Event Extraction for Criminal Legal Text

Qingquan Li[1], Qifan Zhang[1], Junjie Yao[1]*, Yingjie Zhang[2]
[1] East China Normal University, Shanghai
[2] Shanghai Electric Vehicle Public Data Collecting, Monitoring and Research Center, Shanghai
10154507139@stu.ecnu.edu.cn, qifanwz@gmail.com, junjie.yao@cs.ecnu.edu.cn, zhangyingjie@shevdc.org

***Abstract*—This paper concerns with the actual problems in the legal work. We apply event extraction technology to the case description part in the Chinese legal text. We define the event type, event argument and event argument role of the larceny case, and construct a larceny case event extraction dataset through data annotation. We divide event extraction into two steps: event trigger word and argument joint extraction and event argument role assignment. We use BERT to obtain Chinese character vectors, use the BiLSTM-CRF model for extraction at the first step, and combine additional features with the extraction results of the first step, then input them to the CRF model of the second step to obtain an improvement in extraction result. We display the extracted event information in time series to realize the litigation visualization. We format Chinese time expressions, sorts the event information in tine series, and develops a Web application to display the timeline of event information.**

***Index Terms*—Chinese legal text, event dataset construction, event extraction, litigation visualization**

I. Introduction

Recently, more and more legal documents and related judicial data have been released to the public. With the increasing complex legal suits and judgements, these legal data becomes a valuable source to push forwards the intelligent legal services and judgement assistance. Recently, there are some tentative work in contradictions and disputes analytics and mining from judgement documents. For example, the combination of the affair graph and the knowledge graph realizes the vision of Wisdom Court. The usage of the graph empowerment to structure the facts involved, achieves the functions of class recommendation, document abstraction, and document generation [1].

Under the hood of these justice data analytics, a suitable structured extraction and representation is the essential part. Event graph is involved in the legal document extraction and understanding. Event extraction is adapted to associate and manage multi-source data records and annotate them with suitable categories or attributes. The relationships between event argument and attributes are well preserved. Event graph has shown substantial improvements in the search, recommendation and inference applications, not only in web based scenarios but also some specific usages. For example, product, finance and personal assistant ones are enriched with event graphs.

*Corresponding author.
This work was supported by NSFC grant 61972151, the Fundamental Research Funds for the Central Universities, and the Open Research Fund of KLATASDS-MOE.

However, the fuzzy discussion and vague words in legal text hinder effective extraction and processing. We have encountered a lot of difficulties in applying event extraction perspective into the legal text understanding. Here in this work, we devote to effective ways to extract five types of events in larceny cases and six types of event arguments.

At the legal level, document disclosure is of great significance: for the legal institution, document disclosure provides a large number of cases and judgments for reference; to the public and society, the disclosure of documents can make judicial decisions transparent. At the same time, the publication of a large number of legal documents also facilitates the application of information technology, especially big data technology, in the legal field.

In this work, we propose a comprehensive event extraction approach. We first fetch the related legel text and update them periodically. For event elements, we follow a pre-designed template for events by ACE2005, and typically define five frequent events, including event arguments, event types, and event argument role. Then we extract the event information with the help of labeled trigger words by conditional random field models. The extracted event information is placed on visualization for clearly showing the basic facts and development of the input larceny case.

In judicial decisions, criminal facts are an important criterion for judges to judge cases. Taking the larceny case as an example, the things and money stolen by the criminal suspect, the consumption of stolen money, the sale of stolen things, and surrender or not are all important basis for the judge to convict and sentence. The judge quantifies the various behaviors and their degree of the criminal suspects, and punishes them according to the existing legal standards. However, the case descriptions in legal texts are often unstructured, narrative style language. Under high-reading work, judges are often possible to miss the case elements, which will affect the judgment of the case. If we extract the case elements manually, the cost of manpower and time is also very high. In recent years, most of the information extraction research for Chinese legal texts has focused on extracting information such as legal language or entity relationship triplets, which is slightly weaker in the systematic presentation of the case. Extracting the events and elements of the case in the legal text can make the case description clearer, which facilitates readers to quantify the degree of the case better and reduce manual costs. In 2019, Li Chuanyi et al. applied event extraction technology to Chinese legal texts for the first time [2]. But their research did not

978-1-7281-8156-1/20/$31.00 ©2020 IEEE
DOI 10.1109/ICBK50248.2020.00086

involve the generation of case timeline. Inspired by their work, we make a little improvement to their event extraction method, and display the extracted events in time series to form a case timeline.

