From Graph to Word Bag: Introducing Domain Knowledge to Confusing Charge Prediction

Ang Li[1], Qiangchao Chen[2], Yiquan Wu[1], Ming Cai[1],
Xiang Zhou[2†] , Fei Wu[1], Kun Kuang[1]
[1]College of Computer Science and Technology, Zhejiang University
[2]GuangHua Law School, Zhejiang University
{liangrex, 22102078, wuyiquan, cm, 0020355, wufei, kunkuang}@zju.edu.cn

Abstract

Confusing charge prediction is a challenging task in legal AI, which involves predicting confusing charges based on fact descriptions. While existing charge prediction methods have shown impressive performance, they face significant challenges when dealing with confusing charges, such as *Snatch* and *Robbery*. In the legal domain, constituent elements play a pivotal role in distinguishing confusing charges. Constituent elements are fundamental behaviors underlying criminal punishment and have subtle distinctions among charges. In this paper, we introduce a novel **F**rom **G**raph to **W**ord **B**ag (FWGB) approach, which introduces domain knowledge regarding constituent elements to guide the model in making judgments on confusing charges, much like a judge's reasoning process. Specifically, we first construct a legal knowledge graph containing constituent elements to help select keywords for each charge, forming a word bag. Subsequently, to guide the model's attention towards the differentiating information for each charge within the context, we expand the attention mechanism and introduce a new loss function with attention supervision through words in the word bag. We construct the confusing charges dataset from real-world judicial documents. Experiments demonstrate the effectiveness of our method, especially in maintaining exceptional performance in imbalanced label distributions.

Keywords: Document Classification, Knowledge Discovery, Legal Artificial Intelligence

1. Introduction

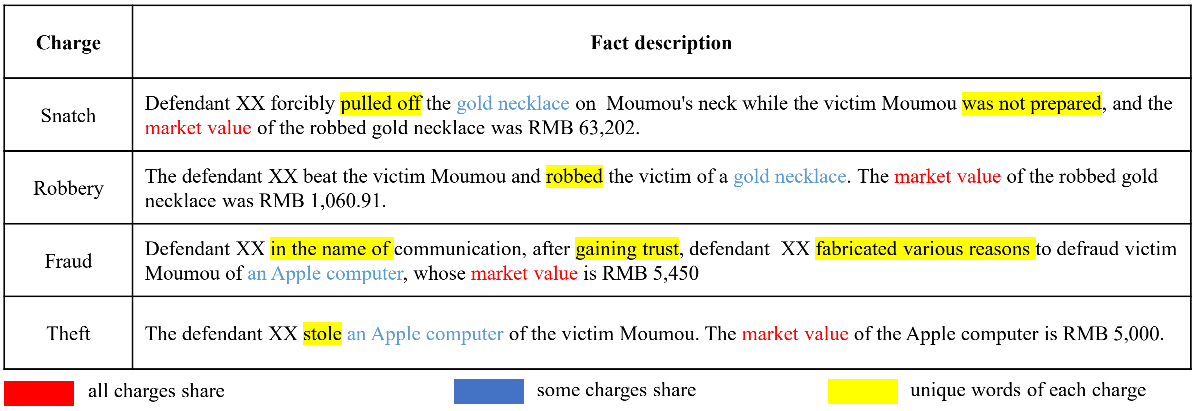
In recent years, artificial intelligence has been applied in the legal domain. Legal artificial intelligence (LegalAI) focuses on applying artificial intelligence methods to benefit legal tasks (Zhong et al., 2020). These advancements have led to increased research on charge prediction (Cui et al., 2022; Luo et al., 2017; Hu et al., 2018; Ye et al., 2018). In all these studies, researchers approach the charge prediction task as a classification problem, utilizing classification models. Substantial progress has been made in these areas over time.

In charge prediction, many methods have been extensively proposed and they have commendable predictive performance. However, this task still faces challenges when dealing with confusing charges in real legal scenarios. Most existing methods for crime prediction primarily focus on the legal system structure (Zhong et al., 2018) or on macro-level semantic knowledge (Hu et al., 2018). These methods utilize legal knowledge to assist the model, but they are not sufficient to enable the model to master the ability to distinguish between confusing charges.

In this work, we focus on the task of confusing charge prediction. which is a subset of charge prediction, specifically addressing cases where the performance is poor. Fig. 1 illustrates a cluster of confusing charges in real cases, and highlights shared words and unique words for each charge. From a legal perspective, the key to distinguishing these charges can be reflected through unique words: *Theft* is characterized by non-violence, *Snatch* involves violence against property, *Robbery* entails violence against individuals, and *Fraud* includes descriptions related to trust. Therefore, the challenge in this task is: How to make the model focus on and understand the critical information that distinguishes confusing charges?

It's worth noting that the constituent elements play a pivotal role in charge prediction. Constituent elements refer to the types of behavior or crimes that serve as the basis for criminal punishment according to abstract provisions of criminal law. To address the aforementioned challenge, we propose a novel **F**rom **G**raph to **W**ord **B**ag (FWGB) approach to leverage these constituent elements. Specifically, we construct a knowledge graph that encompasses distinguishing constituent elements. Through a graph-based keyword selection method, we automatically extract words highly relevant to the constituent elements in the knowledge graph, thus forming a word bag. Subsequently, to make the model pay more attention to the distinguishing information for each charge within the context, we propose a multi-attention supervision method. Specifically, we expand the attention mechanism and introduce a new loss function with attention

[†] Corresponding author.



