

# Recognition of Requisite Part and Effectuation Part in Law Sentences

**Ngo Xuan Bach**                      **Nguyen Le Minh**                      **Akira Shimazu**  
 School of Information Science School of Information Science School of Information Science  
 JAIST                                      JAIST                                      JAIST  
 bachnx@jaist.ac.jp    nguyenml@jaist.ac.jp    shimazu@jaist.ac.jp

## Abstract

Analyzing a logical structure of a sentence is important for understanding natural language. In this paper, we present a task of *Recognition of Requisite Part and Effectuation Part in Law Sentences*, or RRE task for short, which is studied in the research of legal engineering. The goal of this task is to recognize the structure of a law sentence. We investigate RRE task in both aspects: linguistic features and problem modeling. We also propose solutions and present experimental results on a Japanese legal text domain. Our model achieved 88.58% in  $F_{\beta=1}$  score on the Japanese National Pension Law corpus.

## 1 Introduction

Legal texts have some specific characteristics that make them different from other daily-used documents. One of the most important characteristics of legal texts is that law sentences have some specific structures. In most cases, a law sentence can roughly be divided into two parts: a *requisite part* and an *effectuation part*. It may also contain some other parts such as subject parts, which describe objects in the law sentence (Nakamura et al, 2007; Tanaka et al, 1993).

Analyzing the logical structure of law sentences is an important task to understand the meaning of legal documents. This task is the preliminary step which supports other problems in automatic legal text processing such as translating a legal sentence into a logical form, legal text summarization, question answering in legal domains, etc (Nakamura et al, 2007).

In this paper, we present a task of *Recognition of Requisite Part and Effectuation Part in Law Sentences* - RRE task. We show how to model RRE task as a sequence learning problem. We de-

scribe an investigation into RRE task in some aspects: linguistic features, problem modeling, and tag settings. We present a discriminative reranking model for RRE task using the perceptron algorithm (Collins, ; Freund et al, 1999). We also show experimental results on the Japanese National Pension Law corpus. Our model achieved 88.58% in  $F_{\beta=1}$  score.

The remains of this paper are organized as follows. First, Section 2 presents how to model RRE task as a sequence labeling problem and show experimental results about the effect of features on the task. Next, we describe another setting for RRE task basing on bunsetsu in Section 3. Then, Section 4 presents a discriminative reranking model for RRE task. Finally, discussion is presented in Section 5.

## 2 RRE as a Sequence Learning Problem

### 2.1 Problem Setting

We model RRE task as a sequence labeling task in which each sentence is a sequence of words. Figure 1 illustrates an example in the IOB notation (Ludtke et al, 2003).

Source Sentence	<R>被保険者期間を計算する場合には、<R><E>月によるものとする。<E> (When a period of an insured is calculated, it is based on a month.)																										
Word Sequence	被	保	險	者	期	間	を	計	算	す	る	場	合	に	は	、	月	に	よ	る	も	の	と	す	る	。	
Tag Sequence	B-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	B-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E

Figure 1: A law sentence in the IOB notation.

In RRE task, we consider two types of law sentences: *implication type* and *equivalence type*. Each type of sentences has some kinds of parts:  $R, E, S_1, S_2$ , and  $S_3$  in implication type;  $EL$  and  $ER$  in equivalence type. Totally, we have 7 kinds of parts.

## 2.2 Feature Investigation

### 2.2.1 Corpus

The Japanese National Pension Law corpus includes 764 annotated Japanese law sentences.<sup>1</sup> We have some remarks. First, about 98.5% sentences belong to the implication type, and only 1.5% sentences belong to the equivalence type. Second, about 83.5% subject parts are  $S_2$ , 15.2% subject parts are  $S_3$ , and only 1.3% subject parts are  $S_1$ . Finally, four main types of parts  $E$ ,  $R$ ,  $S_2$ , and  $S_3$  make up more than 98.3%.

### 2.2.2 Evaluation Method and Baseline Model

We divided the annotated corpus into 10 sets and did 10-fold cross-validation tests. The results were evaluated using  $F_{\beta=1}$  score.

