Treatment of Legal Sentences Including Itemized and Referential Expressions – Towards Translation into Logical Forms

Yusuke Kimura, Makoto Nakamura, and Akira Shimazu

School of Information Science
Japan Advanced Institute of Science and Technology
1-1, Asahidai, Nomi, Ishikawa, 923-1292, Japan
{mnakamur,shimazu}@jaist.ac.jp

Abstract. This paper proposes a framework for analyzing legal sentences including itemized or referential expressions. Thus far, we have developed a system for translating legal documents into logical formulae. Although our system basically converts words and phrases in a target sentence into predicates in a logical formula, it generates some useless predicates for itemized and referential expressions. We propose a front end system which substitutes corresponding referent phrases for these expressions. Thus, the proposed system generates a meaningful text with high readability, which can be input into our translation system. We examine our system with actual data of legal documents. As a result, the system was 73.1% accurate in terms of removing itemized expressions in a closed test, and 51.4% accurate in an open test.

1 Introduction

A new research field called *Legal Engineering* was proposed in the 21st Century COE Program, Verifiable and Evolvable e-Society [1,2,3]. Legal Engineering serves for computer-aided examination and verification of whether a law has been established appropriately according to its purpose, whether there are logical contradictions or problems in the document per se, whether the law is consistent with related laws, and whether its revisions have been modified, added, and deleted consistently. One approach to verifying law sentences is to convert law sentences into logical or formal expressions and to verify them based on inference [4].

This paper reports our ongoing research effort to build up a system for automatically converting legal documents into logical forms. The system analyzes law sentences, determines logical structures, and then generates logical expressions. Thus far, we have shown our system provides high accuracy in terms of generating logical predicates corresponding to words and their semantic relations [5]. However, some predicates generated concerned with itemization and reference were meaningless, because predicates converted from words and phrases, such as "the items below," "Article 5," and so on are not intrinsic to a logical representation of the sentence. These words should be replaced with appropriate phrases

H. Hattori et al. (Eds.): JSAI 2008, LNAI 5447, pp. 242–253, 2009.

before the process of translation. Accordingly, our purpose in this paper is to propose a method to rewrite legal sentences including itemization or reference into an independent, plain sentence. We consider that this system is useful not only for the front end processor of our main system for translating legal sentences into logical forms, but also for assistance for reading legal documents.

In this paper, we introduce our current system and its problems in Section 2. In Section 3 we show analysis of law sentences including itemization or reference, and we propose a method to rewrite the law sentences into plain sentences in Section 4. We also examine our new method and report its results in Section 5. Finally, we conclude and describe our future work in Section 6.

2 The Current System and Problems

In this section, we describe our current system for translating legal documents into logical forms, and its problems. We call our system WILDCATS[1].

2.1 Work Related to Wildcats

Acquisition of knowledge bases by automatically reading natural language texts has widely been studied. Because the definition of semantic representation differs depending on what the language processing systems deal with, a few systems try to generate logical formulae based on first order predicate logic [6]. A study of knowledge acquisition by Mulkar et al. [7,8] is one of those systems. They extracted well-defined logical formulae from textbooks of biology and chemistry. As a result, their model succeeded in solving some high school AP exam questions. Legal documents are different from the textbooks in that they are described with characteristic expressions in order to avoid ambiguous description. Therefore, we take into account analysis of the expressions based on the linguistic investigation.

In most cases, a law sentence in Japanese Law consists of a law requisite part and a law effectuation part, which designate its legal logical structure [9,10]. Structure of a sentence in terms of these parts is shown in Fig. 1. The law requisite part is further divided into a subject part and a condition part, and the law effectuation part is divided into an object, content, and provision part.

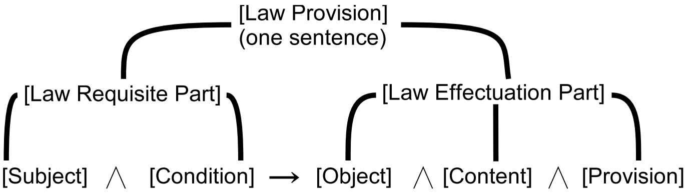


Fig. 1. Structure of requisition and effectuation [9]

[1] WILDCATS is an abbreviation of " 'Wildcats' Is a Legal Domain Controller As a Translation System."