Charge	Fact description
Snatch	Defendant XX forcibly pulled off the gold necklace on Moumou's neck while the victim Moumou was not prepared, and the market value of the robbed gold necklace was RMB 63,202.
Robbery	The defendant XX beat the victim Moumou and robbed the victim of a gold necklace. The market value of the robbed gold necklace was RMB 1,060.91.
Fraud	Defendant XX in the name of communication, after gaining trust, defendant XX fabricated various reasons to defraud victim Moumou of an Apple computer, whose market value is RMB 5,450
Theft	The defendant XX stole an Apple computer of the victim Moumou. The market value of the Apple computer is RMB 5,000.

🟥 all charges share 🟦 some charges share 🟨 unique words of each charge

Figure 1: Confusing charges in real legal cases. Red words indicate the words all charges share, blue words indicate the words some charges share, and the yellow highlighted words indicate the unique words of each charge.

supervision through words in the word bag.

To verify the effectiveness of our method, we construct the confusing charge dataset by selecting easily confusing charges based on real-world data. The comparison with numerous powerful baselines demonstrates the effectiveness of our method, as it outperforms many strong baselines. Ablation experiments show that using a legal knowledge graph with constituent elements can enable the model to learn more distinguishing knowledge about the confusion charges. The multi-attention supervision can help the model focus on distinguishing information in the context. It is worth noting that we are the first to use a legal knowledge graph with constituent elements to assist in charge prediction.

In summary, we make the following contributions:

- We investigate the task of confusing charge prediction by taking the domain knowledge into consideration.
- We propose a novel **F**rom **G**raph to **W**ord **B**ag (FWGB) approach. Specifically, we construct an expert knowledge graph with constituent elements and then form the word bag, combining multi-attention supervision to guide the model in distinguishing between confusing charges.
- We construct the confusing charge dataset from real-world data. Our experiments evaluate the effectiveness of our proposed method. We make the code and dataset publicly available [1] for reproducibility.

2. Related Work

2.1. AI and Law

AI and law is an emerging interdisciplinary field of law and computer science. Currently, several scholars focus on the regulation of AI by law (Wachter et al., 2021), while others choose to study the application of AI techniques in the field of law. The most studied tasks of the applications are legal judgment prediction(Hachey and Grover, 2006; Lyu et al., 2022; Liu et al., 2023; Wu et al., 2023), legal question answering(Taniguchi and Kano, 2017), legal case retrieval(Xiao et al., 2019), legal information extraction(Ji et al., 2020) and legal summarization(Bhattacharya et al., 2019). Legal judgment prediction aims to provide legal consequences, including the charges, prison terms, and so on, for professionals to lighten their workload or for laymen to learn about the case they are concerned about. Our work focuses on confusing charge prediction, which is one of the aspects of legal judgment prediction.

2.2. Charge Prediction

Charge prediction is a subtask of legal judgment prediction that takes fact descriptions as the input of the model and charges as the output of the model. Early work focused on predicting charges through artificial intelligence analysis of charge features (Mochales and Moens, 2009) or through manually designed methods (Lin et al., 2012). However, due to the large amount of feature engineering of these methods, Some researchers proposed leveraging the legal system's structure or incorporating emerging technologies like graph neural networks to enhance task performance (Zhong et al., 2018; Yue et al., 2021b; Xu et al., 2020; Kang et al., 2019; Yue et al., 2021a; Dong and Niu, 2021). Simultaneously, with pretrained models similar to BERT (Devlin et al., 2019) achieving outstanding performance in many classification tasks, some models specifically pretrained for the legal domain have also been introduced (Xiao et al., 2021; Cui et al., 2020). Though this task has been explored for a long time, confusing charge prediction still needs to be improved. An et al. (2022) define

[1] https://github.com/LIANG-star177/FWGB

confusing charges: If two charges differ in only one constitutive element, they are considered confusing charges to each other. In this work, We enhance confusing charge prediction by introducing a word bag formed by the knowledge graph with constituent elements, coupled with a multi-attention mechanism for model supervision. Moreover, Our model gets interpretability from inherent legal knowledge, allowing it to make predictions in a lawyer's way.

3. Method

As illustrated in Fig. 2, our method FGWB comprises three key components: Charge predictor, Word bag former, and Multi-attention supervisor.

3.1. Word Bag former

3.1.1. Using Constituent Elements to Form Expert Knowledge Graphs

One legal domain knowledge frequently used in real legal scenarios is constituent elements. Constituent elements refer to the preconditions that a certain behavior should have to be evaluated by law. In brief, when a person's behavior in life meets the constituent elements of the law, the person may assume legal responsibility. For example, constituent elements of fraud are: (1) The suspect commits a fraudulent act; (2) The victim falls into a wrong understanding, and so on. However, expert domain knowledge such as constituent elements is too obscure for laymen. Inspired by Bi et al. (2022), we construct the knowledge graph to use constituent elements easily, as shown in Fig. 3. Specifically, we use a set of charges as nodes and employ the constituent elements as the connections between nodes. To allow people to clearly understand significant features, we downplay some elements in the graph. For instance, the constituent elements of fraud contain five while theft contains different two. But legal experts can distinguish the two crimes only relying on the differences in the disposal act, which is indicted through "fraudulent act" and "stealing act".

3.1.2. Graph-based Keyword Selection

To make the model focus on differentiating information, a crucial prerequisite is to identify distinguishing keywords in fact descriptions, and thus address the issue of charge confusion. A simple method to find keywords is based on data statistics. However, it has two issues: (1) Different criminal charges may share a set of common keywords, and focusing on these words does not resolve the confusion problem. (2) Some words may not have a genuine relationship with the criminal charge, leading to misconceptions.

We notice that all the constituent elements along the path from the starting node to the leaf nodes in the knowledge graph can comprehensively describe a criminal charge. For example, in Fig. 3, the crime of *Snatch* in the graph includes the *violence* and *violence against people* constituent elements. Therefore, to find actual keywords, we use the constituent elements in the knowledge graph to select keywords obtained from data statistics.

