A Study on Statistical Machine Translation of Legal Sentences

by

BUI THANH HUNG

submitted to

Japan Advanced Institute of Science and Technology

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Supervisor: **Professor AKIRA SHIMAZU**

School of Information Science

Japan Advanced Institute of Science and Technology

August, 2013

Abstract

Machine translation is the task of automatically translating a text from one natural language into another. Statistical machine translation (SMT) is a machine translation paradigm where translations are generated on the basis of statistical models whose parameters are derived from the analysis of bilingual text corpora (Philipp Koehn, 2010). Many translation models of statistical machine translation such as word-based, phrase-based, syntax-based, a combination of phrase-based and syntax-based translation, and hierarchical phrase-based translation are proposed. Phrase-based and hierarchical-phrase-based model (tree-based model) have become the majority of research in recent years, however they are not powerful enough to legal translation. Legal translation is the task of how to translate texts within the field of law. Translating legal texts automatically is one of the difficult tasks because legal translation requires exact precision, authenticity and a deep understanding of law systems. The problem of translation in the legal domain is that legal texts have some specific characteristics that make them different from other daily-use documents as follows:

- Because of the meticulous nature of the composition (by experts), sentences in legal texts are usually long and complicated.
- In several language pairs such as English-Japanese the target phrase order differs significantly from the source phrase order, selecting appropriate synchronous context-free grammars translation rule (SCFG) to improve phrase-reordering is especially hard in the hierarchical phrase-based model
- The terms (name phrases) for legal texts are difficult to translate as well as to understand.

Therefore, it is necessary to find ways to take advantage to improve legal translation. To deal with three problems mentioned above, we propose a new method for translating a legal sentence by dividing it based on the logical structure of a legal sentence, using rule selection to improve phrase-reordering for the tree-based machine translation, and propose sentence paraphrasing and named entity to increase translation.

A legal sentence represents a requisite and its effectuation (Tanaka et al. 1993). If each part of the legal sentence is shown separately, the readability will increase especially for a long

sentence as seen in administrative laws. Such parts are recognized automatically by dividing a legal sentence according to the requisite-effectuation structure as described in this thesis. Furthermore, each fragment obtained by the dividing is shorter than the original sentence and the translation quality is expected to be improved. For the first problem mentioned above, we propose dividing and translating legal text basing on the logical structure of a legal sentence. The existing methods for dividing a sentence are mainly based on clause splitting and not be based on the requisite-effectuation structure. We recognize the logical structure of a legal sentence using statistical learning model with linguistic information. Then we segment a legal sentence into parts of its structure and translate them with statistical machine translation models. In this study, we applied the phrased-based and the tree-based models separately and evaluated them with baseline models.

Rule selection is important to tree-based statistical machine translation systems. This is because a rule contains not only terminals (words or phrases), but also nonterminals and structural information. During decoding, when a rule is selected and applied to a source text, both lexical translations (for terminals) and reorderings (for nonterminals) are determined. Therefore, rule selection affects both lexical translation and phrase reorderings. For the second problem, we propose a maximum entropy-based rule selection model for the tree-based model, the maximum entropy-based rule selection model combines local contextual information around rules and information of sub-trees covered by variables in rules.

For the last problem, we propose sentence paraphrasing and named entity approaches. We apply a monolingual sentence paraphrasing method for augmenting the training data for statistical machine translation systems by creating it from data that is already available. We generate named-entity recognition (NER) training data automatically from a bilingual parallel corpus, employ an existing high-performance English NER system to recognized named entities at the English side, and then project the labels to the Japanese side according to the word alignment. We split the long sentence into several block areas that could be translates independently.

We integrate dividing a legal sentence based on its logical structure into the first step of rule selection as well as sentence paraphrasing and named entity. With this method, our experiments on legal translation show that the method achieves better translations.

Keywords: *phrase-based machine translation*; *tree-based machine translation*; *logical structure of a legal sentence*; *CRFs*; *Maximum Entropy Model*, *rule selection*; *linguistic and contextual information*; *sentence paraphrasing*, *NER*

Acknowledgments

Firstly, I would like to thank my supervisor, Professor Akira Shimazu for his kindly guidance, warn encouragement and helpful support. He has given me much invaluable knowledge not only how to formulate research ideal or to write a good paper but also the vision and much useful experiment in the academic life.

I would like to thank Professor Kiyoaki Shirai, who has been discussing and giving me inspirations.

I would like to thank Professor Hiroyuki Iida for his help in my sub-theme research. He has given me as good as possible conditions for my work during this time.

I would like to thank Associate Professor Nguyen Le Minh. He is a respectable dedicated person. He always gave me all the time and supported everything I needed from using software tools to listening to my problems, making kind suggestion.

I also appreciate the help and the encouragement from professor Ho Tu Bao, professor Duong Anh Duc, professor Le Hoai Bac, professor Dinh Dien and many other faculty members of Ho Chi Minh University of Science and Ha Noi University of Technology.

A special thank to colleagues and friends in Shimazu-Lab, Shirai-Lab and in JAIST from the first day I came to Japan. I have received a lot of help from them. They gave me invaluable advices, comments, and most importantly cheered me up all the time.

I am deeply indebted to the Ministry of Education and Training of Vietnam for granting me a scholarship. Thanks also to the JAIST Foundation for providing me with their travel grants which supported me to attend and present my work at international conferences

I would like to thank my friends, all members of my family for sharing my happiness, difficulties all the time and supporting me as always. Finally I have to give a big thank you to my wife, my son and my daughter, without their encouragements I would never have began, and much less completed this thesis.

Content

List of Figures

List of Tables

1 Introduction

In this chapter we briefly address the research context, the research motivations, as well as the major contributions of the thesis. First, we introduce the Machine Translation approaches. Second, we state the research motivation which the thesis focuses to solve. Third, we present the main contribution of the thesis. Finally, we outline the structure of the thesis

1.1 Machine Translation

Machine translation (MT) is the task of automatically translating a text from one natural language into another. The ideal of machine translation can be traced back to the seventeenth century, but it became realistically possible only in the middle of the twentieth century (Hutchins, 2005). Soon after the first computers were developed, researchers began on MT algorithms. The earlier MT systems consisted primarily of large bilingual dictionaries and sets of translation rules. Dictionaries were used for word level translation, while rules controlled higher level aspects such as word order and sentence organization. Starting from a restricted vocabulary or domain, rule based systems proved useful. But as the study progressed, researchers found that it is extremely hard for rules to cover the complexity of natural language, and the output of the MT systems were disappointing when applied to larger domains. Little breakthrough was made until the late 1980's, when the increase in computing power made statistical machine translation (SMT) based on bilingual language corpora possible. In the beginning, much scepticism about SMT existed from the traditional MT community because people doubled whether statistical methods based on counting and mathematical equations can be used for the sophisticated linguistic problem. However, the potential of SMT was justified by pioneering experiments carried out at IBM in the early 1990s (Brown et al., 1993). Since then the statistical approach has become the dominant method in MT research.

Several criteria can be used to classify machine translation approaches, yet the most popular classification is done attending to the level of linguistic analysis (and generation) required by the system to produce translations. Usually, this can be graphically expressed by the machine translation pyramid in Figure 1.1.

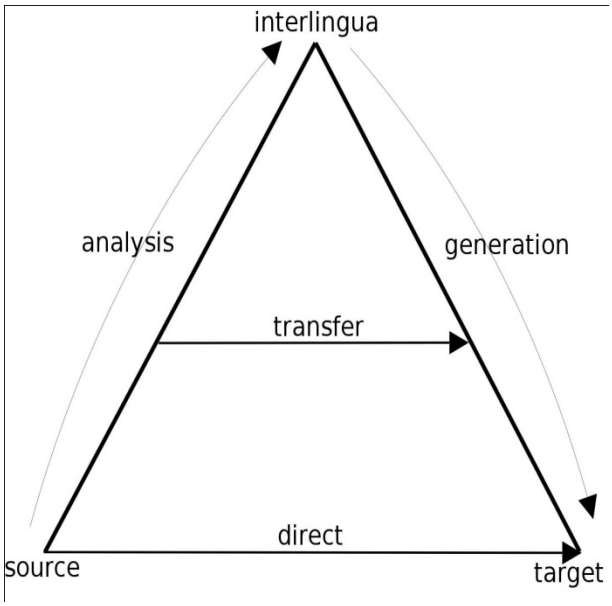


Figure 1.1: The machine translation pyramid

Generally speaking, the bottom of the pyramid represents those systems which do not perform any kind of linguistic analysis of the source sentence in order to produce a target sentence. Moving upwards, the systems which carry out some analysis (usually by means of morphosyntax-based rules) are to be found. Finally, on top of the pyramid a semantic analysis of the source sentence turns the translation task into generating a target sentence according to the obtained semantic representation.

Aiming at a bird's-eye survey rather than a complete review, next each of these approaches is briefly discussed, before delving into the statistical approach to machine translation.

