. Each case contains a fact description and the corresponding charge. Three sub-datasets with different sizes are constructed: **Criminal-S**, **Criminal-M** and **Criminal-L**. **CAIL** is a legal dataset for competition. Since it is a multi-label dataset, we perform pre-processing on it (details in Appendix A.3).

We compare CECP with the following ordinary text classification (OTC) baselines and legal knowledge based (LKB) baselines. **The OTC baselines** just take the fact description as input and totally neglect legal knowledge. **TextCNN** [Kim, 2014] and **DPCNN** [Johnson and Zhang, 2017] are two convolutional neural network based models and **HARNN** [Yang *et al.*, 2016] is a recurrent neural network based model. Based on the transformer, **BERT** [Devlin *et al.*, 2019] can benefit from pre-trained models. **SAttCaps** [Le *et al.*, 2020] is the state-of-the-art OTC baseline, which captures long-range dependencies of lengthy fact descriptions. **The LKB baselines** exploit law articles or legal attributes as auxiliary information to predict charges. **FewShot** [Hu *et al.*, 2018] is an attribute-attentive charge prediction model. **FLA** [Luo *et al.*, 2017] and the state-of-the-art LKB baseline, **LADAN** [Xu *et al.*, 2020], are two models which exploit law articles. LADAN cannot be applied to the Criminal dataset due to lack of required information (details in Appendix A.3).

Following the work of [Le *et al.*, 2020; Xu *et al.*, 2020], we employ Accuracy (Acc.), Macro-Precision (MP), Macro-Recall (MR), and Macro-F1 (F1) as the evaluation metrics. Note that Criminal and CAIL are imbalanced datasets. Hence, Acc. might be dominated by high-frequency charges, while MP, MR, F1 are fairer. For CEs construction and more experimental settings, we describe them in Appendix A.3.

4.1 Results

Table 2 shows the experimental results on the Criminal and CAIL datasets. We can obtain the following observations: (1) the models (SAttCaps, FLA, FewShot, LADAN, CECP)

³<https://wenshu.court.gov.cn/>

Methods		TextCNN	DPCNN	HARNN	BERT	SAttCaps	FLA	FewShot	LADAN	CECP
Criminal-S	Acc.	0.9438	0.9353	0.9429	0.9447	<u>0.9448</u>	0.8965	0.9425	-	0.9497
	MP	0.6438	0.5804	0.6476	0.5251	<u>0.6700</u>	0.6018	0.6514	-	0.7107
	MR	0.6390	0.5843	0.6466	0.5039	<u>0.6766</u>	0.5039	<u>0.6798</u>	-	0.7031
	F1	0.6288	0.5687	0.6377	0.5038	<u>0.6637</u>	0.5352	0.6492	-	0.6940
Criminal-M	Acc.	0.9516	0.9481	0.9525	<u>0.9571</u>	0.9563	0.9114	0.9562	-	0.9591
	MP	0.7040	0.6766	0.6897	0.6581	<u>0.7329</u>	0.6547	0.7129	-	0.7617
	MR	0.6838	0.6595	0.6891	0.6318	<u>0.7163</u>	0.5197	<u>0.7230</u>	-	0.7516
	F1	0.6805	0.6543	0.6778	0.6369	<u>0.7146</u>	0.5657	<u>0.7038</u>	-	0.7450
Criminal-L	Acc.	0.9567	0.9554	0.9594	0.9630	0.9600	0.9604	0.9627	-	0.9618
	MP	0.7639	0.7425	0.7856	<u>0.7697</u>	0.7823	0.7590	<u>0.7799</u>	-	0.8087
	MR	0.7177	0.7034	0.7350	0.6839	0.7392	0.7227	<u>0.7564</u>	-	0.7792
	F1	0.7277	0.7072	0.7463	0.7068	0.7467	0.7281	<u>0.7511</u>	-	0.7794
CAIL	Acc.	0.8435	0.8431	0.8550	0.8502	0.8465	0.8156	<u>0.8572</u>	0.8400	0.8646
	MP	0.8260	0.8224	0.8324	<u>0.8415</u>	0.8261	0.7871	0.8327	0.8302	0.8563
	MR	0.8271	0.8195	0.8347	0.8157	0.8318	0.7612	<u>0.8470</u>	0.8356	0.8599
	F1	0.8205	0.8145	0.8291	0.8195	0.8243	0.7693	<u>0.8341</u>	0.8262	0.8520