For a given criminal charge i, we first use the data statistics method to filter out a candidate keyword set $C_i^{'}$. We then obtain the constituent elements set R_i of criminal charge i from the knowledge graph. We feed both the candidate keywords and the constituent elements into a legal pretrained model f_θ to obtain their corresponding vector representations. For each word in $C_i^{'}$, we calculate its cosine similarity with each word in R_i. If the average similarity between the word and the constituent element set exceeds the threshold η, we consider that word as a keyword of true keyword set C_i.

$$\frac{\sum_{w \epsilon C_i^{'}, r \epsilon R_i} sim(f_\theta(w), f_\theta(r))}{|R_i|} > \eta \rightarrow w \epsilon C_i \tag{1}$$

Subsequently, we filter out words that genuinely characterize the constituent elements, resulting in the final word bag $B = \{C_1, ..., C_N\}$, where N is the number of charge labels. It's worth noting that the automatic formation of the word bag is independent of the model and only needs to be executed once.

3.2. Charge Predictor

3.2.1. Encoder

To assess the generality of our approach, we implemented the attention mechanism with supervision on two different encoders. Specifically, we employed LSTM (Liu and Guo, 2019) trained from scratch and Electra (Cui et al., 2020), which was pretrained on legal data. When a L-length word sequence x=$\{x_1, x_2, ..., x_L\}$ is put into the encoder, every word $x_i \in$ x is converted to its hidden state h_i according to the following formulas:

$$\{h_1, h_2, ..., h_L\} = Encoder\{x_1, x_2, ..., x_L\} \tag{2}$$

3.2.2. Multi-attention Mechanism

Because the keywords in the word bag contain the key information that determines the charge prediction, we naturally think of using the attention mechanism to let the model pay attention to the key information. To get independent attention, we introduce a multi-attention mechanism on top of the basic attention mechanism. Here, we introduce the acquisition of the keyword attention matrix, and

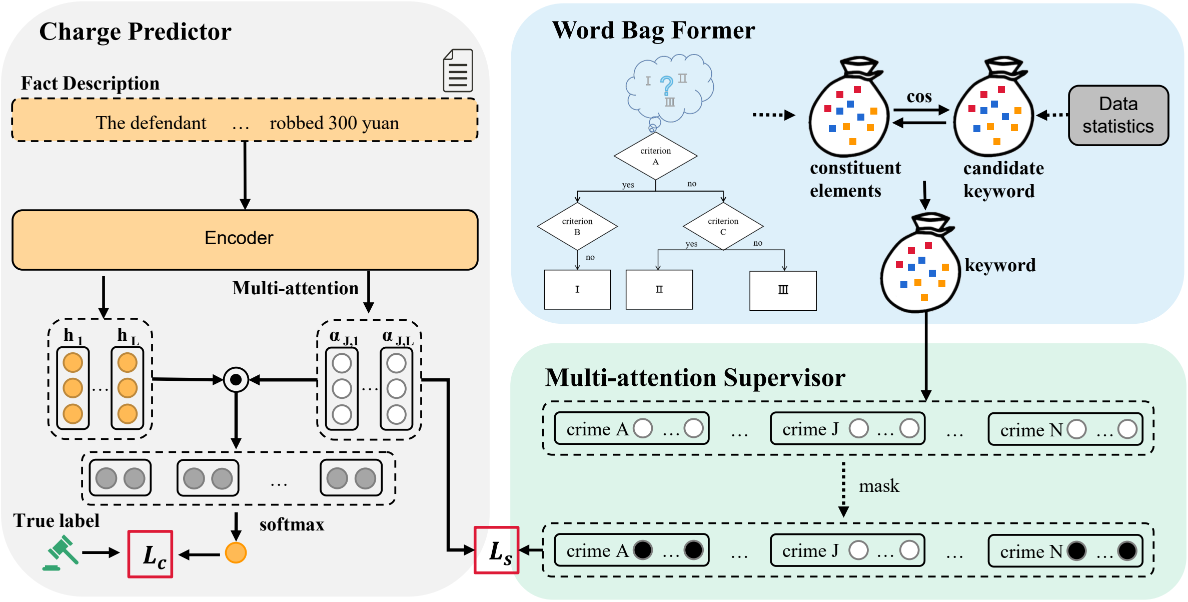


Figure 2: Structure of Our Model. The charge predictor uses LSTM to encode fact descriptions, employs a multi-attention mechanism for label-independent attention scores, and derives probability distributions for each label. The word bag former transforms expert knowledge graphs into prerequisites, selecting genuine keywords from statistical data to create a word bag. The Multi-attention supervisor assumes high attention values for label-related keywords, masking out irrelevant ones to guide the attention mechanism. Here, L_c is the loss of classification, and L_s is the loss associated with attention supervision.

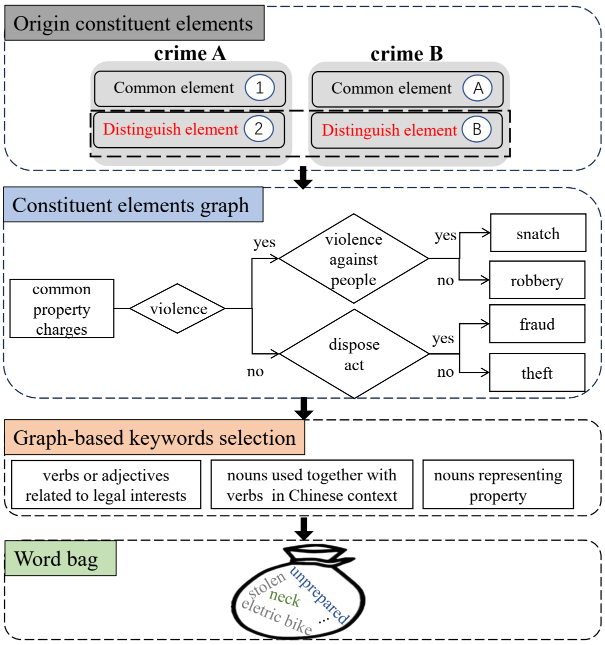


Figure 3: Construction and utilization of expert knowledge graphs.

the supervision methods are implemented through training loss functions in Sec. 3.3.