Direct translation

This approach solves translation on a word-by-word basis, and it was followed by the early MT systems, which included a very shallow morphosyntactic analysis. Today, this preliminary approach has been abandoned, even in the framework of corpus-based approaches.

Transfer-based translation

The rationale behind the transfer-based approach is that, once we grammatically analyze a given sentence, we can pass this grammar on to the grammatical representation of this sentence in another language. In order to do so, rules to convert source text into some

structure, rules to transfer the source structure into a target structure, and rules to generate target text from it are needed. Lexical rules need to be introduced as well.

Usually, rules are collected manually, thus involving a great deal of expert human labour and knowledge of comparative grammar of the language pair. Apart from that, when several competing rules can be applied, it is difficult for the systems to prioritize them, as there is no natural way of weighing them.

This approach was massively followed in the 1980s, and despite much research effort, high-quality MT was only achieved for limited domains (Hut, 1992).

Interlingua-based translation

This approach advocates for the deepest analysis of the source sentence, reaching a language of semantic representation named Interlingua. This conceptual language, which needs to be developed, has the advantage that, once the source meaning is captured by it, in theory we can express it in any number of target languages, so long as a generation engine for each of them exists.

Though conceptually appealing, several drawbacks make this approach unpractical. On the one hand, the difficulty of creating a conceptual language capable of bearing the particular semantics of all languages is an enormous task, which in fact has only been achieved in very limited domains. Apart from that, the requirement that the whole input sentence needs to be understood before proceeding onto translating it, has proved to make these engines less robust to the grammatical incorrectness of informal language, or which can be produced by an automatic speech recognition system.

Corpus-based approaches

In contrast to the previous approaches, these systems extract the information needed to generate translations from parallel corpora that include many sentences which have already been translated by human translators. The advantage is that, once the required techniques have been developed for a given language pair, in theory it should be relatively simple to transpose them to another language pair, so long as sufficient parallel training data is available.

Among the many corpus-based approaches that sprung at the beginning of the 1990s, the most relevant ones are example-based (EBMT) and statistical (SMT), although the differences between them are constantly under debate. Example-based MT makes use

of parallel corpora to extract a database of translation examples, which are compared to the input sentence in order to translate. By choosing and combining these examples in an appropriate way, a translation of the input sentence can be provided.

In SMT, this process is accomplished by focusing on purely statistical parameters and a set of translation and language models, among other data-driven features. Although this approach initially worked on a word-to-word basis and could therefore be classified as a direct method, nowadays several engines attempt to include a certain degree of linguistic analysis into the SMT approach, slightly climbing up the aforementioned MT pyramid.

The following section further introduces about the statistical approach to machine translation.

1.1.1 Statistical Machine Translation

Statistical machine translation (SMT) is a machine translation paradigm where translations are generated on the basis of statistical models whose parameters are derived from the analysis of bilingual text corpora.

The first ideas of statistical machine translation were introduced by Warren Weaver in 1949, including the ideas of applying Claude Shannon's information theory. Statistical machine translation was re-introduced in 1991 by researchers at IBM's Thomas J. Watson Research Center and has contributed to the significant resurgence in interest in machine translation in recent years.

A statistical machine translation system based on the noisy channel model consists of three components: a language model (LM), a translation model (TM), and a decoder. For a system which translates from a foreign language F to English E, the LM gives a prior probability $P(E)$ and the TM gives a channel translation probability $P(F|E)$. These models are automatically trained using monolingual and bilingual corpora. A decoder then finds the best English sentence given a foreign sentence that maximizes $P(E|F)$, which also maximizes $P(F|E)P(E)$ according to Bayes' rule. That is, the most appropriate foreign translation is obtained by:

$$E^* = \arg\max_E P(E \mid F) = \arg\max_E \frac{P(F \mid E)P(E)}{P(F)} \qquad (1.1)$$

Since $P(F)$ is constant for the given F, it can be rewritten as Equation 1.2:

$$E^* = \arg\max_E P(F \mid E)P(E) \qquad (1.2)$$

Here, $P(F|E)$ is the translation model and $P(E)$ is the language model. Fig. 1.2 shows the structure of typical statistical machine translation system. Architecture of the statistical machine translation approach based on Bayes' rule is shown in Fig. 1.3.

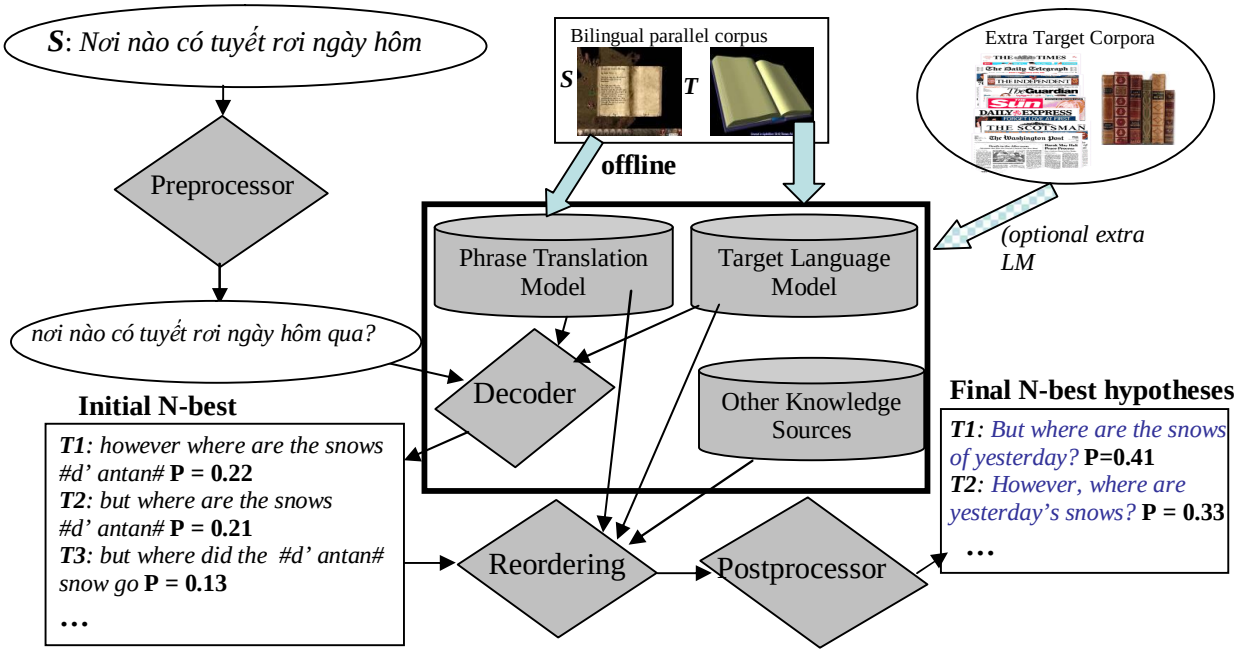


Figure 1.2: Structure of typical statistical machine translation system

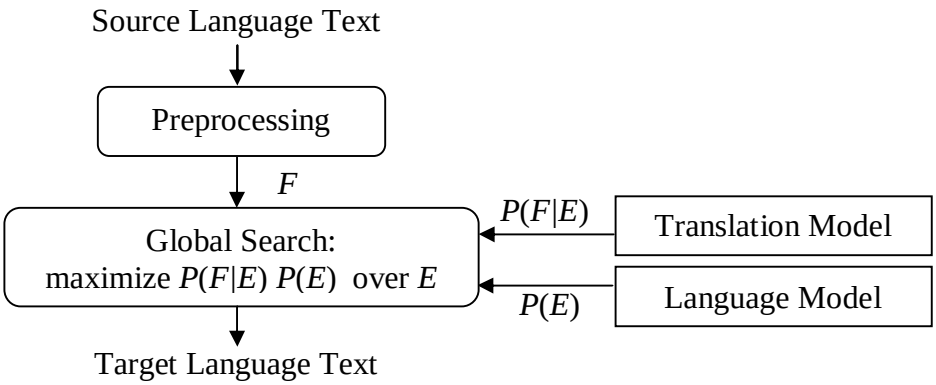


Figure 1.3: Architecture of the statistical machine translation approach based on Bayes' decision rule.

1.1.2 Machine Translation in Legal Domain

In recent year, a new research field called Legal Engineering was proposed in order to achieve a trustworthy electronic society.

The legal domain has continuous publishing and translation cycles, large volumes of digital content and growing demand to distribute more multilingual information. It is necessary to handle a high volume of translations quickly. Currently, a certified translation of a legal judgment takes several months to complete. Afterwards, there is a significant delay between the publication of a judgment in the original language and the availability of its human translation into the other official language.

Because the high quality of the machine translation system obtained, developed and trained specifically on the legal corpora, opens further opportunities. Machine translations could be considered as first drafts for official translations that would only need to be revised before their publication. This procedure would thus reduce the delay between the publication of the decision in the original language and its official translation. It would also provide opportunities for saving on the cost of translation.

