Contrastive Learning for Legal Judgment Prediction

HAN ZHANG, School of Information, Renmin University of China, China
ZHICHENG DOU, Gaoling School of Artificial Intelligence, Renmin University of China, China
YUTAO ZHU, University of Montreal, Canada
JI-RONG WEN, Engineering Research Center of Next-Generation Intelligent Search and Recommendation, Ministry of Education, China, and Gaoling School of Artificial Intelligence, Renmin University of China, China

Legal judgment prediction (LJP) is a fundamental task of legal artificial intelligence. It aims to automatically predict the judgment results of legal cases. Three typical subtasks are relevant law article prediction, charge prediction, and term-of-penalty prediction. Due to the wide range of potential applications, LJP has attracted a great deal of interest, prompting the development of numerous approaches. These methods mainly focus on building a more accurate representation of a case's fact description in order to improve the performance of judgment prediction. They overlook, however, the practical judicial scenario in which human judges often compare similar law articles or possible charges before making a final decision. To this end, we propose a supervised contrastive learning framework for the LJP task. Specifically, we train the model to distinguish (1) various law articles within the same chapter of a Law and (2) similar charges of the same law article or related law articles. By this means, the fine-grained differences between similar articles/charges can be captured, which are important for making a judgment. Besides, we optimize our model by identifying cases with the same article/charge labels, allowing it to more effectively model the relationship between the case's fact description and its associated labels. By jointly learning the LJP task with the aforementioned contrastive learning tasks, our model achieves better performance than the state-of-the-art models on two real-world datasets.

CCS Concepts: • **Applied computing** → *Law;*

Additional Key Words and Phrases: Deep learning, legal judgment prediction, supervised contrastive learning, legal artificial intelligence, law

ACM Reference format:
Han Zhang, Zhicheng Dou, Yutao Zhu, and Ji-Rong Wen. 2023. Contrastive Learning for Legal Judgment Prediction. *ACM Trans. Inf. Syst.* 41, 4, Article 113 (April 2023), 25 pages.
https://doi.org/10.1145/3580489

This work was supported by the National Natural Science Foundation of China No. 62272467 and No. 61872370, Beijing Outstanding Young Scientist Program NO. BJJWZYJH012019100020098, the Fundamental Research Funds for the Central Universities, the Research Funds of Renmin University of China NO. 22XNKJ34, Public Computing Cloud, Renmin University of China, and Intelligent Social Governance Platform, Major Innovation & Planning Interdisciplinary Platform for the "Double-First Class" Initiative, Renmin University of China. The work was partially done at Beijing Key Laboratory of Big Data Management and Analysis Methods, and Key Laboratory of Data Engineering and Knowledge Engineering, MOE.
Authors' addresses: H. Zhang, School of Information, Renmin University of China, Beijing, China; email: zhanghanjl@ruc.edu.cn; Z. Dou (corresponding author), Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China; email: dou@ruc.edu.cn; Y. Zhu, University of Montreal, Montreal, Canada; email: yutaozhu94@gmail.com; J.-R. Wen, Engineering Research Center of Next-Generation Intelligent Search and Recommendation, Ministry of Education, Beijing, China, and Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China; email: jrwen@ruc.edu.cn.

1046-8188/2023/04-ART113 $15.00
https://doi.org/10.1145/3580489

1 INTRODUCTION

In recent years, there has been an increasing interest in the application of artificial intelligence approaches to assist with legal judgment. As illustrated in Table 1, **legal judgment prediction (LJP)** attempts to predict a case's judgment results, such as the applicable law article, charge, and term of penalty, based on the fact description. Legal judgment prediction can not only increase the efficiency of the judges' work but also provide valuable legal advice to the general public.

In the literature, legal judgment prediction is typically formulated as three text classification tasks: relevant law article prediction, charge prediction, and term-of-penalty prediction. Various methods have been proposed and achieved promising results [5, 34–36, 38] with the application of state-of-the-art neural networks and text representation models [4, 12, 14, 28]. However, these approaches still have limits in identifying cases involving confusing law articles or charges.

On the one hand, previous works [16, 34, 36] have incorporated law articles as external information into the fact description representation in order to improve the relevant law article prediction. Nevertheless, these methods neglect the practical judgment process, in which human judges usually need to compare and analyze similar law articles to determine the most relevant one, especially when dealing with very similar law articles. Let us use an example to illustrate this. As shown in Figure 1, *Article #114* and *Article #115* belong to Chapter II, *Crimes of Endangering Public Security*, of "Criminal Law of the People's Republic of China." Both law articles explain several **identical** charges (such as *Crime of Arson* and *Crime of Breaching Dikes*), whereas *Article #115* further introduces a group of **similar** charges (such as *Crime of Negligently Causing a Fire* and *Crime of Negligently Endangering Public Security by Dangerous Means*). To determine which of these analogous law articles is applicable to a particular legal case, the judge has to analyze and compare the specific provisions contained in the text of each law article. As shown in the example in Table 1, the fact description of the case is that a criminal suspect sets fire and causes property damage. The potentially applicable law articles include *Article #114* and *Article #115*, since they both contain *Crime of Arson*. The judge needs to compare and analyze the specific provisions of the two articles (e.g., different consequences as marked in Figure 1) and may determine that *Article #114* is more applicable.

On the other hand, the majority of existing legal judgment prediction approaches [34, 36, 38] rely on the fact description of a single case to predict its judgment result. They overlook the commonalities between similar cases. In practice, historical cases involving similar charges may serve as a source of reference and comparison for making a judgment. It is common for human judges to compare and analyze similar situations before making a decision. As shown in Figure 1, *Article #115* stipulates *Crime of Arson* and *Crime of Negligently Causing a Fire*, and the provisions of these two charges are very similar. The judge can compare and analyze existing cases of *Crime of Arson* and *Crime of Negligently Causing a Fire* that have the same law article label *Article #115* in order to distinguish between them.

To tackle these challenges, we propose simulating the judgment process of human judges, which entails evaluating similar articles and cases with similar charges in order to make final judgments. We find that contrastive learning can naturally fit our goal. The basic idea of contrastive learning is to pull close the vector representations of positive pairs in a high-dimensional space, as well as to push apart the ones of negative pairs. In our work, we investigate the hierarchical structure of "chapter→article→charge" and develop three contrastive learning tasks for the legal judgment prediction task:

• **First**, we treat the fact description of a legal case and the associated law articles as a positive pair, whereas other law articles from *the same chapter* are used to construct negative pairs. By pulling close the vector representations of the positive pair and pushing apart the ones of the

Table 1. An Illustration of the Legal Judgment Prediction Task

Fact Description
On XX, XXX, in order to vent his emotions due to family chores, the defendant Zhou **deliberately ignited curtains, sheets, and other objects in the kitchen** of room XX, building XX, XX community, XX town, XX District, XX City, and **fled the scene after the fire could not be controlled**. After that, the people in the community put out the fire. After identification, the value of the burned items was YYY yuan. The public prosecution believes that the defendant Zhou deliberately burned public and private property, endangering public safety...
Relevant Law Article
Criminal Law of the People's Republic of China Chapter II: Crimes of Endangering Public Security *Article #114* **[Crime of Arson] Whoever commits arson**, breaches a dike, causes an explosion, spreads toxic, radioactive, infectious disease pathogens and other substances, or endangers public security by other dangerous methods, but has not caused serious consequences, shall be sentenced to **fixed-term imprisonment of not less than three years but not more than 10 years.**
Charge: **Crime of Arson**
Term of Penalty: **A fixed-term imprisonment of 36 months**

A judge should analyze and reason on the fact description of a legal case and then select relevant law articles, charges, and term of penalty to convict the offender. In this legal case, the judge finally selects *Article #114* of Chapter II (Crimes of Endangering Public Security) in Criminal Law of the People's Republic of China as the relevant law article. The charge of the offender is convicted as Crime of Arson in *Article #114*, and the term of penalty of the offender is 36 months according to *Article #114*.

negative pair, our model can learn to capture the fine-grained clues in the fact descriptions for identifying similar law articles in a Law, which is important for the relevant article prediction task.

• **Second**, we treat the fact description of the current legal case and a case's fact description with the same charge label as a positive pair, while the fact descriptions of cases whose charges are under the same law article or related law articles are utilized to construct the negative pair. We employ the fact description of a case rather than the associated charge label to create a positive pair because charge label texts are much shorter than law articles and lack sufficient identification information. Similarly, with contrastive learning, our method can learn the fact description details to distinguish cases with similar charge labels, which is beneficial for charge prediction.

• **Third**, in order to fully utilize the article/charge label information, we design a label-aware contrastive learning task that operates within training batches. Cases with identical article/charge labels are treated as positive pairs, while the remaining cases in the batch serve as negatives. This task can aid in modeling the relationship between the case's fact description and the article/charge label.

We train our model by learning the three proposed contrastive learning tasks concurrently with the three sub-tasks of legal judgment prediction. Our method is called CL4LJP, which stands for **C**ontrastive **L**earning framework for **L**egal **J**udgment **P**rediction. We conduct experiments on two real-world datasets. Experimental results show that our proposed method outperforms the existing methods significantly, which demonstrates the effectiveness of our approach. Our further experiments validate the flexibility of our method and its robustness in tail categories.

Our contributions are three fold:

(1) We propose learning better representation for fact description of a legal case by leveraging similar law articles and cases involving similar law articles or charges. The specific provisions of

