# CPEE: Civil Case Judgment Prediction centering on the Trial Mode of Essential Elements

Lili Zhao
Anhui Province Key Laboratory of Big Data Analysis and Application, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
liliz@mail.ustc.edu.cn

Linan Yue
Anhui Province Key Laboratory of Big Data Analysis and Application, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
lnyue@mail.ustc.edu.cn

Yanqing An
Anhui Province Key Laboratory of Big Data Analysis and Application, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
anyq@mail.ustc.edu.cn

Yuren Zhang
Anhui Province Key Laboratory of Big Data Analysis and Application, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
yr160698@mail.ustc.edu.cn

Jun Yu
IFLYTEK
Hefei, China
junyu@iflytek.com

Qi Liu*
Anhui Province Key Laboratory of Big Data Analysis and Application, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
qiliuql@ustc.edu.cn

Enhong Chen
Anhui Province Key Laboratory of Big Data Analysis and Application, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
cheneh@ustc.edu.cn

## ABSTRACT

Civil Case Judgment Prediction (CCJP) is a fundamental task in the legal intelligence of the civil law system, which aims to automatically predict the judgment results on each plea of the plaintiff. Existing studies mainly focus on making judgment predictions only on a certain civil cause (e.g., the divorce dispute) by utilizing the fact descriptions and pleas of the plaintiff, which still suffer from the various causes and complicated legal essential elements in the real court. Thus, in this paper, we formalize CCJP as a multi-task learning problem and propose a CCJP method centering on the trial mode of essential elements, CPEE, which explores the practical judicial process and analyzes comprehensive legal essential elements to make judgment predictions. Specifically, we first construct three tasks (i.e., the predictions on the *civil causes*, *law articles*, and the *final judgment on each plea*) necessary for CCJP, that follow the judgment process and exploit the results of intermediate subtasks to make judgment predictions. Then we design a logic-enhanced network to predict the results of three tasks and conduct a comprehensive study of civil cases. Finally, owing to the interlinked and dependent relationships among each task, we adopt the cause prediction result to help predict law articles and incorporate them into final judgment prediction through a gate mechanism. Furthermore, since the existing dataset fails to provide sufficient case information, we construct a real-world CCJP dataset that contains various causes and comprehensive legal elements. Extensive experimental results on the dataset validate the effectiveness of our method.

*Corresponding Author.


*CIKM '22, October 17–21, 2022, Atlanta, GA, USA*

ACM ISBN 978-1-4503-9236-5/22/10…$15.00
https://doi.org/10.1145/3511808.3557273

## CCS CONCEPTS

• **Applied computing → Law**.

## KEYWORDS

civil case judgment prediction, multi-task learning, trial mode

**ACM Reference Format:**
Lili Zhao, Linan Yue, Yanqing An, Yuren Zhang, Jun Yu, Qi Liu, and Enhong Chen. 2022. CPEE: Civil Case Judgment Prediction centering on the Trial Mode of Essential Elements. In *Proceedings of the 31st ACM International*

*Conference on Information and Knowledge Management (CIKM '22), October 17–21, 2022, Atlanta, GA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3511808.3557273

# 1 INTRODUCTION

Based on the case narrative, the goal of **C**ivil **C**ase **J**udgment **P**rediction (**CCJP**) is mainly to make judgment predictions about whether a certain plea (e.g., the plaintiff demands a refund of wages by the defendant) in a given case would be supported, partially supported, or rejected. In countries with civil law systems[1], a human judge focuses on case controversies and ascertains the case facts through the statements of the plaintiff and the defendant during the trial. The judge makes final judgments based on the relevant law articles and all-sided case information, which is a complex set of procedures full of the logic of judges [2, 41].

In the literature, CCJP has been formalized as a text classification task for predicting the judgments on the pleas of the plaintiff, and massive efforts have been made in this area [16, 19, 38, 44]. Among them, Long et al. [16] proposed a legal reading comprehension method for divorce cases. Ma et al. [19] showed a framework based on real court debate data of the private lending dispute. Despite these efforts in designing progressive frameworks for judgment predictions, existing studies fail to discover the universality and integrity of CCJP. In order to break through the bottleneck of existing methods, we are confronted with two major challenges:

**Various civil causes in the law system:** Most existing studies concentrate on a certain type of civil cases like the private lending dispute [19] or the divorce one [16]. However, in real court scenarios, judges need to handle a wide variety of civil cases (e.g., there are about 928 prescribed types of civil cases in China[2]). If we apply the current models of CCJP (one model for one cause) to the real court, it means that every time a civil case needs to be handled, the judges need to find a particular one from various models, which is cumbersome and unrealistic. Therefore, it is challenging to design a model that could handle various causes. Furthermore, the relationships among the civil causes are quite complex. For example, in China, the causes are designed into a four-level hierarchical structure [18], realizing the evolution of the causes from top-level general to bottom-level specific as shown in Figure 2 (a). Each case has its particular cause, the corresponding applicable laws and case circumstances of different causes often vary widely.

**Complicated legal elements in civil cases:** Recent studies focus on the case facts to make judgment on each plea of the plaintiff [16, 32]. However, they ignore the legal essential elements including litigant statements and law articles, which are important for CCJP. On the one hand, it is difficult to determine case controversies of both parties directly based on only facts and ignoring the statements between the plaintiff and the defendant in real court scenarios [2]. Therefore, it is significant for CCJP to capture case controversies from litigant statements. On the other hand, civil law articles work as references in civil cases and can be applied in all cases with different judgments [16]. It is hard to follow one-to-one matching between law articles and cases, and predict directly among numerous articles [5].

[1]Civil law courts generally decide cases using codal provisions on a case-by-case basis, without reference to other (even superior) judicial decisions in the civil law system.
[2]https://www.court.gov.cn/shenpan-xiangqing-282031.html

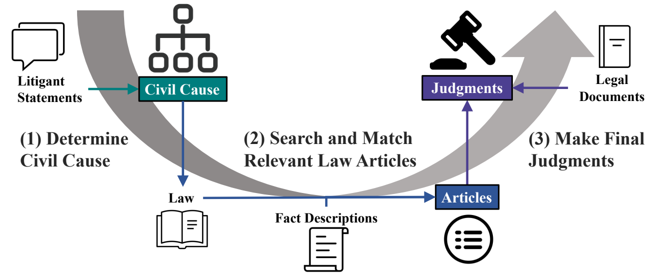


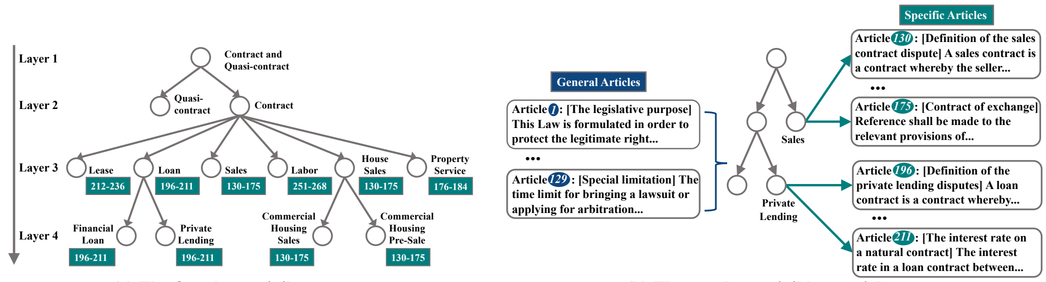

**Figure 1: The logic of judges in the actual adjudication (Three line segments in different colors are considered to be the judgment process of three tasks).**