However, translating legal texts automatically is one of the difficult tasks and there is little research about it as Atefeh and Guy (2008, 2009). Almost all of this research only focused on building the system based on open baseline systems and evaluating the result of the systems.

In this research, we propose a new method for translating a legal sentence by dividing it based on the logical structure of a legal sentence, using rule selection to improve phrase-reordering for the tree-based machine translation, and propose sentence paraphrasing and named entity for legal translation. Our experiment shows that our proposed method archives better translation quality.

1.2 Motivation and Problem

Because the high quality of the machine translation system obtained, developed and trained specifically on the legal corpora. Machine translation in legal domain is increasing in recent years. Building a high-quality machine translation to help official translations before their publication is necessary. It would also provide opportunities for saving on the cost of translation, reduce the time, propagate public as well as support understanding law.

However, translating legal texts automatically is one of the difficult tasks and there is little research about it. The problem of translation in the legal domain is that legal texts have some specific characteristics that make them different from other daily-use documents as follows:

- Because of the meticulous nature of the composition (by experts), sentences in legal texts are usually long and complicated.
- In several language pairs such as English-Japanese the target phrase order differs significantly from the source phrase order, selecting appropriate synchronous context-free grammars translation rule (SCFG) to improve phrase-reordering is especially hard in the hierarchical phrase-based model
- The terms (name phrases) for legal texts are difficult to translate as well as to understand.

Therefore, it is necessary to find ways to take advantage to improve legal translation. To deal with three problems mentioned above, we propose a new method for translating a legal sentence by dividing it based on the logical structure of a legal sentence, using rule selection to improve phrase-reordering for the tree-based machine translation, proposing sentence paraphrasing and named entity to improve machine translation.

Because machine translation can work well for simple sentences but a machine translation system faces difficulty while translating long sentences, as a result the performance of the system degrades. Most legal sentences are long and complex, the translation model has a higher probability to fail in the analysis, and produces poor translation results. One possible way to overcome this problem is to divide long sentences to smaller units which can be translated separately. There are several approaches on splitting long sentences into smaller segments in order to improve the translation. Splitting can be done either at the translation testing phase or translation model training phase. These approaches are different in a method.

Our approach is different from those of previous works. We propose a new method using the logical structure of a legal sentence to split legal sentences. We use characteristics and linguistic information of legal texts to split legal sentences into logical structures.

Bach et al. (2010) used Conditional Random Fields (CRFs) to recognize the logical structure of a Japanese legal sentence. We use the same way to recognize the logical

structure of a legal text sentence for Japanese. For an English sentence, we propose new features to recognize its logical structure. The logical structure of a legal sentence by the recognition task will be used to split long sentences. Our approach is useful for legal translation. It will reserve a legal sentence structure, reduce the analysis in deciding the correct syntactic structure of a sentence, remove ambiguous cases in advanced and promise results.

The syntax-based statistical machine translation model uses rules with hierarchical structures as translation knowledge, which can capture long-distance re-orderings. Typically, a translation rule consists of a source side and a target side. However, the source side of a rule usually corresponds to multiple target-sides in multiple rules. Therefore, during decoding, the decoder should select correct target-side for a source side. This is rule selection.

Rule selection is of great importance to syntax-based statistical machine translation systems. This is because that a rule contains not only terminals (words or phrases), but also non-terminals and structural information. During decoding, when a rule is selected and applied to a source text, both lexical translations (for terminals) and re-orderings (for non-terminals) are determined. Therefore, rule selection affects both lexical translation and phrase re-orderings. However, most of the current tree-based systems ignore contextual information when they select rules during decoding, especially the information covered by non-terminals. This makes the decoder hardly to distinguish rules. Intuitively, information covered by non-terminals as well as contextual information of rules is believed to be helpful for rule selection.

In this work, we present rule selection for tree-based statistical machine translation, we propose a maximum entropy-based rule selection model for tree-based statistical machine translation. The maximum entropy-based rule selection model combines local contextual information around rules and information of sub-trees covered by variables in rules. Therefore, our model allows the decoder to perform context-dependent rule selection during decoding. We integrate dividing a legal sentence based on its logical structures into the rule selection as the first step and incorporate the maximum entropy-based rule selection model into a state-of-the-art linguistically tree-based English-Japanese statistical

machine translation model. Experiments show that our approach archives significant improvements over the baseline system.

Statistical machine translation (SMT) systems learn how to translate by analyzing bilingual parallel corpora. Generally speaking, high-quality translations can be produced when ample training data is available. However, because of low density of legal language pairs that do not have large-scale parallel corpora, limited amount of training data usually leads to a problem of low coverage in that many phrases encountered at run-time have not been observed in the training data. This problem becomes more serious for higher-order n-grams, and for morphologically richer languages. To overcome the coverage problem of SMT we investigate using sentence paraphrasing and named entity approaches. We propose a monolingual sentence paraphrasing method for augmenting the training data for statistical machine translation systems by creating it from data that is already available. The terms (name phrases) for legal texts are difficult to translate as well as to understand, so we apply named entity for splitting the long sentence into several named entities that could be translates independently. We generate NER training data automatically from a bilingual parallel corpus, employ an existing high-performance English NER system to recognized NEs at the English side, and then project the labels to the Japanese side according to the word alignment. We integrate dividing a legal sentence based on its logical structures into sentence paraphrasing and named entity as the first step. Our proposed method improves the translation quality.

1.3 Main Contribution

We propose three methods to deal with three problems of legal translation mentioned above.

Firstly, to solve the first problem: sentences in legal texts are usually long and complicate, we propose a novel method for translating a legal sentence by dividing it based on the logical structure of a legal sentence. We first recognize the logical structure of a legal sentence using statistical learning model with linguistic information. Then we segment a legal sentence into parts of its structure and translate them with statistic machine translation models. In this study, we applied the phrased-based and the tree-based models separately and evaluated them with baseline models. With this method, our experiments on

Japanese-to-English and English-to-Japanese translations show that the method achieves better translations on measuring by the BLEU, NIST and TER scores. The subjective evaluation also shows better results.

Secondly, solving the problem in several language pairs such as English-Japanese the target phrase order differs significantly from the source phrase order, selecting appropriate synchronous context-free grammars translation rule (SCFG) to improve phrase-reordering is especially hard in the tree-based model, we propose using rich linguistic and contextual information for rule selection specifically:

- We integrate dividing a legal sentence based on its logical structures into the rule selection as the first step .
- We use rich linguistic and contextual information for both non-terminals and terminals. Linguistic and contextual information around terminals have never been used before, we see that these new features are very useful for selecting appropriate translation rules if we integrate them with the features of non-terminals.
- We propose a simple and sufficient algorithm for extracting features in rule selection.
- We use Moses-chart to extract translation rules with rich linguistic and contextual information. Moses-chart system is a tree-based model developed by many machine translation experts and used in many systems, so that, our model is more generic.
- We use a simple way to classify features by using maximum entropy-based rule selection model and incorporate this model into a state-of-the-art syntax-based SMT model, the tree-based model (Moses-chart system). We obtain substantial improvements over the Moses-chart system.

Lastly, with the problem the terms (name phrases) for legal texts are difficult to translate as well as to understand, we propose sentence paraphrasing and named entity approaches. We apply a monolingual sentence paraphrasing method for augmenting the training data for statistical machine translation systems by creating it from data that is already available. We generate NER training data automatically from a bilingual parallel

corpus, employ an existing high-performance English NER system to recognized NEs at the English side, and then project the labels to the Japanese side according to the word alignment. We apply splitting the long sentence into several block areas that could be translates independently. We integrate dividing a legal sentence based on its logical structures into the sentence paraphrasing and named entity as the first step. Our proposed method achieves better translation quality.

1.4 Thesis Structure

This chapter presents an overview of the thesis, including an introduction of statistical machine translation, the motivation and problems of the thesis and our contributions. The rest of this thesis is organized as follows:

Chapter 2 presents the important background information for the thesis, such as the theory of statistical machine translation, reviewing the most widely used approaches since its introduction in the early 1990s until our days. This chapter introduces phrase-based and tree-based model with using synchronous context-free-grammars. We also review related work and provide detail about our approach.

In chapter 3, we present a new method dividing and translating legal text based on the logical structure of a legal sentence. Translating legal texts automatically is one of the difficult tasks because legal translation requires exact precision, authenticity and a deep understanding of law systems. The problem of translation in the legal domain is that legal texts have some specific characteristics that make them different from other daily-use documents and a legal text is usually long and complicated. In order to improve the legal text translation quality, splitting an input sentence becomes mandatory. This chapter presents a novel method which divides a legal sentence in Japanese/English based on its logical structure and translates into sentences in English/Japanese. Characteristics and linguistic information of legal texts are used to split legal sentences into logical structures. A statistical learning method - Conditional Random Fields (CRFs) with rich linguistic information is used to recognize the logical structure of legal sentence. New features are proposed for recognizing the logical structure of English sentences. The logical structure of a legal sentence is adopted to divide the sentence. The experiments and evaluation are given with promising results.