We aim to extract and visualize the events and related elements in the description of the case in the Chinese legal text, so as to facilitate the legal workers to evaluate and judge the case. The main work of this paper includes data acquisition and annotation, training, testing and adjustment of deep learning models, and visual display of extraction results. Fig.1 shows the visualization of the event extraction results in a case.

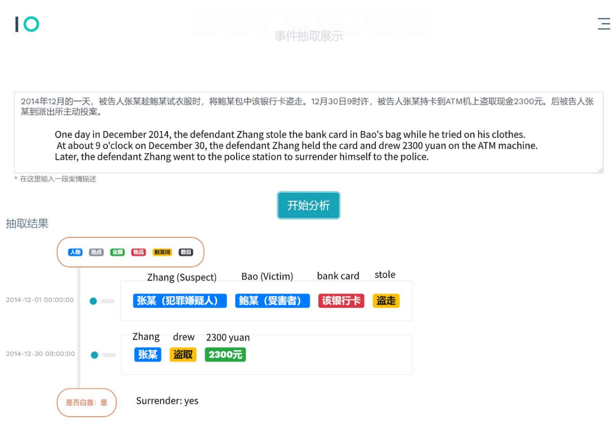

Fig. 1. Visualization of Event Extraction Result.

The contribution of this work can be summarized as follows:

- we formalize the problem of legal event extraction and present the necessary extraction pipeline.
- we demonstrate the usefulness of the proposed extraction approach and constructed a clear visualization for larceny case review.
- we utilize the joint extraction under multi-source elements and improve the event extraction performance on legal text.

II. Related Work

A. Event Extraction

The purpose of researching event extraction technology is to extract event information of interest from unstructured natural language text and display it in a structured form. The ACE conference defines event extraction tasks as: identifying specific types of events and extracting information related to the events (mainly including the type and subtype of event, trigger word, event argument and their role). Fig.2 shows an example of event extraction

Event extraction technology based on deep learning has gradually become a research hotspot. Compared with traditional machine learning methods, deep learning methods can obtain the rich text features through character vectors or word vectors, reducing the dependence on external NLP tools. At the same time, extraction can be achieved automatically without

Fig. 2. An Example of Event Extraction.

specifying extraction rules through the neural network. In 2015, Yubo Chen et al. proposed DMCNN [3]. The authors regarded the event extraction task as a two-step text multi-classification task of trigger word extraction and argument extraction, and uses DMCNN for extraction. DMCNN achieved a breakthrough in performance at that time.

Pre-trained models have also been applied to the field of event extraction. In 2019, Yang, Sen et al. applied BERT word embedding to event extraction to solve the overlapping problem of event argument roles [4]. They achieved the state-of-art performance on the ACE2005 dataset [5]. In addition, the research also proposed a method of using BERT for event extraction data enhancement.

A large part of the current research on event extraction technology is based on English. The research of Chinese event extraction technology has just started. In 2018, Lin Hongyu et al. proposed NPNs [6]. The research constructed a trigger block centered on each word to solve the problem of difficult matching of trigger words caused by the difficulty of Chinese word segmentation. In 2018, Yang Hang et al. proposed the document-level Chinese event extraction system DCFEE [7] in the financial field . The research used distant supervision to expand the dataset through the structured financial event knowledge base. By determining the central sentence of the document and looking for the supplement of event elements in the document, the research extracted events at the document level.

B. Event Graph Construction

Event graph is mentioned in various NLP applications, and the pipeline of construction of it develops rapidly in recent years. Li et al. [8] proposed a narrative event evolutionary graph to describes event evolutionary principles and patterns based on the extracted event chains. Yang and Feng et al. [9] separate the argument prediction in terms of roles and utilize pre-trained language models to propose an event extraction model, solving the roles overlap problem. After that, they take samples as prototypes for generation, including argument replacement and token rewriting. Nguyen and Grishman [10] investigate a graph convolutional neural network based on dependency trees to realize event detection. They propose a pooling method and integrate syntax into event detection.

The limitations of traditional knowledge graph are mainly reflected in two aspects. Firstly, there is few dynamic attribute in traditional knowledge graph. Traditional knowledge graph describes certain fact that is static and unchanged. In our real world, most knowledge is dynamic, typically events in various kinds of domain scenario. Knowledge itself will be corrected due to the change of external situations. Secondly, traditional knowledge graph has less applications due to the static knowledge and common structure. Traditional knowledge graph can only perform easy tasks such as what or which recorded in itself. For knowledge reasoning and prediction, traditional knowledge graph acts weak.