To tackle the above two challenges, we propose an approach centering on the trial mode of essential elements for CCJP (i.e. **CPEE**) by simulating the real judicial process. As shown in Figure 1, there are rigorous relationships among judicial processes in actual adjudication, and the logic for judgment is divided into three steps. First, the judge determines the cause based on litigant statements. Next, the relevant law articles can be found and applied based on the cause, where the cause could help narrow the scope of a legal search as shown in Figure 2. Finally, the judge makes judgments on pleas after a series of rigorous deliberations. Inspired by the above judgment process, we construct three tasks for CCJP (i.e., *Civil Cause Prediction (CCP)*, *Civil Law Articles Prediction (CLAP)*, and *Final Judgment Prediction(FJP)*) to simulate the logic of judges.

Across the above three tasks, we design a logic-enhanced network to make complete judgment predictions. On the one hand, we design task-specific decoders in the network for three tasks. Specifically, in the *CCP*, we set a local aware global classifier to handle various civil causes by introducing the hierarchical structure of cases (i.e., the structure is shown in Figure 2 (a)). In the *CLAP*, we explore the hierarchy in civil law articles, where the articles are structured in two layers including the general and specific articles [5] as shown in Figure 2 (b). Based on this two-layer structure, we design two modules with label embedding methods to reduce the complexity of direct predictions on total articles. Furthermore, in order to focus more on case controversies between plaintiff and defendant, we adopt the co-attention mechanism and mutual information regularizer to capture more relationships and pivotal information of a case in *FJP*.

On the other hand, there is a logical advancement among tasks. The steps in these tasks are interlinked and dependent as illustrated in Figure 1. Since cases on different causes are governed by the same general articles but different specific articles as shown in Figure 2 (b), we adopt the predicted cause of *CCP* to fix the scope of specific articles, which could reduce the total amount and complexity of predictions of *CLAP*. Subsequently, the predicted general and specific articles are integrated to help *FJP* through the gate mechanism to provide legal reference and ensure standardization.

Although some datasets are currently available for civil cases [19, 32], they ignore various causes and comprehensive legal elements. Therefore, we construct a real-world dataset that contains 158,625 civil cases of sufficient case information (the detailed description of our dataset is shown in Section 5.1), and extensive experimental results on the dataset show the effectiveness of our method CPEE.




(a) The four-layer civil causes. (b) The two-layer civil law articles.

**Figure 2: (a) The four-layer hierarchical structure of civil causes. (The numeric range under the cause name represents the scope of specific articles applicable to each cause). (b) The laws are divided into two layers: general articles (i.e., articles 1 through 129 are applied to all causes) and specific articles (i.e., different causes apply different scopes of specific articles).**

## 2 RELATED WORK

At present, there was a great deal of research on the prediction of criminal cases which was called **L**egal **J**udgment **P**rediction(**LJP**). From single-task learning [17] to multi-task learning [37, 39], extensive studies have succeeded in predicting charges [30], articles [34], and terms of penalty [35]. They have greatly contributed to improving the accuracy, logicality, and interpretability of judgment predictions on criminal cases [36, 40]. Extensive work has succeeded for LJP, but few studies focused on modeling complex civil cases. In this section, we review some excellent studies on civil case judgment prediction as well as related technologies about hierarchical text classification and label embedding.

### 2.1 Civil Case Judgment Prediction

In the civil case judgment prediction, many researchers have gradually explored the structure and nature of civil cases. Long et al. [16] proposed a reading comprehension network to handle multiple and complex inputs (i.e., fact description, pleas of the plaintiff, law articles) of divorce disputes. Zhou et al. [44] modeled the judgment prediction of e-commerce disputes from buyer-seller interactions and transaction data online. Gan et al. [8] represented declarative legal knowledge as a set of first-order logic rules and integrated them into a network in order to make the model more interpretable in private lending disputes. Ma et al. [19] took a different approach to CCJP by simulating the interaction between claims, facts, and arguments in court debate data of private lending disputes. Since the above methods were established on the basis of studying one kind of dispute, Zhao et al. [38] proposed to solve the problem of different civil causes by introducing an external knowledge base.

Nevertheless, these efforts neglected to handle various causes and legal essential elements. They all performed independent tasks to proceed, which lacked the logic and integrity of the real judicial procedure. We try to make up for that in the paper.

### 2.2 Hierarchical Text Classification

In **H**ierarchical **T**ext **C**lassification (HTC), there were flat, local, and global methods [23]. The flat methods usually treated HTC as a simple multi-class classification problem, ignored the hierarchical structure, and only predicted classes at the leaf nodes [7]. The local approaches could be divided into local per node (i.e., train a classifier on each class), local per parent (i.e., train a multi-class classifier on all child nodes of a parent node), and local per level (i.e., train a multi-class classifier for each level) [3, 12, 15]. The global ones built only one classifier to discriminate all categories in a hierarchy [14, 27]. Clearly, the above works mainly focused on either the local regions or the overall structure of the category hierarchy, so some studies have combined the advantages of local and global approaches for learning the dependencies among the different levels in the hierarchy [31]. In addition, some researchers tried to adopt label embeddings in the HTC [10]. Motivated by these efforts, we integrated predictions for each level of the hierarchy with those for the overall by combining local and global approaches.

**Table 1: Main mathematical notations.**

| Symbol | Description |
|---|---|
| $S^c = \left\{w_1^c, \ldots, w_{l_c}^c\right\}$ | a word sequence of the plaintiff's claims |
| $S^g = \left\{w_1^g, \ldots, w_{l_g}^g\right\}$ | a word sequence of the defendant's arguments |
| $S^f = \left\{w_1^f, \ldots, w_{l_f}^f\right\}$ | a word sequence of the fact description |
| $P = \{p_1, \ldots, p_k\}$ | the $k$ pleas of the plaintiff |
| $S^{p_i} = \left\{w_1^{p_i}, \ldots, w_{l_{p_i}}^{p_i}\right\}$ | a word sequence of an arbitrary plea |
| $S^l = \{S^c, S^g, P\}$ | a collection of litigant statements |
| $Y_{p_i} = \{0, 1, 2\}$ | the labels of a plea (i.e., reject, partially support, support) |
| $Y_c = \{c_1, \ldots, c_K\}$ | the $K$ civil cause labels |
| $S^{c_t} = \left\{w_1^{c_t}, \ldots, w_{l_{c_t}}^{c_t}\right\}$ | a word sequence of an arbitrary civil cause description |
| $Y_a = \{a_1, \ldots, a_m\}$ | a set of law articles |
| $S^{a_j} = \left\{w_1^{a_j}, \ldots, w_{l_{a_j}}^{a_j}\right\}$ | a word sequence of an arbitrary article description |

### 2.3 Label Embedding

The methods of label embedding have been popular in various domains and tasks that investigated the rich information behind class labels. In NLP, Tang et al. [25] constructed a heterogeneous text network and jointly embedded words, documents, and labels based on word-word and word-document co-occurrences as well as labeled documents. Wang et al. [28] embedded words and labels in the latent space and designed an attention framework to compute the compatibility between the labels and texts. A text embedding model CatE [20] was proposed to learn discriminative text embeddings for category representative term retrieval given a set of category

**Table 2: Specific inputs for three tasks.**

| Task | $S^l$ | $S^f$ | $\hat{Y}_c$ | $\hat{Y}_a$ |
|---|---|---|---|---|
| **CCP** | ✓ | | | |
| **CLAP** | | ✓ | ✓ | |
| **FJP** | ✓ | ✓ | | ✓ |

names as user guidance. We adopt label embeddings for learning semantics about law articles for obtaining more information between the case and formulated articles.

