Legal Knowledge and Information Systems
M. Araszkiewicz and V. Rodríguez-Doncel (Eds.)

doi:10.3233/FAIA190331

On Constructing a Knowledge Base of Chinese Criminal Cases

Xiaohan WU [a,1], Benjamin L. LIEBMAN [a], Rachel E. STERN [b],
Margaret E. ROBERTS [c] and Amarnath GUPTA [c]

[a] *Columbia Law School, USA*
[b] *University of California Berkeley, USA*
[c] *University of California San Diego, USA*

Abstract. We are developing a knowledge base over Chinese judicial decision documents to facilitate landscape analyses of Chinese Criminal Cases. We view judicial decision documents as a mixed-granularity semi-structured text where different levels of the text carry different semantic constructs and entailments. We use a combination of context-sensitive grammar, dependency parsing and discourse analysis to extract a formal and interpretable representation of these documents. Our knowledge base is developed by constructing associations between different elements of these documents. The interpretability is contributed in part by our formal representation of the Chinese criminal laws, also as semi-structured documents. The landscape analyses utilizes these two representations and enables a law researcher to ask legal pattern analysis queries.

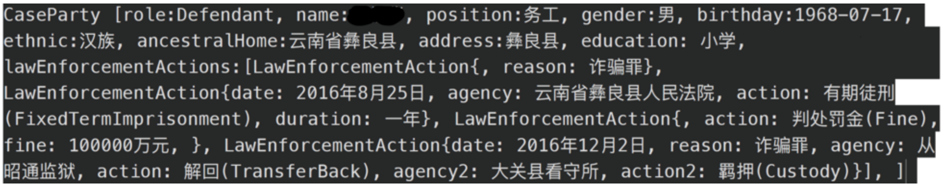
Keywords. landscape analysis, Chinese criminal cases, Information Extraction, discourse analysis, context-sensitive grammar, knowledge representation

1. Introduction

Our long-term goal is to develop a knowledge-based information system that would capture the "general knowledge" about a legal universe and the way law is practised in that universe. We use the term "general knowledge" in the sense that it can maintain enough information to enable a user infer "what usually happens" in a given legal scenario and what makes some case exceptional. For example, the one should be able to infer from the system that no defense argument is usually presented for drunk driving cases, and in an exceptional situation where there is one, only a leniency in the punishment is requested. We call these class of questions *legal landscape analyses*.

Prior Work. The primary corpus for our study is the Judicial Decision Documents (JDD) available from the Supreme People's Court (SPC) [1]. As Gupta et al [2] showed, parts of the data, such as the parties to the lawsuit including the plaintiffs and defendants, together with their legal representation, are represented as structurable text, stored in a relational database. However, [2] did not analyze the unstructured part such as the facts found by the court.

[1] Corresponding Authors: Wu E-mail: xw2510@columbia.edu, Gupta E-mail: a1gupta@ucsd.edu



```
CaseParty [role:Defendant, name:[redacted], position:务工, gender:男, birthday:1968-07-17,
ethnic:汉族, ancestralHome:云南省彝良县, address:彝良县, education: 小学,
lawEnforcementActions:[LawEnforcementAction{, reason: 诈骗罪},
LawEnforcementAction{date: 2016年8月25日, agency: 云南省彝良县人民法院, action: 有期徒刑
(FixedTermImprisonment), duration: 一年}, LawEnforcementAction{, action: 判处罚金(Fine),
fine: 100000万元, }, LawEnforcementAction{date: 2016年12月2日, reason: 诈骗罪, agency: 从
昭通监狱, action: 解回(TransferBack), agency2: 大关县看守所, action2: 羁押(Custody)}], ]
```

Figure 1. The semi-structured output of a party involved in a case.