Attention. Firstly, we input the hidden state into the single-layer neural network to obtain a vector, then multiply the transpose of the vector and the context vector, and obtain the i-th token's attention weight after softmax normalization. The formula is shown as follows:

$$a_i = \frac{exp(tanh(Wh_i + b)^T u)}{\sum_i^L exp(tanh(Wh_i + b)^T u)} \tag{3}$$

where h_i is the hidden state, W, b are trainable parameters; u is the context vector. $a = \{a_1, ..., a_i, ..., a_L\}$ is attention sequence.

Multi-attention. Traditional attention pays attention to every word in the word bag while in legal practice, not all words relate to a certain charge. Thus, we propose the idea to use multi-attention to meet the needs of legal practice.

Given the Word bag $B = \{C_1, ..., C_N\}$ contains a total of R words, among which there are N charges. In order to independently compute the attention for each charge, we expand the context vector u to N dimensions, corresponding to the number of charges. For the n-th charge, $n \in [1, N]$, the corresponding attention weights are calculated as follows:

$$\alpha_{i,n} = \frac{exp(tanh(Wh_i + b)^T u_n)}{\sum_i^L exp(tanh(Wh_i + b)^T u_n)} \tag{4}$$

where u_n is the context vector corresponding to the n-th charge, $a = \{a_{1,1}, ..., a_{i,n}, ..., a_{L,N}\} \in \mathbb{R}^{L \times N}$ is attention matrix.

3.2.3. Predictor

The predictor calculates the weighted average hidden produced by the hidden state from the encoder and attention weights, for attention:

$$s = \sum_{i}^{L} \alpha_i h_i, \tag{5}$$

for multi-attention:

$$s = \sum_{i}^{L} \sum_{n}^{N} a_{i,n} h_i. \tag{6}$$

The charge predictor predicts the distribution of y overall charges through one fully connected layer and a softmax function:

$$y = softmax(Ws + b) \tag{7}$$

3.3. Training

3.3.1. Multi-attention Supervisor

To guide the model's focus toward crucial information that can distinguish between confusing charges, we design an attention supervision loss in the training stage. Similarly, we first introduce the basic attention supervision and then proceed to introduce our multi-attention supervision.

Attention Supervision Loss. First, we prepare a target attention sequence $\hat{a} = [\hat{a}_1, ..., \hat{a}_i, ..., \hat{a}_L]$, L is the length of the input. Specifically, if a word from the input is present in the word bag, we set its target attention value to 1; otherwise, it is set to 0. Then, to guide the model's focus on keywords, we calculate the loss between the target attention sequence and the calculated attention values $a = [a_1, ..., a_i, ..., a_L]$ as follows:

$$\mathcal{L}_s = -\sum_{i=1}^{L} [\hat{a}_i log(a_i) + (1 - \hat{a}_i) log(1 - a_i)] \tag{8}$$

Multi-attention Supervision Loss. To enable the model to independently direct its attention to distinct key information when dealing with various criminal charges, we supervise the multi-attention values by a loss function. Here, we establish a target attention as an $N * L$ matrix, where $\hat{a}_{i,n}$ represents the attention value of the i-th word corresponding to the n-th criminal charge label. Similarly, the calculated attention is also an $N*L$ matrix. Although the dimensions of these attention matrices are $N * L$, we mask the $(N - 1) * L$ attention weights that do not belong to the current charge label according to the true label:

$$a_i = \mathrm{MASK}(n)(a_{1,1}, ..., a_{i,n}, ..., a_{L,N}) = a_{i,n} \tag{9}$$

Where $\mathrm{MASK}(n)$ indicates that the current sample is labeled as n, only the attention value $a_{i,n}$ is retained. That's to say, after the masking operation, the matrix we need to supervise remains a sequence, and the dimension is the same as that of the traditional attention supervision, so the loss function $\mathcal{L}_s$ is also the same as Eq. 8.

Type	Property Set	Drug Set
# Train set	34529	11391
# Valid set	3836	1266
# Test set	4000	2000
Avg. # Tokens in Fact	339	347

Table 1: Data Set Collection

Property Set	Number	Drug Set	Number
Theft	28535	DS	5345
Fraud	8426	PVFDU	4789
Robbery	1048	IPOD	1668
Snatch	356	DT	855

Table 2: Label Distribution

3.3.2. Total Loss

The total loss of our model contains two parts: $\mathcal{L}_s$ and $\mathcal{L}_c$. $\mathcal{L}_s$ is the loss function to supervise the attention of keywords, while $\mathcal{L}_c$ is the loss function to minimize the cross entropy between the ground-truth y and predicted charge label $\hat{y}$ as follow:

$$\mathcal{L}_c = -y ln \hat{y} \tag{10}$$

The total loss is the sum of $\mathcal{L}_s$ and $\mathcal{L}_c$, λ is the adjustment coefficient for attention supervision:

$$\mathcal{L} = \mathcal{L}_c + \lambda \cdot \mathcal{L}_s \tag{11}$$

4. Experiment

4.1. Dataset Description

We collect our data from 12309 China Procuratorate Website [2]. Considering that our study focuses on confusing charges in real law scenes, we choose common property charges, including *Theft*, *Fraud*, *Snatch*, and *Robbery*, which are easily confused. After disposing of accusations with garbled or incomplete contents and multiple charges or multiple defendants, we finally got 38365 cases to form a data set. Further, we observe the distribution of charges is imbalanced. The number of *Theft* is 80 times that of *Snatch*, which indicates the imbalance between charges. To fairly assess the model's performance, we additionally construct a balanced test set from CAIL2018[3] (Xiao et al.,