## 3 PROBLEM DEFINITION

To sum up, we formulate CCJP as a multi-task learning problem that contains three tasks: the predictions on the *civil cause*, the *relevant law articles*, and the *final judgment on each plea.* The main mathematical notations about inputs on three tasks are shown in Table 1. Our goal is to learn a classifier $\xi$ that can predict these three tasks cooperatively as follows:

$$\{\mathrm{Y}_c, \mathrm{Y_a}, Y_P\} \Leftarrow \xi\left(S^c, S^g, S^f, P, \mathrm{Y}_c, \mathrm{Y_a}\right). \tag{1}$$

Besides, in the Civil Cause Prediction (CCP), we formulate the task as a hierarchical text classification problem. Given a collection of litigant statements $S^l = \{S^c, S^g, P\}$, the defined hierarchical possible categories in H hierarchical levels $\mathbb{C} = (C_1, C_2, \ldots, C_H)$, where $\mathrm{C}_i = \left\{c_1, \ldots, c_{|c_i|}\right\}$ is the set of possible categories in the $i$-th hierarchical level, the $|c_i|$ is the number of categories in the $i$-th hierarchical level. The total number of causes is $K^3$. Our goal of CCP is to integrate the document texts $S^l$ and the hierarchical category structure $\gamma$ of causes to learn a classifier that could be used to predict $Y_c$ for different cases.

## 4 THE CPEE FRAMEWORK

In this section, we will introduce the technical details of CPEE. As shown in Figure 3, CPEE mainly contains three parts, i.e., CCP, CLAP, and FJP. Besides the legal elements inputs and document encoder layer, we design a logic-enhanced network across three tasks. Specifically, we first adopt specific legal elements inputs for each task. Then, we employ encoders to generate the semantic vectors of task-specific inputs. Finally, we exploit the logic-enhanced network that contains task-specific decoders to predict causes, law articles, and final judgments. In the whole framework, the results of cause prediction contribute to predicting articles, while law articles help to make final judgment predictions through the gate mechanism.

### 4.1 Legal Elements Inputs

In a real court, almost all sentences, facts, and debates are used for or against both main parties in the case [2], which indicates it is necessary to take full advantage of the text data in civil judgment documents[4]. Therefore, we perform different inputs for three tasks as shown in Table 2 by availing data adequately to simulate the real judicial process and analyze the case from multiple perspectives.

[3]In our work, the $H$ is 3, the root node represents the contract dispute. The $|c_2|$ is 6, the $|c_3|$ is 4, the total number of causes is 10.
[4]https://wenshu.court.gov.cn

### 4.2 Document Encoder

We design document encoders to generate the vector representations of the fact descriptions, litigant statements, cause labels, and article labels. The common Bi-directional LSTM [9] is adopted as our encoder. In detail, given a word sequence of fact descriptions $S^f$, we map each word of $S^f$ into its word embeddings by adopting pre-trained word vectors, the word2vec [21], and get the word embedding sequence of fact descriptions $E^f = \left\{e_1^f, \ldots e_{l_f}^f\right\}, e_i^f \in \mathbb{R}^{d_w}$, where $d_w$ is the dimension of word embedding. Then, we embed $E^f$ into continuous hidden states by Bi-LSTM encoder:

$$H^f = \mathrm{Bi}-LSTM\left(E^f\right), \tag{2}$$

where $H^f = \left\{h_1^f, h_2^f \ldots h_{l_f}^f\right\} \in \mathbb{R}^{l_f \times d_s}, d_s$ is the double size of hidden state. Similarly, given the claims $S^c$ of the plaintiff, the arguments $S^g$ of the defendant, and an arbitrary plea $S^{p_i}$, we can obtain their continuous representations $H^c \in \mathbb{R}^{l_c \times d_s}$, $H^g \in \mathbb{R}^{l_g \times d_s}$, and $H^{p_i} \in \mathbb{R}^{l_{p_i} \times d_s}$. The $H^c$, $H^g$, and all $H^{p_i}$ are concated as $H^l$ for the representation of litigant statements.

Moreover, the label descriptions of causes and law articles are embedded for label embeddings. First, we input the hierarchical causes structure $\gamma$ into a matrix $E^C = \left(E^{C_1}, E^{C_2}, \ldots, E^{C_H}\right)$, where $E^{C_i} \in \mathbb{R}^{|c_i| \times d_c}$ is a randomly initialized matrix that represents the embedding of the $i$-th hierarchical category level with the $d_c$ dimension. Besides, take the general articles descriptions as an example, the label embeddings are $E^{a_g} = \left(e_1^{a_g}, e_2^{a_g}, \ldots, e_{|a_g|}^{a_g}\right)$, where $E^{a_g} \in \mathbb{R}^{|a_g| \times d_s}$, $\left|a_g\right|$ is the number of general articles. We can also get the representation of specific articles $E^{a_s}$ in a similar way.

### 4.3 Logic-enhanced Network

As mentioned in Section 1, our goal is to build an integrated framework for CCJP on various causes by simulating the logic of judges. To achieve that, we propose a logic-enhanced network that contains task-specific decoders for three tasks. Following, we describe these tasks in order.

*4.3.1* ***Task 1: Civil Cause Prediction (CCP).*** Actually, the judge determines the case type according to litigant statements to brief the case quickly and guide the following trial. The guidance of a good cause could reduce the scope of legal search, and quickly apply relevant law articles [22]. Practically, civil causes are defined as a four-layer hierarchy that represents the evolution from top-level general to bottom-level specific [18, 22] shown in Figure 2 (a).

To model the hierarchy of causes, we first choose ten of the **most frequent and representative** causes shown in Figure 2 (a). Then, we propose a **local aware global classifier** to model the dependencies among different layers by exploiting the hierarchy gradually. We set a local classifier at each level and integrate them into the global classifier to obtain the final prediction.