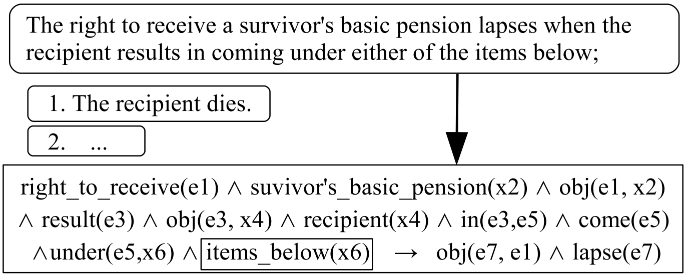

The right to receive a survivor's basic pension lapses when the recipient results in coming under either of the items below;
1. The recipient dies.
2. ...
right_to_receive(e1) ∧ suvivor's_basic_pension(x2) ∧ obj(e1, x2) ∧ result(e3) ∧ obj(e3, x4) ∧ recipient(x4) ∧ in(e3,e5) ∧ come(e5) ∧under(e5,x6) ∧ items_below(x6) → obj(e7, e1) ∧ lapse(e7)

Fig. 2. Converting a law sentence including a reference phrase

Dividing a sentence into these two parts in the pre-processing stage makes the main procedure more efficient and accurate. Nagai et al. [10] proposed an acquisition model for this structure from Japanese law sentences. Dealing with strict linguistic constraints of law sentences, their model succeeded in acquiring the structures at fairly high accuracy using a simple method, which specifies the surface forms of law sentences. Our approach is different from theirs in that we consider some semantic analyses in order to represent logical formulae.

2.2 Wildcats

Here, we explain an outline of our current system. The following list is the procedure for one sentence. We repeat it when we process a set of sentences.

1. Analyzing morphology by JUMAN [11] and parsing a target sentence by KNP [12].
2. Splitting the sentence based on the characteristic structure of a law sentence.
3. Assignment of modal operators with the cue of auxiliary verbs.
4. Making one paraphrase of multiple similar expressions for unified expression.
5. Analyzing clauses and noun phrases using a case frame dictionary.
6. Assigning variables and logical predicates. We assign verb phrases and *sahen*-nouns[2] to a logical predicate and an event variable, e_i, and other content words to x_j, which represents an argument of a logical predicate.
7. Building a logical formula based on fragments of logical connectives, modal operators, and predicates.

The procedure is roughly divided into two parts. One is to make the outside frame of the logical form (Step 1 to 3 and 7), which corresponds to the legal logical structure shown in Fig. 1. The other (Step 4 to 6) is for the inside frame. We assign noun phrases to bound variables and predicates using a case frame dictionary. We show an example of input and output in Fig. 2.

2.3 Problems of Wildcats

When our system converts a law sentence including referential phrases, they are not interpreted correctly. For example, in Fig. 2, the enclosed predicate

[2] A *sahen*-noun is a noun which can become a verb with the suffix -*suru*.

"items_below(x6)" is useless. This is because the generated predicates lack information which must be referred. These phrases should be replaced with appropriate phrases in the items before the process of translation into logical forms. Therefore, substituting corresponding referent phrases for these expressions appropriately, our proposed system in this paper generates a meaningful text with high readability, and then the generated text can be input to the translation system. For example, the system should process the following instead of the input sentences in Fig. 2; "The right to receive a survivor's basic pension lapses when the recipient dies."

We have found other kinds of related problems such as treatment of tables in National Pension Law. However, the scope of the study in this paper is restricted to itemized and referential expressions. Therefore, in the following sections we show analysis of law sentences and explain our methodology, which is based on the previous study by Ogawa et al. [13], who proposed a method for rewriting texts using regular expressions in order to consolidate legal sentences and amendment sentences.

3 Analysis of Law Sentences

In this section, we analyze sentences in National Pension Law, which is often picked up in the field of Legal Engineering as one of laws in which law enforcement information systems have been developed, such as Income Tax Law, Road Traffic Law, and so on.