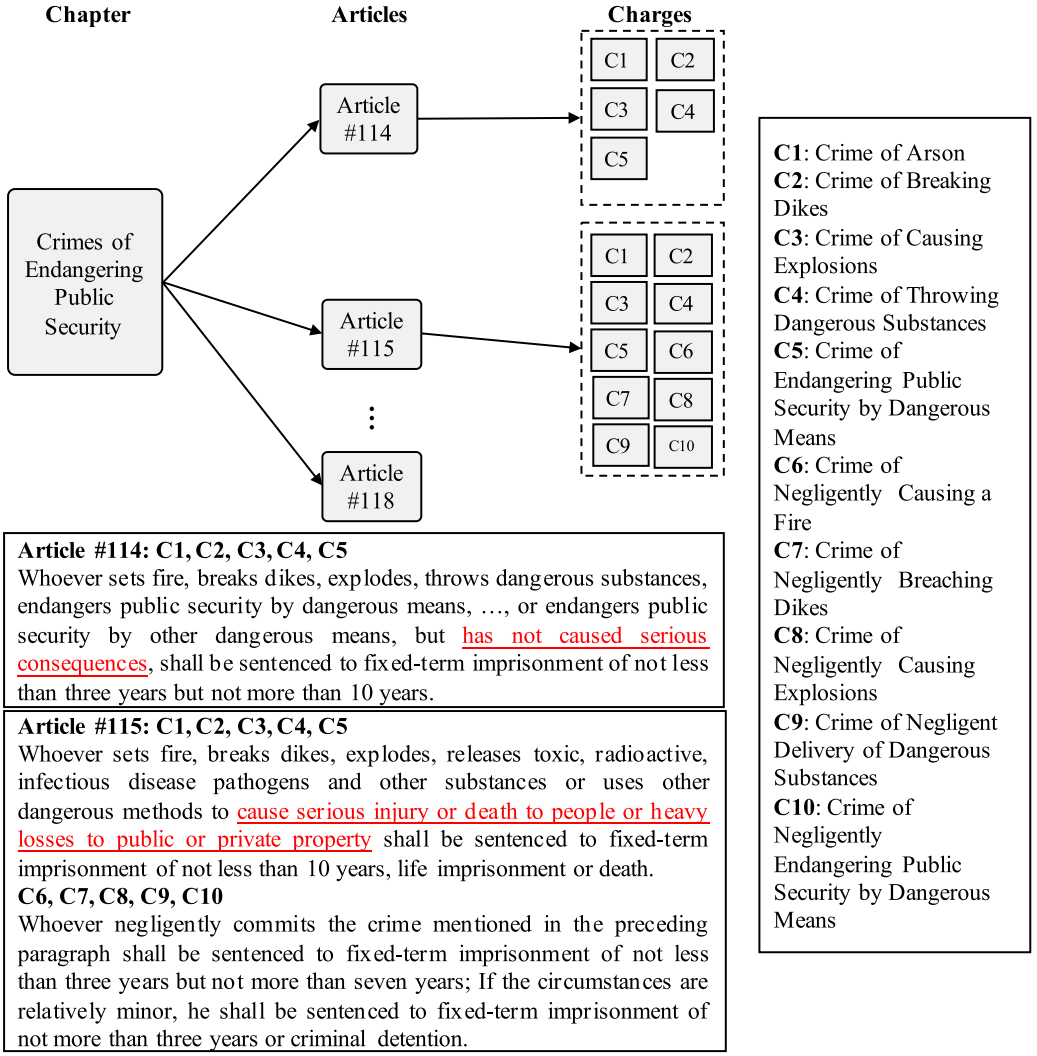

Fig. 1. An example of the structure of the law articles of "Criminal Law of the People's Republic of China." Article #114, Article #115, and Article #118 belong to the same chapter, *Crimes of Endangering Public Security* (Chapter II), of this Law. Article #114 and Article #115 jointly stipulate the same charges (C1–C5), while Article #115 stipulates more charges (C6–C10). It can be seen from the figure that the law articles under the same chapter are usually similar. The charges under the same law articles or the same chapter are also similar and difficult to distinguish.

these law articles and fact descriptions of these similar cases can provide fine-grained clues for identifying the associated law articles or charges.

(2) We design a contrastive learning framework with three learning tasks. These tasks can assist the model in differentiating between similar law articles and charges and capturing the relationship between fact descriptions and law article/charge labels.

(3) We conduct experiments on two large-scale real-world legal judgment prediction datasets. The better performance confirms the effectiveness of applying contrastive learning in legal

judgment prediction. Our further study investigates the effect of various encoders and the performance of tail classes, and the results demonstrate the robustness of our method.

2 RELATED WORK

2.1 Legal Judgment Prediction

Early studies [7, 13, 20, 21, 25] focused on applying mathematical and statistical methods to legal case analysis. These methods heavily rely on manual rules or handcrafted features, thus limiting their applicability in reality. Due to the rapid growth of neural networks in recent years, various neural network models have been proposed for analyzing legal cases.

Some studies concentrated on how to make better use of the semantic information of law articles and charges. In light of the fact that judges must determine the suspect's charges based on the law articles, and the law articles can provide important supplemental information for predicting legal judgments. Luo et al. [16] utilized the attention mechanism to incorporate law articles into the representation of the legal case's fact description, hence enhancing the model's representation capability. As the descriptions of similar charges in the law articles are alike and the performance of the few-shot charges in the previous work is very poor, Hu et al. [10] introduced artificial features of charges to improve the performance on few-shot charges. Wang et al. [29] utilized the hierarchical structure between articles and charges to improve the performance of multi-label charge prediction, taking into account that the laws and articles are organized in a tree-like hierarchy. Considering that the descriptions of some law articles are difficult to distinguish for the model, Xu et al. [34] proposed an attention-based model. It applies a graph distillation operator to learn the differences between confusing law articles.

Thereafter, researchers started to consider the relationship between the three subtasks of legal judgment prediction, namely, the relevant law article prediction, charge prediction, and term-of-penalty prediction. For instance, in the Statutory Law system, judges often identify the applicable law articles, assess the charges, and then determine the term of penalty based on the law articles. Zhong et al. [38] exploited such explicit dependencies between subtasks and developed a topological framework for multitask learning. After evaluating the relevant law articles, charges, and terms of penalty in real work, judges should reconfirm whether the charges and terms of penalty are consistent with the descriptions of the relevant law articles. This paradigm was expanded by Yang et al. [35], using multi-perspective forward prediction and backward verification. Dong and Niu [5] addressed the legal judgment prediction subtasks as a graph node classification problem and utilized BERT [4] to represent the case description, based on the fact that the law article, charge, and term-of-penalty labels are organized in a large graph.

More recently, many studies have attempted to simulate human judges and consider the real judgment process. Yue et al. [36] used the intermediate results of subtasks to divide the fact description into distinct conditions and create predictions, taking into account the fact that judges use different sections of the fact description in a case to make a judgment in the actual legal scenario. Considering that plaintiffs' claims and court debate data are also important for the legal judgment prediction task, Ma et al. [17] separated the case description and further incorporated these data to facilitate the prediction of legal judgments.

In addition, a number of studies accounted for the fact that each prior legal case is a unique application of law articles. Some cases may expand the scope of the law article (called expanded cases), while others may contract the scope of the law article (called contracted cases). Valvoda et al. [26] introduced an outcome prediction legal task of contracted cases and designed a model based on the dynamics of a court process. This model improves the prediction performance of both contracted cases and expanded cases.

In contrast to existing studies, our method is inspired by the human judges' behavior in comparing and analyzing similar law articles and cases under similar law articles or charges. We develop a framework based on supervised contrastive learning. By training the model to distinguish ambiguous law articles and ambiguous cases with similar charge labels, it can capture fine-grained clues and generate a more accurate representation of the fact description, hence enhancing the performance of legal judgment prediction.

2.2 Contrastive Learning

Contrastive learning aims at learning effective representation of data by pulling semantically close neighbors together and pushing apart other non-neighbors [8, 30]. It has been widely applied in computer vision [2, 9, 11, 27], natural language processing [6, 32], and information retrieval [40, 41], due to its high efficiency in leveraging the training data without the need for annotation. The key to contrastive learning is to identify semantically close neighbors. Commonly, in visual representation, neighbors are formed by performing two random transformations on the same image (such as flipping, cropping, rotation, and distortion) [2]. Similarly, in text representation, data augmentation techniques such as word deletion, reordering, and substitution can be employed to generate text neighbors from a given text sequence [18, 32]. Contrastive learning can also be combined with supervised learning [6, 11], where the label information can significantly improve the models' performance.

In this article, we propose a supervised contrastive learning framework to optimize fact representation and improve legal judgment prediction. The promising results indicate the great potential of applying contrastive learning to legal judgment prediction.

3 BACKGROUND AND PROBLEM STATEMENT

Before diving into the details of our method, we first briefly introduce several key notations, concepts, and definitions in LJP.

Law articles are generally organized in a hierarchical structure. As shown in Figure 1, in a law document, articles are organized into different chapters according to their commonalities, and charges are grouped into different articles similarly. Such structural information inspires us to design contrastive learning tasks at different levels, which will be introduced later. Formally, we represent all m law articles and all n different charges as

$$A = \{a_1, a_2, \ldots, a_m\}, \tag{1}$$

$$C = \{c_1, c_2, \ldots, c_n\}. \tag{2}$$

Since law articles and charges are usually texts, we use a_i and c_j to denote them, respectively.

Legal cases in our study consist of a fact description and a judgment result made by human judges. The fact description contains the suspected criminal behavior of a suspect. It is also a text, and we use f to denote it. The judgment result of a legal case includes the relevant law article, the final charge, and the term of penalty. They are accordingly denoted as y_a, y_c, and y_p. For example, as illustrated in Table 1, given the fact description f, the relevant law article y_a is *Article #114*, the charge y_c is *Crime of Arson*, and the term of penalty y_p is *A fixed-term imprisonment 36 months*. With the above notations, a legal case can be represented by a quadruple (f, y_a, y_c, y_p).

Following previous studies [34, 36, 38], we consider solving the three subtasks in legal judgment prediction simultaneously: **relevant law article prediction**, **charge prediction**, and **term-of-penalty prediction**. Such a multi-tasking learning paradigm is proved to be more effective than separate modeling of each subtask [34, 38]. Given the dataset D, our target is to train a model $F(\cdot)$ such that for a new fact description f_{test} of a legal case, the model can predict the labels of the three subtasks, namely $F(f_{\text{test}}) = (\hat{y}_a, \hat{y}_c, \hat{y}_p)$, where $\hat{y}_a$, $\hat{y}_c$, and $\hat{y}_p$ represent the predicted relevant law

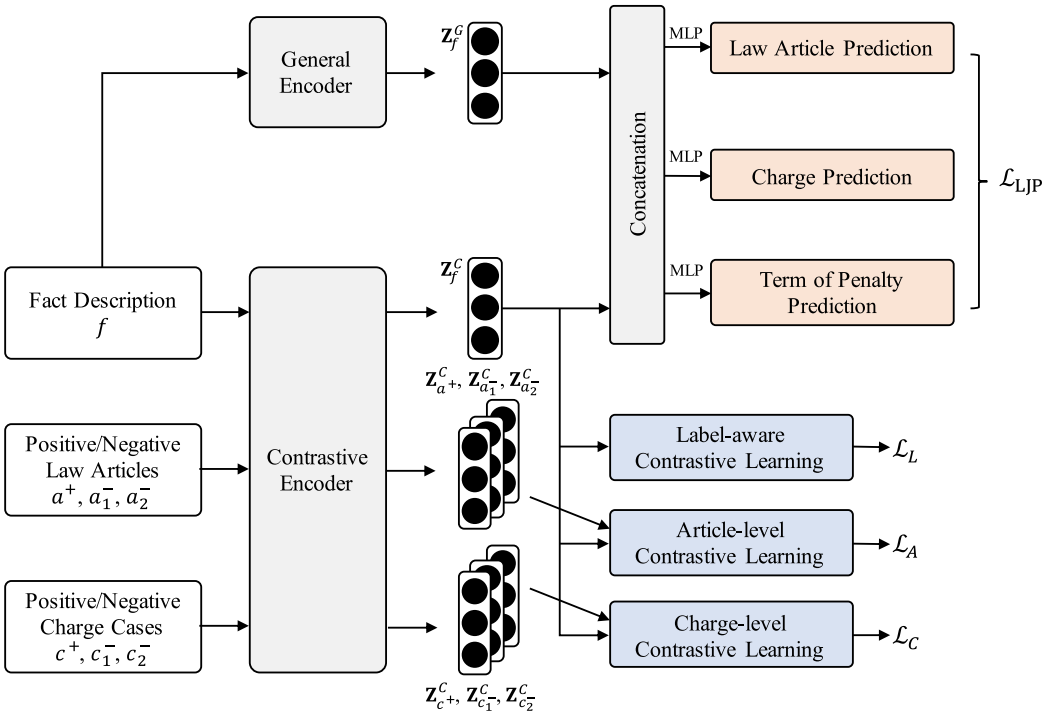

Fig. 2. The contrastive learning framework for legal judgment prediction. f represents the fact description of a legal case. a^+ represents the legal text of the law article label of the case, and a_1^-, a_2^- represent two negative law articles of a^+. c^+ represents a case with the same charge label as the case with fact description f, and c_1^-, c_2^- represent two negative cases with similar charge labels. Superscript C and G denote the contrastive encoder and the general encoder, respectively.

article, charge, and term of penalty, respectively. Consistent with existing studies [34, 38], only the legal cases in which each subtask has only one label are considered in the dataset. We will consider multi-label prediction in our future work.