Specifically, in each layer, given the representations of litigant statements $H^l$, the embedding of the $h$-th cause level $E^{C_h} \in \mathbb{R}^{|c_h| \times d_c}$, we could perform $|c_h|$ different causes of attention as:

$$O_h = \tanh\left(w_l \cdot {H^l}^T\right), \tag{3}$$

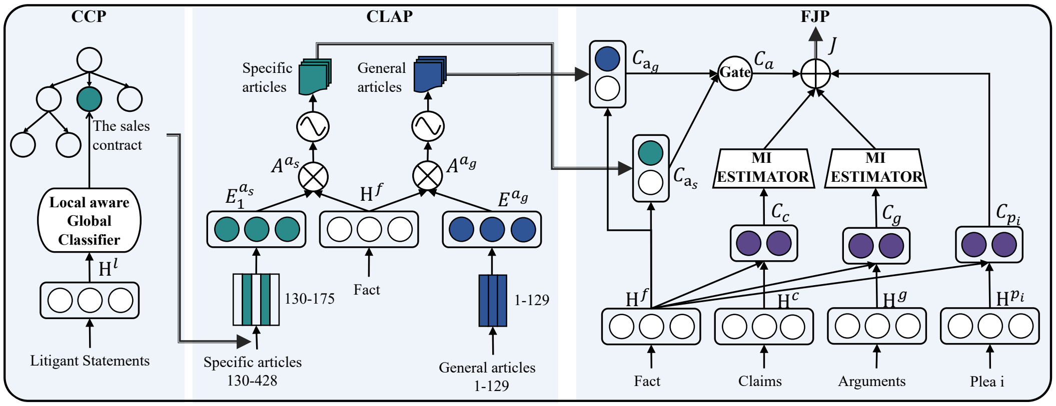



**Figure 3: The framework of CPEE represents three tasks on the trial mode from left to right: (1) CCP: a local aware global classifier for predicting hierarchical causes. (2) CLAP: Two modules for predicting two-layer articles. (3) FJP: The final judgment prediction with comprehensive case relationships.**

$$W_{att}^h = \text{softmax}\left(E^{C_h} \cdot O_h\right), \tag{4}$$

where $w_l \in \mathbb{R}^{d_c \times d_s}$ is a randomly initialized weight matrix, the softmax() ensures all the computed weights sum up to 1 for each cause. The $W_i^h$ in $W_{att}^h = \left(W_1^h, W_2^h, \ldots, W_{|c_h|}^h\right) \in \mathbb{R}^{|c_h| \times l_l}$ represents the attention score of the text with the $i$-th cause in the $h$-th level after normalization. Then we compute $|c_h|$ weighted sums to obtain the text-cause representation $V_{att}^h \in \mathbb{R}^{|c_h| \times d_s}$ with each cause in the $h$-th level:

$$V_{att}^h = W_{att}^h \cdot H^l, \tag{5}$$

the text-cause representation $A_{att}^h \in \mathbb{R}^{d_s}$ for whole $h$-th cause level can be modeled by averaging $V_{att}^h$ in cause-dims:

$$A_{att}^h = \text{avg}\left(V_{att}^h\right). \tag{6}$$

Next, we concat the average text representation $\widetilde{H^l}$ and the hierarchical text-cause representation $A_{att}^h$ to get local representation at $h$-th level $A_c^h \in \mathbb{R}^{|c_h|}$:

$$A_c^h = \varphi\left(w_c^h \cdot \left[\widetilde{H^l} \oplus A_{att}^h\right] + b_c^h\right), \tag{7}$$

where $w_c^h \in \mathbb{R}^{|c_h| \times 2d_s}$ is the weighted matrix and $b_c^h \in \mathbb{R}^{|c_h|}$ is the corresponding bias vector, $\oplus$ denotes vector concatenation operation and $\varphi$ is a non-linear activation function (e.g. RELU).

In the next layer, in order to introduce the hierarchy information of previous level to the global information, we replace $H^l$ in Equation (3), (5), and (7) with $H_h^l$:

$$H_h^l = W_{att}^h \otimes H^l, \tag{8}$$

where the $\otimes$ denotes the entry-wise product operation.

After we obtain the local representation $A_c^h$ in each layer, it is important to generate both the local and global information to predict the final cause. Thus, we concat all text-cause representations at each layer $A_c^h$ together as $A_c^H \in \mathbb{R}^K$ and predict the cause $\hat{y}_c$ by:

$$\hat{y}_c = \text{softmax}\left(A_c^H\right). \tag{9}$$

4.3.2 ***Task 2: Civil Law Articles Prediction (CLAP)***. After determining the civil cause, the judge can apply relevant law articles which provide the legal basis and improve the rigor and fairness of judgments. In the real civil system, civil law has a two-layer structure that divides into general articles and specific articles[5] [5]. The general stipulates the principles and basic spirit of civil law which are **universal**, the specific regulates the concrete situations of different disputes which are **particular** [29]. Therefore, cases on different causes apply to the same general articles but different specific articles as shown in Figure 2 (b). We could adopt the results of CCP to help predict specific articles.

Moreover, in our dataset, there are a total of 198 articles and 3 articles per case, which is hard for an algorithm to predict directly. Therefore, we propose two modules to predict the general articles and specific articles respectively. In each module, we take the straightforward approach to alleviate the complexity of civil law articles by enhancing the interaction between article labels and case facts through label embedding.

In the module for predicting general articles, we input the fact descriptions $H^f$ and the general articles embedding $E^{a_g} \in \mathbb{R}^{|a_g| \times d_s}$ to compute the interaction matrix $I^{a_g} \in \mathbb{R}^{l_f \times |a_g|}$ as follows:

$$I_{a_g} = H^f \cdot {E^{a_g}}^T. \tag{10}$$

To calculate the attention scores between the fact and general articles descriptions, we have:

$$\begin{aligned} G_{a_g} &= \text{softmax}\left(I_{a_g}\right), \\ A_{a_g} &= {I_{a_g}}^T \cdot G_{a_g}, \end{aligned} \tag{11}$$

where $A_{a_g} \in \mathbb{R}^{|a_g| \times |a_g|}$, then the general articles can be predicted by sigmoid operation:

$$\hat{y}_{a_g} = \sigma\left(w_{a_g} \cdot A_{a_g} + b_{a_g}\right), \tag{12}$$

where $\sigma$ is the sigmoid activation function, $w_{a_g} \in \mathbb{R}^{|a_g| \times |a_g|}$ is a randomly initialized weight matrix, and $b_{a_g} \in \mathbb{R}^{|a_g|}$ is the corresponding bias vector.

[5]The law used in the paper is The Contract Law of the People's Republic of China.

Different from predicting general articles, since the cases on different causes apply to different specific articles, it is essential to predict specific articles with the help of the causes predicted in Task 1. We define the specific articles on $K$ causes as a collection of $E^{a_s} = \{E_1^{a_s}, \ldots, E_K^{a_s}\}$, where $E_i^{a_s} \in \mathbb{R}^{|a_{s_i}| \times d_s}$ means the corresponding specific articles on the $i$-th cause, $|a_{s_i}|$ means the number of corresponding specific articles. We replace $E^{a_g}$ with $E_i^{a_s}$ to predict the specific articles $\hat{y}_{a_s}$ by the above computations.

*4.3.3* ***Task 3: Final Judgment Prediction (FJP).*** After applying law articles, the judge makes final judgments on the plaintiff's pleas based on litigant statements, confirmed facts, and relevant law articles, i.e., the comprehensive information from multiple perspectives. The logical and comprehensive processes motivate judges to make fairer and more accurate judgments. Thus, we adopt the co-attention mechanism [33] to capture sufficient information. Besides, both the claims from the plaintiff and the arguments from the defendant are expressed from different perspectives, but both aim at one case. We adopt mutual information regularizer [26, 43] to capture more case information from both parties.