3.1 Analysis of Reference in Law Sentences

There are reference phrases in law sentences, for example "X *-ni kitei-suru* Y (Y which is prescribed in X)." In National Pension Law, typical reference phrases are shown in Table 1.

In these phrases, X acts as a pointer to another law sentence. We show some examples of reference phrases found in National Pension Law, as follows:

- Item *a*, Paragraph *b*, Article *c* (absolute pointer)
- the previous paragraph (relative pointer)

Table 1. Typical reference phrases in National Pension Law

Reference phrases		Frequency
X-*ni kitei-suru* Y_{noun} Y_{noun} which is prescribed in X	(X に規定する Y(名詞))	103
X-*no kitei-niyoru* Y_{noun} Y_{noun} which is prescribed in X	(X の規定による Y(名詞))	71
X-*no kitei-niyori* Y_{verb} Y_{verb} as prescribed in X	(X の規定により Y(動詞))	109

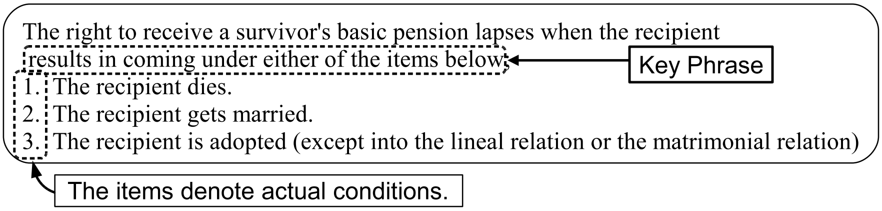

The right to receive a survivor's basic pension lapses when the recipient
results in coming under either of the items below:
1. The recipient dies.
2. The recipient gets married.
3. The recipient is adopted (except into the lineal relation or the matrimonial relation)
Key Phrase
The items denote actual conditions.

Fig. 3. Itemization of conditions in the law requisite part

- Paragraph *b* in the previous article (combination)
- the same article
- the previous *n* articles
- a proviso in the previous article
- this law (self-reference)

We also examined frequency in the use of the noun Y in the sentence indicated by a pointer X, because we consider that the noun Y is explained in detail in the sentence indicated by X. For example, a phrase "the postponement of issuance which is prescribed in Paragraph 1, Article 28" implies that we can find a more detailed phrase "postponement of issuance of the old age basic pension" in Paragraph 1, Article 28. Therefore, we regarded the sentence indicated by X as an explanation of the noun Y.

We targeted the phrase "X-*ni kitei-suru* Y (Y which is prescribed in X)," as it appears most frequently among reference phrases where Y is a noun phrase in National Pension Law (see Table 1). A pointer X indicates another law document in 21 cases out of 103, and we examined the remaining 82 cases. As a result, in 49 cases the noun Y appears only once in the sentence indicated by X, and twice or more in 24 cases, while the sentence indicated by X does not contain the noun Y in only 9 cases. Therefore, it is easy to find the part of the explanation, which is located near the noun Y. With this idea, we consider a method to extract an explanation from a sentence indicated by X in Section 4.1.

3.2 Analysis of Itemization in Law Sentences

Some law sentences include itemization of conditions in the law requisite part, an example of which is shown in Fig. 3. The enclosed phrase should be replaced with one of the items denoting actual conditions. When one or more conditions are satisfied, the description in the law effectuation part becomes effective. We found 34 sentences of such a style in National Pension Law. Therefore, we considered a method to embed itemized conditions instead of cue phrases of itemization.