Actually, there is a huge demand for application in knowledge reasoning and deduction, such as financial risk control, which demands to capture external events and make sure the authenticity and accuracy of received information by knowledge reasoning, deduction and prediction based on the logical relationship of events. For marketing management, there is a need to predict the follow-up impact of some event and find or deduct the cause of the event, such as "what are the possible reasons for the sudden increase of the price of pork, and whether this situation will be continued?", etc.

The knowledge in traditional knowledge graph is static and often framed as triplets. It enhance chatbots and efficient queries but lack of dynamic information. Event graph record a whole process of the events in specific domain. It produces richer information than traditional knowledge graph for knowledge reasoning and event graph can be customized to improve the performance in domain.

C. Legal Intelligence Research

The purpose of legal intelligence research is to give machines the ability to understand legal text. In recent years, with the continuous disclosure of judicial big data represented by judge documents and the continuous breakthrough of natural language processing technology, how to apply artificial intelligence technology in the legal field and efficiency help judicial person in different case processing has gradually become a hot topic of legal intelligence research.

China AI & Law Challenge(CAIL) [1] has successfully held twice and CAIL2020 is coming soon. CAIL provides a large amount of labeled legal text, aiming to provide academic exchange platform for researchers and promote the application of language understanding and artificial intelligence technology in the field of law. CAIL proposes different tasks under legal text, such as accusation prediction, extraction of legal elements, legal cases understanding and matching, etc. In the latest CAIL2020, it even attempts judicial examination and argument mining, which walks closer to legal environment in real world.

There are a few studies on information extraction of Chinese legal texts. In 2018, Yin et al. proposed NE-Reasoner [11], which is a framework to introduce global consistency of recognized entities into Neural Reasoner over Named Entity Recognition task. Their experimental dataset contains a Chinese legal dataset. In 2019, Liu and Chen extracted the gist of Chinese judgement documents with machine learning and deep learning methods [12]. Li et al. proposed two labelling model to extract events in marriage case [2]. They try to apply the technology of event extraction to faster capture the focus of legal case but find that there is no proper definition of events that contains types of focus in the judicial field. So they propose a two-level labeling approach, solving multiple events sharing the same argument or trigger words.

Information extraction of Chinese events is limited to simple NER or event extraction task. Refining the event information and forming the context of events in the case needs to be researched.

III. Approach

A. Definition of Event Elements

Existing event extraction data sets, such as ACE2005, have a pre-designed template for events. The template defines event types, event arguments, and argument roles. We selected the larceny cases as the research object of this paper. With reference to the existing event definition of event extraction dataset, we construct a larceny case event dataset. Through reading and analyzing a large number of larceny cases, we define five frequent events, which practical impact on the sentencing work in larceny cases. Those events are Steal, Draw, Spend Money, Handle Things and Surrender. The definition of event type is shown in Table I, the definition of event argument is shown in Table II, and the definition of event argument roles is shown in Table III.

TABLE I
Definition of Event Type

Event Type	Meaning
Steal	Take something from a person or an organization secretly
Draw	Withdraw or transfer money from a stolen bank account/card
Spend Money	Spend stolen money
Handle Things	Handle stolen goods, such as selling, discarding, etc.
Surrender	The criminal surrender himself/herself

TABLE II
Definition of Event Argument

Event Argument	Meaning
Time	Time of the event
Person	Participants of the event
Location	Where the event occurred
Amount	Amount involved in the event
Thing	Things involved in the event
Number	Number of the things involved in the event

TABLE III
DEFINITION OF EVENT ARGUMENT ROLES

Event Type	Event Argument Role
Steal	Steel Time, Crime Suspect, Victim, Steal Location, Steal Thing, Steal Number, Steal Amount
Draw	Draw Time, Depositor, Draw Location, Draw Amount
Spend Money	Spend Time, Spender, Spend Location, Spend Amount
Handle Things	Handle Time, Handler, Handled Thing, Handle Amount
Surrender	Crime Suspect

B. Data Processing

The raw text data of this research comes from many data sources. The main source is the CAIL 2018 [1]. The organizer of the contest provided the largest Chinese judicial dataset at that time, containing a total of 2.6 million criminal cases published by the Supreme People's Court of China on the Chinese Judgement Document Website. We also obtained texts of criminal cases from other platforms, such as news websites.

The text data we obtained is the original text of various criminal cases in the judgment documents. In order to select texts that are consistent with the research field of our work and reduce the impact of noise on the deep learning model, data cleaning and processing is required. We select 3,000 larceny case description texts from the raw data. In order to reduce the length of a single sample and improve model performance, we split the texts into sentences. In order to remove noise samples, we remove sentences that do not contain the predifined events. Finally, we obtained 6,538 sentences containing predefined events.