Specifically, after encoding text representations $H^f$, $H^c$, $H^g$, and $H^P$, we first compute the affinity matrix which contains affinity scores corresponding to all pairs of the fact and the claims:

$$L_{cf} = \mathrm{H}^c \cdot {\mathrm{H}^f}^T, \tag{13}$$

where $L_{cf} \in \mathbb{R}^{l_c \times l_f}$. Then, the affinity matrix is normalized row-wise to produce attention weights $A_f$ across the claims for each word in the fact, and column-wise to produce the attention weights $A_c$ across the fact for each word in the claims.

$$A_f = \mathrm{softmax}(L_{cf}) \text{ and } A_c = \mathrm{softmax}\left(L_{cf}^T\right), \tag{14}$$

where $A_f \in \mathbb{R}^{l_c \times l_f}$ and $A_c \in \mathbb{R}^{l_f \times l_c}$. Next, we compute the attention contexts of the fact in light of the word of the claims:

$$C_f = {\mathrm{H}^c}^T \cdot \mathrm{A}_f, \tag{15}$$

where $C_f \in \mathbb{R}^{d_s \times l_f}$, and finally we get the co-dependent representation of the fact and the claims by:

$$C_c = {\mathrm{A}_c}^T \cdot \left[H^f; {\mathrm{C}_f}^T\right], \tag{16}$$

where $C_c \in \mathbb{R}^{l_c \times 2d_s}$. Similarly, given the arguments $H^g$ and an arbitrary plea $H^{p_i}$, the co-attention representation $C_g$ shows the interaction between fact and arguments, $C_{p_i}$ denotes the interaction between fact and the $i$-th plea.

Furthermore, the judge needs to synthesize information from general and specific articles to make judgments in real court settings. Therefore, we use a gate mechanism to control the weight between both and integrate specific and general articles. First, we take the semantic vector $G_{a_g}$ of general articles and vector $G_{a_s}$ of specific articles as inputs with fact $H^f$ to compute the interaction matrix $C_{a_g} \in \mathbb{R}^{|a_g| \times d_s}$ and $C_{a_s} \in \mathbb{R}^{|a_s| \times d_s}$ as follows:

$$C_{a_g} = {\mathrm{G}_{\mathrm{a_g}}}^T \cdot \mathrm{H}_f \text{ and } C_{a_s} = {\mathrm{G}_{\mathrm{a_s}}}^T \cdot \mathrm{H}_f. \tag{17}$$

Then, the weights in the gate mechanism are calculated by:

$$\begin{aligned} \mathrm{w}_a &= \sigma\left(w_s \cdot \mathrm{C}_{a_s} + w_g \cdot \mathrm{C}_{a_g} + b_a\right), \\ \mathrm{C}_a &= w_a \cdot \left(w_s \cdot \mathrm{C}_{a_s}\right) \oplus (1 - w_a) \cdot \left(w_g \cdot \mathrm{C}_{a_g}\right), \end{aligned} \tag{18}$$

where $C_a \in \mathbb{R}^{d_s \times 2d_s}$ is the integrated representation of law articles, $w_g$, $w_s$, and $b_a$ are trainable parameters.

Besides, as we all know, the higher the mutual information [26], the higher relationship between the two variables. Since the claims could be seen as $H^c = wH^f$, which means the claims from the plaintiff focus on a part of case facts and there may be some deviation from fact. We adopt the mutual information regularizer to capture the part that is consistent with the fact to maximize the case fact information, capture case controversies, and ignore the part that is biased. The mutual information $\mathrm{I}(H^c; H^f)$ between claims $H^c$ and fact $H^f$ could be defined as follows:

$$I(H^c; H^f) = \mathbb{E}_{p(H^c, H^f)}\left[\log \frac{p(H^c, H^f)}{p(H^c)p(H^f)}\right]. \tag{19}$$

Similarly, we could obtain the common parts between arguments $H^g$ and fact $H^f$ by maximizing the mutual information $I(H^g; H^f)$.

## 4.4 Prediction and Training

With the predictions of CCP and CLAP (i.e., Equation (9) and (12)), we consider FJP based on multi-perspective representations. We first apply per-dimension mean-pooling over the concated representation $J$. Then, we apply an affine transformation followed by softmax to obtain the final prediction as:

$$\begin{aligned} J &= \mathrm{avg}\left(\left[\mathrm{C_c}; \mathrm{C_g}; \mathrm{C_{p_i}}; \mathrm{C_a}\right]\right), \\ \hat{y}_J &= \mathrm{softmax}\left(w_j \cdot J + b_j\right), \end{aligned} \tag{20}$$

here, $w_j$ and $b_j$ are parameters to learned, and ";" denotes the concatenate operation. In practice, we employ the cross-entropy loss function for each task. The mutual information regularization is taken for FJP and the weighted sum is used as an overall loss:

$$\mathcal{L} = -\sum_{j=1}^{3} \lambda_j \sum_{k=1}^{|Y_j|} y_{j,k} \log\left(\hat{y}_{j,k}\right) - I(H^c; H^f) - I(H^g; H^f), \tag{21}$$

where $|Y_j|$ denotes the number of labels for task j, and $\lambda_j$ is the weight factor which is the hyperparameter for each task.

# 5 EXPERIMENTS

In this section, in order to demonstrate the effectiveness of our model, we first compare CPEE with some baselines on FJP, then we conduct some baselines and variants experiments on CCP and CLAP. Finally, we assess the contribution of different components and the performance of CPEE on single causes. Furthermore, we make some visualization analyses.

## 5.1 Dataset Description

At present, existing research on CCJP all focuses on one cause [16, 19, 44] and neglects all-sided case information. In order to formulate a complete trial mode of CCJP, we collect 158,625 cases

**Table 3: The statistics of dataset.**

| #Pleas | #Supported pleas | #Partially supported pleas | #Rejected pleas | #Law articles | #Specific articles | #General articles |
|---|---|---|---|---|---|---|
| 320,003 | 216,815 | 37,381 | 65,807 | 198 | 76 | 122 |
| Avg. #Tokens in fact | Avg. # Tokens in claims | Avg. #Tokens in arguments | Avg. #Tokens in pleas | Avg. #Law articles | Avg. #Specific articles | Avg. #General articles |
| 156 | 99 | 32 | 49 | 3 | 1.7 | 1.9 |