We defined *Key Phrases*, which always appear in sentences before itemization[3]. As we analyzed sentences from all 215 articles of the National Pension Law, the set of Key Phrases can be expressed as a regular expression, the diagram of which is shown in Fig. 4. For example, the phrase *"Tsugi no kaku gou*

[3] There may be a proviso between the sentence and itemization

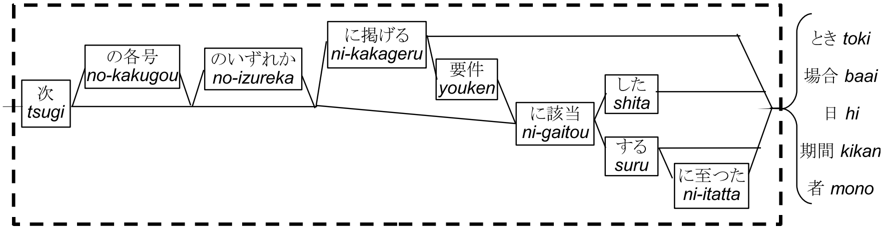


Fig. 4. Key phrases for itemization

Table 2. Frequency of Key Phrases

(**KP**: Key Phrase)

Format of KPs / Frequency		
KP + *toki*	(とき)	9
KP + *baai*	(場合)	9
KP + *mono*	(者)	6
KP + *hi*	(日)	3
KP + *kikan*	(期間)	1
KP + *youken*	(要件)	1
KP + a noun		5
Total		34

Table 3. Frequency of Condition Items

CI: Condition Items

Format of CIs / Frequency		
CI + *toki*	(とき)	106
CI + *koto*	(こと)	4
CI + *mono*	(もの)	3
CI + *mono*	(者)	2
CI + a noun		9
Total		124

ni gaitou suru ni itatta," meaning "to result in coming under either of the items below[4]," which is derived from the generative rule in Fig. 4, is regarded as a Key Phrase.

Itemized condition sentences appear next to sentences which contain Key Phrases. The last words of these sentences are "*Toki* (time)," "*Mono* (person)," and so on. In this paper, we call these sentences excluding the last words *Condition Items.* Key Phrases and Condition Items appearing in National Pension Law are shown in Table 2 and Table 3, respectively.

We will describe a method to remove itemization using Key Phrases and Condition Items in Section 4.2.

4 Method for Substituting Referent Phrases

4.1 Extracting an Explanation from Referent

As was mentioned in Section 3.1, we show a method to extract a detailed explanation of a reference phrase, such as "X-*ni kitei-suru* Y (Y which is prescribed in X)," from a referent sentence. The procedure is shown as follows;

[4] If we do not care about word-to-word translation for the Japanese law sentence, the following phrase is more appropriate; "to be included in one of the following cases."

1. Identifying a reference expression
2. Searching for the same words in the reference expression and the referent sentence
3. Syntactic analysis of the referent sentence and extraction of supplements

In the first step, if the sentence includes one of the phrases in Table 1, the system recognizes the phrase as a reference expression. We show an example of a reference expression in Fig. 5-A. The phrase "which is prescribed in" is the reference phrase, and the referent sentence is shown in Paragraph 1, Article 28.

In the next step, the system searches for a phrase in the referent sentence, which is matched with the noun phrase corresponding to Y described in the reference sentence. A difficult thing is to determine the region of words as an identified phrase. The system recognizes the longest matched words as the noun phrase Y. In Fig. 5, the system extracted a phrase corresponding to Y as "apply for postponement of issuance."[5]

Finally, the system analyzes the referent sentence with the Japanese morphological analyzer, JUMAN, and Japanese dependency analyzer, KNP. We regard elements which modify Y in the dependency tree as supplements for the word Y. Then, we replace the phrase "X-*ni kitei-suru*" with the supplements for the word Y. In this example shown in Fig. 6, "which is prescribed in Paragraph 1, Article 28" is replaced with "of the old age basic pension."

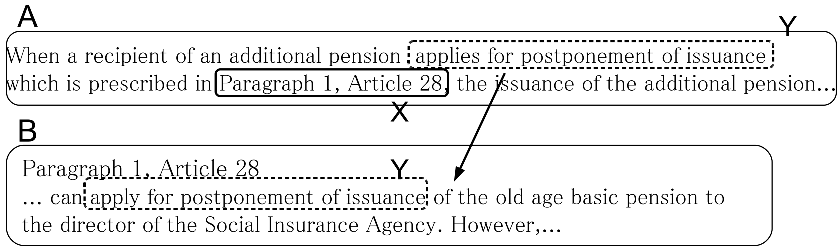

A
Y
When a recipient of an additional pension applies for postponement of issuance
which is prescribed in Paragraph 1, Article 28, the issuance of the additional pension...
X
B
Paragraph 1, Article 28
Y
... can apply for postponement of issuance of the old age basic pension to
the director of the Social Insurance Agency. However,...