[2] https://www.12309.gov.cn

[3] http://cail.cipsc.org.cn/index.html

Method	Property Set					Drug Set				
	Ma-P	Ma-R	Ma-F	Acc	Ma-F*	Ma-P	Ma-R	Ma-F	Acc	Ma-F*
TextRNN	0.789	0.784	0.769	0.783	0.769	0.913	0.910	0.909	0.909	0.895
LSTM	0.857	0.815	0.808	0.814	0.798	0.914	0.901	0.909	0.924	0.914
TextCNN	0.846	0.799	0.776	0.800	0.743	0.903	0.880	0.878	0.884	0.908
DPCNN	0.882	0.864	0.856	0.865	0.896	0.931	0.934	0.927	0.937	0.947
C3VG	0.882	0.862	0.860	0.868	0.985	0.920	0.920	0.920	0.914	0.939
Electra	0.903	0.889	0.881	0.888	0.892	0.936	0.930	0.928	0.928	0.943
Topjudge	0.891	0.877	0.874	0.878	0.886	0.926	0.923	0.922	0.917	0.940
LADAN	0.905	0.893	0.892	0.895	0.908	0.933	0.930	0.929	0.940	0.966
NeurJudge	0.907	0.897	0.902	0.905	0.917	0.939	0.940	0.935	0.950	0.968
R-Former	0.905	0.895	0.894	0.901	0.918	0.931	0.948	0.941	0.951	0.970
FGWB (LSTM)	0.888	0.880	0.876	0.880	0.880	0.934	0.933	0.930	0.929	0.942
FGWB (Electra)	**0.923**	**0.925**	**0.924**	**0.925**	**0.928**	**0.957**	**0.955**	**0.956**	**0.955**	**0.979**

Table 3: Experiment results for property charges and drug charges, the best is **bolded** and the second best is underlined.

2018). Specifically, we set the number of cases for each charge in the test set to 1000. Then for the rest of the cases, we randomly divided them into a training set and validation set at the ratio of 9:1. To assess the model's ability to handle imbalanced distributions, we used the validation set as an imbalanced test set for comparison.

To verify the generality, we apply the same processing steps to another cluster of common drug charges to get the second data set, containing *Drugs Selling (DS)*, *Providing Venues For Drug Users (PVFDU)*, *Illegal Possession Of Drugs (IPOD)* and *Drugs Transportation (DT)* charges. we set the number of cases for each charge in the test set to 500. The details of the dataset are shown in Tab. 1 and Tab. 2.

4.2. Baselines

To evaluate the performance and interpretability of our model, we implemented several baselines to compare these two aspects. **TextRNN** (Graves, 2013) is a traditional recurrent neural network model for text classification. **LSTM** (Zhou et al., 2015) incorporates both forward and backward information flow through LSTM units to capture contextual information effectively. **TextCNN** (Lai et al., 2015) is a traditional convolutional neural network model for text classification. **DPCNN** (Johnson and Zhang, 2017) is a low-complexity word-level deep convolutional neural network architecture for text categorization. **C3VG** (Yue et al., 2021b) is a model following a two-stage architecture which is from extraction to generation. **Electra** (Cui et al., 2020) is a pretrained model that has been adjusted to Chinese. **Topjudge** (Zhong et al., 2018) is a model that incorporates multiple subtasks and DAG dependencies into judgment prediction. **LADAN** (Xu et al., 2020) is a model that attentively extracts features from law cases' fact descriptions to distinguish confusing law articles. **NeurJudge** (Yue et al., 2021a) splits the fact description into two parts and encodes them separately. **R-Former** (Dong and Niu, 2021) formalizes LJP as a node classification problem.

To further validate our proposed FWGB, we conducted three sets of ablation experiments for each of the two encoder methods: **w/o SV** means not using attention supervision but retaining the multi-attention configuration. **w/o Multi-Attn** uses traditional attention mechanisms and applies attention supervision. **w/o KG** employs the multi-attention supervision mechanism but does not use the knowledge graph to filter the word bag, instead using a word bag composed of high-frequency words.

4.3. Experimental Settings

Our experiment is carried out on two V100 GPUs, and all the baseline models adopt the settings in their original papers. For the models without pre-trained models, we adopt Gensim (Řehůřek and Sojka, 2010) on the training corpus to initialize the word embeddings, which are in the dimension of 300. For samples with long input, we truncate them to 512 tokens. We set the coefficient λ to the best-performing 0.7 and explore the impact of different values of λ on performance.

To evaluate the performance of the prediction, we calculate the Macro precision (Ma-P), Macro recall(Ma-R), and Macro F1 score (Ma-F) and accuracy (Acc). Ma-F* is used to represent the macro-F1 score tested on the imbalanced test set.