Chapter 4 presents about rule selection for tree-based statistical translation model. We focus on selecting appropriate translation rules to improve phrase-reordering for the tree-based statistical machine translation, the model operates on synchronous context-free grammars basing on linguistic and contextual information. We propose a simple and sufficient algorithm for extracting features in rule selection. We use Moses-chart to extract translation rules with rich linguistic and contextual information. A simple way is used to classify features by using maximum entropy-based rule selection model. We integrate dividing a legal sentence based on its logical structure into the rule selection and incorporate this model into a tree-based model (Moses-chart). The experiment results with English-Japanese legal sentence pairs show that our method outperforms the baseline Moses-chart, the state-of-the-art syntax-based SMT.

Chapter 5 presents about sentence paraphrasing and named entity for legal translation. In this chapter, we introduce a monolingual sentence paraphrasing method for augmenting the training data for statistical machine translation systems and generating NER training data automatically from a bilingual parallel legal corpus. We create augmenting the training data from data that is already available. We employ an existing high-performance English NER system to recognized NEs at the English side, and then project the labels to the Japanese side according to the word alignment. We apply splitting the long sentence into several block areas that could be translates independently. We integrate dividing a legal sentence based on its logical structures into the sentence paraphrasing and named entity as the first step.

Chapter 6 summarizes the main tasks of the thesis including the main achievements and contributions, as well as the remaining problems. Open problems that are interesting to be solved from this thesis will be mentioned as the future research directions.

2 Background

This Chapter presents the important background information for the thesis, such as the theory of statistical machine translation, reviewing the most widely used approaches since its introduction in the early 1990s until our days. Two famous models: phrase-based and tree-based are introduced. We also review related work and provide detail about our approach.

2.1 Translation Model

2.1.1 Word-Based Translation Model

In word-based translation, the fundamental unit of translation is a word in some natural language. Figure 2.1 illustrates word-based translation

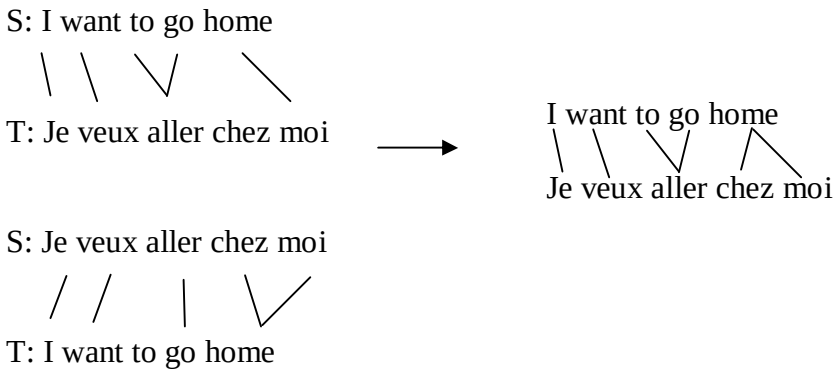


Figure 2.1: The process of word-based translation

Typically, the number of words in translated sentences are different, because of compound words, morphology and idioms. An example of a word-based translation system is the freely available GIZA++ package which includes the training program for IBM models and HMM model and Model 6. The word-based translation is not widely used today; phrase-based systems are more common.

2.1.2 Phrase-Based Translation Model

The phrase-based statistical machine translation extends a basic translation unit from words to phrases. The basic idea of phrase-based translation is to segment a given source sentence into phrases, then translate each phrase and finally compose the target sentence from these phrase translations. Fig. 2.2 illustrates the process of phrase-based translation. The input is segmented into a number of sequences of consecutive words (so-called phrases). Each phrase is translated into an English phrase, and English phrases in the output are reordered.

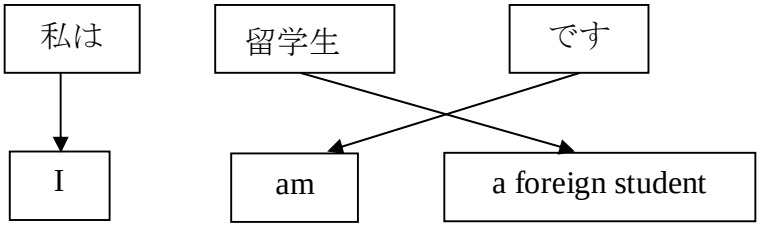


Figure 2.2: Phrase-based machine translation: The input is segmented into phrases, translated one-to-one into phrases in English and possibly reordered.

The phrase translation model is based on the noisy channel model. This model uses Bayes rule to reformulate the translation probability for translating a foreign sentence **f** into English **e** as

$$\text{argmax}_{\mathbf{e}}\, p(\mathbf{e}|\mathbf{f}) = \text{argmax}_{\mathbf{e}}\, p(\mathbf{f}|\mathbf{e})\, p(\mathbf{e}) \qquad (2.1)$$

This allows for a language model e and a separate translation model p(f|e).

During decoding, the foreign input sentence f is segmented into a sequence of *I* phrases f_1^I assuming a uniform probability distribution over all possible segmentations.

Each foreign phrase f_i in f_1^I is translated into an English phrase e_i. The English phrases may be reordered. Phrase translation is modeled by a probability distribution $\varphi(f_i|e_i)$. Recall that due to the Bayes rule, the translation direction is inverted from a modeling standpoint.

Reordering of the English output phrases is modeled by a relative distortion probability distribution $d(start_i,end_{i-1})$, where $start_i$ denotes the start position of the foreign phrase that was translated into the *i*th English phrase, and end_{i-1} denotes the end position of the foreign phrase that was translated into the (*i-1*)th English phrase.

A simple distortion model $d(start_i, end_{i-1}) = \alpha^{|start_i - end_{i-1} - 1|}$ with an appropriate value for the parameter α is used. In order to calibrate the output length, a factor ω (called word cost) is used for each generated English word in addition to the trigram language model p_{LM}. This is a simple means to optimize performance. Usually, this factor is larger than 1, biasing toward longer output.

In summary, the best English output sentence e_{best} given a foreign input sentence f according to this model is

$$e_{best} = \text{argmax\_e } p(e|f) = \text{argmax}_e \, p(f|e) \, p\_LM(e) \, \omega^{length(e)} \quad (2.2)$$

where p(f|e) is decomposed into

$$p(f_1^I|e_1^I) = \Phi_{i=1}^I \varphi(f_i|e_i) \, d(start_i, end_{i-1}) \quad (2.3)$$

2.1.3 Syntax-based Translation Model

Syntax-based translation is based on the idea of translating syntactic units, rather than single words or strings of words (as in phrase-based MT), i.e. (partial) parse trees of sentences/utterances. The idea of syntax-based translation is quite old in MT, though its statistical counterpart did not take off until the advent of strong stochastic parsers in the 1990s. Examples of this approach include DOP-based MT and, more recently, synchronous context-free grammars.

2.1.4 Tree-Based Translation Model

The Tree-based model or the hierarchical phrase-based model (Chiang, 2005; Chiang, 2007) is built on a weighted synchronous context-free grammar (SCFG).

This is a statistical machine translation model that uses hierarchical phrases-phrases that contain subphrases. The model is formally a synchronous context-free grammar but is learned from a parallel text without any syntactic annotations. Thus it can be seen as combining fundamental ideas from both syntax-based translation and phrase-based translation.

A SCFG rule has the following form:

$$X \rightarrow (\alpha, \ \gamma, \sim)$$

Where X is nonterminal, α is an LHS (left-hand side) string consissts of terminal and nonterminal, γ (RHS right-hand side) is the translation of α, ~ defines a one-one correspondence between nonterminals in α and γ. For examples,

(1) $X \rightarrow$ (*phát triển kinh tế, economic development*)

(2) $X \rightarrow$ (X_1 *của* X_2, *the* X_2 *of* X_1)

Rule (1) contains only terminals, which is similar to phrase-to-phrase translation in phrase-based SMT models. Rule (2) contains both terminals and nonterminals, which causes a reordering of phrases.

The tree-based model uses the maximum likehood method to estimate translation probabilities for a phrase pair (α, γ), independent of any other context information.

To perform translation, Chiang uses a log-linear model (Och and Ney, 2002) to combine various features. The weight of a derivation D is computed by:

$$W(D) = \Pi\phi_i(D)^{\lambda}{}_1 \qquad (2.4)$$

Where $\phi_i(D)$ is a feature function and λ_i is the feature weight of $\phi_i(D)$.

During decoding, the decoder searches the best derivation with the lowest cost by applying SCFG rules. However, the rule selections are independence of context information, except the left neighboring *n-1* target words for computing n-gram language model.