4 THE PROPOSED FRAMEWORK

In the actual judgment process, judges usually compare and analyze similar law articles and cases with similar charges. Inspired by this process, we propose a supervised contrastive learning framework (CL4LJP) based on the law article structure information. In the following sections, we first provide an overview of our proposed framework, and then we describe the details of each component in our framework. The optimization process is introduced in the last part of this section.

4.1 Overview

The overview of our framework is shown in Figure 2. In general, CL4LJP is a multi-task learning framework that jointly performs three legal judgment prediction subtasks with our proposed three contrastive learning tasks. The main components and training process of CL4LJP can be described as follows:

(1) The **general encoder** encodes the fact description f into a vector representation $\mathbf{Z}_f^G$.

(2) The **contrastive encoder** encodes the fact description f into a vector representation $\mathbf{Z}_f^C$. In addition, it also encodes similar law articles and cases associated with similar charges as $\mathbf{Z}_a^C$ and $\mathbf{Z}_c^C$.

(3) Three **contrastive learning tasks**, i.e., article-level contrastive learning, charge-level contrastive learning, and label-aware contrastive learning, are performed based on the representation $\mathbf{Z}_f^C$, $\mathbf{Z}_a^C$, and $\mathbf{Z}_c^C$.

(4) Two representations of the fact description are **concatenated** as $\mathbf{Z}_f = [\mathbf{Z}_f^G; \mathbf{Z}_f^C]$ and used to perform **three legal judgment prediction subtasks**.

(5) The model is optimized by the loss of all tasks.

During the inference stage, CL4LJP only employs the two encoders to compute the concatenated representation $\mathbf{Z}_f$ and then uses three **multi-layer perceptrons (MLPs)** to predict the judgment results of three subtasks. The contrastive learning tasks are only performed during the training stage to optimize the encoders.

4.2 Encoders

As shown in Figure 2, we use the same neural structure to encode the fact description of legal cases and law articles. For efficiency, we follow previous work [37] and employ a CNN-based encoder. Although CNN is used, our framework CL4LJP is flexible with the choice of the encoder. Other neural structures, such as **recurrent neural networks (RNNs)** or **pre-trained language models (PLMs)** can also be used for encoding the fact description. We have also tried BERT [4] as the encoder in our experiments and more details are included in Section 5.6.

Specifically, the fact description $f = (w_1, \ldots, w_n)$ with n words is first represented as a word embedding sequence $\mathbf{f}$ by looking up the pre-trained word embedding table $\mathbf{E}$:

$$\mathbf{f} = [\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n], \tag{3}$$

$$\mathbf{e}_i = \text{Look-Up}(\mathbf{E}(w_i)), \tag{4}$$

where $\mathbf{f} \in \mathbb{R}^{n \times d}$, and $\mathbf{e}_i \in \mathbb{R}^d$ is the embedding of the ith word w_i. Then, to extract the information of the fact description in different granularities, we apply four 1D-CNNs with different kernels and compute the representation $\mathbf{Z}'$ of the fact description f as follows:

$$\mathbf{Z}'_f = \left[\mathbf{Z}_f^1; \mathbf{Z}_f^2; \mathbf{Z}_f^3; \mathbf{Z}_f^4\right], \tag{5}$$

$$\mathbf{Z}_f^1 = \text{1D-CNN}(\mathbf{f}, k_1), \tag{6}$$

$$\mathbf{Z}_f^2 = \text{1D-CNN}(\mathbf{f}, k_2), \tag{7}$$

$$\mathbf{Z}_f^3 = \text{1D-CNN}(\mathbf{f}, k_3), \tag{8}$$

$$\mathbf{Z}_f^4 = \text{1D-CNN}(\mathbf{f}, k_4), \tag{9}$$

where k_t is the kernel size of the tth 1D-CNN, and $[;]$ is the concatenation operation. By using different kernels, the semantic information in n-grams can be captured. The concatenation operation is further applied to integrate such information. Law articles can be represented as $\mathbf{Z}_a$ in a similar way; we omit the calculation details.

In our framework, we employ two encoders, i.e., a **general** encoder and a **contrastive** encoder, for different purposes.

The general encoder only encodes the fact description of a legal case, and its parameters are tuned by the objective in legal judgment prediction subtasks. Such an encoder is optimized to generate representations specific to the legal judgment prediction task (denoted as $\mathbf{Z}_f^G$). The fact description f of a legal case after the general encoder is represented as follows:

$$\mathbf{Z}_f^G = \left[\mathbf{Z}_f^{G1}; \mathbf{Z}_f^{G2}; \mathbf{Z}_f^{G3}; \mathbf{Z}_f^{G4}\right], \tag{10}$$

where $\mathbf{Z}_f^{G1}$, $\mathbf{Z}_f^{G2}$, $\mathbf{Z}_f^{G3}$, and $\mathbf{Z}_f^{G4}$ are calculated according to Equations (6) through (9).

Another contrastive encoder is applied to encoding the fact description, the corresponding positive/negative law articles, and the fact descriptions in sampled positive/negative cases (we denote them as $\mathbf{Z}_f^C$, $\mathbf{Z}_a^C$, and $\mathbf{Z}_c^C$, which will be introduced later). It is optimized by both contrastive learning and legal judgment prediction objectives. The representation generated by the contrastive encoder is more accurate at capturing the relationship between articles, charges, and legal cases and is also effective for the legal judgment prediction task. Consequently, we integrate the representations calculated by both encoders into our CNN-based model, and our experimental results will demonstrate their effectiveness.

4.3 Article-level Contrastive Learning

As introduced in Section 1, when making a judgment for a legal case, it is common for the judge to compare similar law articles, assess their differences, and finally determine which law article is the most applicable to the case. For example, as shown in Figure 1, both Articles #114 and #115 contain similar stipulations for *Crime of Arson*; however, Article #115 pays more attention to the serious consequences caused by the case. If a suspect sets a fire and causes serious personal and property damage, Article #115 should be applied instead of Article #114.

To simulate such a process, we design an article-level contrastive learning task to enhance the model's representation capability by distinguishing similar articles. This learning task can be generally described as pulling close the representation of the case and that of the relevant law article while pushing apart the representation of other similar articles.

We design a rule-based contrastive sampling for each case as shown in Figure 3. For the fact description of a case, we first couple the fact with its corresponding article and construct a **positive** pair (f, a^+). To construct **negative** pairs, we leverage the hierarchical structure of law articles. Concretely, we randomly sample a group of articles **from the same chapter** as a^-. According to our observation, law articles in the same chapter have some commonalities; hence, learning to distinguish them may enable the model to capture more fine-grained information in the law articles, which is advantageous for legal judgment prediction. Let us use the example in Figure 1 to illustrate this idea: Both Article #114 and Article #115 belong to the same chapter, *Crimes of Endangering Public Security*. They stipulate the same charges (C1–C5), the main difference being the severity of the consequences. Therefore, learning to distinguish them can allow the model to focus more on the consequence description of cases.

Formally, supposing that the fact description of a legal case is f, the relevant law article label is y_a, and the corresponding legal text of the relevant law article label is a, we obtain a set of negative articles $\{a_1^-, \ldots, a_l^-\}$, where l is a hyper-parameter. They are encoded using the contrastive encoder and represented as $\mathbf{Z}_f^C$, $\mathbf{Z}_{a^+}^C$, and $\{\mathbf{Z}_{a_i^-}^C\}_{i=1}^l$. According to Equation (5), the calculation is as follows:

$$\mathbf{Z}_f^C = \left[\mathbf{Z}_f^{C1}; \mathbf{Z}_f^{C2}; \mathbf{Z}_f^{C3}; \mathbf{Z}_f^{C4}\right], \tag{11}$$

$$\mathbf{Z}_{a^+}^C = \left[\mathbf{Z}_{a^+}^{C1}; \mathbf{Z}_{a^+}^{C2}; \mathbf{Z}_{a^+}^{C3}; \mathbf{Z}_{a^+}^{C4}\right], \tag{12}$$

$$\mathbf{Z}_{a_i^-}^C = \left[\mathbf{Z}_{a_i^-}^{C1}; \mathbf{Z}_{a_i^-}^{C2}; \mathbf{Z}_{a_i^-}^{C3}; \mathbf{Z}_{a_i^-}^{C4}\right]. \tag{13}$$

We apply a contrastive learning loss to minimize the distance between $\mathbf{Z}_f^C$ and $\mathbf{Z}_{a^+}^C$, while maximizing those between $\mathbf{Z}_f^C$ and $\{\mathbf{Z}_{a_i^-}^C\}_{i=1}^l$, which is defined as

$$\mathcal{L}_A = -\log \frac{\exp\left(\delta\left(\mathbf{Z}_f^C, \mathbf{Z}_{a^+}^C\right)\right)}{\exp\left(\delta\left(\mathbf{Z}_f^C, \mathbf{Z}_{a^+}^C\right)\right) + \sum_{j=1}^l \exp\left(\delta\left(\mathbf{Z}_f^C, \mathbf{Z}_{a_i^-}^C\right)\right)}, \tag{14}$$