4.4. Experiment Results

Result of Charge Prediction: From Tab. 3 of Property Set, we observe that: (1) FGWB (Electra) model significantly and consistently outperforms all the baselines. The result proves that FWGB

Method	property charges					drug charges				
	Ma-P	Ma-R	Ma-F	Acc	Ma-F*	Ma-P	Ma-R	Ma-F	Acc	Ma-F*
FGWB (LSTM)	**0.888**	**0.880**	**0.876**	**0.880**	**0.880**	**0.934**	**0.933**	**0.930**	**0.929**	0.942
w/o SV	0.862	0.839	0.857	0.848	0.872	0.914	0.897	0.911	0.904	0.924
w/o Multi-Attn	0.866	0.842	0.861	0.854	0.862	0.918	0.908	0.917	0.921	0.931
w/o KG	0.874	0.869	0.863	0.865	0.872	0.930	0.923	0.921	0.919	**0.943**
FGWB (Electra)	**0.923**	**0.925**	**0.924**	**0.925**	**0.938**	**0.957**	**0.955**	**0.956**	0.955	**0.979**
w/o SV	0.901	0.892	0.907	0.897	0.905	0.942	0.904	0.920	0.919	0.957
w/o Multi-Attn	0.907	0.897	0.917	0.902	0.911	0.948	0.927	0.934	**0.959**	0.965
w/o KG	0.912	0.902	0.910	0.895	0.919	0.950	0.948	0.935	0.948	0.977

Table 4: Ablation Experiment Results

NeurJudge	Robbery	Snatch	Theft	Fraud
Robbery	709	7	0	2
Snatch	48	836	1	2
Theft	108	41	989	6
Fraud	2	10	10	992

Electra	Robbery	Snatch	Theft	Fraud
Robbery	888	244	1	0
Snatch	4	355	2	0
Theft	101	376	982	10
Fraud	7	25	15	990

FGWB (SV)	Robbery	Snatch	Theft	Fraud
Robbery	872	71	0	0
Snatch	68	862	2	0
Theft	35	34	**997**	6
Fraud	25	33	1	994

FGWB (MSV)	Robbery	Snatch	Theft	Fraud
Robbery	**902**	29	0	0
Snatch	181	**942**	1	0
Theft	45	98	996	3
Fraud	5	37	3	**995**

Table 5: Confusing Matrices for Different Models. “SV" stands for using attention supervision, while “MSV" stands for using multi-attention supervision. They are both implemented on Electra.

effectively draws the model's attention to specific parts of fact descriptions that can help to make the right predictions. (2) Compared with baselines without attention supervision mechanism, FGWB (Electra) gets 1.6% more scores on Ma-P, 0.020 more scores on recall, 2.8% more scores on Ma-R, 2.2% more scores on Ma-F and 2.0% more scores on Acc than the best-performed baseline (NeurJudge). (3) Compared with FGWB (LSTM), FGWB (Electra) gets 4.8% more scores on Ma-F. This indicates that pretrained models acquire knowledge that is beneficial for helping the model understand input information.

Comparing the result of the imbalanced test set (valid set) and the balanced test set, we conclude that: (1) Ma-F* values are higher than those of Ma-F, which indicates that an imbalanced data set is a huge challenge for charge prediction. (2) By comparing the difference between Ma-F* and Ma-F for each method, we find that the FGWB (Electra) is the most suitable model for an imbalanced data set with the fact that it gets the least decrease of 0.4%.

From Tab. 3 of Property Set, we get the similar observations: (1) Ma-P, Ma-R, Ma-F and Acc value for FGWB (Electra) win. the best-performed baseline R-Former, which is a corroboration for the validity of our method. (2) Our FWGB method exhibits significant improvements in both implementation approaches (LSTM and Electra). In summary, our approach performs well on the dataset containing drug-related confusing charges, highlighting its generalizability and applicability to various situations.

Result of Confusing Matrices: From the confusing matrices of different models shown in Tab. 5, We can conclude that: (1) Compared FGWB with the model without attention supervision mechanism(LSTM, NeurJudge), methods that use attention supervision are generally effective at improving the model's performance for confusing charges. On low-frequency charges, FGWB (MSV) gets 106 more right predictions on *Snatch*. This indicates that our approach can better handle label imbalance compared to other methods. (2) The confusing matrix of FGWB (MSV) outperforms that of FGWB (SV) on all charges, which shows the attention trained under the supervision of legal knowledge is better than the traditional attention.

Result of Ablation Experiment: From Tab. 4, We conclude that: (1) “w/o SV" results in a significant decrease in performance for both implementation methods compared to FGWB suggesting that attention supervision is effective in enhancing the model's ability to distinguish between confusing charges. Additionally, the decrease is more pronounced for LSTM, which is due to the inferior performance of LSTM compared to Electra. This

Model	LSTM-attn	FGWB (SV)	FGWB (MSV)
Attention distribution	At about 02:02 on June 18, 2019, when the defendant Zhao Moumou went to the toilet in Tuqiao Village, Guandu District, Kunming City, he saw the victim Li Moumou put a red OPPO R17 mobile phone at his feet when he went to the toilet. Defendant Zhao XX took advantage of the victim Li XX's unpreparedness and snatched away the mobile phone that was at his feet. The value of the OPPO R17 mobile phone identified as involved in the case was 2190 yuan and has been returned to the victim.	At about 02:02 on June 18, 2019, when the defendant Zhao Moumou went to the toilet in Tuqiao Village, Guandu District, Kunming City, he saw the victim Li Moumou put a red OPPO R17 mobile phone at his feet when he went to the toilet. Defendant Zhao XX took advantage of the victim Li XX's unpreparedness and snatched away the mobile phone that was at his feet. The value of the OPPO R17 mobile phone identified as involved in the case was 2190 yuan and has been returned to the victim.	At about 02:02 on June 18, 2019, when the defendant Zhao Moumou went to the toilet in Tuqiao Village, Guandu District, Kunming City, he saw the victim Li Moumou put a red OPPO R17 mobile phone at his feet when he went to the toilet. Defendant Zhao XX took advantage of the victim Li XX's unpreparedness and snatched away the mobile phone that was at his feet. The value of the OPPO R17 mobile phone identified as involved in the case was 2190 yuan and has been returned to the victim.
prediction	**Theft**	**Theft**	**Snatch**

Figure 4: Attention distribution from different models for a *Snatch* case. “SV" stands for using attention supervision, while “MSV" stands for using multi-attention supervision. They are both implemented on LSTM.