Fig. 5. (A) a reference expression, and (B) a referent sentence

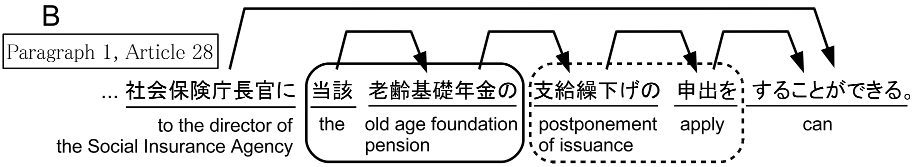
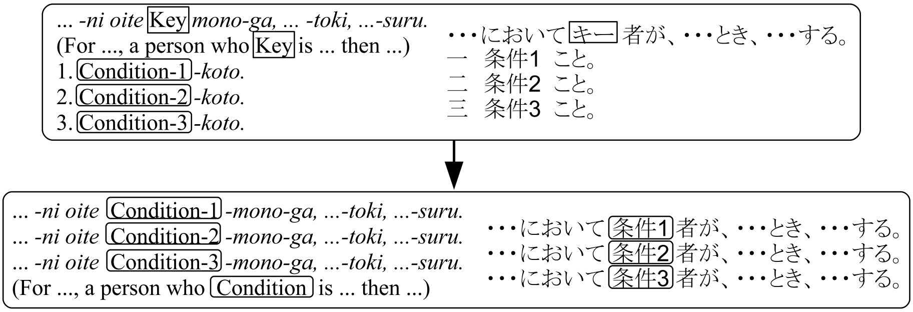


Fig. 6. The dependency tree of the referent sentence

4.2 Removing Itemization

In Section 3.2, we defined Key Phrases as cue phrases that always appear with itemization, like "*tsugi-no kaku gou no izureka ni gaitou-suru* ((something) to

[5] In Japanese, the verb 'apply' is expressed as a *sahen*-noun.



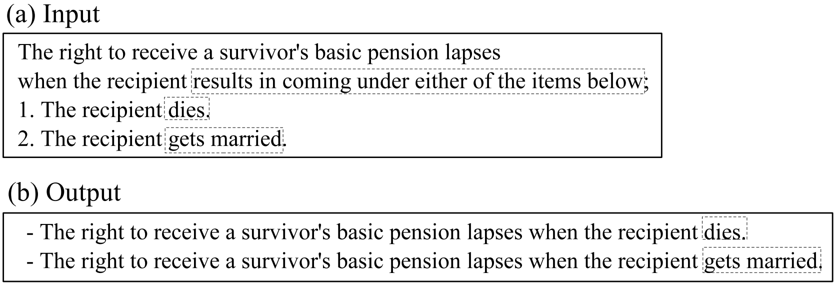
... *-ni oite* Key *mono-ga, ... -toki, ...-suru.*
(For ..., a person who Key is ... then ...)
1. Condition-1 *-koto.*
2. Condition-2 *-koto.*
3. Condition-3 *-koto.*

・・・においてキー者が、・・・とき、・・・する。
一　条件1 こと。
二　条件2 こと。
三　条件3 こと。

... *-ni oite* Condition-1 *-mono-ga, ...-toki, ...-suru.*
... *-ni oite* Condition-2 *-mono-ga, ...-toki, ...-suru.*
... *-ni oite* Condition-3 *-mono-ga, ...-toki, ...-suru.*
(For ..., a person who Condition is ... then ...)

・・・において条件1者が、・・・とき、・・・する。
・・・において条件2者が、・・・とき、・・・する。
・・・において条件3者が、・・・とき、・・・する。

Fig. 7. Removing itemization



(a) Input

The right to receive a survivor's basic pension lapses
when the recipient results in coming under either of the items below;
1. The recipient dies.
2. The recipient gets married.