We designed five sets of features (based on CaboCha tool (Kudo, )), Each of these feature sets contains one kind of feature. With each kind of feature, we got unigram, bigram, and trigram features in a window size 2. All experiments were conducted using Conditional random fields, a powerful model for sequence labeling tasks (Kudo, ; Lafferty et al, 2001).

We considered the model using only word features as the baseline model. The results of the baseline model are shown in Table 1.

Table 1: Results of the baseline model

Tag	Precision(%)	Recall(%)	$F_{\beta=1}$ (%)
$E$	90.25	91.95	91.09
$EL$	0.00	0.00	0.00
$ER$	0.00	0.00	0.00
$R$	89.29	85.55	87.38
$S_1$	100.00	22.22	36.36
$S_2$	85.02	89.86	87.37
$S_3$	60.00	38.24	46.71
Overall	<b>87.27</b>	<b>85.50</b>	<b>86.38</b>

### 2.2.3 Experimental Results on Feature Sets

In order to investigate the effect of each type of feature on the task, we respectively conducted experiments on four other feature sets combining with the word features. The experimental results are shown in Table 2. Only the model 3 with *word* and *pos* features made an improvement of 0.28% comparing with the baseline model.

<sup>1</sup>The corpus consists of mainly the first sentence of each article.

## 2.3 RRE Using Head Words and Functional Words

### 2.3.1 Basic Idea

Our idea is that: 1) first, we reduce an original sentence to a reduction sentence which contains only head words and functional words of all the bunsetsu (Murata et al, 2000) in the original sentence; 2) then we do the recognition task on the new reduction sentence. We illustrate this process in Figure 2.

Source Sentence	<R>被保険者期間を計算する場合には、<R><E>月によるものとする。</E> (When a period of an insured is calculated, it is based on a month.)																										
Original Sequence	被	保	険	者	期	間	を	計	算	す	る	場	合	に	は	、	月	に	よ	る	も	の	と	す	る	。	
Original Tag	B-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	B-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E
Bunsetsu	1			2			3			4			5			6											
Head Word				Yes			Yes	Yes				Yes			Yes		Yes		Yes		Yes		Yes		Yes		
Functional Word					Yes							Yes				Yes		Yes		Yes		Yes		Yes		Yes	
New Sequence				期	間	を		す	る	場	合	に	は		月	に	よ	る	も	の	と	す	る				
New Tag				B-R	I-R			I-R	I-R			I-R			B-E	I-E		I-E		I-E		I-E					

Figure 2: Sentence reduction.

### 2.3.2 Experimental Results

Experimental results on new reduction sentences are described in Table 3. In HFW model we only use head and functional words while in HFWP model we add their POS tags. Note that in both models, in addition to head and functional words, we remain punctuations, which are important signals in a sentence.

Table 3: Experiments on reduction sentences

Model	Precision(%)	Recall(%)	$F_{\beta=1}$ (%)
Baseline	87.27	85.50	86.38
HFW	<b>88.09</b>	<b>86.30</b>	<b>87.19(+0.81)</b>
HFWP	87.74	86.52	87.12(+0.74)

Experimental results showed that using reduction sentences is better than using full sentences. It demonstrates the important role of head and functional words in RRE task. HFW model improves 0.81% in  $F_{\beta=1}$  score (5.9% in error rate) comparing with the baseline model.

## 3 RRE Basing on Bunsetsu

### 3.1 Basic Idea

Our idea is that instead of considering elements as words we will consider elements as bunsetsu. By doing this, we can reduce the length of sequences

Table 2: Experiments with feature sets

Model	Feature Sets	Precision(%)	Recall(%)	$F_{\beta=1}(\%)$
Model1(Baseline)	Word	87.27	85.50	86.38
Model2	Word + Katakana, Stem	87.02	85.39	86.20(-0.18)
Model3	Word + POS	<b>87.68</b>	<b>85.66</b>	<b>86.66(+0.28)</b>
Model4	Word + Bunsetsu	86.15	84.86	85.50(-0.88)
Model5	Word + NE	87.22	85.45	86.32(-0.06)

significantly. The process of getting new setting from the previous one is illustrated in Figure 3.