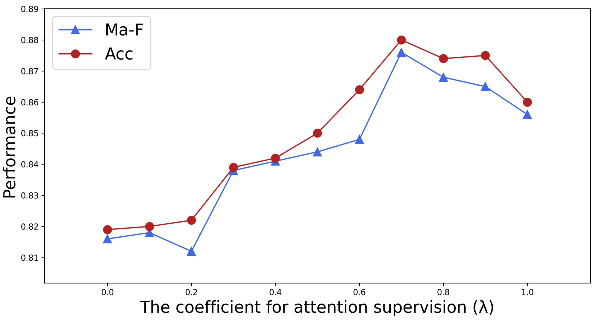


Figure 5: Model performance by the coefficient λ for attention supervision.

implies that attention supervision has a more substantial impact on improving LSTM's performance. (2) When it comes to “w/o Multi-Attn", there is a significant performance decrease compared to FWGB. This implies that the multi-attention mechanism successfully provides separate attention spaces for each criminal charge, avoiding mutual interference and achieving better supervision results. (3) When it comes to “w/o KG", there is a performance decrease compared to FWGB. This highlights the significance of knowledge graph assistance in constructing the word bag. The knowledge graph, summarized by legal experts, retains crucial elements that can differentiate between criminal charges. Filtering keywords from the knowledge graph's components is, in fact, an effective form of external knowledge incorporation, aiding the model in learning expert knowledge to distinguish between confusing charges.

Performance by the Coefficient for Attention Supervision λ: We investigate the impact of changing the coefficient λ, which controls the attention supervision loss, on the model FGWB (LSTM) in terms of Ma-F and Acc metrics. As shown in Fig. 5, as λ gradually increases, the model's performance exhibits an initial improvement followed by a decline, reaching its optimal performance at $\lambda = 0.7$. This outcome indicates that attention supervision is effective in confusing charge prediction.

4.5. Case Study

Fig. 4 shows the heat maps of a real *Snatch* case when predicting the charges by the LSTM-attn model, FGWB (SV), and FGWB (MSV) respectively. Words with a deeper background color have higher attention weights. We observe that: (1) In the LSTM-attn model, we observe that it pays attention to many irrelevant details, such as the crime time ('June 18, 2019'; '02:02'). Due to the model's attention not being focused on crucial information, it made a prediction error, classifying the crime as *Theft*. (2) FGWB (LSTM+SV) identifies keywords related to the charge like *value*, *identified*, and *snatch away*. However, it assigns incorrect weights to these words, leading to a false theft charge prediction. This occurs because when all charges are supervised with a shared attention mechanism, they tend to influence each other, emphasizing information they have in common while potentially neglecting differentiating details. (3) FGWB (MSV) places emphasis on *unpreparedness* and increases the attention weight on *snatch away*, both of which are constituent elements of *Snatch* associated with *violence against people*. As a result, FGWB, following the logic akin to that of a judge, correctly predicts *Snatch*.

5. Conclusion

In this paper, we address the challenging task of confusing charge prediction within the legal domain. Existing charge prediction methods often fall short of effectively distinguishing between easily confused charges. Our innovative approach, the "From Graph to Word Bag (FWGB)" model, leverages constituent elements within a legal knowledge graph to enhance predictive accuracy. We introduce a multi-attention supervision mechanism to ensure that the model focuses on critical information within the context, leading to substantial improvements in performance. Through extensive experimentation, we have validated the effectiveness of our approach using real-world judicial documents.

6. Ethical Discussion

Automatic charge prediction is a sensitive field of AI. While our goal is to surpass the performance of existing approaches, it's essential to acknowledge that these technologies are not yet ready for practical implementation. Legal cases often contain sensitive personal information, highlighting the importance of protecting privacy when processing datasets (Xu et al., 2023). Ensuring the ethical deployment of artificial intelligence systems in legal decision-making requires strict safeguards, transparency, and sustained ethical considerations to protect individual rights and maintain trust in the legal system. Additionally, exploring more suitable encoding methods to mitigate biases introduced by data distributions can promote fairness (Wang et al., 2017).

7. Acknowledgements

This work was supported in part by National Key Research and Development Program of China (2022YFC3340900), National Natural Science Foundation of China (No. 62376243, 62037001, U20A20387), the StarryNight Science Fund of Zhejiang University Shanghai Institute for Advanced Study (SN-ZJU-SIAS-0010) and Project by Shanghai AI Laboratory (P22KS00111).

8. Bibliographical References

Zhenwei An, Quzhe Huang, Cong Jiang, Yansong Feng, and Dongyan Zhao. 2022. Do charge prediction models learn legal theory?

Paheli Bhattacharya, Kaustubh Hiware, Subham Rajgaria, Nilay Pochhi, Kripabandhu Ghosh, and Saptarshi Ghosh. 2019. A comparative study of summarization algorithms applied to legal case judgments. In *Advances in Information Retrieval*, pages 413–428, Cham. Springer International Publishing.

Sheng Bi, Zafar Ali, Meng Wang, Tianxing Wu, and Guilin Qi. 2022. Learning heterogeneous graph embedding for chinese legal document similarity. *Knowledge-Based Systems*, 250:109046.

Shi Cheng, Yuhui Shi, and Quande Qin. 2012. Particle swarm optimization based semi-supervised learning on chinese text categorization. *2012 IEEE Congress on Evolutionary Computation, CEC 2012*.