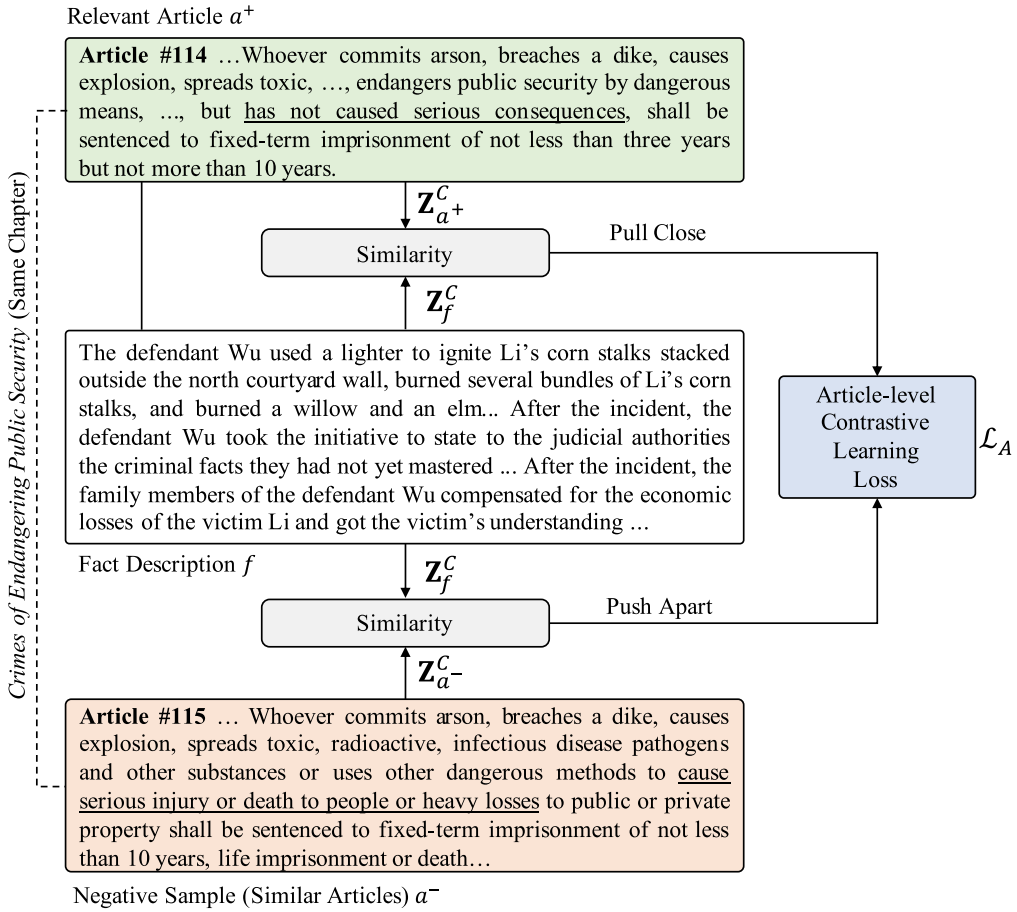

Fig. 3. An example of article-level contrastive sampling. In this example, Article #114 and Article #115 are two very similar law articles. They both stipulate the crimes of endangering public security, and the difference lies in the severity of the consequences. For a legal case with the law article label of Article #114 and the charge label of the crimes of endangering public security, the article-level contrastive learning subtask should pull close the vector representation $\mathbf{Z}_f^C$ of the fact description to the vector representation $\mathbf{Z}_{a^+}^C$ of Article #114, and pull apart the vector representation $\mathbf{Z}_f^C$ of the fact description from the vector representation $\mathbf{Z}_{a^-}^C$ of Article #115.

where the function $\delta(\cdot)$ represents the similarity of two vectors, which is implemented by cosine similarity in our article.

4.4 Charge-level Contrastive Learning

In addition to analyzing similar articles, the judge should also compare similar charges under the same article or similar articles of the same chapter for judging a legal case. In comparison to articles, charges are finer-grained descriptions. However, as shown in Figure 4, the charge text is usually very short. Directly comparing two short charge texts is insufficient to learn a good representation. To tackle this problem, we propose to leverage the cases associated with the charge for the comparison. As can be seen, the fact description f is associated with the charge *crime of arson*. We randomly sample another case c^+ that is also associated with this charge (as the surrogate

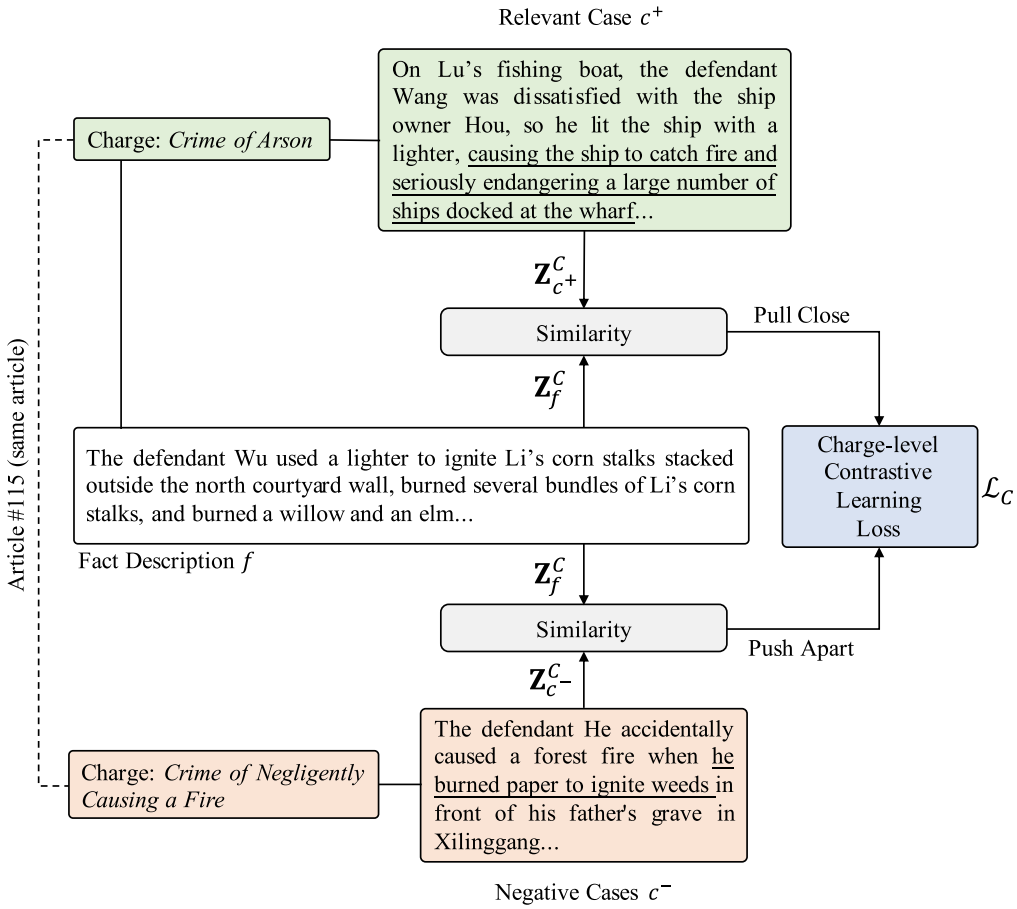

Fig. 4. An example of charge-level contrastive sampling. In this example, the Crime of Arson and the Crime of Negligently Causing a Fire are two very similar charges, as shown in Figure 1. The difference between the two charges lies in whether the suspect intentionally or unintentionally led to serious consequences. For a legal case with a charge label of the Crime of Arson and a charge label of the Crime of Negligently Causing a Fire, the charge-level contrastive learning subtask should pull close the vector representation $\mathbf{Z}_f^C$ of the fact description to the vector representation $\mathbf{Z}_{c^+}^C$ of a relevant case with a charge label of Crime of Arson and pull apart the vector representation $\mathbf{Z}_f^C$ of the fact description from the vector representation $\mathbf{Z}_{c^-}^C$ of a negative case with the charge label of Crime of Negligently Causing a Fire.

of the charge) and treat (f, c^+) as a **positive** pair. For the negative charges, we consider the charge *crime of negligently causing a fire* under the same article (#115). The cases c^- associated with such charges can be randomly sampled for constructing **negative** pairs (f, c^-). By comparing the cases c^+ under the same charge and those c^- under similar charges, the model can learn the basis for applying a charge, which is valuable for legal judgment prediction tasks. Note that in the Law, some articles have only one charge. For these charges, we use the charges under the law articles of the same chapter as their negatives.

Formally, the fact description of a case is denoted as f, and the charge label is y_c. The corresponding sampled positive case's fact description is c^+, and the fact descriptions of randomly sampled negative cases are $\{c_1^-, \ldots, c_{l'}^-\}$, where l' is also a hyper-parameter. We use the contrastive encoder

to compute their representation as $\mathbf{Z}_f^C$, $\mathbf{Z}_{c^+}^C$, and $\{\mathbf{Z}_{c_i^-}^C\}_{i=1}^{l'}$, respectively. Following Equation (11), the calculation can be described as:

$$\mathbf{Z}_{c^+}^C = \left[\mathbf{Z}_{c^+}^{C1}; \mathbf{Z}_{c^+}^{C2}; \mathbf{Z}_{c^+}^{C3}; \mathbf{Z}_{c^+}^{C4}\right], \tag{15}$$

$$\mathbf{Z}_{c_i^-}^C = \left[\mathbf{Z}_{c_i^-}^{C1}; \mathbf{Z}_{c_i^-}^{C2}; \mathbf{Z}_{c_i^-}^{C3}; \mathbf{Z}_{c_i^-}^{C4}\right]. \tag{16}$$

The charge contrastive learning loss of each case is devised to minimize the distance between the vector $\mathbf{Z}_f^C$ and $\mathbf{Z}_{c^+}^C$ and simultaneously maximize the distances between the vector $\mathbf{Z}_f^C$ and $\{\mathbf{Z}_{c_i^-}^C\}_{i=1}^{l'}$. The calculation is formulated as follows:

$$\mathcal{L}_C = -\log \frac{\exp\left(\delta\left(\mathbf{Z}_f^C, \mathbf{Z}_{c^+}^C\right)\right)}{\exp\left(\delta\left(\mathbf{Z}_f^C, \mathbf{Z}_{c^+}^C\right)\right) + \sum_{i=1}^{l'} \exp\left(\delta\left(\mathbf{Z}_f^C, \mathbf{Z}_{c_i^-}^C\right)\right)}. \tag{17}$$

Again, $\delta(\cdot)$ is the similarity function.

4.5 Label-aware Contrastive Learning

In the aforementioned article-level and charge-level contrastive learning, we actually use "hard" negative samples. These hard negative samples can help learn the subtle differences between similar articles or cases with similar charges and train the model to focus on the subtle fact description information that is useful for distinguishing these similar articles and charges.

In addition to these hard negatives, to fully exploit the supervision signals of law article and charge labels, and inspired by the supervised contrastive learning [11], we leverage the cases with the same law article and charge labels in a mini-batch to form positive pairs, whereas other cases in the same mini-batch are used for constructing negative pairs. This allows for more accurate modeling of the relationship between the fact description and its corresponding label.

As shown in Figure 5 and Figure 6, for a specific case in a mini-batch, we first pair it with each case that has the same article label as the positive pair, and other cases in the batch with different article labels are deemed as negatives. Formally, we define this label-aware contrastive learning loss for the article label as

$$\mathcal{L}_{AL} = -\frac{1}{|B_k|} \sum_{f_i \in B_k} \log \frac{\sum_{f_j \in B_k, j \neq i} \mathbb{1}_{y_{a,c_i} = y_{a,c_j}} \exp\left(\delta\left(\mathbf{Z}_{c_i}^C, \mathbf{Z}_{c_j}^C\right)\right)}{\sum_{f_j \in B_k, j \neq i} \exp\left(\delta\left(\mathbf{Z}_{c_i}^C, \mathbf{Z}_{c_j}^C\right)\right)}, \tag{18}$$

where B_k is the kth mini-batch of training data. $Z_{c_i}^C$ and $Z_{c_j}^C$ are the representation vectors of the anchor case and the positive cases in the batch calculated by the contrastive encoder. y_{a,c_i} and y_{a,c_j} are the relevant law article labels, respectively.