Source Sentence	<R>被保険者期間を計算する場合には、</R><E>月によるものとする。</E> (When a period of an insured is calculated, it is based on a month.)																										
Original Sequence	被	保	険	者	期	間	を	計	算	す	る	場	合	に	は	、	月	に	よ	る	も	の	と	す	る	。	
Original Tag	B-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	I-R	B-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E	I-E
Bunsetsu	1				2				3				4				5				6						
New Tag	B-R				I-R				I-R				B-E				I-E				I-E						

Figure 3: New setting for RRE task.

### 3.2 Experiments on Bunsetsu Setting

We use features about head words, functional words, punctuations, and co-occurrence of head words and functional words in a window size 1. Experimental results on new setting are shown in Table 4, in which BC model (Based on Chunks model) is the model in new setting (based on bunsetsu). The results show that modeling sequences basing on bunsetsu is suitable for RRE task.

Table 4: Experiments on new setting

Model	Precision(%)	Recall(%)	$F_{\beta=1}(\%)$
Baseline	87.27	85.50	86.38
HFW	88.09	86.30	87.19(+0.81)
BC(Bunsetsu)	<b>88.75</b>	<b>86.52</b>	<b>87.62(+1.24)</b>

BC model improves 1.24% in  $F_{\beta=1}$  score (9.1% in error rate) comparing with the baseline model.

### 3.3 Experiments with Different Tag Settings

To investigate RRE task in different tag settings, we conducted experiments with BC model using four kinds of tag settings (Ludtke et al, 2003). In all experiments below (see Table 5) we consider an element of a sequence is a bunsetsu (except for the baseline model) and used the same feature set (see Section 3.2).

The best tag setting is IOE (it improves 1.80% in  $F_{\beta=1}$  score, 13.2% in error rate comparing with the baseline model).

Table 5: Experiments on four tag settings

Model	Precision(%)	Recall(%)	$F_{\beta=1}(\%)$
Baseline	87.27	85.50	86.38
BC-IOB	88.75	86.52	87.62(+1.24)
BC-IOE	<b>89.35</b>	<b>87.05</b>	<b>88.18(+1.80)</b>
BC-FILC	88.75	86.09	87.40(+1.02)
BC-FIL	88.87	86.30	87.57(+1.19)

## 4 A Reranking Model for RRE

### 4.1 Discriminative Reranking with Linear Models

In reranking approach (Collins et al, 2005; Collins, ), first, a set of candidates is generated using a component GEN. GEN can be any model for the task. For example, in POS tagging problem, GEN may be a model that generates all possible POS tags for a word basing on a dictionary. Then, candidates are reranked using a linear score function:

$$score(y) = \Phi(y) \cdot W \quad (1)$$

where  $y$  is a candidate,  $\Phi(y)$  is the feature vector of candidate  $y$ , and  $W$  is a parameter vector. The output of reranking model will be the candidate with the highest score:

$$F(x) = \operatorname{argmax}_{y \in GEN(x)} \Phi(y) \cdot W. \quad (2)$$

### 4.2 Feature Representation

From a candidate, we first extracted a tag sequence and a part sequence. Tag sequence is the output of the candidate after removing the second tag if there are two adjacent same tags. For a candidate, we extracted following features: probability of the candidate outputted by GEN, unigram, bigram, trigram, and number of parts in the candidate.

### 4.3 Experiments

The architecture of our reranking model for RRE task is illustrated in Figure 4. First, the annotated

Table 6: Comparison between BC-IOE model and reranking model

Tag	Number	BC-IOE model			Reranking model		
		Precision(%)	Recall(%)	$F_{\beta=1}$	Precision(%)	Recall(%)	$F_{\beta=1}$
<i>E</i>	745	90.99	92.21	91.60	91.50	92.48	<b>91.99(+0.39)</b>
<i>EL</i>	11	0.00	0.00	0.00	0.00	0.00	0.00
<i>ER</i>	11	0.00	0.00	0.00	0.00	0.00	0.00
<i>R</i>	429	91.87	86.95	89.34	91.28	87.88	<b>89.55(+0.21)</b>
<i>S<sub>1</sub></i>	9	100.00	22.22	36.36	100.00	22.22	36.36
<i>S<sub>2</sub></i>	562	86.66	91.28	88.91	87.90	91.81	<b>89.82(+0.91)</b>
<i>S<sub>3</sub></i>	102	78.79	50.98	61.90	74.67	54.90	<b>63.28(+1.38)</b>
Overall	1869	89.35	87.05	88.18	<b>89.42</b>	<b>87.75</b>	<b>88.58(+0.40)</b>