(b) Output

- The right to receive a survivor's basic pension lapses when the recipient dies.
- The right to receive a survivor's basic pension lapses when the recipient gets married.

Fig. 8. An example of removing itemization

which either of the following items is applicable)," and we search for itemization with it. If a Key Phrase is found, we regard the following items as Condition Items, and replace the Key Phrase with one of the Condition Items for each. Then we have sentences which are understandable separately[6], as shown in Fig. 7. We show an example of the pair of input and output in Fig. 8.

5 Experiments and Results

5.1 Reference Expressions in National Pension Law

We tested our system on reference phrases "X-*ni kitei-suru* Y (Y which is prescribed in X)" in National Pension Law. The result is shown in Table 4. The system derived correct information from 41.5 percent of reference phrases in National Pension Law. For 20.8 percent of reference expressions, generated sentences were ungrammatical or not enough, since some necessary words or phrases were not expressed in output sentences. For example, some referent sentences contain a number of reference expressions. An example is shown in Fig. 9. In

[6] Even though the converted logical formulae are repetitive, there is no problem as long as the system gives the same logical predicates and variables to the repetitive phrases.

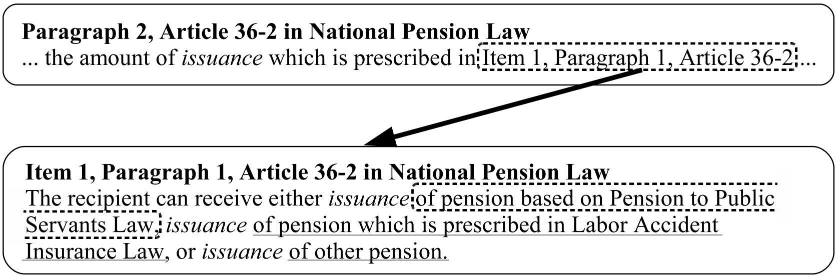

Paragraph 2, Article 36-2 in National Pension Law
... the amount of issuance which is prescribed in Item 1, Paragraph 1, Article 36-2 ...
Item 1, Paragraph 1, Article 36-2 in National Pension Law
The recipient can receive either issuance of pension based on Pension to Public Servants Law, issuance of pension which is prescribed in Labor Accident Insurance Law, or issuance of other pension.

Fig. 9. An example of partially extracted reference expressions

Table 4. Result for the reference expression "X-*ni kitei-suru* Y"

	Sentence	%
Extracted correctly	22	41.5%
Extracted partially	11	20.8%
Extracted nothing	20	37.7%
Total	53	100%

Table 5. Result for identifying itemization

Identifying	Itemization Frequency	Conditions
Succeeded	33	119
Failed	1	5
Total	34	124
Misidentify	1	2

Paragraph 2, Article 36-2 in National Pension Law, there exists a phrase referring to Item 1, Paragraph 1, Article 36-2, in which three phrases should be referred such as (1) issuance of pension based on Pension to Public Servants Law, (2) issuance of pension which is prescribed in Labor Accident Insurance Law, and (3) issuance of other pension. Even though all of them are an explanation of the reference expression, the system extracts only the one of them which appears first in the sentence, and ignores the rest of expressions. Therefore, the generated phrase became "the amount of issuance of pension based on Pension to Public Servants Law." We judged the result to be partially extracted.

5.2 Experiment for Itemization

We tested our system on itemization in National Pension Law. From the point of view of identifying itemization, our system found most itemization structures, shown in Table 5. The result of removing itemization is shown on the left hand side of Table 6.