Similarly, we can get the label-aware contrastive learning loss $\mathcal{L}_{CL}$ for the charge label. Then, the total label-aware contrastive learning loss is

$$\mathcal{L}_L = \mathcal{L}_{AL} + \mathcal{L}_{CL}. \tag{19}$$

4.6 Optimization

We use the fact representation output by the encoder module to predict the judgment results as follows:

$$\mathbf{Z}_f = \left[\mathbf{Z}_f^G; \mathbf{Z}_f^C\right], \tag{20}$$

$$y_i = \mathrm{MLP}_i(\mathbf{Z}_f), \tag{21}$$

where MLP_i is the multi-layer perceptron for the ith legal judgment prediction subtask.

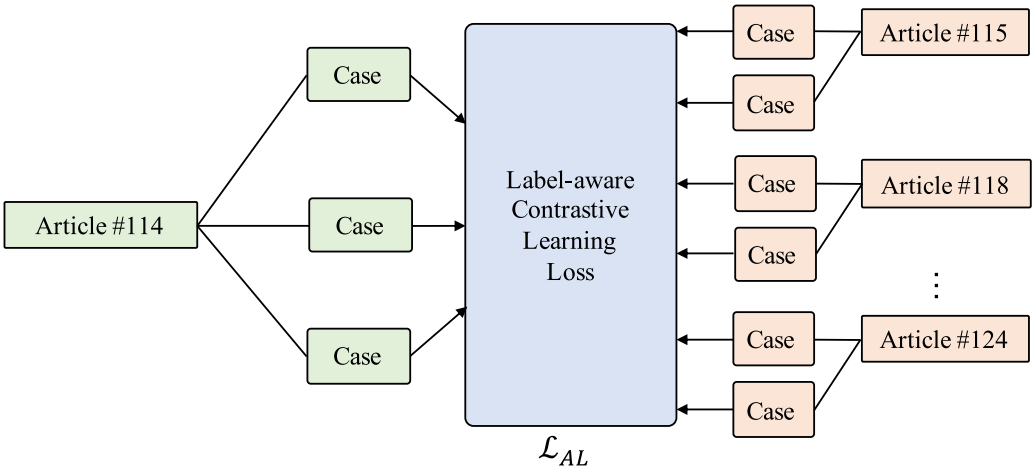

Fig. 5. An example of label-aware contrastive learning for law articles. For example, in a training batch, for the cases with the label of Article #114, other cases with different law article labels are regarded as negative cases, such as the cases with Article #115 and Article #118, as well as the cases with other law article labels in the same batch.

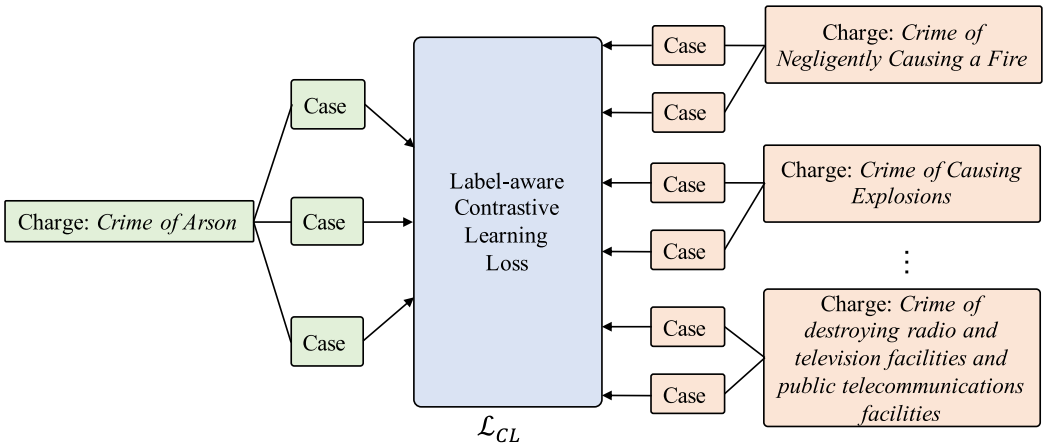

Fig. 6. An example of label-aware contrastive learning for charges. For example, in a training batch, for the cases with the charge label of "Crime of Negligently Causing a Fire," other cases with different charge labels are regarded as negative cases, such as the cases with the charge label of "Crime of Causing Explosions" and the charge label of "Crime of destroying radio and television facilities and public telecommunications facilities," as well as the cases with other charge labels in the same batch.

Legal judgment prediction loss. For each subtask of legal judgment prediction, we use the cross-entropy loss function to train the model. The overall legal judgment prediction loss is computed as

$$\mathcal{L}_{\text{LJP}} = -\sum_{i=1}^{3} \lambda_i \sum_{j=1}^{|Y_j|} y_{i,j} \log(\hat{y}_{\text{i,j}}), \tag{22}$$

Table 2. The Statistics of the CAIL Dataset

Dataset	CAIL-small	CAIL-big
# Training Set Cases	106,750	1,648,600
# Testing Set Cases	25,652	200,449
# Law Articles	94	115
# Charges	109	129
# Terms of Penalty	11	11

The CAIL dataset includes two sub-datasets: the CAIL-small and the CAIL-big dataset.

where $|Y_j|$ denotes the total number of labels for subtask i. λ_i is the hyper-parameter weight of each subtask in the legal judgment prediction task.

Total loss. Finally, we optimize our model jointly by the legal judgment prediction loss and three contrastive learning task losses as follows:

$$\mathcal{L}_{\text{Total}} = \frac{1}{|D|} \sum_{f_i \in D} (\mathcal{L}_{\text{LJP}} + \alpha_1 \mathcal{L}_A + \alpha_2 \mathcal{L}_C + \alpha_3 \mathcal{L}_L), \tag{23}$$

where α_i is a hyper-parameter to balance the weight of each contrastive learning task, D represents the dataset, and $|D|$ is the number of legal cases in the dataset.

5 EXPERIMENTS

We introduce the experimental settings and results in this section.

5.1 Datasets and Preprocessing

To validate the effectiveness of our method, we conduct experiments on the **C**hinese **AI** and **L**aw challenge (CAIL2018) dataset [33], which consists of two sub-datasets, namely, CAIL-small and CAIL-big. CAIL2018 contains legal cases published by the Supreme People's Court of China. Each case contains a fact description and a corresponding judgment result (i.e., relevant law articles, charges, and term of penalty). The detailed statistics of the datasets are shown in Table 2.

Following previous studies [34, 36, 38], we first filter out the cases with missing or confusing labels (e.g., some cases have no relevant law article/charge label, or the law article label is inconsistent with the charge label) and then filter out the cases with multiple article/charge labels. Then, we filter out the infrequent law articles and charges and only keep those with more than 100 cases. The term of penalty is divided into non-overlapping intervals.

5.2 Baseline Models

In order to verify the effectiveness and evaluate the performance of our model on legal judgment prediction, we select several representative baseline models as follows:

(1) SVM is a typical machine learning model widely used in various classification problems. It is selected as a representative traditional machine learning baseline model. We first represent the words in fact description by looking up the word embedding table pre-trained by Word2vec [19]. Then, we train an SVM [24] model to predict the results of the judgment.

(2) FLA [16] is a simple deep learning model based on the attention mechanism. It assumes that the law articles can provide useful supplementary information for legal judgment prediction, particularly in the Statutory Law system, where judges must determine the suspect's charges based on the law articles. To achieve this, FLA first introduces a retrieval method that selects the k most relevant law articles for each case. Then, the information from the selected law articles is integrated

into the representation of the fact description via the attention mechanism. Finally, the fused vector representation is employed to predict the judgment results.

(3) Topjudge [38] is a neural-network-based model that investigates the relationship between three subtasks in legal judgment prediction. In the Statutory Law system, judges often assess first the law articles that a suspect may have violated, then the charges against the suspect, and finally the term of penalty based on the provisions of the law articles. Therefore, the three subtasks (relevant law article prediction, charge prediction, and term-of-penalty prediction) in legal judgment prediction have a sequential relationship in the real judgment scenario. This model considers the order of three subtasks and builds a topological multi-task learning framework to improve the performance of legal judgment prediction.

(4) MPBFN-WCA [35] is also a neural-network-based model that considers the relationship between three subtasks inside the legal judgment prediction task. Different from the Topjudge [38] model, it assumes that after making a judgment, judicial personnel should analyze and confirm whether the charges and terms of penalty are consistent with the provisions of the law articles. In order to capture the dependency of the predicted results for the three subtasks, this model develops a framework for multi-perspective forward prediction and backward verification. By introducing human behavior patterns, the legal judgment prediction task can be enhanced.

(5) Attribute-Att [10] is a deep neural-network-based model with an attention mechanism. It intends to improve the performance of few-shot and confusing charges. Manually annotated discriminative attributes of charges are introduced to help model the relationship between the fact description and the charges. To achieve this, an attribute-attentive module is designed to enhance the fact representation through labeled attributes. This artificial knowledge has been shown to improve the performance of legal judgment prediction.

(6) LADAN [34] is a neural network framework with a graph distillation operator. This method, similar to Attribute-Att [10], tries to improve the model's capability of discriminating between confusing law articles. It introduces a graph distillation operator for determining the differences between similar law articles. By this means, the confusing law articles of a Law can be better identified and the performance of the legal judgment prediction task can be improved effectively.

(7) CPTP [1] is a charge-based model dedicated to the term-of-penalty prediction (it cannot predict the results of the other two subtasks of the legal judgment prediction task). This model seeks to exclude parts of the fact description that are irrelevant to the term-of-penalty prediction. It uses a deep gating network to refine and aggregate charge-specific information from the fact description.

(8) Neurjudge [36] is a neural network model considering the division of fact description in a legal case. In previous studies, features for predicting the judgment results are typically extracted from the entire statement of the case's fact description. In the actual legal scenario, however, judges use different aspects of the fact description in a case to make a judgment (i.e., some content in the fact description is related to charge prediction, while other content is related to the term-of-penalty prediction). This model utilizes the intermediate results of subtasks to split the fact description into distinct circumstances and exploits them to predict the results of other subtasks. This is one of the state-of-the-art methods for legal judgment prediction.

5.3 Experimental Setup

We use THULAC [23] for word segmentation and Word2vec [19] for pre-training the word embeddings. The embedding size is set as 200. The maximum text length of the fact description is set as 1,500. The kernel sizes of four 1D-CNNs are set as 2, 4, 8, and 12. The output channels are all set as 75. For the weights of LJP's subtasks, we set the weights λ_i as 1. For the three contrastive learning tasks, we set α_1, α_2, and α_3 as 0.5. l and l' are set as 2. When training our model, we use

Table 3. Experimental Results on the CAIL-small Datasets

	Law Articles				Charges				Terms of Penalty			
Method	Acc.	MP	MR	F1	Acc.	MP	MR	F1	Acc.	MP	MR	F1
SVM	86.55	82.72	77.59	79.70	85.09	81.91	78.38	79.32	33.62	30.48	28.34	28.78
FLA	88.53	84.63	80.67	81.88	87.32	84.14	81.34	81.19	35.66	32.79	31.76	31.04
TOPJUDGE	89.40	85.78	83.48	84.30	88.19	85.13	83.31	83.79	36.68	32.96	34.94	32.75
MPBFN-WCA	89.44	86.00	84.34	84.78	88.20	85.37	83.93	84.25	36.77	34.17	33.46	33.57
Attribute-Att	89.10	84.90	83.57	83.96	88.96	85.87	83.43	84.50	36.86	33.55	32.88	32.46
LADAN	90.16	87.11	85.56	86.04	88.71	85.88	84.51	84.64	37.18	34.96	34.88	33.83
CPTP	-	-	-	-	-	-	-	-	38.26	37.15	33.90	34.95
Neurjudge	91.12	88.53	86.61	87.20	89.13	86.63	84.86	85.12	**40.64**	37.80	**36.41**	**36.56**
CL4LJP	**91.42**†	**90.31**†	**87.95**†	**88.06**†	**89.90**†	**88.13**†	**85.79**	**86.80**	38.50	**38.11**	34.16	34.50

"Acc.," "MP," and "MR" are abbreviations for "Accuracy," "Macro Precision," and "Macro Recall," respectively. The best results are in **bold**, and the second best results are underlined. † indicates our CL4LJP achieves significant improvements over all existing methods in paired t-test with p-value < 0.05.