An example about partial derivation of a synchronous CFG

If we have a rule:

có X_1 *với* X_2, *have* X_2 *with* X_1

Alignment phrases as:

[Úc] [là] [một] [trong số ít nước] [có] [quan hệ ngoại giao] [với] [Triều Tiên]

[Australia] [is] [one of the new countries] [that have] [diplomatic relations] [with] [North Korea]

We can get the derivation as:

$< S_1 , S_1 >$

$\rightarrow <S_2 \ X_3 , \ S_2 \ X_3>$

$\rightarrow <S_4 \ X_5 \ X_3 , S_4 \ X_5 \ X_3>$

→ <X_6 X_5 X_3 , X_6 X_5 X_3>

→ <*Úc* X_5 X_3 , *Australia* X_5 X_3>

→ <*Úc là* X_5 X_3 , *Australia is* X_5 X_3>

→ <*Úc là một trong số ít nước* X_3 , *Australia is one of the new countries* X_3>

→ <*Úc là một trong số ít nước có* X_7 *với* X_8,
Australia is one of the new countries that have X_7 *with* X_8>

Another example, consider the following input and translation rules:

Input: 彼女は歌手です

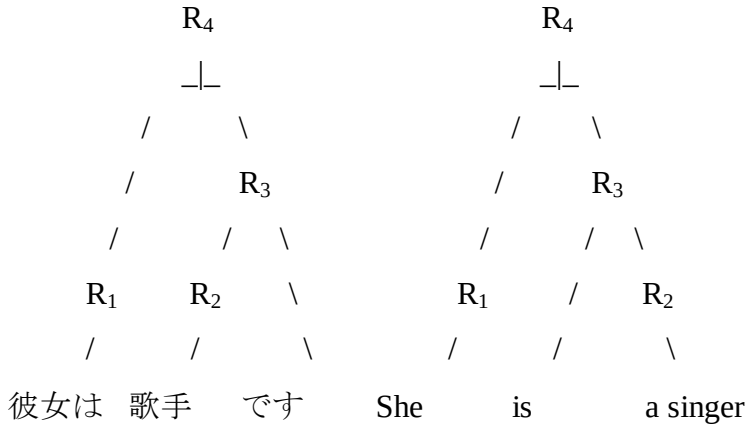
Rules: R_1: 彼女は → *she*

R_2: 歌手 → *a singer*

R_3: X_1 です→ *is* X_1

R_4: X_1 X_2→ X_1 X_2

By applying these rules in the given order, we produce the translation s*he is a singer* in the following fashion:



First the simple phrase mappings (R_1) *彼女は* to s*he* and (R_2) *歌手* to *a singer* are carried out. This allows for the application of the more complex rule (R_3) X_1 です to *is* X_1. The non-terminal which covers the input spanning over 歌手 is *a singer* replaced by a known translation. Finally, the glue rule (R_4) X_1 X_2 to X_1 X_2 combines the two fragments into a complete sentence. Here is how the spans over the input words are filled in:

| 4 --- she is a singer ---------- |
	3--- is a singer ---
1 she	2 a singer	
彼女は	歌手	です

2.1.5 Proposed Model

Though the phrase-based and tree-based translation models have become popular, they are not powerful enough to legal translation. The phrase-based and tree-based translation models work well for simple sentences but for long and complex legal sentences they face difficulty and as a result the performance of the system degrades. In this research, we propose a new model for legal translation by dividing and translating a legal sentence based on the logical structure of a legal sentence. We recognize the logical structure of a legal sentence using statistical learning model with linguistic information. We segment a legal sentence into the parts of its structure. We build the legal translation model in both phrase-based and tree-based translation models, and translate split sentences with these models.

We propose a maximum entropy-based rule selection model for the tree-based model, the maximum entropy-based rule selection model combines local contextual information around rules and information of sub-trees covered by variables in rules.

We use sentence paraphrasing and named entity to improve machine quality. A monolingual sentence paraphrasing method is proposed for augmenting the training data for statistical machine translation systems by creating it from data that is already available. We generate NER training data automatically from a bilingual parallel corpus, employ an existing high-performance English NER system to recognized NEs at the English side, and then project the labels to the Japanese side according to the word alignment. We apply splitting the long sentence into several block areas that could be translates independently.

We integrate dividing a legal sentence based on its logical structures as the first step of the rule selection as well as sentence paraphrasing and named entity.

With this method, our experiments in legal domain show that the method achieves better translations.

2.2 Word Alignment

When describing the phrase-based translation model so far, we did not discuss how to obtain the model parameters, especially the phrase probability translation table that maps foreign phrases to English phrases.

Most recently published methods on extracting a phrase translation table from a parallel corpus start with a word alignment. Word alignment is an active research topic. For instance, this problem was the focus as a shared task at a recent data driven machine translation workshop.

At this point, the most common tool to establish a word alignment is to use the toolkit Giza++ (Och and Ney, 2000). This toolkit is an implementation of the original IBM Models that started statistical machine translation research. However, these models have some serious draw-backs. Most importantly, they only allow at most one English word to be aligned with each foreign word. To resolve this, some transformations are applied.

First, the parallel corpus is aligned bidirectionally, e.g., Vietnamese to English and English to Vietnamese. This generates two word alignments that have to be reconciled. If we intersect the two alignments, we get a high-precision alignment of high-confidence alignment points. If we take the union of the two alignments, we get a high-recall alignment with additional alignment points. See the figure below for an illustration.