For each sentence, we label the event trigger words, data arguments and their roles according to the predefined event template. We use the open source tool YEDDA [13] to label our data. The labeling scheme is BIO scheme.

C. Event Extraction

At present, the research on the event extraction task, especially based on the ACE2005 dataset, mostly regards event extraction as three sub-tasks: event trigger word extraction, event argument extraction, and event argument role assignment. Subtasks can be processed jointly. In the legal text, the trigger word of the event has the characteristics of few words and clear expressions. So the trigger word can be extracted jointly with the argument.

Through statistical analysis of out annotated event dataset, We find that the distribution of argument types and event types in the dataset is unbalanced. Fig.3 shows the distribution of event argument in our dataset. Fig.4 shows the distribution of event types in our dataset.

The unbalanced distribution of events and arguments is related to the characteristics of larceny cases. For example, there are often multiple criminal suspects or victims in larceny cases, so there are many "Person" arguments. In the larceny cases, there must be a "Steal" event, but not necessarily other events such as "Draw", "Surrender". Therefore, there are many "Steal" events.

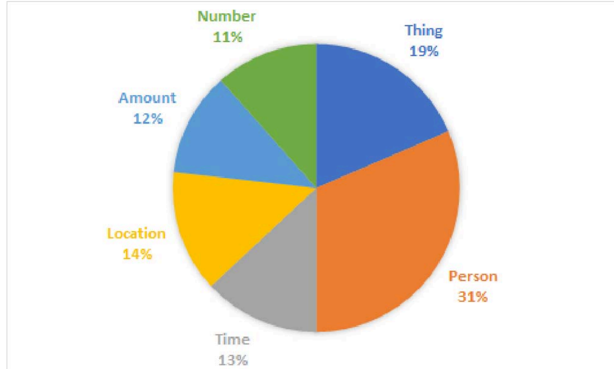

Fig. 3. Distribution of Event Argument.

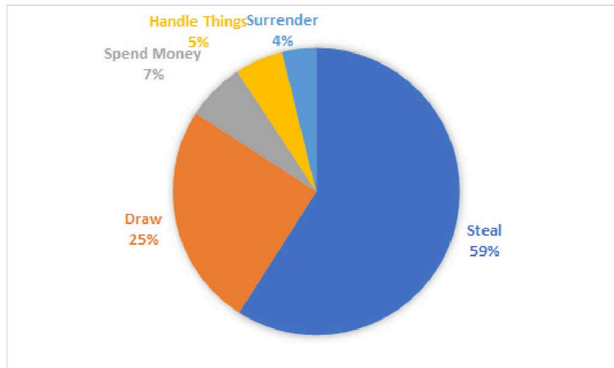

Fig. 4. Distribution of Event Types.

In order to improve the accuracy of extraction and reduce the impact of unbalanced label distribution, we divide the event extraction task of larceny cases into two steps: the joint extraction of trigger word and argument, and the assignment of event argument roles. Fig.5 shows the event extraction flow of this paper.

a) Joint Extraction of Trigger Word and Argument: In this step, the input is the text which has been split into characters. The pre-trained BERT [14] embeds the text to obtain character vectors. The character vectors are input into the BiLSTM-CRF [15] model to get the extraction result. Fig.6 shows the model structure of this step.

b) Assignment of Event Argument Roles: In this step, the arguments extracted in the first step are mapped to the argument roles defined in Table III. In other words, we determine what type of event those arguments belongs to.

The model at this step is same as the second labeling step of the two labeling model [2]. As explained in this section, the distribution of argument types and event types in the dataset we used are unbalanced. If the roles of arguments are directly assigned, a good accuracy cannot be guaranteed. Therefore, in order to improve the accuracy of extraction, in this step, the input of CRF [16] sequence labeling model is as follows:

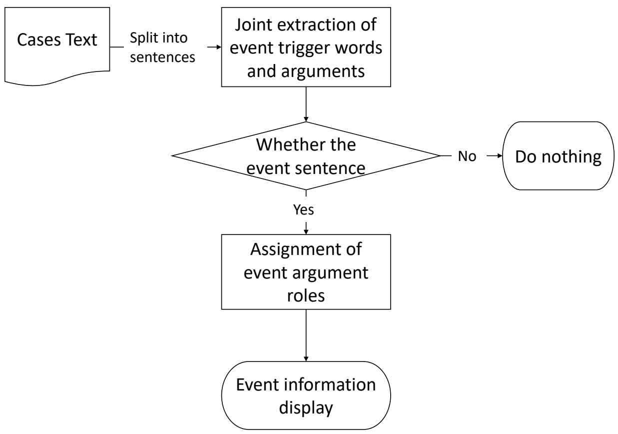

Fig. 5. Event Extraction Flow.