2. Landscape Analysis of Legal Documents - A First Formal Model

We model a collection **C** of JDDs as a triple $(\mathbf{S},\mathbf{D},\mathbf{M})$ where **S** is a heterogeneous relation, **M** is a *k*-dimensional matrix and **D** is a mapping between elements of **S** and the indices of **M**. Here, a *heterogeneous relation* refers to a relation whose attributes can take different forms of semi-structured values. For example, `case-type` is a string valued (e.g., 'criminal' or 'administrative') attribute, while `parties` is a complex value as shown in Fig. 1. Notice how the parser output includes the criminal history of the defendant under the element `LawEnforcementActions` containing a hierarchy of subelements like the duration of the defendant's imprisonment.

The matrix **M** is derived from our analysis of the text-valued `Fact` element. Using parsing methods described in the next section, sentences in the fact can be classified into 8 classes: case background, arguments from plaintiff/prosecutor, evidences provided from plaintiff/prosecutor, requests/opinions from plaintiff/prosecutor, arguments from defendant, evidences from defendant, reviewed facts from court, and evidences accepted by court. In a typical JDD document, multiple consecutive sentences may belong to each class. The sentences in these sections can be further decomposed into an *action schema* given by `[subject, action, object, action_modifier]`. For example, the sentence (translated) "The defendant surrendered himself at police station in Binjiang on Feb.13th, 2017, where he admitted his crime honestly." has the actions: `['name of defendant', 'went to', 'Binjiang police station','voluntarily'], ['name of defendant', 'stated', 'criminal action','later','honestly']`. In the sentence (translated)(The total value of stolen items is 25,920 yuan.), the system detects the variable `damage`: ['25,920 yuan'] A similar representation of the court decision leads to a structure of the punishment issued by the court. For criminal cases `punishment` is represented by the numeric vector

`{Exemption(免于刑事处罚), Public Surveillance(管制),Detention(拘役), Fixed-Term Imprisonment(有期徒刑), Probation(缓刑), Fine(罚金), Political Rights Deprivation(剥夺政治权利), Confiscation(没收), Life Imprisonment(无期徒刑), Death(死刑), Political Rights Deprivation For Life(剥夺政治权利终身)}` where `Death, Exemption, LifeInprisonment, PoliticalRightsDeprivationForLife` are represented in binary code and other vector elements are represented by a quantified "degree of punishment" either in terms of time or in terms of monetary value.

The representation enables us to represent more than one punishment (e.g., prison time and fine) for a crime. Integrity constraints are applied to ensure that specific combinations of punishments (e.g., `FixedTermImprisonment` and `lifeImprisonment`) do not co-occur. We construct the matrix **M** as a product `action` × `damage` × `punishment-bucket` where a `punishment-bucket` is a discretized representation of