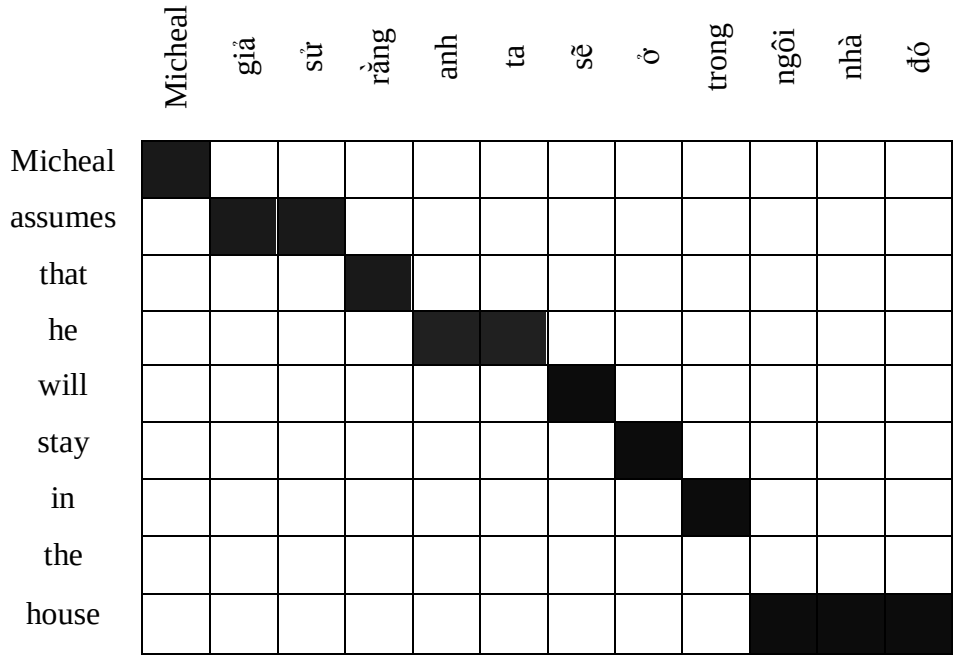


Figure 2-3: Word alignment from English to Vietnamese

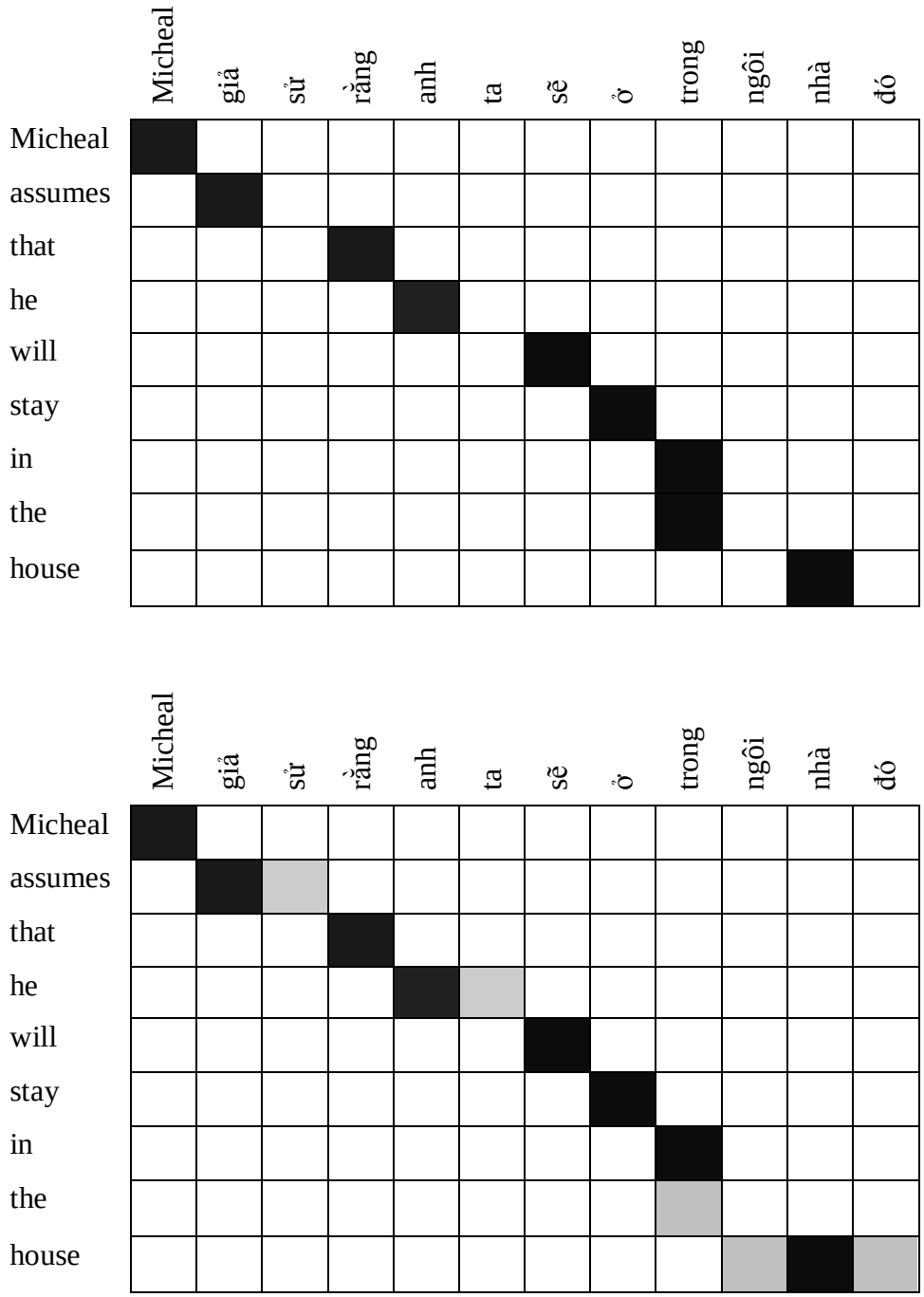


Figure 2-5: Intersection/Union of word alignment

2.3 Language Model

One essential component of any statistical machine translation is the language model, which measures how likely it is that a sequence of word would be uttered by an English speaker. It is easy to see the benefits of such a model. Obviously, we want a machine translation system not only to produce output words that are true to the original in meaning, but also to string them together in fluent English sentences.

In fact, the language model typically does much more than just enable fluent output. It supports difficult decisions about word order and word translation. For instance, a probabilistic language model P_{LM} should prefer correct word order to incorrect word order:

$$P_{LM}(\text{the house is small}) > P_{LM}(\text{small the is house})$$

Formally, a language model is a function that takes an English sentence and returns the probability that it was produced by an English speaker. According to the example above, it is more likely that an English speaker would utter the sentence *the house is small* than the sentence *small the is house*. Hence, a good language model P_{LM} assigns a higher probability to the first sentence.

This preference of the language model helps a statistical machine translation system to find the right word order. Another area where the language model aids translation is word choice. If a foreign word has multiple translations, lexical translation probabilities already give preference to the more common translation. But in specific contexts, other translations maybe preferred. Again, the language model steps in. It gives higher probability to the more natural word choice in context, for instance

$$P_{LM}(\text{I am going home}) > P_{LM}(\text{I am going house})$$

The dominant language modeling methodology is n-gram models. N-gram language models are based on statistics of how likely words are to follow each other. Recall the last example. If we analyze a large amount of text, we will observe that the word home follows the word going more often than the word house does. We will be exploiting such statistics.

Formally, in language modeling, we want to compute the probability if a string $W= w_1, w_2,\ldots, w_n$. Intuitively, p(W) is the probability that if we pick a sequence of English words at random it turns out to be W.

How can we compute p(W)? The typical approach to statistical estimation calls for first collecting a large amount of text and counting how often W occurs in it. So we have to

break down the computation of p(W) into smaller steps, for which we can collect sufficient statistics and estimate probability distributions.

2.4 Decoding

We have a model and estimate for all of our parameters, we can translate new input sentences. This is called decoding. In principle, decoding corresponds solving the maximization problem in Equation:

$$\text{argmax}_{\mathbf{e}}\ p(\mathbf{e}|\mathbf{f}) = \text{argmax}_{\mathbf{e}}\ p(\mathbf{f}|\mathbf{e})\ p(\mathbf{e}) \qquad (2.5)$$

We call this the decision rule. Equation is not the only possible decision rule, although it is by far the most common.

Finding the sentence which maximizes the translation model probability p(f|e) and langue model probability p(**e**) is a search problem, and decoding is thus a kind of search. This is different optimization. Therefore, a primary objective of decoding is to search this space as efficiently as possible. Decoders in machine translation are based on best-first search, a kind of heuristic or informed search; these are search algorithms that are informed by knowledge from the problem domains. Best-first search algorithms select a node n in the search space to explore based on an evaluation function f(n). Machine translation decoder are variants of a specific kind of best-first search called A* search, which based on A* search for speech recognition (Jelinek, 1969). A* search and its variants are commonly called stack decoding in speech recognition and sometime also stack decoding in machine translation.

Following the stack decoding algorithms (Wang and Waibel, 1997) were for word-based statistical machine translation, Pharaoh (Koehn, 2004) were for phrase-based SMT decoder in the publicly available MT. For each number of foreign words covered, a hypothesis stack in created. The initial hypothesis is placed in the stack for hypothesis with no foreign words covered. Starting with this hypothesis, new hypothesis are generated by committing to phrasal translation that covers previous unused foreign words. Each derived hypothesis is placed in a stack based on the number of foreign words it covers. After a new hypothesis is placed into a stack, the stack may have to be pruned by threshold or

histogram pruning, if it has become too large. In the end, the best hypothesis of the ones that cover all foreign words in the final state of the best translation. We can read off the target words of the translation by following the backtracking in each hypothesis.

Decoding with SCFG models is equivalent to CFG parsing (Melamed, 2004). The goal is to infer the highest-scoring tree that generates the input sentence using the source side of the grammar, and the road off the tree in target order. Most practical syntax based decoders are straightforward extensions of dynamic programming algorithm for parsing monolingual context-free grammars (Yamada and Knight, 2002). Further detail please read in (Lopex, 2008).

2.5 Evaluation

It is important to evaluate the accuracy of machine translation against fixed standards, so that the effect of different models can be seen and compared. The obvious difficulty in setting a standard for MT evaluation is the flexibility of natural language usage. For an input sentence, there can be many perfect translations. Knight and Marcu (2004) showed 12 independent English translations by human translators, given the same Vietnamese sentence. All of the 12 are different, yet all correct.

The most accurate evaluation is human evaluation, and it is frequently used for new MT theories. However, this method is far more time consuming than automatic methods. It is difficult for human evaluators to evaluate a large sample of translated sentences. Research has shown that certain machine evaluation methods correspond reasonably well with human evaluators, and thus they are usually used for the evaluation of large test sets. This section introduces three most common automatic evaluation methods, which are Bleu metrics, NIST metric and F-measure.

The Bleu metrics

The Bleu metrics (Papineni et al., 2001) evaluates machine translation by comparing the output of an MT system with correct translations. Therefore, a test corpus is needed for this method, giving at least one manual translation for each test sentence. During a test, each test sentence is passed to the MT system, and the output is scored by comparison with the correct translations. This score is called the *Bleu score*. The output sentence is called the *candidate* sentence, and the correct translations are called *references*.

The Bleu score is evaluated by two factors, concerning the precision and the length of candidates, respectively. *Precision* refers to the percentage of correct n-grams in the candidate. In the simplest case, unigram (*n=1*) precision equals to the number of words from the candidate that appear in the references divided by the total number of words in the candidate.

The standard n-gram precision is sometimes inaccurate in measuring translation accuracy. Take the following candidate translation for example:

Candidate: *a a a*.

Reference: *a good example*.

In the above case, the standard unigram precision is 3/3=1, but the candidate translation is inaccurate with duplicated words. Because of this problem, Bleu uses a modified n-gram precision measure, which consumes a word in the references when it is matched to a candidate word. The modified unigram precision of the above example is 1/3, for the word 'a' in the reference is consumed by the first 'a' in the candidate.

Similar to unigrams, modified n-gram precision applies to bigrams, trigrams and so forth. In mathematical form, the n-gram precision is as follows:

$$P_n = \frac{\sum_{C \in \{Candidat\}} \sum_{n-gram \in C} Matched(n-gram)}{\sum_{C \in \{Candidate\}} \sum_{n-gram \in C} Count(n-gram)} \qquad (2.6)$$

Apart from modified n-gram precision, a factor of candidate length is also included in the Bleu score. The main aim of this factor is to penalise short candidates, because long candidates will be penalised by low modified n-gram precisions. Take the following candidate for example:

Candidate: *C++ runs*.

Reference: *C++ runs much faster than Python*.