**Table 4: The experimental results of Final Judgment Prediction (FJP) and Civil Cause Prediction (CCP).**

| **FJP** | | | | | **CCP** | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Methods | ACC | Mac.P | Mac.R | Mac.F1 | Methods | ACC | Mac.P | Mac.R | Mac.F |
| SVM+word2vec [24] | 0.678 | 0.332 | 0.327 | 0.311 | HMC [27] | 0.536 | 0.482 | 0.488 | 0.484 |
| AutoJudge [16] | 0.751 | 0.417 | 0.334 | 0.287 | Hdltex [15] | 0.474 | 0.079 | 0.167 | 0.107 |
| MSJudge [19] | 0.661 | 0.329 | 0.399 | 0.344 | HARNN [10] | 0.136 | 0.093 | 0.160 | 0.052 |
| CCJudge [38] | 0.738 | 0.369 | 0.497 | 0.423 | Flat-cause | 0.500 | 0.250 | 0.333 | 0.278 |
| BERT-Civil [42] | 0.735 | 0.418 | 0.341 | 0.302 | Hie+fact | 0.667 | 0.513 | 0.489 | 0.487 |
| **CPEE** | **0.836** | **0.423** | **0.500** | **0.458** | **CPEE (std)** | **0.706**±0.07 | **0.556**±0.07 | **0.640**±0.07 | **0.589**±0.08 |

containing ten kinds of frequent causes[6]. Every case includes the litigant statements, ascertained facts, corresponding cause, law articles, and final judgment on each plea. Note that the collected cases based on raw civil legal documents[7] include special typographical signals, making extracting labeled data with regular expressions easy. Following the hierarchical structures of civil causes and law articles, we make some data processing. Specifically, we handle the civil causes as described in Section 3, and divide the law articles into general articles and specific articles based on the scope of application. Besides, we filter out some mislabeled cases from the initial data. For example, we find some specific articles of lease contract dispute which should be labeled in 212-236, but are labeled in 237-250. these financial lease contracts are mislabeled as lease contract disputes. Due to the division of specific articles, the efficiency and accuracy of data review have improved greatly. We also remove personally identifiable information in each case.

After collecting the dataset as shown in Table 3, we randomly separate the dataset into a training set, a validation set, and a test set according to a ratio of 8: 1: 1. The distributions of civil causes and articles are roughly the same in each set. On average, there are about 2 pleas and 3 law articles per case in our dataset, the detailed statistics are shown in Table 3. Similar to the discoveries of Zhao et al. [38] and Wu et al. [32], there exists data imbalance in civil datasets, the ratio of category labels of our dataset(i.e., support, partially support, and reject) in FJP is 5.8: 1: 1.7.

## 5.2 Comparison Methods

To evaluate the performance of our model on CCJP, we adopt some representative approaches including the traditional method, large-scale pre-trained method, and other deep learning-based CCJP models. Among them, as the task inputs, definitions, and goals of LJP [37, 39] are different from CCJP, we do not implement LJP models as comparison methods. In the following, we introduce some baselines on FJP, where the existing methods that are available for CCJP are single-task learning to predict the final judgments. Therefore, we compare CPEE with those CCJP models only on FJP.

- **Word2Vec+SVM** employs the word2vec [21] to represent word features and utilizes SVM [24] for text classification.
- **AutoJudge** [16] formalizes CCJP as a reading comprehension task on the divorce dispute where the fact, claims, and relevant articles as inputs. Besides they selected the most relevant 5 articles according to the fact as inputs. We remove the part in our experiments.
- **MSJudge** [19] proposes a multi-task learning framework by predicting the hand-labeled fact and the final judgments based on the private lending debate data. Since our data comes from civil judgment documents, we replace the debate data with our claims and arguments to predict the final judgments with ground truth facts.
- **CCJudge** [38] takes advantage of multi-perspective information and introduces the legal knowledge base of various civil causes. For a fair comparison, the predictions are conducted without legal knowledge in our experiments.
- **BERT-Civil** [42] is a variant of BERT which is pre-trained with civil case data. As we all know, BERT [6] is a language representation built on deep bidirectional transformers. It outperforms state-of-the-art models on a wide range of NLP tasks. We use BERT-Civil as our baseline method.

Then, to further validate the performance of CCP and CLAP, we adopt some baselines and design some variants, including:

- **HMC** [27] is a global approach that constructs a single decision tree to classify all categories.
- **Hdltex** [15] is a local approach that trains one multi-class classifier for each class level.
- **HARNN** [10] is a model proposed for hierarchical multi-label text classification, they focus on modeling the dependencies among class levels and the text-label compatibility.
- **Flat-cause** is a method of Bi-directional LSTM with mean pooling, without using the hierarchical structure of causes.

[6]The ten causes are all contractual disputes, they are distributed on the third and fourth layer in Figure 2 (a).
[7]https://wenshu.court.gov.cn

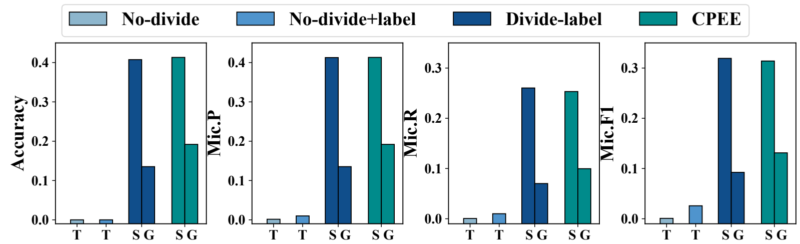



**Figure 4: The experimental results on the variants of CLAP. The T, S, G represent the predictions on total, specific, general articles, respectively.**

- **Hie+fact** utilizes fact descriptions and hierarchical structure other than the litigant statements as inputs.
- **No-divide** directly predicts the total articles without the division of general articles and specific articles.
- **No-divide+label** predicts the total articles with label embeddings of total articles.
- **Divide-label** divides the predictions of articles into the predictions on the general articles and specific articles but without the label embedding.

## 5.3 Experimental Setup

For methods based on CNN or RNN, we first employ the JieBa word segmentation tool[8] for word segmentation as the legal documents are written in Chinese with no space. Afterward, we adopt the word2vec tool [21] to generate word embeddings of embedding size 300 for each word in a document, which is trained on texts of documents in the dataset. Meanwhile, the size of embedding representations for causes is 300. Besides, we set the max sequence length to 400, and all hidden sizes to 128. For methods based on BERT, we adopt the pre-trained model of Chinese which was trained by Cui et al. [6] and Zhong et al. [42], the maximum document length is 512 tokens. For training, the learning rate of the Adam optimizer [13] is initialized as $10^{-3}$. We implement the proposed model with Tensorflow [1] on a single V100 GPU, and train each model for 10 epochs with batch size 16. Finally, we employ accuracy (ACC), macro-precision (Mac.P), macro-recall (Mac.R), and macro-F1 (Mac.F1) as evaluation metrics to evaluate the final model on CCP and FJP. Besides, we adopt micro-precision (Mic.P), micro-recall (Mic.R), and micro-F1 (Mic.F1) in CLAP to evaluate top 3 civil law articles prediction[9].

## 5.4 Experimental Results

To evaluate the performance of CPEE, we export the experimental results on three tasks. As CCJP is ultimately concerned with FJP, here we report the results of FJP first, followed by CCP and CLAP.