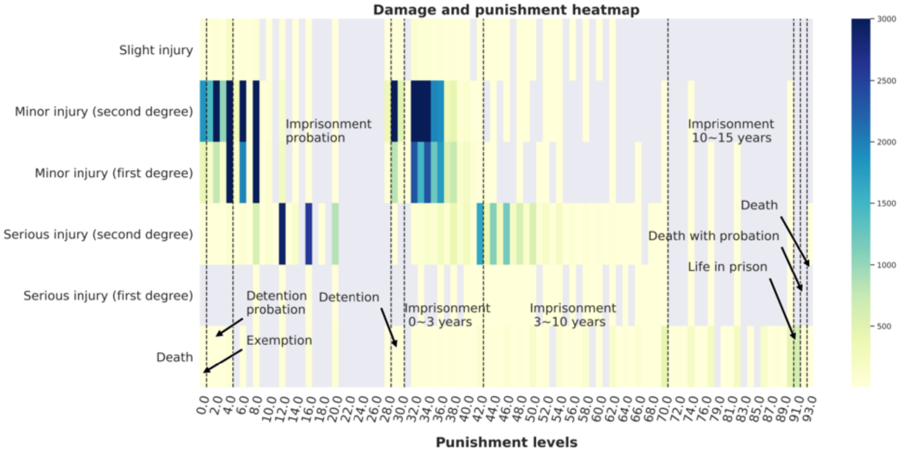


Figure 2. Damage and punishment heat map for assault and battery cases

the punishments. A cell of the matrix represents the number of cases that fall in the action-damage-punishment construct. **M** is partitioned by crime type so that theft is considered separately from murder. While this partitioning introduces some inaccuracy for cases where multiple crimes occur, we tolerate the inaccuracy for landscape analyses where the goal is to understand general properties of the distribution. Figure 2 shows a fragment of this matrix as a heatmap. Note that the color in this map indicates the number of cases for the corresponding combination. Gray means zero case. The unit for punishment levels is 3 months except for *Exemption, life in prison, death with probation and death penalty*, each of which takes one unit. Figure 2 shows how some combination of damages and punishment are more dense while some other combinations are empty, indicating combinations that although theoretically plausible occur rarely in practice. For example, according to Criminal law article 234, "whoever intentionally inflicts injury upon another person,causing severe injury to another person, shall be sentenced to fixed-term imprisonment of not less than three years but not more than 10 years". However, in practice, many assaulters were sentenced to fixed-term imprisonment of less than three years with probation – indicating judges' discretion in deciding punishments.

The mapping **D** between **S** and **M**, which is used for information retrieval, is a collection of indices. The forward indices serve as a pointer from a schema element like: `JDD.prosecutorArgument.sentence.actions.drunk-driving` to **M**`.traffic-misconduct[3]` where [3] indicates the axis of the matrix where `drunk-driving` is mapped. Similarly, `JDD.prosecutorArgument.sentence.drunk-driving.punishment` may map to **M**`.traffic-misconduct[3][2]` which is the action-punishment slice of the `traffic-misconduct` partition of **M**. In contrast, the reverse index behaves similarly as an inverted index in an information retrieval system where every cell of the matrix is mapped back to a list of case identifiers that populate the cell. Thus, the retrieval function `getCases(`**M**`[3][2][4])` will retrieve the drunk driving cases resulting in property damage up to 1000 yuan where a fine was imposed.

3. Information Extraction

To extract our analytical primitives, we have developed a parsing strategy for linguistic patterns that are characteristically observed in JDDs. The information extraction module assumes that the names of plaintiffs, defendants and their legal counsel are available to the system. In the following, we present a method for extracting the "action" part from the unstructured `Facts` of a JDD. The linguistic patterns observed include:

Long flowing sentences. The flowing sentence is a unique sentence pattern in Chinese. It contains so-called 链式结构(chain structure) – the relationship between 逗断(dòuduàn) was usually indicated by the order of events. Wang [3] defined dòuduàn as the basic unit of Chinese text and dòuduàn can be used as the index to specific communication event. We use dòuduàn as the minimum text processing unit for parsing and discourse analysis to reduces computation and improves parsing accuracy rate [4].

Action-focused defendant-centered description. The majority of sentences in facts, especially arguments from prosecutor and reviewed facts, are descriptions of actions. Even if the description is in passive voice, the subject of an action is usually the defendant. For example, 'The defendant has already obtained the victim's families' forgiveness.' is more common than 'The victim's family has already forgiven the defendant.'

Extracting action triggers. Verbs have been used as triggers in open information extraction [5,6] and news events extraction [7]. These relation patterns, however, is only applicable to English text. Open information extraction research in Chinese is still relatively inadequate[8]. We extract central actions where the subjects are the defendant or the police using the following rules for trigger verb extraction.

1. *Rule 1*. verbs in paths that originated from ROOT in constituency tree and only contains {*’IP’,’VP’,’VV’,’VRD’*}
2. *Rule 2*. verbs that are {*’conjunct’,’clausal complement’*} dependents of trigger verbs obtained by Rule 1.

For example, in dòuduàn 被告人在15号车厢当面接收张某某发送的手机微信红包(The defendant received Wechat red pockets sent by Zhang in person in car No.15), part-of-speech tagging identified two verbs: 接收(receive) and 发送(send). The central action in this dòuduàn is, [[’The defendant’], ’receive’, [’wechat red pocket’], [’in person’]]. Therefore, the trigger verb is ”receive” rather than ”send” by *Rule 1*.