Both the unigram precision and the bigram precision for the above candidate are 1 (i.e. 100%), but the candidate contains much less information than the reference. To penalise such short candidates, a *brevity penalty* score is used. Suppose that the length of

the reference sentence is *r*, and the length of the candidate is *c*. In equation form, the brevity penalty score is as follows:

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases} \qquad (2.7)$$

When there are many references, *r* takes the length of the reference that is the closest to the length of the candidate. This length is called the effective reference length.

The Bleu score combines the modified n-gram score and the brevity penalty score. When there are many test sentences in the test set, one Bleu score is calculated for all candidate translations. This is done is two steps. Firstly, the geometric average of the modified n-gram precisions p_n is calculated for all n from 1 to N, using positive weights w_n which sum up to 1. Secondly, the brevity penalty score is computed with the total length of all candidates and total effective reference length for all candidates. In equation form,

$$BLEU = BP.\exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \qquad (2.8)$$

By default, the Bleu score includes the unigram, bigram, trigram and 4-gram precisions, each having the same weight. This is done by using N=4 and w_n=1/N in the above equation.

Experiments have shown that the Blue metrics are generally consistent with human evaluators, and thus are useful indicators for the accuracy of machine translation.

The NIST metric

The NIST metric (Doddington, 2002) was developed on the basis of the Bleu metrics. It focuses mainly on improving two problems of the Bleu score. Firstly, the Bleu metrics use the geometric average of modified n-gram precisions. However, because current MT systems have not reached considerable fluency, the modified n-gram precision scores may become very small for long phrases (i.e. big *n*). Such small scores have a potential negative effect on the overall score, which is not desired. To solve this problem, the NIST score uses the arithmetic average instead of geometric average. In this way, all modified n-gram precisions make zero or positive contribution to the overall score. Secondly, the Bleu metrics weigh all n-grams equally in the modified n-gram precision

score. However, some n-grams carry more useful information than others. For example, the bigram "washing machine" is considered more useful for the evaluation than the bigram "of the". The NIST metric gives each n-gram an information weight, which is computed by:

$$Info(w_{1\ldots}w_n) = \log_2\left(\frac{the\ \#\ of\ \ occurences\ of\ \ w_{1\ldots}w_{n-1}}{the\ \#\ of\ \ occurences\ of\ \ w_{1\ldots}w_n}\right) \qquad (2.9)$$

Besides the above two differences, the NIST score also uses a special brevity penalty score. In equation form, it can be written as:

$$BP = \exp\left(\beta \log^2(\min(\frac{L_{sys}}{L_{ref}},1))\right), \qquad (2.10)$$

where L_{ref} is the average number of words in the references, L_{sys} is the number of words in the candidate, and β is chosen to make $BP=0.5$ when the number of words in the candidate is $2/3$ of the average number of words in the references.

In summary, the NIST score for MT evaluation can be written as:

$$Score = BP.\sum_{n=1}^{N}\left(\frac{\sum_{w_1\ldots w_n \in Matched} Info(w_1\ldots w_n)}{\sum_{w_1\ldots w_n \in Candidate}(1)}\right) \qquad (2.11)$$

The F-measure

The F-measure (Turian et al., 2003) is an MT evaluation method developed independently from the Bleu and NIST metrics. In the domain of natural language processing, the term *F-measure* refers to a combination of *precision* and *recall*. It is commonly used for the evaluation of information retrieval systems. Suppose that the set of candidates is Y and the set of references is X, the precision, recall and F-measure are defined as follows:

$$precision(Y \mid X) = \frac{|X \cap Y|}{|Y|} \qquad (2.12)$$

$$recall(Y \mid X) = \frac{|X \cap Y|}{|Y|} \qquad (2.13)$$

$$F - measure = \frac{2 \times precision \times recall}{precision + recall} \tag{2.14}$$

In the simplest case, the F-measure for a MT translation candidate can be based on unigram precision and recall. See Fig. 2.6 for an illustration of this method.

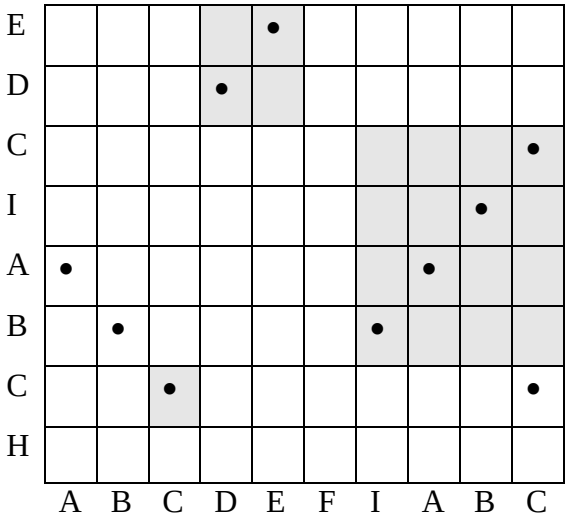


Figure 2.6: Unigram matches; adapted from (Turian et al., 2003).

In the above figure, each row represents a unigram (i.e. word) from the candidate translation (*C*), and each column represents a unigram from a reference (*R*). A dot (•) highlights the matching between a row and a column, which is called a *hit*. A *matching* is a subset of hits in which no two are in the same row or column. For the unigram case, the size of a matching can be defined as the number of hits in it. A matching with the biggest size is called a *maximum matching*, and is used as $R \cap C$ for precision and recall computations. Fig. 2.6 shows a maximum matching with dark background.

Denote the size of a maximum matching as MMS. In equation form, we have:

$$precision(C \mid R) = \frac{|MMS(C,R)|}{|C|} \tag{2.15}$$

$$recall(C \mid R) = \frac{|MMS(C,R)|}{|C|} \tag{2.16}$$

Therefore, from the above definitions, the unigram F-measure can be calculated.

The unigram form of the F-measure treats each sentence as a bag of words. This method ignores the evaluation of the word order in the candidate translations. One way to include the word order information is weighing continuous hits (i.e. phrases) more heavily than discontinuous hits. In formal definition, a *run* is a sequence of hits in which both the row and the column are contiguous. For example, the matching in Fig. 2.6 contains three runs, each with length 1, 2 and 4 respectively. Denote a matching with *M*, and a run in *M* with *r*. To give longer runs more weight, the size of matching *M* can be calculated by:

$$size(M) = \sqrt[e]{\sum_{r \in M} length(r)^e} \tag{2.17}$$

In the above equation, *e* is the weighing factor which favours longer runs when *e>1*. When *e=1*, the F-measure is reduced to the unigram case.

Experiments have shown that automatic evaluation methods are useful indicators of the quality of MT. However, they are not always consistent with human evaluators. Also, among different evaluation methods, some may perform comparatively better in certain cases but worse in others. For example, with the reference "programming methods", the candidate "methods of programming" would have a comparatively low Bleu score, because it does not contain matching bigrams. The same candidate may have a better score by the unigram F-measure, because word order information is not considered by this method. Therefore, the unigram F-measure is more consistent with human evaluators in this particular example. In contrast, the candidate "methods programming of" will not be penalised by the unigram F-measure by the same reason. Therefore, the Bleu metrics will be more consistent with human evaluators in this case.

Translation Edit Rate (TER)

TER (Matthew et al., 2006) is defined as the minimum number of edits needed to change a hypothesis so that it exactly matches one of the references, normalized by the average length of the references. Since we are concerned with the minimum number of edits needed to modify the hypothesis, we only measure the number of edits to the closest reference (as measured by the TER score). Specifically:

$$TER = \frac{\#\text{ of edits}}{\text{average}\,\#\,\text{of reference words}} \tag{2.18}$$

Possible edits include the insertion, deletion, and substitution of single words as well as shifts of word sequences. A shift moves a contiguous sequence of words within the hypothesis to another location within the hypothesis. All edits, including shifts of any number of words, by any distance, have equal cost. In addition, punctuation tokens are treated as normal words and mis-capitalization is counted as an edit. Consider the reference/hypothesis pair below, where differences between the reference and hypothesis are indicated by upper case:

REF: SAUDI ARABIA denied THIS WEEK
information published in the
AMERICAN new york times
HYP: THIS WEEK THE SAUDIS denied
information published in the
new york times

Here, the hypothesis (HYP) is fluent and means the same thing (except for missing "American") as the reference (REF). However, TER does not consider this an exact match. First, we note that the phrase "this week" in the hypothesis is in a "shifted" position (at the beginning of the sentence rather than after the word "denied") with respect to the hypothesis. Second, we note that the phrase "Saudi Arabia" in the reference appears as "the Saudis" in the hypothesis (this counts as two separate substitutions). Finally, the word "American" appears only in the reference.

If we apply TER to this hypothesis and reference, the number of edits is 4 (1 Shift, 2 Substitutions, and 1 Insertion), giving a TER score of 4/13 = 31%. BLEU also yields a poor score of 32.3% (or 67.7% when viewed as the error-rate analog to the TER score) on the hypothesis because it doesn't account for phrasal shifts adequately. Clearly these scores do not reflect the acceptability of the hypothesis, but it would take human knowledge to determine that the hypothesis semantically matches the reference.