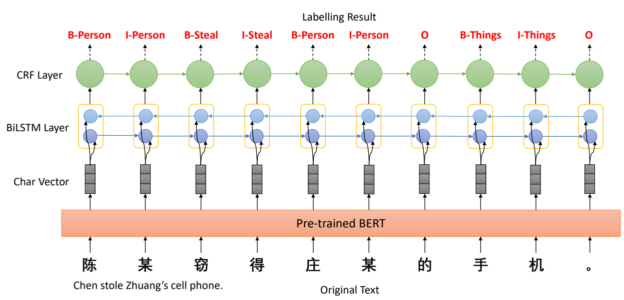

Fig. 6. Model Structure of Event Trigger Word and Argument Joint Extraction.

- **Argument Extraction Result.** We process the argument extraction result of the model in the first step, map the labels to id.
- **Concerned Trigger Vector.** The vector has five dimensions, and each dimension represents a type of trigger. If there is no such event in the text, the value of the dimension is 0, otherwise 1.
- **Distance Vector.** For all characters in the text, we calculate the distance between them and the first word of the trigger word in the current text to build a distance embedding. The vector also has five dimensions, and each dimension represents a type of event. If there is no such event in the text, the value of the dimension is positive infinity.

We combine the three vectors, input them into the CRF model to extract the roles of arguments. Fig.7 shows the model structure of this step.

Taking Fig.7 as an example, the text in this figure contains "Steal" events, so the "Steal" dimension of the concerned trigger vector is 1, and the larceny dimension of the distance vector is the distance between the current character and the trigger word. The remaining dimensions of the concerned trigger vector are 0, and the remaining dimensions of the distance vector are positive infinity, which is not shown in the figure.

The larceny event extraction model defined in this paper consists of the above two-step model. After the extraction of the two-step model, for a piece of text, the event trigger word, event arguments and their roles can be extracted.

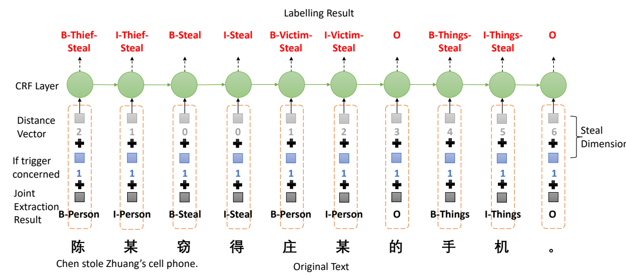

Fig. 7. Model Structure of Event Argument Roles Assignment.

D. Visualization of Event Elements

After extracting the event elements from the case description, we need to present them in a clearly form to support legal decision-making better. "Litigation visualization" is a method of presentation. Litigation visualization uses time charts to display event elements, which can clearly show the basic facts and development of the case. In this paper, the extracted events are sorted and displayed in time series after being preprocessed. Fig.8 shows the event elements visualization flow of this paper.

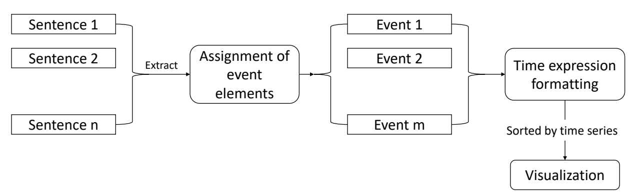

Fig. 8. Event Elements Visualization Flow.

a) Assignment of Event Elements: One or more events may be included in a event sentence. For the event sentence containing only one event, extract it directly to get an event. For an event sentence containing multiple events, the event elements in the sentence need to be assigned to the corresponding events to get mutiple event information. The method of event elements assignment is as follows:

- If multiple events included in the event sentence are different types of events, we directly allocate them according to the event type of the event element.
- If multiple events included in the event sentence are events of the same type, we calculate the distance between the event element and the event trigger word in the text, and the event element is assigned to the closest trigger word.

After the assignment of event elements, for an event information, it should contain the event trigger word and the assigned event elements.

b) Chinese Time Expression Formatting: After obtaining multiple pieces of event information, we need to sort them in time series. The time information extracted from the text is all Chinese time expressions, which need to be formatted and converted into the same form to be sorted.

We divide the Chinese time expressions in our dataset into four types. Table IV shows the types of Chinese time expression.