Extracting elements of actions. In addition to action trigger verb, we defined *Subject, Object* and *action_modifier* in *action schema*. We extracted these elements based on universal dependencies (a multiliguial generalization of the dependency relationships from the Stanford Dependency parser) of trigger verbs:

- *Subject* extraction has two rules: *Rule 1* extracts nouns that are *’nominal subject’* of the trigger verb. *Rule 2* inherits *Subject* from the latest dòuduàn if *Rule 1* fails.
- *Objects* are usually *direct objects* of trigger verbs. Note that dòuduàn containing ‘被‘,’将‘ and ’把‘are treated as exceptions.
- *action_modifier* are trigger verb's *adverb modifier*. We also excluded *(遂,并,且,后,但)* because they turned out to be less important in our landscape analysis.

Extract damages, criminal charges, convicted crime charges and punishments. We extract monetary damages by applying named entity recognition(NER). There are five

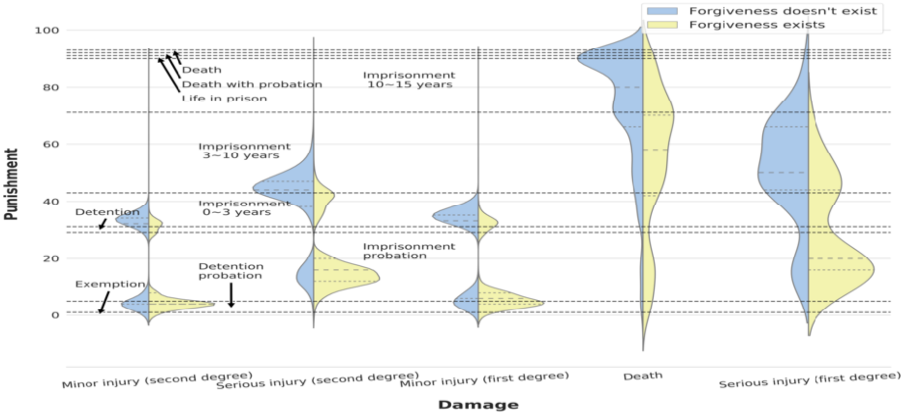


Figure 3. Probability density of punishment levels for battery cases with/without victims' forgiveness

injury levels in Chinese legal system. Since the injury levels are fixed and finite, we extracted human health damages by keyword matching. We use regular expression to extract the name of 469 crimes and convert extracted crime names to standard names to eliminate variations. Since the decision part of criminal cases is more structured, we chose the extraction keywords according to the principal and supplementary punishments in Chinese criminal law Article 33.

4. Answering Analytical Questions

Question 1. What is the distribution of punishments for cases where the defendant received the victims'(or victim families') forgiveness versus where they did not, conditioned by the damage caused by the crime?

We define $C1$ as a subset of cases where the action includes a lemmatized version of the term "forgiveness" with positive *action_modifier* and $C2$ where the cases do not. $C1$ contains 75655 battery cases while $C2$ contains 60627 cases. In Figure 3, the yellow part is probability density of punishments for cases where forgiveness exist while blue part is for cases where forgiveness don't exist. Evidently, judges tend to give lenient punishments to defendants who received forgiveness regardless of the damage severity.