Table 2: Results on the Criminal and CAIL datasets. The underlined values denote the optimal results of baselines.

that are carefully designed for legal tasks usually perform better than universal models (TextCNN, DPCNN, HARNN, BERT). This demonstrates a specialized model is essential in the legal domain. (2) SAttCaps shows good performance on Criminal-S and Criminal-M. The reason might be that SAttCaps uses the self-attentive dynamic routing to explicitly capture charge-related contents in fact descriptions. Our CECP also achieves this by selecting crucial sentences. (3) BERT does not achieve satisfactory performance mainly because the maximum length that BERT can handle is 512, which is much less than the length of legal texts. (4) We use t-test with significance level 0.05 to test the significance of performance difference. Results show that in most cases CECP significantly outperforms all the baselines on both datasets.

4.2 Analysis

In this subsection, we analyze CECP from three perspectives, i.e., ablation experiments, case study and ethical analysis.

We conduct ablation experiments on the CAIL dataset to demonstrate the effectiveness of two mechanisms of CECP: (1) the PFI layer. We remove the PFI layer to form a variant called No-PFI. (2) the logical order exploited by the RL Module. We randomly choose two new orders and construct two variants (Order-1: subjective→object→objective→subject and Order-2: object→subject→subjective→objective). From Figure 3 (left) (details in Appendix A.4) we can see that the performance of these variants drops apparently, which confirms the effectiveness of the PFI layer and the logical order.

We select a representative case for the charge of *abuse of power* (AP) to explore what CECP can mine from the fact description. In Figure 3 (right), we illustrate the selected sentences with maximum aggregation weights for each CE type. Simply put, AP’s objective element is the behavior of overstepping authority, subject element is the functionary of state organ, subjective element is the deliberate destruction of normal management order, and object element is the serious damage to the interests of state and people. Here we can find that the CECP can effectively mine the instances of CEs.

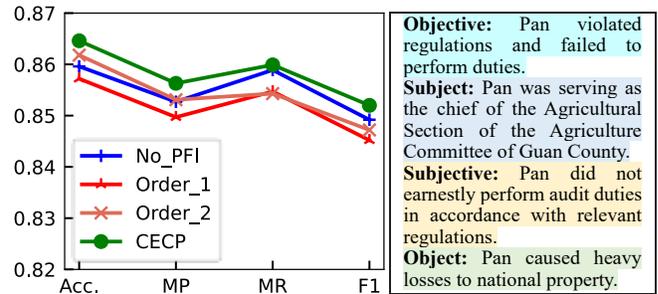


Figure 3: Ablation experiments and case study.

Ethical issues such as gender bias and racial discrimination may be learned by embedding-based legal models. The prediction error produced by these models may even be fatal to the suspect. However, it is still an open problem in this field. For CECP, the goal is not to replace legal professionals, but to ease their work and help non-professionals get help. Therefore, we should only use the result of CECP as a reference.

5 Conclusion

In this paper, we propose a novel method for charge prediction, CECP. We design a legal agent to mimic the human’s charge identification process to extract potential instances of CEs. It utilizes the sequential decision-making ability of RL and avoids laborious labeling of matching instances of CEs. Experimental results confirm the effectiveness of CECP compared to competitive baselines.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (Grant Nos.62133012, 61936006, 61876144, 61876145, 62073255, 62103314), the Key Research and Development Program of Shaanxi (Program No.2020ZDLGY04-07), and Innovation Capability Support Program of Shaanxi (Program No. 2021TD-05).