Junyun Cui, Xiaoyu Shen, Feiping Nie, Zheng Wang, Jinglong Wang, and Yulong. Chen. 2022. A survey on legal judgment prediction: Datasets, metrics, models and challenges. Preprint at `https://arxiv.org/abs/2204.04859`.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for chinese natural language processing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 284–298, . Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Qian Dong and Shuzi Niu. 2021. Legal judgment prediction via relational learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 983–992.

Alex Graves. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.

Ben Hachey and Claire Grover. 2006. Extractive summarisation of legal texts. *Artificial Intelligence and Law*, 14(4):305–345.

Zikun Hu, Xiang Li, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. 2018. Few-shot charge prediction with discriminative legal attributes. In Emily M. Bender, Leon Derczynski, and Pierre Isabelle, editors, *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 487–498. Association for Computational Linguistics, .

Fuqiong Huang, Mei Li, Yuchuan Ma, Yanyan Han, Lei Tian, Wei Yan, and Xiaofan Li. 2017. Studies on earthquake precursors in china: A review for recent 50 years. *Geodesy and Geodynamics*, 8(1):1–12.

Donghong Ji, Peng Tao, Hao Fei, and Yafeng Ren. 2020. An end-to-end joint model for evidence information extraction from court record document. *Information Processing and Management*, 57(6):102305.

Rie Johnson and Tong Zhang. 2017. Deep pyramid convolutional neural networks for text categorization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 562–570.

Liangyi Kang, Jie Liu, Lingqiao Lfriu, Qinfeng Shi, and Dan Ye. 2019. Creating auxiliary representations from charge definitions for criminal charge prediction. *CoRR*, abs/1911.05202.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Wan-Chen Lin, Tsung-Ting Kuo, Tung-Jia Chang, Chueh-An Yen, Chao-Ju Chen, and Shou-de Lin. 2012. Exploiting machine learning models for Chinese legal documents labeling, case classification, and sentencing prediction. In *International Journal of Computational Linguistics & Chinese Language Processing, Volume 17, Number 4, December 2012-Special Issue on Selected Papers from ROCLING XXIV*.

Gang Liu and Jiabao Guo. 2019. Bidirectional lstm with attention mechanism and convolutional layer for text classification. *Neurocomputing*, 337:325–338.

Yifei Liu, Yiquan Wu, Yating Zhang, Changlong Sun, Weiming Lu, Fei Wu, and Kun Kuang. 2023. ML-LJP: multi-law aware legal judgment prediction. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 1023–1034. ACM.

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*, pages 2727–2736, Copenhagen, Denmark. Association for Computational Linguistics.

Yougang Lyu, Zihan Wang, Zhaochun Ren, Pengjie Ren, Zhumin Chen, Xiaozhong Liu, Yujun Li, Hongsong Li, and Hongye Song. 2022. Improving legal judgment prediction through reinforced criminal element extraction. *Information Processing and Management*, 59(1):102780.

Raquel Mochales and Marie-Francine Moens. 2009. Argumentation mining: The detection, classification and structure of arguments in text. *Proceedings of the International Conference on Artificial Intelligence and Law*, pages 98–107.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. `http://is.muni.cz/publication/884893/en`.

Ryosuke Taniguchi and Yoshinobu Kano. 2017. Legal yes/no question answering system using case-role analysis. In *New Frontiers in Artificial Intelligence*, pages 284–298, Cham. Springer International Publishing.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. 2021. Why fairness cannot be automated: Bridging the gap between eu non-discrimination law and ai. *Computer Law and Security Review*, 41:105567.

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 2019, Paris, France, July 21-25, 2019*, pages 325–334. ACM.

Xiao Wang, Peng Cui, Jing Wang, Jian Pei, Wenwu Zhu, and Shiqiang Yang. 2017. Community preserving network embedding. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 203–209. AAAI Press.

Yiquan Wu, Siying Zhou, Yifei Liu, Weiming Lu, Xiaozhong Liu, Yating Zhang, Changlong Sun, Fei Wu, and Kun Kuang. 2023. Precedent-enhanced legal judgment prediction with LLM and domain-model collaboration. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12060–12075, Singapore. Association for Computational Linguistics.

Chaojun Xiao, Xueyu Hu, Zhiyuan Liu, Cunchao Tu, and Maosong Sun. 2021. Lawformer: A pre-trained language model for chinese legal long documents. *AI Open*, 2:79–84.

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.

Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Tianyang Zhang, Xianpei Han, Zhen Hu, Heng Wang, and Jianfeng Xu. 2019. Cail2019-scm: A dataset of similar case matching in legal domain.

Jimin Xu, Nuanxin Hong, Zhening Xu, Zhou Zhao, Chao Wu, Kun Kuang, Jiaping Wang, Mingjie Zhu, Jingren Zhou, Kui Ren, Xiaohu Yang, Cewu Lu, Jian Pei, and Harry Shum. 2023. Data-driven learning for data rights, data pricing, and privacy computing. *Engineering*, 25:66–76.

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*, pages 3086–3095, Online. Association for Computational Linguistics.

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

Linan Yue, Qi Liu, Binbin Jin, Han Wu, Kai Zhang, Yanqing An, Mingyue Cheng, Biao Yin, and Dayong Wu. 2021a. Neurjudge: A circumstance-aware neural framework for legal judgment prediction. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 973–982.

Linan Yue, Qi Liu, Han Wu, Yanqing An, Li Wang, Senchao Yuan, and Dayong Wu. 2021b. Circumstances enhanced criminal court view generation. In *SIGIR*, pages 1855–1859.

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*, pages 3540–3549.

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*, pages 5218–5230, Online. Association for Computational Linguistics.

Chunting Zhou, Chonglin Sun, Zhiyuan Liu, and Francis Lau. 2015. A c-lstm neural network for text classification. *arXiv preprint arXiv:1511.08630*.

9. Language Resource References

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.