Table 4. Experimental Results on the CAIL-big Datasets

	Law Articles				Charges				Terms of Penalty			
Method	Acc.	MP	MR	F1	Acc.	MP	MR	F1	Acc.	MP	MR	F1
SVM	92.99	82.15	75.61	78.58	92.46	81.38	75.62	77.69	51.70	40.85	37.93	38.09
FLA	94.36	84.71	78.70	80.91	93.83	83.90	77.65	79.93	53.38	42.23	40.33	40.97
TOPJUDGE	95.02	86.48	80.21	82.46	94.61	86.43	79.43	82.01	55.74	45.83	40.40	42.06
MPBFN-WCA	95.07	87.33	80.54	82.91	94.57	86.56	79.57	81.89	55.83	44.29	41.10	41.21
Attribute-Att	95.12	87.87	78.49	81.37	94.69	87.59	78.21	81.48	55.03	45.52	39.41	41.26
LADAN	95.30	87.19	81.41	83.45	94.27	86.07	80.70	82.63	57.99	48.33	43.34	44.13
CPTP	-	-	-	-	-	-	-	-	57.91	48.42	44.50	45.95
Neurjudge	95.68	88.41	83.07	84.97	95.05	87.07	81.97	83.56	**58.05**	48.51	**46.11**	**46.38**
CL4LJP	**96.33**†	**89.91**†	**84.56**†	**86.39**†	**95.87**†	**88.71**†	**83.15**†	**85.42**†	58.00	**48.57**	45.44	45.79

"Acc.," "MP," and "MR" are abbreviations for "Accuracy," "Macro Precision," and "Macro Recall," respectively. The best results are in **bold**, and the second best results are underlined. † indicates our CL4LJP achieves significant improvements over all existing methods in paired t-test with p-value < 0.05.

the AdamW [15] optimizer with a learning rate of 1e-3. We train our model on a single Nvidia Tesla V100 GPU for 20 epochs, and the batch size is set as 100.

We employ **Accuracy (Acc.)**, **Macro Precision (MP)**, **Macro Recall (MR)**, and **Macro F1 (F1)** as evaluation metrics to measure the performance of all models.

5.4 Overall Results

The experimental results of three subtasks (relevant law article prediction, charge prediction, and term-of-penalty prediction) are shown in Table 3 and Table 4.

Compared with the state-of-the-art model Neurjudge, our CL4LJP improves the F1-score of the law article and charge prediction on the CAIL-small dataset by 0.98% and 1.97%, respectively. On the CAIL-big dataset, this improvement is 1.67% and 2.17%. This result clearly demonstrates the superiority of our method. It is interesting to see that our CL4LJP performs worse than Neurjudge with respect to the term-of-penalty prediction on the CAIL-small dataset. Indeed, Neurjudge simulates a practical judicial scenario in which the facts are meticulously dissected and modeled. When data are limited, this prior human knowledge is incredibly useful. Our CL4LJP's smaller gap on the CAIL-big dataset suggests that this kind of human knowledge can be implicitly learned with sufficient training data.

By comparing CL4LJP with other baseline models for the legal judgment prediction task, we can make the following observations:

Table 5. Ablation Study of Encoders on the CAIL-small Dataset

	Law Articles		Charges		Terms of Penalty	
Method	Acc	F1	Acc	F1	Acc	F1
CL4LJP (Full)	**91.42**	**88.06**	**89.90**	**86.80**	**38.50**	**34.50**
w/o General	90.54	85.82	88.75	84.45	37.03	32.10
w/o Contrastive	88.31	81.13	87.12	80.55	35.41	30.94

We remove the general encoder and the contrastive encoder in our framework respectively to investigate their influence. The best results are in bold.

(1) SVM is inferior to all neural approaches. This reflects neural networks' better capability of extracting features and modeling the relationship between facts and labels.

(2) The performance of FLA is poor because it directly integrates information from top-k law articles into the fact representation of a legal case. The lack of differentiation between confusing law articles may introduce noise to the model.

(3) Both Topjudge and MPBFN-WCA leverage the topological order of the three subtasks in a real judgment situation to improve the fact representation. The better performance achieved by CL4LJP indicates that using rule-based contrastive sampling to learn fine-grained differences between articles/charges can improve the representation capability of the model.

(4) The performance of our model is better than that of Attribute-Att and Ladan, showing that our proposed rule-based contrastive learning tasks can help the model better extract the discriminative fact description features for the legal judgment prediction task.

(5) We notice that the F1 scores of all methods in the law article prediction and charge prediction tasks on the CAIL-big dataset are lower than those on the CAIL-small dataset, but the accuracy is higher. We attribute this to the fact that the categories of law articles and charges in the CAIL-big dataset are highly unbalanced.

5.5 Ablation Studies

To validate the effectiveness of the two encoders in the framework and the three contrastive learning tasks, we conduct an ablation study by removing them from the full model respectively. Note that multi-task learning has been reported to be beneficial for the legal judgment prediction task [38], so we do not investigate the influence of each subtask in this study.

5.5.1 Experiments with Encoders. We first investigate the impact of two encoders. We respectively remove the general encoder and the contrastive encoder from the full model and refer to them as **w/o General** and **w/o Contrastive**. The results are shown in Table 5.

In general, removing either encoder leads to performance degradation. This indicates that both encoders are effective in our method. Specifically, the performance drops more when removing the contrastive encoder. This result clearly reveals that our proposed contrastive learning tasks are effective for learning better representations for legal judgment prediction tasks.

5.5.2 Experiments with Contrastive Tasks. We also explore the influence of the proposed three contrastive learning tasks. The article-level, charge-level, and label-aware contrastive learning subtasks are eliminated, and the corresponding models are denoted as w/o article, w/o charge, and w/o label, respectively.

The results are shown in Figure 7. We can see:

(1) The performance on three subtasks decreases when any task is removed. This indicates the effectiveness of applying contrastive learning for legal judgment prediction.

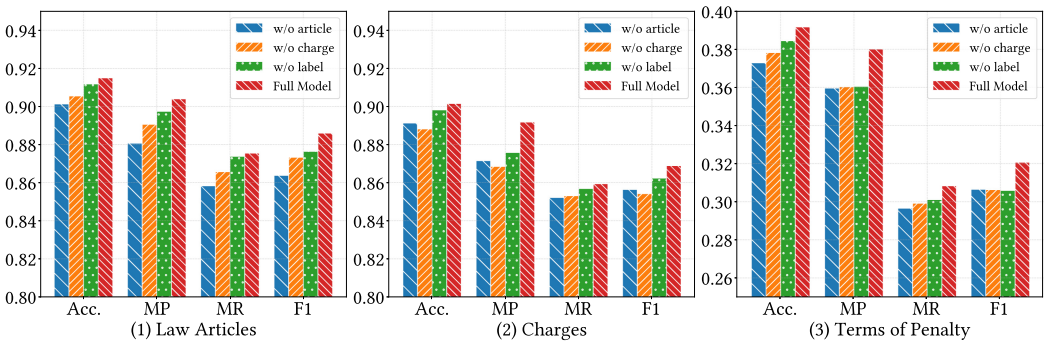

Fig. 7. Ablation study of three contrastive learning subtasks of our framework on the CAIL-small dataset. To intuitively show the influence of the proposed contrastive learning tasks, we remove the article-level, charge-level, and label-aware contrastive learning subtasks. They are denoted as w/o article, w/o charge, and w/o label, respectively.

(2) Among the three variants, the performance drops the least when the label-aware contrastive learning task is not used. Indeed, this is a general learning task for all classification tasks. Compared with the other two rule-based contrastive learning tasks, it does not take into account the characteristics of the legal judgment prediction task, so it brings less improvement.

(3) Removing the article-level or charge-level contrastive learning task has a considerable impact on the performance of the corresponding subtask. This is consistent with our expectations. These two contrastive learning tasks enhance the encoder's capability by distinguishing similar law articles and charges, so they are the most beneficial for the associated subtasks.

(4) Overall, the article-level contrastive learning task contributes most to legal judgment prediction (the sum of F1 scores on three subtasks decreases 2.4% when it is removed). The potential reason is that the articles provide the basis for a judgment and also contain charge information, so they play the most important role in the legal judgment prediction task.

5.6 Experiments with Alternative Encoders

Pre-trained language models have achieved great performance on various natural language processing tasks [22, 42, 43]. As a result, we employ the pre-trained language model, BERT [4], as the encoder. It is worth noting that, due to the computing complexity of BERT, we only use the representation computed by the contrastive encoder for legal judgment prediction.

Specifically, for the fact description f, following the design of BERT, we add special tokens [CLS] and [SEP] at the head and tail of the text sequence as follows: $f' = [CLS]\ f\ [SEP]$. Then, we feed it into BERT and use the representation of the [CLS] token as the representation of the fact description:

$$\mathbf{Z}_f^B = \text{BERT}(f')_{[CLS]} \in \mathbb{R}^{768}. \tag{24}$$

Similarly, law articles can be represented as $\mathbf{Z}_a^B$ and the fact description of sampled positive/negative cases can be represented as $\mathbf{Z}_c^B$:

$$\mathbf{Z}_a^B = \text{BERT}(a')_{[CLS]}, \tag{25}$$

$$\mathbf{Z}_c^B = \text{BERT}(c')_{[CLS]}, \tag{26}$$

where a' is a law article with special tokens, and c' is the fact description of a sampled case with special tokens.

Table 6. Performance of BERT-based Neural Network Methods on the CAIL-small Dataset

	Law Articles		Charges		Terms of Penalty	
Method	Acc	F1	Acc	F1	Acc	F1
BERT	92.38	88.59	91.39	87.92	40.83	34.25
BERT-Crime	92.35	88.72	91.45	88.38	41.00	34.41
Neurjudge+BERT	93.14	90.64	92.30	90.10	**41.26**	**36.70**
CL4LJP	91.42	88.06	89.90	86.80	38.50	34.50
CL4LJP+BERT	**93.50**	**91.24**	**92.91**	**90.84**	41.03	35.71

We compare the performance of our BERT-based model with those of the BERT-based baselines. The best results are in bold.

Table 7. Time Consumed per Epoch of Our Framework with Different Encoders

Model	Training Time per Epoch	Testing Time per Epoch
BERT-based encoder	38,404s	3,800s
CNN-based encoder	445s	50s

We count the time consumed by our model on the training and testing sets of the CAIL-small dataset.