TABLE IV
TYPE OF CHINESE TIME EXPRESSION

Type	Example in English
Standard	August 15, 2016
Fuzzy	In the evening
Offset	4 days later
Range	July to August 2016

The overall idea for formatting Chinese time expressions is to create an array of length 6, each digit representing the formatted year, month, day, hour, minute, and second numbers. For the extracted time expression, we extract the corresponding number of the time unit in bits and fill in the corresponding position of the array. If there is no corresponding number in the expression, fill in the initial value according to the reference time unit (month and day are 1, and the rest are 0). After formatting, the time information is output in a form similar to ”2020-04-10 11:45:00”.

For the above four types of time expressions, the processing method is as follows:

- **Standard.** Directly extract the number of the corresponding time unit.
- **Fuzzy.** Correspond to a fixed fuzzy expression to a certain value. For example: ”in the evening” corresponds to 18 o’clock .
- **Offset.** The offset expression generally does not appear in the first time of all events. Therefore, for each event expression other than the first time, the last time is used as the ”base time”, and the current time is calculated based on the base time and the offset value.
- **Range.** For the range expression, the two time expressions in the range are extracted to form a time interval $[Time_1, Time_2]$. During formatting, the first time is used as the base time for the second time. During sorting, the time at the left end of the interval is used as the sorting criterion.

After formatting the time expression, each event information is sorted according to the formatted time to generate a event sequence in time series.

IV. EXPERIMENTS

A. Evaluation Indicators

We use precision, recall and F1-score as evaluation indicators, which are defined as follows:

$$Precision = \frac{S}{S_t} \tag{1}$$

$$Recall = \frac{S}{S_p} \tag{2}$$

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{3}$$

S_t is the number of characters labeled with positive examples in the extraction result. S is the number of characters that are correctly labeled and are positive examples in the extraction result. S_p is the number of all positive example characters in the original dataset.

B. Baseline

The baseline model we choose to compare with is the two-labeling model [2]. The model is divided into two steps. In the first step, word2vec [17] is used to generate word vectors after word segmentation. They used BiLSTM to extract semantic information, then combined it with part-of-speech tagging results and IsTrigger Vector, used CRF model to extract the transition label. In the second step, they took the embedding of transition label and trigger word, trigger word of current concern and position embedding as the input, used CRF model to extract the trigger types and argument types and roles.

The two-labeling model needs to use the trigger word matching method to label the event trigger words before prediction. We integrate the extraction of trigger words into the first step model, and automatically determine whether the sentence is an event sentence and its event type according to the trigger word extraction results. At the same time we optimize its first step model. We use BERT character vectors as the text representation. And we use BiLSTM-CRF as the extraction model.

We conduct a comparative experiment on the overall extraction effect of the two models. We use the model of the first step to extract the original text and obtain the joint extraction results of the argument and the trigger word. After that, we input the extraction results of the first-stage model and corresponding information into the second-stage model to obtain the extraction results and evaluate them based on the correct results. TableV shows the comparison of the extraction results of the baseline and our model.

TABLE V
OVERALL EXTRACTION RESULT

Method	Precision	Recall	F1-Score
Two-labeling (Baseline)	0.8444	0.8487	0.8466
Our Model	0.8544	0.8562	0.8553

C. Experimental Results

a) Joint Extraction of Trigger Word and Argument: This section introduces the experimental results of the first stage event argument and trigger word joint extraction model in this paper. In the training process, we use the ten-fold cross-validation method. We divide the training set into ten parts on average, randomly select one part as the verification set in each epoch training, and the remaining nine parts as the training set. In the iterative process, we use Adam [18] optimization algorithm to update the weights of the neural network. Table VI shows the overall extraction result of this step. Fig 9 shows the result of event trigger word extraction. Fig 10 shows the result of event argument extraction.

TABLE VI
RESULT OF EVENT TRIGGER WORD AND ARGUMENT JOINT EXTRACTION

Method	Precision	Recall	F1-Score
Word2Vec+BiLSTM+POS +Trigger+CRF (Baseline)	0.9104	0.9187	0.9146
BERT Char Embedding +BiLSTM-CRF (Our Model)	0.9592	0.9663	0.9628

The next part of this section shows our model's performance in trigger word extraction and argument extraction.

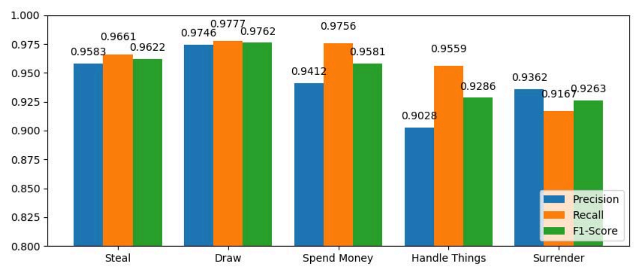

Fig. 9. Result of Event Trigger Word Extraction.