*5.4.1 Comparison against baselines on FJP.* Specifically, we compare CPEE with some baselines, and the results as shown in Table 4. We could find our proposed CPEE performs the best overall which demonstrates that CPEE could effectively make predictions on various causes and capture the relationships from multiple perspectives. Furthermore, from the results, we could get the following observations. (1) SVM+word2vec does not perform as well as other baselines based on deep learning overall, it is possible that the SVM fails to model the deep interactions between multi-perspective data and labels. (2) MSJudge performs life-cycle admissibility by injecting court data. The performance comes primarily from the ground truth fact descriptions with manual fact labels. By failing to provide fact labels for each cause, MSJudge performs poorly on various-cause data. (3) AutoJudge mainly makes judgment predictions on the ascertained facts but ignores the statements from the plaintiff and defendant, our model beats it which also indicates the comprehensive judicial process helps make judgments on various causes. (4) CCJudge performs well than other baselines because it also builds for the predictions on various causes. However, the overall performance is poor for predicting a series of complicated contract disputes with multiple pleas. At the same time, we guess that the lack of legal knowledge base in the experiment is also a reason for the poor effect. (5) We also choose the Chinese BERT trained by [6] to make experiments, the accuracy of BERT is **0.760** and the Mac.F1 is **0.288**. Compared to the BERT, BERT-Civil performs better but behaves worse than CPEE. It further demonstrates the effectiveness of our model. (6) At last, we choose the most intricate contract disputes, which include a large number of the property, contractual, personal relations, etc. The case circumstances of leaf causes belonging to the same parent cause are similar. It is hard to predict in such complex relationships and confusing cases.

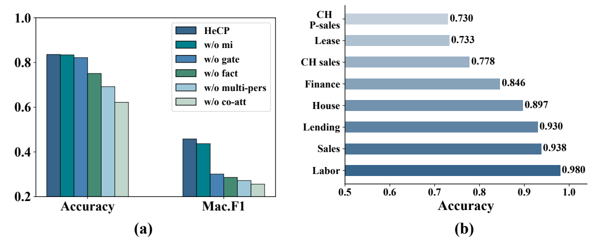
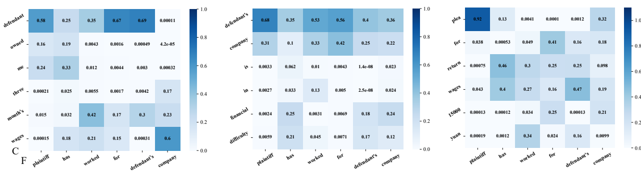



**Figure 5: (a) The experimental results of ablation tests. (b) CPEE performance on different single causes.**

*5.4.2 Comparison against variants on CCP and CLAP.* **In CCP**, we formalize the task as a Hierarchical Text Classification (HTC) problem by introducing the original structure of causes. In camera-ready version, we perform 5 experiments and obtain the average performances and the standard deviations to verify the effectiveness of CPEE. Contrary to expectations, as shown in Table 4, the HTC methods based on deep learning perform worst than the traditional method as HMC based on the decision tree. Hdltex and HARNN do not learn the characteristics of legal data well compared to HMC, which is better suited to multi-domain data. To be obvious, the results are worst when the task becomes a flat classification problem, which illustrates the importance of introducing the hierarchical

[8]https://github.com/fxsjy/jieba
[9]https://github.com/bigdata-ustc/CPEE



(a) The $A^f$ scores across the Claims in the Fact. (b) The $A^f$ scores across the arguments in the Fact. (c) The $A^f$ scores across the Plea i in the Fact.

**Figure 6: Examples of attention scores between the fact (horizontal axis) and the litigant statements (vertical axis).**

structure. Besides, the performance of Hie+fact is poor, which indicates that multi-perspective information is more conducive for CCP than fact descriptions. The local aware global classifier of CPEE performs well on CCP, which not only helps us predict law articles but also extends to numerous causes ahead.

**In CLAP**, it is a multi-label classification task, results are shown in Figure 4. First, No-divide does not perform anything in predicting a total of 198 law articles, after all, it is difficult to predict articles directly. Then, we adopt label embeddings for the total predictions with a little improvement. Next, we divide law articles into specific articles and general articles, it is clear that both performances of predictions on specific and general articles are improved a lot in the variant Divide-label. Finally, CPEE improves a bit with the label embeddings. Since civil articles do not contain fine-grained ground truth articles[10] but just basic principles, they are for reference in all judgments. It further demonstrates the CPEE could extract relevant articles with the help of two modules and CCP.

## 5.5 Comparative Analysis

**Ablation tests.** We conduct ablation tests to evaluate the contribution of different components in the FJP. Figure 5 (a) reports the accuracy and Mac.F1 scores when training on all features except the particular one. To validate the influence of data information, we remove the multi-perspective information and fact as "w/o fact" and "w/o multi-pers", respectively. Similarly, the importance of co-attention, mutual information regularizer, and gate mechanisms are demonstrated shown as "w/o co-att", "w/o mi", and "w/o gate" respectively. Figure 5 (a) clearly tells that all the components contribute positively to the results. Specifically, the co-attention layer shows a significant influence on judgment predictions to capture comprehensive information. Besides, multi-perspective data has contributed a lot to the performance, as the data and mutual information regularizer complement each other which impacts a lot. Also, we could find the removal of the gate mechanism results in the reduction of Mac.F1 scores, which further validates the importance of incorporating articles into FJP.

**Experiments on single causes.** Furthermore, we conduct some experiments on every single cause for verifying the effect of CPEE as shown in Figure 5 (b). CPEE performs well in some common causes like the sales contract dispute (Sales) and the house sales contract dispute (House). In the labor contract dispute (Labor), CPEE has an accuracy rate of 0.980, mainly because the legal relationship is relatively simple in the dispute involving whether the employer and the employee have a legal labor contract. However, on some of the inferior performing causes like the commercial housing presale contract dispute (CH P-sales), and commercial housing sales contract dispute (CH Sales), the legal relationships usually include not only the establishment of a contract but also the amount of compensation and personal relationships. Besides, the circumstances of their cases are similar. On the whole, the judgment predictions on a single cause perform well which illustrates the CPEE could capture all-sided case information.

**Visualization study.** We visualize the heat maps of co-attention results as shown in Figure 6, the attention $A_f$ score is calculated in Eq. (14). For example in Figure 6 (a), we take the average of the $l_c \times l_f$ over the time dimension to obtain attention values for each word. The visualization demonstrates that the attention mechanism can capture deep relationships between the facts and litigant statements.

[10] Fine-grained articles are in the Juridical Interpretations, giving detailed explanations according to some circumstances.

## 5.6 Ethical Discussion

It is well known that the opacity of artificial intelligence algorithms has always been one of the controversial points of its applications in the field of litigation [4]. However, due to the fewer people and more cases in the civil field [41], it is necessary to make reasonable use of artificial intelligence to provide judges with judgment assistance.

We simulated a real trial mode and constructed interdependent judgment tasks centering on the trial mode of essential elements. The correlation between the three tasks not only provides judges with suggestions but also gives them more initiative and room to think. Besides, in the data preprocessing, we anonymized the data by removing sensitive information (e.g., name, gender, race, etc.) [19]. In the future, we need to pay more attention to key legal AI issues, such as fairness, interpretability, judicial impartiality, and judicial diversity [11, 32].

# 6 CONCLUSIONS

In this paper, we proposed a civil case judgment prediction method centering on the trial mode of essential elements (i.e. CPEE), which simulated the comprehensive trial mode to tackle the problem of various causes and complicated legal essential elements. To be specific, we divided the judicial steps into three tasks which contain civil cause, civil law articles, and the final judgment predictions. In the logic-enhanced network, we modeled the hierarchical legal elements in the first two tasks by building a local aware global classifier and two modules with label embeddings. Besides, we adopted co-attention and mutual information regularizer to capture more relationships from multiple perspectives. Moreover, we constructed a real-world dataset that contained 158,625 civil cases with various civil causes and comprehensive legal elements. Extensive experiments on our dataset demonstrated the superiority of CPEE. Finally, we made ethical discussions of our work since the sensitivity and particularity of Legal AI.