corpus was divided into three parts: training set (80%), development set (10%), and test set (10%). The training set was used for training a BC-IOE model. Then, this model was tested on the development set to learn a parameter vector using the Perceptron algorithm (Collins, ; Freund et al, 1999). We also trained another BC-IOE model (using both the training set and the development set), and used this model as the GEN component of the reranking model. With each sample, we got 20-best outputs as candidates.

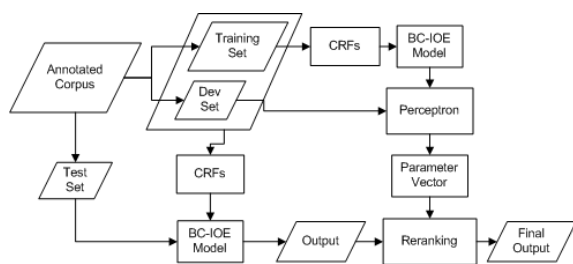


Figure 4: Reranking model.

Experimental results are shown in Table 6, in which iteration number is set to 10. The reranking model improves 0.40% in  $F_{\beta=1}$  score (3.4% in error rate) comparing with the best model before (BC-IOE), and 2.2% in  $F_{\beta=1}$  score (15.9% in error rate) comparing with the baseline model.

## 5 Discussion

We described a study on RRE task in some aspects: linguistic features, problem modeling, and tag settings. We also presented a discriminative reranking model for RRE task using the Perceptron algorithm. Because our corpus is quite small, the results of some parts were not good especially the results of equivalence sentences. In the future, we will investigate RRE task more deeply on a

bigger corpus.

## Acknowledgments

This research was partially supported by the Ministry of Education, Science, Sports and Culture, COE Research and Grant-in-Aid for Scientific Research (B).

## References

- Collins, M., Koo, T. 2005. *Discriminative Reranking for Natural Language Parsing*. In *Computational Linguistics*, Vol. 31, Iss. 1, pp.25-70.
- Collins, M. *Discriminative Reranking for NLP*. <http://www.clsp.jhu.edu/ws2005/calendar/documents/CollinsJuly7.pdf>
- Freund, Y., Schapire, R. 1999. *Large Margin Classification using the Perceptron Algorithm*. In *Machine Learning*, Vol. 37, Iss. 3, pp.277-296.
- Kudo, T. *Yet Another Japanese Dependency Structure Analyzer*. <http://chasen.org/~taku/software/cabocha/>
- Kudo, T. *CRF++: Yet Another CRF toolkit*. <http://crfpp.sourceforge.net/>
- Lafferty, J., McCallum, A., Pereira, F. 2001. *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data*. In *Proc. of ICML*, pp.282-289.
- Lüdtke, D., Sato, S. 2003. *Fast Base NP Chunking with Decision Trees-Experiments on Different POS Tag Settings*. In *Proc. of CICLing*, pp. 139-150.
- Murata, M., Uchimoto, K., Ma, Q., Isahara, H. 2000. *Bunsetsu identification using category-exclusive rules*. In *Proc. of the 18th conf. on Computational linguistics - Vol. 1*, pp.565-571.
- Nakamura, M., Nobuoka, S., Shimazu, A. 2007. *Towards Translation of Legal Sentences into Logical Forms*. In *Proc. of the 1st Int. Workshop on JURISIN*.
- Tanaka, K., Kawazoe, I., Narita, H. 1993. *Standard structure of legal provisions - for the legal knowledge processing by natural language - (in Japanese)*. In *IPJSJ Research Report on NLP*, pp.79-86.