Section III-C mentions the problem of unbalanced distribution of events in larceny cases. The extraction result of smaller number events, such as spend money, handle things and surrender, is not as good as the result of steal event and draw event. But the overall extraction effect is quite good.

The result of event argument extraction is generally good. Among them, the extraction result of the location entity is relatively poor because the location entity has a strong irregularity in expression.

b) Joint Extraction of Trigger Word and Argument: In the experimental results of this section, only the assignment of event argument roles is considered, and the labeling results of trigger words are ignored. Table VII shows the model performance at this step. This result is only focused on the role assignment of event arguments.

TABLE VII
RESULT OF 2ND STEP MODEL

Method	Precision	Recall	F1-Score
Arguments+Concerned Trigger +Distance+CRF (Char-Seg)	0.9334	0.9332	0.9333
Arguments+Concerned Trigger +Distance+CRF (Word-Seg)	0.9438	0.9438	0.9438

In this step, there are many types of event argument roles (20 types), which are unevenly distributed. And the expressions of different event argument roles may be the same, causing confusion in extraction. After adding the concerned trigger embedding and the distance trigger position embedding , the

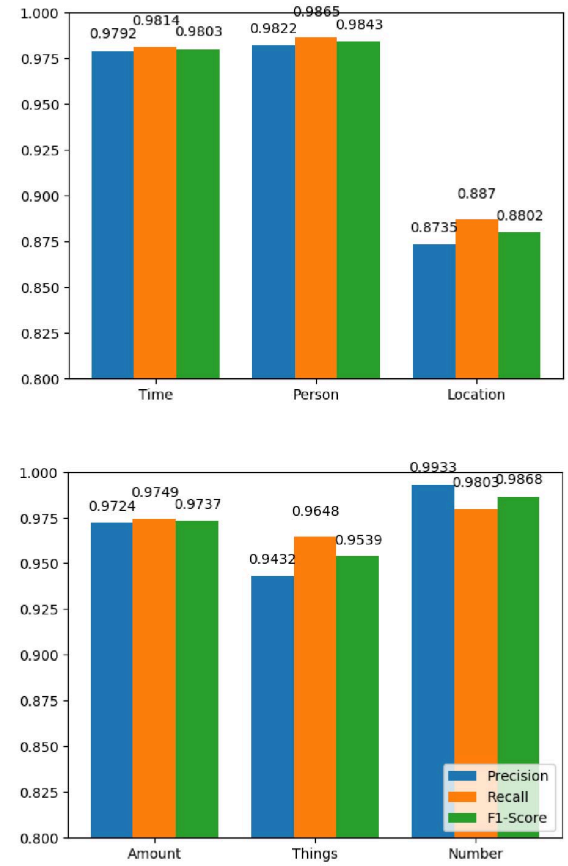

Fig. 10. Result of Event Argument Extraction.

model can more accurately allocate events to the argument, and the recall rate has been significantly improved.

The word segmentation dataset performs better on the CRF sequence labeling than the character segmentation dataset. The reason is that when a word contains many characters in the character segmentation dataset, the distance between the character at the endpoint of the word and the trigger word may be far.

D. Visualization of Event Elements

We develop a web application demo to realize the visualization of event elements in the case. The back end process the text of larceny cases passed in from the front end. Afer processing, the returned results are sent back to the front end to display the event line through dynamically generating page elements. The visualization result is shown in Fig.1.

V. CONCLUSION

In the legal field, the events in the case description are important standards for conviction and sentencing. The Chinese legal text has the characteristics of unstructured and strong narrative. It is possible for readers to misread event elements.

Our research aims to solve the above problems and apply the event extraction technology to the Chinese legal text.

We define five types of events in larceny cases and six types of event arguments. And we define the role of argument in the event. We use BIO scheme as the labelling scheme, and get a larceny case event dataset with the size of 6538 labelled sentences.

The dataset used in this paper has the problem of uneven label distribution. Inspired by the two-labeling model [2], we divide event extraction into two sub-tasks: joint extraction of trigger word and argument, and assignment of argument role. For the first step, we use pre-trained BERT model to embed the character vector, and use the BiLSTM-CRF model complete the extraction. In order to solve the problem that there are many types of event argument role labels and uneven distribution, for the second step, the extraction result of the first part is combined with the concerned trigger vector and the distance vector, and we input them into the CRF model for role assignment.