All of the errors are items which denote a combination of a Condition Item and an object part in the law effectuation part, which are separated by space. In other words, the objects of these sentences change depending on the Condition Items. An example is shown in Fig. 10. This article determines the revision of the rate after the base year about the national pension. An important thing here is that each item consists of a condition part and its result. That is, the first Key Phrase denoting "In the case of the following items," enclosed corresponds to the first phrase of each item, while the second Key Phrase denoting "on the

Table 6. Result for removing itemization

	National Pension Law		Income Tax Law	
	conditions	%	conditions	%
Succeeded	87	73.1%	219	51.4%
Wrong sentence	21	17.7%	123	28.9%
Error	11	9.2%	84	19.7%
Total	119	100%	426	100%

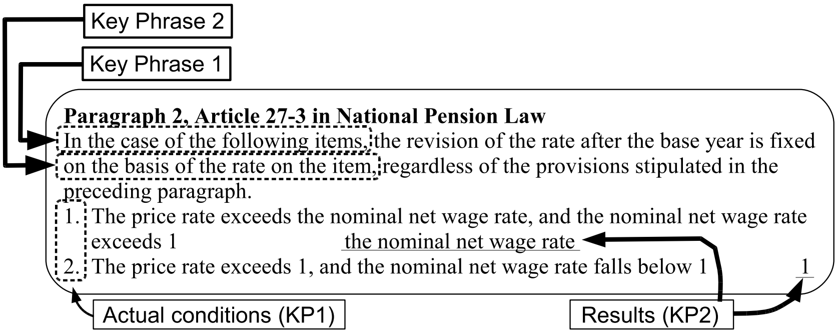

Key Phrase 2
Key Phrase 1
Paragraph 2, Article 27-3 in National Pension Law
In the case of the following items, the revision of the rate after the base year is fixed
on the basis of the rate on the item, regardless of the provisions stipulated in the
preceding paragraph.
1. The price rate exceeds the nominal net wage rate, and the nominal net wage rate
exceeds 1
the nominal net wage rate
2. The price rate exceeds 1, and the nominal net wage rate falls below 1
1
Actual conditions (KP1)
Results (KP2)

Fig. 10. An example of wrong sentence

basis of the rate on the item" corresponds to the second phrases underlined of each item. Therefore, the first item should be interpreted as follows: "When the price rate exceeds the nominal net wage rate, and the nominal net wage rate exceeds 1, the revision of the rate after the base year is fixed to the nominal net wage rate." Our system did not deal with this type of itemization.

We also inspected the system with Income Tax Law as an open test, shown on the right hand side of Table 6. The system was 51.4% accurate in terms of removing itemized expressions, while it was 73.1% accurate in the closed test. There seems to be some difference in notation between National Pension Law and Income Tax Law. Particularly, we found the increase of itemization consisting of a combination of a Condition Item and an object part as mentioned in Fig. 10. Results will be improved after an analysis of the mistakes.

6 Discussion

Our purpose is to transform law sentences into logical forms which are able to be provided for advanced inference in Legal Engineering. We can think of alternative ways to solve the problem which was dealt with in this paper. Thus, it could be a method that the expansion of itemized expressions is done on the logical forms instead of on natural language sentences as this paper. That is, as a first step, each referent sentence such as "The recipient dies." is transformed into a logical form, then the predicate transformed from an itemized expression such as "items_below(x6)" is replaced with the transformed referents. The expansion in this method might be easier because the reference and referent expressions are normalized as logical forms before the expansion process. A conceivable problem would occur in terms of how to associate variables of referent logical predicates with reference ones.

Let us consider advantages of both this alternative way and our proposed method. Advantages of our method are as follows:

1. We can independently develop our method from the main system 'Wildcats.'
2. Our system can generate a natural language text with high readability. It could be a spin-off dealing with other problems like a text-to-speech system.

In fact, the first item was the most important reason because our main system, 'Wildcats,' has been under development.

Meanwhile, the alternative method could have the following advantages:

1. The system need not care about grammar of Japanese unlike our method which sometimes generated ungrammatical sentences.
2. Generated logical forms could be more accurate than the current system, because the system need not analyze generated long sentences with dependency parser.

The best way would be to merge these two approaches. Anyway, it is effective to extract reference phrases by the pattern-match with a regular expression.

7 Conclusion

In this paper we proposed a method to rewrite legal sentences including itemization or reference into independent, plain sentences. In the experiments, we showed that the system successfully extracted itemized expressions with some exceptions. For referential expressions, focusing on a referential phrase "X-*ni kitei-suru* Y" in National Pension Law, we showed the system worked well for extracting reference expressions. We consider that the system is useful not only for the front end of our main system, Wildcats, but also for assistance in reading legal documents.