# 7 ACKNOWLEDGEMENTS

This research was partially supported by grants from the National Natural Science Foundation of China (Grant No. 61922073 and U20A20229).

## REFERENCES


[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467* (2016).

[2] Zou Bihua. 2010. *The Nine Steps of Trial of Essential Items.* Law Press.

[3] Nicolò Cesa-Bianchi, Claudio Gentile, and Luca Zaniboni. 2006. Incremental Algorithms for Hierarchical Classification. *J. Mach. Learn. Res.* 7 (2006), 31–54.

[4] Cary Coglianese and David Lehr. 2016. Regulating by robot: administrative decision making in the machine-learning era. *Geo. LJ* 105 (2016), 1147.

[5] NPC Standing Committee. 2021. *Civil Law.* Press of Chinese Democratic Legal System.

[6] Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2021. Pre-training with whole word masking for chinese bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29 (2021), 3504–3514.

[7] Caspar J. Fall, A. Törcsvári, K. Benzineb, and G. Karetka. 2003. Automated categorization in the international patent classification. *SIGIR Forum* 37, 1 (2003), 10–25.

[8] Leilei Gan, Kun Kuang, Yi Yang, and Fei Wu. 2021. Judgment prediction via injecting legal knowledge into neural networks. In *Proceedings of AAAI*. 12866–12874.

[9] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*. Ieee, 6645–6649.

[10] Wei Huang, Enhong Chen, Qi Liu, Yuying Chen, Zai Huang, Yang Liu, Zhou Zhao, Dan Zhang, and Shijin Wang. 2019. Hierarchical multi-label text classification: An attention-based recurrent network approach. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1051–1060.

[11] Alon Jacovi and Yoav Goldberg. 2020. Towards Faithfully Interpretable NLP Systems: How Should We Define and Evaluate Faithfulness?. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 4198–4205.

[12] Carlos Nascimento Silla Jr. and Alex Alves Freitas. 2009. Novel Top-Down Approaches for Hierarchical Classification and Their Application to Automatic Music Genre Classification. In *2009 IEEE International Conference on Systems, Man and Cybernetics*. IEEE, 3499–3504.

[13] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations (ICLR)*, Yoshua Bengio and Yann LeCun (Eds.).

[14] Svetlana Kiritchenko, Stan Matwin, Richard Nock, and A Fazel Famili. 2006. Learning and evaluation in the presence of class hierarchies: Application to text categorization. In *Conference of the Canadian Society for Computational Studies of Intelligence*. Springer, 395–406.

[15] Kamran Kowsari, Donald E Brown, Mojtaba Heidarysafa, Kiana Jafari Meimandi, Matthew S Gerber, and Laura E Barnes. 2017. Hdltex: Hierarchical deep learning for text classification. In *2017 16th IEEE international conference on machine learning and applications (ICMLA)*. IEEE, 364–371.

[16] Shangbang Long, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. 2019. Automatic judgment prediction via legal reading comprehension. In *China National Conference on Chinese Computational Linguistics*. Springer, 558–572.

[17] 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)*. Association for Computational Linguistics, 2727–2736.

[18] Dongchuan Luo and Jianzhong Huang. 2008. Understanding and application of The Provisions on Cause of Action in Civil Cases. *The people's Judicature* 5 (2008), 18–23.

[19] 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 *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 993–1002.

[20] Yu Meng, Jiaxin Huang, Guangyuan Wang, Zihan Wang, Chao Zhang, Yu Zhang, and Jiawei Han. 2020. Discriminative topic mining via category-name guided text embedding. In *Proceedings of The Web Conference 2020*. 2121–2132.

[21] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[22] Pascoe Pleasence, Nigel Balmer, and Alexy Buck. 2006. *Causes of action: civil law and social justice.* The Stationery Office.

[23] Carlos N Silla and Alex A Freitas. 2011. A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery* 22, 1 (2011), 31–72.

[24] Johan AK Suykens and Joos Vandewalle. 1999. Least squares support vector machine classifiers. *Neural processing letters* 9, 3 (1999), 293–300.

[25] Jian Tang, Meng Qu, and Qiaozhu Mei. 2015. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 1165–1174.

[26] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation Learning with Contrastive Predictive Coding. *CoRR* abs/1807.03748 (2018).

[27] Celine Vens, Jan Struyf, Leander Schietgat, Sašo Džeroski, and Hendrik Blockeel. 2008. Decision trees for hierarchical multi-label classification. *Machine learning* 73, 2 (2008), 185–214.

[28] Guoyin Wang, Chunyuan Li, Wenlin Wang, Yizhe Zhang, Dinghan Shen, Xinyuan Zhang, Ricardo Henao, and Lawrence Carin. 2018. Joint Embedding of Words and Labels for Text Classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*. 2321–2331.

[29] Liming Wang and Yegang Wang. 2016. *Contract Law.* City University of HK Press.

[30] Pengfei Wang, Ze Yang, Shuzi Niu, Yongfeng Zhang, Lei Zhang, and ShaoZhang Niu. 2018. Modeling dynamic pairwise attention for crime classification over legal articles. In *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*. 485–494.

[31] Jonatas Wehrmann, Ricardo Cerri, and Rodrigo Barros. 2018. Hierarchical multi-label classification networks. In *International Conference on Machine Learning*. PMLR, 5075–5084.

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

[33] Caiming Xiong, Victor Zhong, and Richard Socher. 2017. Dynamic Coattention Networks For Question Answering. In *The 5th International Conference on Learning Representations (ICLR)*.

[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)*. Association for Computational Linguistics, 3086–3095.

[35] Zhuopeng Xu, Xia Li, Yinlin Li, Zihan Wang, Yujie Fanxu, and Xiaoyan Lai. 2020. Multi-task Legal Judgement Prediction Combining a Subtask of the Seriousness of Charges. In *China National Conference on Chinese Computational Linguistics*. Springer, 415–429.

[36] 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*. 4085–4091.

[37] 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*. ACM, 973–982.

[38] Lili Zhao, Linan Yue, Yanqing An, Ye Liu, Kai Zhang, Weidong He, Yanmin Chen, Senchao Yuan, and Qi Liu. 2021. Legal Judgment Prediction with Multiple Perspectives on Civil Cases. In *CAAI International Conference on Artificial Intelligence*. Springer, 712–723.

[39] 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, Brussels*. Association for Computational Linguistics, 3540–3549.

[40] Haoxi Zhong, Chaojun Xiao, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, and Jianfeng Xu. 2018. Overview of CAIL2018: Legal Judgment Prediction Competition. *CoRR* abs/1810.05851 (2018).

[41] Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. How Does NLP Benefit Legal System: A Summary of Legal Artificial Intelligence. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault (Eds.). 5218–5230.

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

[43] Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020. S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*. 1893–1902.

[44] Xin Zhou, Yating Zhang, Xiaozhong Liu, Changlong Sun, and Luo Si. 2019. Legal Intelligence for E-commerce: Multi-task Learning by Leveraging Multiview Dispute Representation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 315–324.