To explore its effectiveness, we compare some BERT-based methods with our CL4LJP using BERT encoders. All of these models are implemented based on the publicly available pre-trained Chinese BERT [3] model. Due to the limited input length of BERT, we set the maximum text length of a fact description in a legal case to 512. Fact descriptions with more than 512 tokens are truncated, while those with fewer than 512 tokens are padded. The Bert-based baselines are as follows:

• **BERT** can be directly fine-tuned on the CAIL datasets for the legal judgment prediction task. It takes the fact description of a legal case as input and uses the vector representation of the "[CLS]" token to predict the relevant law article, charge, and term of penalty.

• **BERT-Crime** [39] further pre-trains BERT [39] on large-scale Chinese legal datasets. The fine-tuned process is the same as BERT.

• **Neurjudge+BERT** replaces the RNN encoder of Neurjudge by BERT. Other parts of the model Neurjudge+BERT are the same as the RNN-based Neurjudge model.

The experimental results are provided in Table 6. First, we can only conduct experiments on the CAIL-small dataset because BERT contains tremendous parameters and takes a long time for training. More specifically, on the CAIL-small dataset, CL4LJP+BERT takes 85.3× more time for training, as shown in Table 7. The testing time is also increased dramatically. Fortunately, CL4LJP+BERT can achieve better performance. This suggests that applying BERT is a good strategy if sufficient computation power is available. Besides, CL4LJP+BERT can still outperform Neurjudge+BERT, which confirms once more the benefit of training with our contrastive learning framework. Finally, this experiment further validates the adaptability of our framework—other advanced neural network encoders are compatible with our method.

5.7 Performance on Tail Classes

It has been reported that the class (category) distribution of law articles and charge labels is extremely unbalanced on CAIL datasets [33, 36]. As shown in Figure 8, the case number follows a long-tail distribution from the perspective of the law article classes in the CAIL-small dataset. The head classes contain more than 3,000 cases, while the tail classes contain fewer than 250 cases. The unbalanced number of cases in different classes (articles) is very challenging for legal judgment prediction models. As a result, neural models are easily affected and biased to predict head law

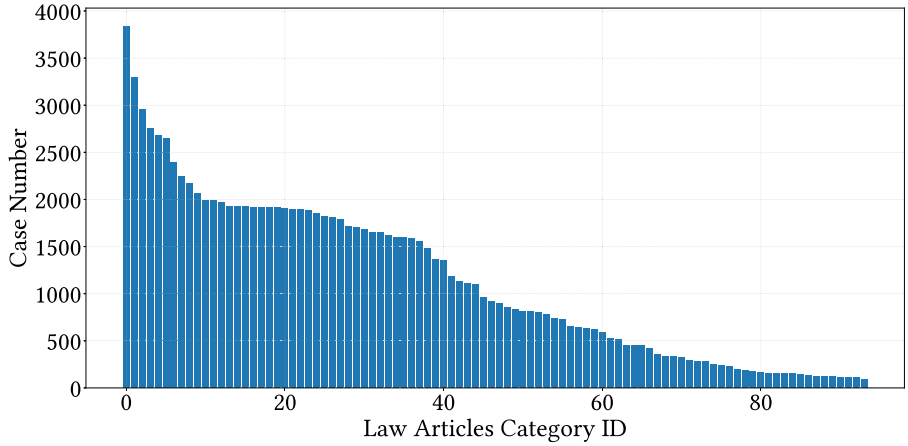

Fig. 8. The case number of each law article category in the CAIL-small dataset. We sort the law article categories descendingly according to the number of cases. The number of cases in head law article categories is more than 3,000, while that of cases in tail law article categories is fewer than 250.

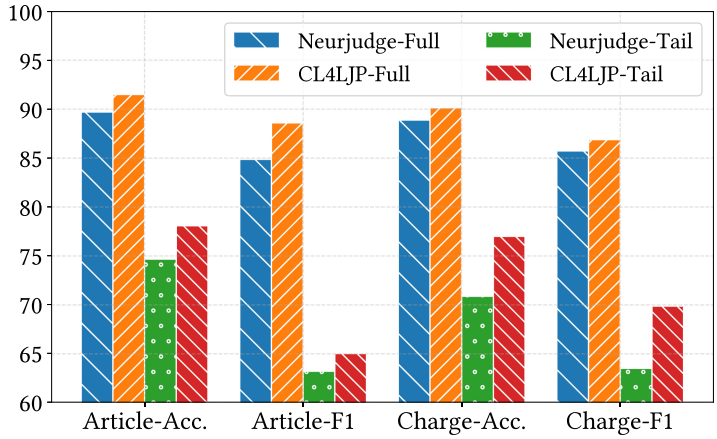

Fig. 9. Performance on the tail classes of the CAIL-small dataset. We compare the performance of our model on the tail classes with Neurjudge. As a comparison, we also provide the performance on the full classes.

articles or charges. However, the prediction of tail classes is an essential ability of a legal judgment prediction system. To investigate our CL4LJP's performance under this circumstance, we test it on the cases of tail law articles and charges, which contain fewer than 200 cases in the CAIL-small dataset. Note that, as introduced in Section 5.1, law articles and charges having fewer than 100 cases are removed to make the results consistent with existing works. Since the distribution of class labels in the term-of-penalty prediction task is much more balanced, i.e., all classes have more than 200 cases, we do not test on this subtask.

We compare the results with that of Neurjudge because it yields the best results in the baselines. The results are shown in Figure 9. We can find:

(1) Compared with the results on full classes, both NeurJudge and CL4LJP perform worse on the tail classes of the CAIL-small dataset. This confirms our hypothesis that predictions on the tail classes are very challenging in the legal judgment prediction task.

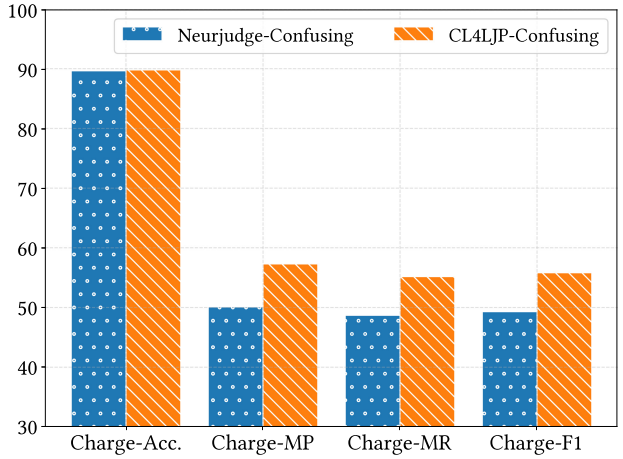

Fig. 10. Performance on cases of confusing charges associated with Article #114 and Article #115 in the CAIL-small dataset. These two articles stipulate some similar charges that are easily confused. We also show the performance of the baseline model Neurjudge as a comparison.

(2) When transitioning from full classes to tail classes, the improvement of CL4LJP over NeurJudge becomes larger. This indicates that CL4LJP handles tail classes more effectively. The advantage may stem from our proposed contrastive learning tasks. By distinguishing confusing cases with similar articles or charges, CL4LJP can learn to capture finer-grained clues for legal judgment prediction.

5.8 Confusing Case Study

To intuitively show the effect of our model in distinguishing confusing cases, we select the cases associated with *Article #114* and *Article #115* from the testing set of CAIL-small as a new tiny testing set and evaluate our model's performance in charge prediction. The applicable law article prediction is not performed because only two law articles are involved. We also report the results of the baseline model Neurjudge as a comparison. Both methods are trained on the CAIL-small dataset.

The results are shown in Figure 10. First, the high accuracy but low precision reflects the confusing nature of the charges (see Figure 1 for the charge text). Second, Neurjudge obtains accuracy comparable to our CL4LJP but significantly lower precision/recall/F1 values. This demonstrates that our proposed contrastive learning of similar articles/charges can effectively improve the model's capability of discriminating ambiguous cases, which is beneficial for legal judgment prediction.

6 ETHICAL DISCUSSION

Given that the outcome of a legal decision is contingent on the litigant's pragmatism and that artificial intelligence for legal judgment prediction in the legal area is a new but sensitive technology, it is worthwhile to investigate certain ethical considerations.

Despite the fact that our method CL4LJP has achieved good performance on real-world datasets, it is worth noting that all judgment documents used are generated in the final stage of the judicial process. They are not involved in other stages, as described in the paper [17]. As a result, our method or the system using similar methods is not designed to replace the rational judgment

made by the judicial personnel, nor can it replace the role of judicial personnel throughout the judicial process.

Our method aims at offering help to human judges by providing relevant law articles, charges, or terms of penalty of cases. Intelligent legal judgment prediction is still in its exploratory stage, and it is possible for mistakes to occur. The accuracy of tail categories is relatively low. Moreover, these analyses do not involve complex debates in the justice process. Therefore, judges must check the judgment results from the algorithm [31].

7 CONCLUSION AND FUTURE WORK

In this article, we first consider that a judge usually compares and analyzes the specific provisions of similar law articles and legal cases with similar charges to distinguish confusing cases before making a judgment in the actual judging process.

We designed a supervised contrastive learning framework to simulate this scenario. Specifically, we design three contrastive learning subtasks, i.e., article-level contrastive learning task, charge-level contrastive learning task, and label-level contrastive learning task, besides the legal judgment prediction multi-task framework. These contrastive learning subtasks are able to improve the representation ability of fact description and that of distinguishing similar law articles and charges. Experimental results on two real datasets show our model is effective and is especially robust on the tail classes.

In the future, we will consider exploring the following directions for the legal judgment prediction task on the basis of our existing work:

(1) Most of the existing legal judgment prediction models only consider the situation of a single suspect and a single charge in a legal case. But in reality, the situation of multiple suspects and multiple charge labels in a legal case is also common. For example, in the cases of the Crime of Illegally Feeling Trees, there are usually several suspects, but each suspect's behavior is usually different. The judge needs to determine the corresponding charges and term of penalty of each suspect according to the specific behavior. So in the future, we will further study the problem of multiple suspects and multiple charge labels in the legal judgment prediction task.

(2) The legal judgment prediction task is often defined as a text classification problem in previous methods. Most of the existing methods can only provide the prediction results without any explanation. However, in practice, the legal judgment results should be explained with clear reasons and clues. For example, in the scenario where a suspect set fire and caused the loss of public property, it may be that the suspect smoked and caused the fire accidentally, or that the suspect deliberately retaliated against society and deliberately ignited and caused serious losses. The charges involved in these two possible situations are different, and the judge needs to specifically and clearly point out the corresponding criminal acts and the situations in the corresponding law articles according to the fact description of the case. Therefore, the model's explainability for the predicted results is another important future direction.

(3) Most of the existing models are only based on the fact description of the current case and the law articles to determine the relevant law articles, charges, and term of penalty. But in reality, lawyers or judicial personnel usually look for historical cases to help identify the charges of a suspect, especially in the Case Law system where historical cases play a similar role as the law articles in the Statutory Law system. How to effectively use historical cases is also a direction worthy of future research.