Some tasks still remain in our future work: (1) As was shown in Section 5, our system failed for some sentences. We can deal with some of the failures easily. (2) We can improve this system by introducing a method for measuring readability of the output sentences. (3) We will test our main system, Wildcats, using the proposed model as the front end system.

Acknowledgment. This research was partly supported by the 21st Century COE Program 'Verifiable and Evolvable e-Society' and Grant-in-Aid for Scientific Research (19650028). We would like to give special thanks to Yoko Shibata, Yoshiko Oyama, and Akiko Hayashi, who analyzed law sentences and their logical representations. We also thank the anonymous three referees for their valuable comments that greatly improved the quality of this paper.

References

1. Katayama, T.: The current status of the art of the 21st COE programs in the information sciences field (2): Verifiable and evolvable e-society - realization of trustworthy e-society by computer science - (in Japanese). IPSJ (Information Processing Society of Japan) Journal 46(5), 515–521 (2005)
2. Katayama, T.: Legal engineering – an engineering approach to laws in e-society age. In: Proc. of the 1st Intl. Workshop on JURISIN (2007)
3. Katayama, T., Shimazu, A., Tojo, S., Futatsugi, K., Ochimizu, K.: e-Society and legal engineering (in Japanese). Journal of the Japanese Society for Artificial Intelligence 23(4), 529–536 (2008)
4. Hagiwara, S., Tojo, S.: Stable legal knowledge with regard to contradictory arguments. In: AIA 2006: Proceedings of the 24th IASTED international conference on Artificial intelligence and applications, Anaheim, CA, USA, pp. 323–328. ACTA Press (2006)
5. Nakamura, M., Nobuoka, S., Shimazu, A.: Towards Translation of Legal Sentences into Logical Forms. In: Satoh, K., Inokuchi, A., Nagao, K., Kawamura, T. (eds.) JSAI 2007. LNCS, vol. 4914, pp. 349–362. Springer, Heidelberg (2008)
6. Hobbs, J.R., Stickel, M., Martin, P., Edwards, D.: Interpretation as abduction. In: Proceedings of the 26th annual meeting on Association for Computational Linguistics, Morristown, NJ, USA. Association for Computational Linguistics, pp. 95–103 (1988)
7. Mulkar, R., Hobbs, J.R., Hovy, E.: Learning from reading syntactically complex biology texts. In: Proceedings of the 8th International Symposium on Logical Formalizations of Commonsense Reasoning. Part of the AAAI Spring Symposium Series (2007)
8. Mulkar, R., Hobbs, J.R., Hovy, E., Chalupsky, H., Lin, C.Y.: Learning by reading: Two experiments. In: Proceedings of IJCAI 2007 workshop on Knowledge and Reasoning for Answering Questions (2007)
9. Tanaka, K., Kawazoe, I., Narita, H.: Standard structure of legal provisions - for the legal knowledge processing by natural language - (in Japanese). In: IPSJ Research Report on Natural Language Processing, pp. 79–86 (1993)
10. Nagai, H., Nakamura, T., Nomura, H.: Skeleton structure acquisition of Japanese law sentences based on linguistic characteristics. In: Proc. of NLPRS 1995, vol. 1, pp. 143–148 (1995)
11. Kurohashi, S., Nakamura, T., Matsumoto, Y., Nagao, M.: Improvements of Japanese morphological analyzer JUMAN. In: Proceedings of the Workshop on Sharable Natural Language Resources, pp. 22–28 (1994)
12. Kurohashi, S., Nagao, M.: KN parser: Japanese dependency/case structure analyzer. In: Proceedings of the Workshop on Sharable Natural Language Resources, pp. 48–55 (1994)
13. Ogawa, Y., Inagaki, S., Toyama, K.: Automatic Consolidation of Japanese Statutes Based on Formalization of Amendment Sentences. In: Satoh, K., Inokuchi, A., Nagao, K., Kawamura, T. (eds.) JSAI 2007. LNCS, vol. 4914, pp. 349–362. Springer, Heidelberg (2008)