References

- [Ashley and Brüninghaus, 2009] Kevin D Ashley and Stefanie Brüninghaus. Automatically classifying case texts and predicting outcomes. *Artificial Intelligence and Law*, 17(2):125–165, 2009.
- [Chen *et al.*, 2016] Danqi Chen, Jason Bolton, and Christopher D Manning. A thorough examination of the cnn/daily mail reading comprehension task. In *ACL*, pages 2358–2367, 2016.
- [Cho *et al.*, 2014] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint*, 2014.
- [Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186, 2019.
- [Gao, 2009] Mingxuan Gao. On the rationality of quadrilateral essentials of crime composition and stick to china’s criminal legal system. *China Legal Science*, 2, 2009.
- [Hu *et al.*, 2018] Zikun Hu, Xiang Li, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. Few-shot charge prediction with discriminative legal attributes. In *COLING*, pages 487–498, 2018.
- [Johnson and Zhang, 2017] Rie Johnson and Tong Zhang. Deep pyramid convolutional neural networks for text categorization. In *ACL*, pages 562–570, 2017.
- [Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. In *EMNLP*, pages 1746–1751, 2014.
- [Kort, 1957] Fred Kort. Predicting supreme court decisions mathematically: A quantitative analysis of the “right to counsel” cases. *American Political Science Review*, 51(1):1–12, 1957.
- [Lange *et al.*, 2012] Sascha Lange, Martin Riedmiller, and Arne Voigtländer. Autonomous reinforcement learning on raw visual input data in a real world application. In *IJCNN*, pages 1–8. IEEE, 2012.
- [Le *et al.*, 2020] Yuquan Le, Congqing He, Meng Chen, Youzheng Wu, Xiaodong He, and Bowen Zhou. Learning to predict charges for legal judgment via self-attentive capsule network. In *ECAI 2020*, pages 1802–1809. 2020.
- [Li *et al.*, 2021] Lin Li, Lingyun Zhao, Peiran Nai, and Xiaohui Tao. Charge prediction modeling with interpretation enhancement driven by double-layer criminal system. *World Wide Web*, pages 1–20, 2021.
- [Lin *et al.*, 2012] Wanchen Lin, Tsungting Kuo, Tungjia Chang, Chuehan Yen, Chaoju Chen, and Shoude Lin. Exploiting machine learning models for chinese legal documents labeling, case classification, and sentencing prediction. *Proceedings of ROCLING*, page 140, 2012.
- [Luo *et al.*, 2017] Bingfeng Luo, Yansong Feng, Jianbo Xu, Xiang Zhang, and Dongyan Zhao. Learning to predict charges for criminal cases with legal basis. *arXiv preprint*, 2017.
- [Mnih *et al.*, 2016] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *ICML*, pages 1928–1937. PMLR, 2016.
- [Ren *et al.*, 2018] Pengjie Ren, Zhumin Chen, Zhaochun Ren, Furu Wei, Liqiang Nie, Jun Ma, and Maarten De Rijke. Sentence relations for extractive summarization with deep neural networks. *ACM Transactions on Information Systems (TOIS)*, 36(4):1–32, 2018.
- [Rennie *et al.*, 2017] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *CVPR*, pages 7008–7024, 2017.
- [Seo *et al.*, 2016] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. *arXiv preprint*, 2016.
- [Shen *et al.*, 2018] Yatian Shen, Jun Sun, Xiaopeng Li, Lei Zhang, Yan Li, and Xiaojong Shen. Legal article-aware end-to-end memory network for charge prediction. In *CSAE*, pages 1–5, 2018.
- [Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.
- [Xiao *et al.*, 2018] Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, and Jianfeng Xu. CAIL2018: A large-scale legal dataset for judgment prediction, 2018.
- [Xu *et al.*, 2020] Nuo Xu, Pinghui Wang, Long Chen, Li Pan, Xiaoyan Wang, and Junzhou Zhao. Distinguish confusing law articles for legal judgment prediction. In *ACL*, pages 3086–3095, 2020.
- [Yang *et al.*, 2016] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alexander J. Smola, and Eduard H. Hovy. Hierarchical attention networks for document classification. In *NAACL*, pages 1480–1489, 2016.
- [Zhao, 2003] Bingzhi Zhao. The logic order of constitutive elements of crime. *The Political Science and Law Tribune*, 6, 2003.
- [Zhong *et al.*, 2020a] Haoxi Zhong, Yuzhong Wang, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. Iteratively questioning and answering for interpretable legal judgment prediction. In *AAAI*, pages 1250–1257, 2020.
- [Zhong *et al.*, 2020b] Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuan-Jing Huang. Extractive summarization as text matching. In *ACL*, pages 6197–6208, 2020.