Question 2. What punishments are rare for crime type X. Find the distribution of circumstances for which the punishment is "exemption". Here, we specify a "circumstance" as a combination of crime types, actions and damages. The steps of query evaluation are: (i) P = getMarginals($\mathbf{M}.X$, 'punishment'), (ii) C = getMarginals($\mathbf{M}.X$, 'punishment'='exemption'), (iii) C' = top-k(C, 20)

We set `case type = ''battery''`. In step (i), We found two types of rare punishments – punishments that are extremely lenient or harsh and punishments where the measurement unit is not a quarter of a year. Notice the C is a 2D histogram with axes action and damage. C' returns a fraction of C that only contains k most important actions defined by user – 20 most frequent action-damage pairs by default. We obtained 2,181 battery cases where defendants were exempted from criminal punishments and 2,033 actions associated with these cases. The importance score for each action is action frequency in C divided by action frequency in $\mathbf{M}.battery$. High exclusiveness can also lead to error actions that had very low frequency in both C and $\mathbf{M}.battery$. So we take

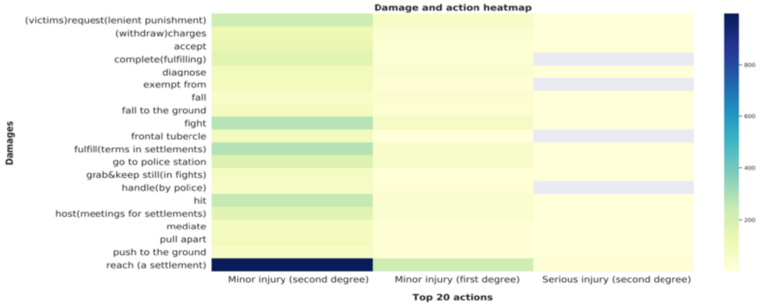


Figure 4. Heat map of damage and top 20 actions in battery cases

5% most frequent actions and select 20 most important actions according to importance score. Figure 4 is the co-occurrence-heat-map of damages and selected actions. This heat map shows that reaching settlements and fulfilling the terms for minor injuries before trial is a key factor for receiving exemption from punishments.

5. Conclusion and Future Work

In this paper, we have sketched our approach to developing a knowledge-base to answer landscape questions revealed by judicial decision documents from Chinese courts. Unlike a facts-and-rules or a graph-based knowledge representation system, we have opted to use heterogeneous relation, a distribution matrix and a mapping between them as our knowledge structure, and showed its usefulness in answering questions. Yet, our representation has taken some simplifying decisions that failed to capture some of the practical nuances of criminal law. In future work, we will refine our representation to accommodate further levels of punishment and action granularity.

Acknowledgment. We acknowledge the NSF RIDIR grant 1738411 for funding.

References

[1] B.L. Liebman, M. Roberts, R.E. Stern and A.Z. Wang, Mass Digitization of Chinese Court Decisions: How to Use Text as Data in the Field of Chinese Law, *S. Science Research Network Collection* (2017).

[2] A. Gupta, A.Z. Wang, K. Lin, H. Hong, H. Sun, B.L. Liebman, R.E. Stern, S. Dasgupta and M.E. Roberts, Toward Building a Legal Knowledge-Base of Chinese Judicial Documents for Large-Scale Analytics, in: *Proc. of Int. Conf. on Legal Knowledge and Information Systems (JURIX)*, 2017, pp. 135–144.

[3] H. Wang and R. Li, On the basic Unit of Chinese Texts and the Causes of the Flowing Sentences, *Essays on Linguistics* (2014), 11–40.

[4] X. Li, C. Zong and R. Hu, A hierarchical parsing approach with punctuation processing for long Chinese sentences, 2005.

[5] A. Fader, S. Soderland and O. Etzioni, Identifying relations for open information extraction, in: *Proceedings of the conference on empirical methods in natural language processing*, Association for Computational Linguistics, 2011, pp. 1535–1545.

[6] N.G. Silveira, Designing Syntactic Representations for NLP: An Empirical Investigation, PhD thesis, Stanford University, 2016.

[7] D. Rusu, J. Hodson and A. Kimball, Unsupervised techniques for extracting and clustering complex events in news, in: *Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, 2014, pp. 26–34.

[8] Q. Bing, L. Anan and L. Ting, Unsupervised Chinese open entity relation extraction, *Journal of Computer Research and Development* **52**(5) (2015), 1029–1035.