We visualize the extracted event information. We format the Chinese time expressions into a unified form and sort them to get time-series event information of a larceny case, and display it through a web page demo.

There are some work needs to be done in the future. In terms of the allocation of event elements, our research does not deal well with the case where there are multiple events in the same sentence and the event elements are distributed across sentences. In addition, there are many types of legal cases. Future work includes breaking the limitations of event templates and applying event extraction techniques to various types of cases.

References

[1] C. Xiao, H. Zhong, Z. Guo, C. Tu, Z. Liu, M. Sun, Y. Feng, X. Han, Z. Hu, H. Wang, and J. Xu, "CAIL2018: A large-scale legal dataset for judgment prediction," *CoRR*, vol. abs/1807.02478, 2018. [Online]. Available: http://arxiv.org/abs/1807.02478

[2] C. Li, Y. Sheng, J. Ge, and B. Luo, "Apply event extraction techniques to the judicial field," in *Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers, UbiComp/ISWC 2019 Adjunct, London, UK, September 9-13, 2019*, 2019, pp. 492–497. [Online]. Available: https://doi.org/10.1145/3341162.3345608

[3] Y. Chen, L. Xu, K. Liu, D. Zeng, and J. Zhao, "Event extraction via dynamic multi-pooling convolutional neural networks," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, 2015, pp. 167–176. [Online]. Available: https://doi.org/10.3115/v1/p15-1017

[4] S. Yang, D. Feng, L. Qiao, Z. Kan, and D. Li, "Exploring pre-trained language models for event extraction and generation," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, 2019, pp. 5284–5294. [Online]. Available: https://doi.org/10.18653/v1/p19-1522

[5] G. R. Doddington, A. Mitchell, M. A. Przybocki, L. A. Ramshaw, S. M. Strassel, and R. M. Weischedel, "The automatic content extraction (ACE) program - tasks, data, and evaluation," in *Proceedings of the Fourth International Conference on Language Resources and Evaluation, LREC 2004, May 26-28, 2004, Lisbon, Portugal*, 2004. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2004/summaries/5.htm

[6] H. Lin, Y. Lu, X. Han, and L. Sun, "Nugget proposal networks for chinese event detection," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, 2018, pp. 1565–1574. [Online]. Available: https://www.aclweb.org/anthology/P18-1145/

[7] H. Yang, Y. Chen, K. Liu, Y. Xiao, and J. Zhao, "DCFEE: A document-level chinese financial event extraction system based on automatically labeled training data," in *Proceedings of ACL 2018, Melbourne, Australia, July 15-20, 2018, System Demonstrations*, 2018, pp. 50–55. [Online]. Available: https://www.aclweb.org/anthology/P18-4009/

[8] Z. Li, X. Ding, and T. Liu, "Constructing narrative event evolutionary graph for script event prediction," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, 2018, pp. 4201–4207. [Online]. Available: https://doi.org/10.24963/ijcai.2018/584

[9] S. Yang, D. Feng, L. Qiao, Z. Kan, and D. Li, "Exploring pre-trained language models for event extraction and generation," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, 2019, pp. 5284–5294. [Online]. Available: https://doi.org/10.18653/v1/p19-1522

[10] T. H. Nguyen and R. Grishman, "Graph convolutional networks with argument-aware pooling for event detection," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, 2018, pp. 5900–5907. [Online]. Available: https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16329

[11] X. Yin, D. Zheng, Z. Lu, and R. Liu, "Neural entity reasoner for global consistency in NER," *CoRR*, vol. abs/1810.00347, 2018. [Online]. Available: http://arxiv.org/abs/1810.00347

[12] C. Liu and K. Chen, "Extracting the gist of chinese judgments of the supreme court," in *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law, ICAIL 2019, Montreal, QC, Canada, June 17-21, 2019*, 2019, pp. 73–82. [Online]. Available: https://doi.org/10.1145/3322640.3326715

[13] J. Yang, Y. Zhang, L. Li, and X. Li, "YEDDA: A lightweight collaborative text span annotation tool," in *Proceedings of ACL 2018, Melbourne, Australia, July 15-20, 2018, System Demonstrations*, 2018, pp. 31–36. [Online]. Available: https://www.aclweb.org/anthology/P18-4006/

[14] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186. [Online]. Available: https://doi.org/10.18653/v1/n19-1423

[15] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," *CoRR*, vol. abs/1508.01991, 2015. [Online]. Available: http://arxiv.org/abs/1508.01991

[16] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001*, 2001, pp. 282–289.

[17] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013. [Online]. Available: http://arxiv.org/abs/1301.3781

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980