(4) Furthermore, each precedent case in the process of judicial practice is a specific use of law articles; thus, a case may expand or narrow the scope of the law's provisions (also known as expanded cases or contracted cases, respectively). However, existing literature does not consider distinguishing these two different kinds of cases. This makes it hard to confirm the application

boundary of law articles. In the future, we will pay more attention to identifying the use of law articles in contracted cases.

(5) Conviction and sentencing are both important for legal judgment prediction in reality. The former process corresponds to the relevant law article prediction and the charge prediction subtasks, while the latter process is related to the term-of-penalty prediction. Several things should be taken into account when determining the term of penalty, such as the suspect's age, whether or not she is pregnant, how much property was stolen, and other specific sentencing information. However, most of the existing research studies usually learn conviction and sentencing simultaneously, and they rarely consider the specific information involved in the sentencing process. In the future, we plan to introduce such information about the sentencing process into the model.

REFERENCES

[1] Huajie Chen, Deng Cai, Wei Dai, Zehui Dai, and Yadong Ding. 2019. Charge-based prison term prediction with deep gating network. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*. Association for Computational Linguistics, 6361–6366. https://doi.org/10.18653/v1/D19-1667

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning (ICML'20), Virtual Event (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 1597–1607. http://proceedings.mlr.press/v119/chen20j.html.

[3] Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2021. Pre-training with whole word masking for Chinese BERT. *IEEE ACM Trans. Audio Speech Lang. Process.* 29 (2021), 3504–3514. https://doi.org/10.1109/TASLP.2021.3124365

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT'19), Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 4171–4186. https://doi.org/10.18653/v1/n19-1423

[5] Qian Dong and Shuzi Niu. 2021. Legal judgment prediction via relational learning. In *The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'21), Virtual Event*. ACM, 983–992. https://doi.org/10.1145/3404835.3462931

[6] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. *CoRR* abs/2104.08821 (2021). arXiv:2104.08821 https://arxiv.org/abs/2104.08821.

[7] Anne von der Lieth Gardner. 1984. *An Artificial Intelligence Approach to Legal Reasoning*. Ph. D. Dissertation. Stanford University.

[8] Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. IEEE Computer Society, 1735–1742. https://doi.org/10.1109/CVPR.2006.100

[9] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20)*. Computer Vision Foundation/IEEE, 9726–9735. https://doi.org/10.1109/CVPR42600.2020.00975

[10] Zikun Hu, Xiang Li, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. 2018. Few-shot charge prediction with discriminative legal attributes. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING'18)*. Association for Computational Linguistics, 487–498. https://aclanthology.org/C18-1041/.

[11] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020 (NeurIPS'20), virtual*. https://proceedings.neurips.cc/paper/2020/hash/d89a66c7c80a29b1bdbab0f2a1a94af8-Abstract.html.

[12] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP'14), A meeting of SIGDAT, a Special Interest Group of the ACL*, Alessandro Moschitti, Bo Pang, and Walter Daelemans (Eds.). ACL, 1746–1751. https://doi.org/10.3115/v1/d14-1181

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

[14] Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, Blai Bonet and Sven Koenig (Eds.). AAAI Press, 2267–2273. http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9745.

[15] Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations (ICLR'19)*. OpenReview.net. https://openreview.net/forum?id=Bkg6RiCqY7.

[16] Bingfeng Luo, Yansong Feng, Jianbo Xu, Xiang Zhang, and Dongyan Zhao. 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'17)*. Association for Computational Linguistics, 2727–2736. https://doi.org/10.18653/v1/d17-1289

[17] Luyao Ma, Yating Zhang, Tianyi Wang, Xiaozhong Liu, Wei Ye, Changlong Sun, and Shikun Zhang. 2021. Legal judgment prediction with multi-stage case representation learning in the real court setting. In *The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'21), Virtual Event*. ACM, 993–1002. https://doi.org/10.1145/3404835.3462945

[18] Yu Meng, Chenyan Xiong, Payal Bajaj, Saurabh Tiwary, Paul Bennett, Jiawei Han, and Xia Song. 2021. COCO-LM: Correcting and contrasting text sequences for language model pretraining. *CoRR* abs/2102.08473 (2021). arXiv:2102.08473 https://arxiv.org/abs/2102.08473.

[19] Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013*. 3111–3119. https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html.

[20] Stuart S. Nagel. 1963. Applying correlation analysis to case prediction. *Tex. L. Rev.* 42 (1963), 1006.

[21] Jeffrey A. Segal. 1984. Predicting supreme court cases probabilistically: The search and seizure cases, 1962-1981. *American Political Science Review* 78, 4 (1984), 891–900.

[22] Zhan Su, Zhicheng Dou, Yutao Zhu, Xubo Qin, and Ji-Rong Wen. 2021. Modeling intent graph for search result diversification. In *The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'21), Virtual Event*, Fernando Diaz, Chirag Shah, Torsten Suel, Pablo Castells, Rosie Jones, and Tetsuya Sakai (Eds.). ACM, 736–746. https://doi.org/10.1145/3404835.3462872

[23] Maosong Sun, Xinxiong Chen, Kaixu Zhang, Zhipeng Guo, and Zhiyuan Liu. 2016. Thulac: An efficient lexical analyzer for Chinese.

[24] Johan A. K. Suykens and Joos Vandewalle. 1999. Least squares support vector machine classifiers. *Neural Process. Lett.* 9, 3 (1999), 293–300. https://doi.org/10.1023/A:1018628609742

[25] S. Sidney Ulmer. 1963. Quantitative analysis of judicial processes: Some practical and theoretical applications. *Law and Contemporary Problems* 28, 1 (1963), 164–184.

[26] Josef Valvoda, Ryan Cotterell, and Simone Teufel. 2022. On the role of negative precedent in legal outcome prediction. *CoRR* abs/2208.08225 (2022). https://doi.org/10.48550/arXiv.2208.08225 arXiv:2208.08225

[27] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *CoRR* abs/1807.03748 (2018). arXiv:1807.03748 http://arxiv.org/abs/1807.03748.

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*. 5998–6008. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

[29] Pengfei Wang, Yu Fan, Shuzi Niu, Ze Yang, Yongfeng Zhang, and Jiafeng Guo. 2019. Hierarchical matching network for crime classification. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*. ACM, 325–334. https://doi.org/10.1145/3331184.3331223

[30] Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning (ICML'20), Virtual Event (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 9929–9939. http://proceedings.mlr.press/v119/wang20k.html.

[31] Yiquan Wu, Kun Kuang, Yating Zhang, Xiaozhong Liu, Changlong Sun, Jun Xiao, Yueting Zhuang, Luo Si, and Fei Wu. 2020. De-biased court's view generation with causality. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP'20), Online*, Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, 763–780. https://doi.org/10.18653/v1/2020.emnlp-main.56

[32] Zhuofeng Wu, Sinong Wang, Jiatao Gu, Madian Khabsa, Fei Sun, and Hao Ma. 2020. CLEAR: Contrastive learning for sentence representation. *CoRR* abs/2012.15466 (2020). arXiv:2012.15466 https://arxiv.org/abs/2012.15466.

[33] Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, and Jianfeng Xu. 2018. CAIL2018: A large-scale legal dataset for judgment prediction. *CoRR* abs/1807.02478 (2018). arXiv:1807.02478 http://arxiv.org/abs/1807.02478.

[34] Nuo Xu, Pinghui Wang, Long Chen, Li Pan, Xiaoyan Wang, and Junzhou Zhao. 2020. Distinguish confusing law articles for legal judgment prediction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL'20), Online*. Association for Computational Linguistics, 3086–3095. https://doi.org/10.18653/v1/2020.acl-main.280

[35] Wenmian Yang, Weijia Jia, Xiaojie Zhou, and Yutao Luo. 2019. Legal judgment prediction via multi-perspective bi-feedback network. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI'19)*. ijcai.org, 4085–4091. https://doi.org/10.24963/ijcai.2019/567

[36] Linan Yue, Qi Liu, Binbin Jin, Han Wu, Kai Zhang, Yanqing An, Mingyue Cheng, Biao Yin, and Dayong Wu. 2021. NeurJudge: A circumstance-aware neural framework for legal judgment prediction. In *The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'21), Virtual Event*. ACM, 973–982. https://doi.org/10.1145/3404835.3462826

[37] Han Zhang, Zhicheng Dou, Yutao Zhu, and Jirong Wen. 2021. Few-shot charge prediction with multi-grained features and mutual information. In *Chinese Computational Linguistics - 20th China National Conference (CCL'21), Proceedings (Lecture Notes in Computer Science, Vol. 12869)*, Sheng Li, Maosong Sun, Yang Liu, Hua Wu, Kang Liu, Wanxiang Che, Shizhu He, and Gaoqi Rao (Eds.). Springer, 387–403. https://doi.org/10.1007/978-3-030-84186-7_26

[38] Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Chaojun Xiao, Zhiyuan Liu, and Maosong Sun. 2018. Legal judgment prediction via topological learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 3540–3549. https://doi.org/10.18653/v1/d18-1390

[39] Haoxi Zhong, Zhengyan Zhang, Zhiyuan Liu, and Maosong Sun. 2019. *Open Chinese Language Pre-trained Model Zoo*. Technical Report. https://github.com/thunlp/openclap.

[40] Yujia Zhou, Zhicheng Dou, Yutao Zhu, and Ji-Rong Wen. 2021. PSSL: Self-supervised learning for personalized search with contrastive sampling. In *The 30th ACM International Conference on Information and Knowledge Management (CIKM'21), Virtual Event*. ACM, 2749–2758. https://doi.org/10.1145/3459637.3482379

[41] Yutao Zhu, Jian-Yun Nie, Zhicheng Dou, Zhengyi Ma, Xinyu Zhang, Pan Du, Xiaochen Zuo, and Hao Jiang. 2021. Contrastive learning of user behavior sequence for context-aware document ranking. In *The 30th ACM International Conference on Information and Knowledge Management (CIKM'21), Virtual Event*. ACM, 2780–2791. https://doi.org/10.1145/3459637.3482243

[42] Yutao Zhu, Jian-Yun Nie, Zhicheng Dou, Zhengyi Ma, Xinyu Zhang, Pan Du, Xiaochen Zuo, and Hao Jiang. 2021. Contrastive learning of user behavior sequence for context-aware document ranking. In *The 30th ACM International Conference on Information and Knowledge Management (CIKM'21), Virtual Event*, Gianluca Demartini, Guido Zuccon, J. Shane Culpepper, Zi Huang, and Hanghang Tong (Eds.). ACM, 2780–2791. https://doi.org/10.1145/3459637.3482243

[43] Yutao Zhu, Kun Zhou, Jian-Yun Nie, Shengchao Liu, and Zhicheng Dou. 2021. Neural sentence ordering based on constraint graphs. In *35th AAAI Conference on Artificial Intelligence (AAAI'21), 33rd Conference on Innovative Applications of Artificial Intelligence (IAAI'21), 11th Symposium on Educational Advances in Artificial Intelligence (EAAI'21), Virtual Event*. AAAI Press, 14656–14664. https://ojs.aaai.org/index.php/AAAI/article/view/17722.

Received 2 August 2022; revised 4 November 2022; accepted 5 January 2023