The four automatic methods (Bleu, NIST, F-measure and TER metrics) are currently the most commonly used for MT evaluation. In the experiments of this thesis, we applied with the BLEU, NIST and TER metrics.

2.6 Conclusion

In this chapter, we have classified and summarized the current approaches of statistical machine translation, and previous work related to our research in this thesis, as well as the methods for translation evaluation.

3 Dividing and Translating Legal Sentence based on Its Logical Structure

A legal sentence represents a requisite and its effectuation (Tanaka et al. 1993). If each part of the legal sentence is shown separately, the readability will increase especially for a long sentence as seen in administrative laws. Such parts are recognized automatically by dividing a legal sentence according to the requisite-effectuation structure as described in this chapter. Furthermore, each fragment obtained by the dividing is shorter than the original sentence and the translation quality is expected to be improved.

The existing methods are mainly based on clause splitting and not be based on the requisite-effectuation structure. So, they are not used for the above purpose.

Dividing a sentence into shorter parts and translating them has a possibility to improve the quality of translation. For a legal sentence with the requisite-effectuation structure (logical structure), dividing a sentence into requisite-and-effectuation parts is simpler than dividing the sentence into its clauses because such legal sentences have specific linguistic expressions that are useful for dividing.

In this chapter, we present a novel method which divides a legal sentence in Japanese/English based on its logical structure and translates into sentences in English/Japanese. Characteristics and linguistic information of legal texts are used to split legal sentences into logical structures. A statistical learning method - Conditional Random Fields (CRFs) with rich linguistic information is used to recognize the logical structure of legal sentence. New features are proposed for recognizing the logical structure of English sentences. The logical structure of a legal sentence is adopted to divide the sentence. The experiments and evaluation are given with promising results.

The method of dividing and translating a legal sentence based on its logical structure follows in three steps:

- Recognition of the logical structure of a legal sentence
- Sentence segmentation
- Translating split sentences with phrase-based and tree-based models

3.1 Recognition of Logical Structure of a Legal Sentence

Though there are an implication type and an equivalence type in the logical structure of a legal sentence, this paper focuses on the implication type. Most law sentences are the implication and the logical structure of a sentence defining a term is the equivalence type. An implication law sentence consists of a law requisite part and a law effectuation part which designate the legal logical structure described by Tanaka et al. (1993) and Nakamura et al. (2007). Structures of a sentence in terms of these parts are shown in Fig. 3.1.

The requisite part and the effectuation part of a legal sentence are generally composed from three parts: a topic part, an antecedent part and a consequent part. In a legal sentence, the consequent part usually describes a law provision, and the antecedent part describes cases in which the law provision can be applied. The topic part describes a subject which is related to the law provision.

There are four cases (illustrated in Fig. 3.1) basing on where the topic part depends on: case 0 (no topic part), case 1 (the topic part depends on the antecedent part), case 2 (the topic part depends on the consequent part), and case 3 (the topic part depends on both the antecedent and the consequent parts). In case 0, the requisite part is the antecedent part and the effectuation part is the consequent part. In case 1, the requisite part is composed from the topic part and the antecedent part, while the effectuation part is the consequent part. In case 2, the requisite part is the antecedent part, while the effectuation part is composed from the topic and the consequent parts. In case 3, the requisite part is composed from the topic and the antecedent parts, while the effectuation part is composed from the topic and the consequent parts. Let's show examples of four cases of legal sentences. The annotation was carried out by a person who was an officer of the Japanese government, and persons who were students of a graduate law school and a law school.

- Case 0:

<A> 被保険者期間を計算する場合、</A> <C> 月によるものとする。</C> <A> When a period of an insured is calculated, </A> <C> it is based on a month. </C>

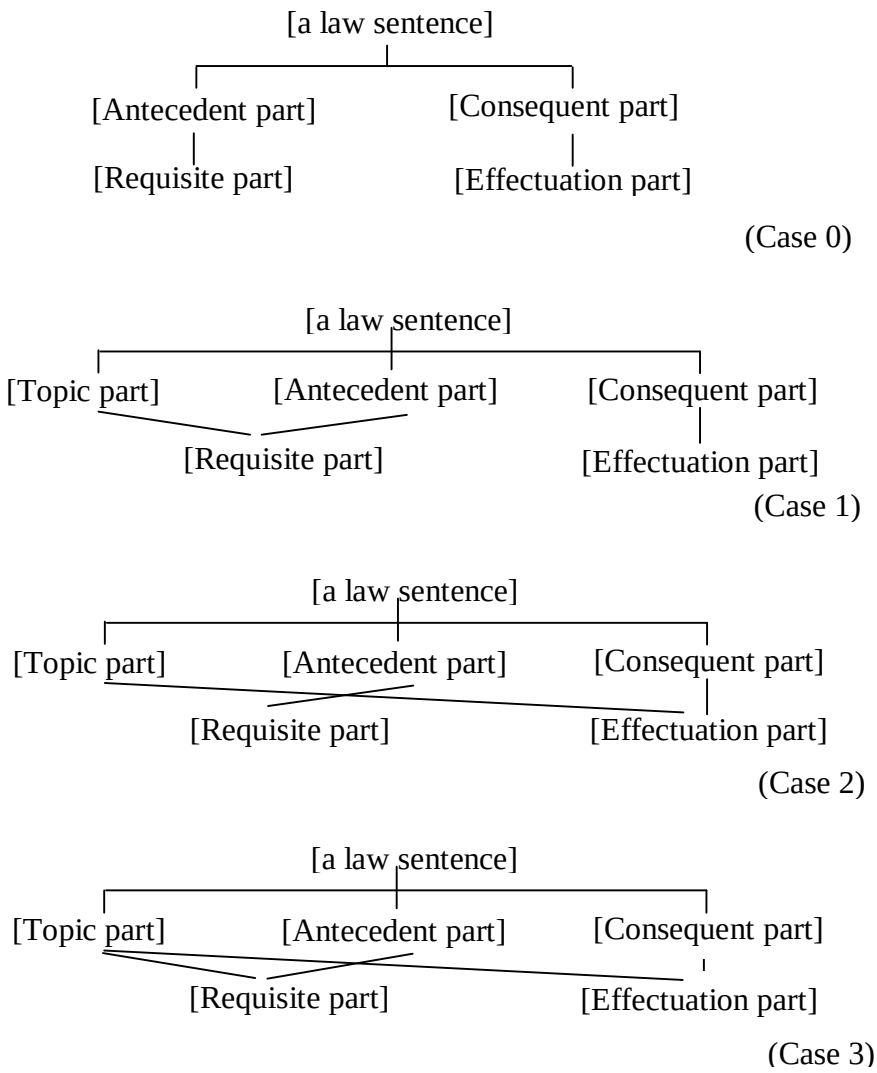


Figure 3.1: Four cases of the logical structure of a legal texts sentence

- Case 1:

<A> 被保険者の資格を喪失した後、さらにその資格を取得した </A> <T1> 者については、</T1> <C> 前後の被保険者期間を合算する。</C>

<T1> For the person </T1>

<A> who is qualified for the insured after s/he was disqualified, </A>

<C> the terms of the insured are added up together. </C>

- Case 2:

<T2> この法律による年金の額は、</T2> <A> 国民の生活に水準その地の諸事情に著しい変動が生じた場合には、</A> <C> 変動後の諸事情に応ずるため、速やかに改定の措置が講ぜられなければならない。</C>

<T2> For the amount of the pension by this law, </T2>

<A> when there is a remarkable change in the living standard of the nation of the other situation, </A>

<C> a revision of the amount of the pension must be taken action promptly to meet the situations. </C>

- Case 3:

<T3> 政府は、</T3> <A> 第―の規定により財政の現況及び見通しを作成したときは、</A> <C> 遅滞なく、 これを公表しなければならない。</C>

<T3> For the Government, </T3>

<A> when it makes a present state and a perspective of the finance, </A>

<C> it must announce it officially without delay. </C>

In these examples, A refers to the antecedent part, C refers to the consequent part, and T1, T2, T3 refer to the topic parts which correspond to case 1, case 2, and case 3.

Recognition of the logical structure of a legal sentence is an important task which has been studied in the research on Legal Engineering described by Katayama (2007). This task is a preliminary step to support other tasks in legal text processing such as legal text summarization, question answering in legal domains, legal article retrieval, legal translation, detection of contradiction in laws, and so on.

The recognition task of the logical structure of a legal sentence is to split a source sentence into some non-overlapping and non-embedded logical parts. This task belongs to the class of phrase recognition problems. Sequence learning is a suitable model for phrase recognition problems which do not allow overlapping and embedded relationships. It has been applied successfully to many phrase recognition tasks such as word segmentation, chunking and name entity recognition. So we choose the sequence learning model for the recognition task of the logical structure of a legal sentence. The framework of the recognition of the logical structure of a legal sentence is shown in Fig. 3.2.

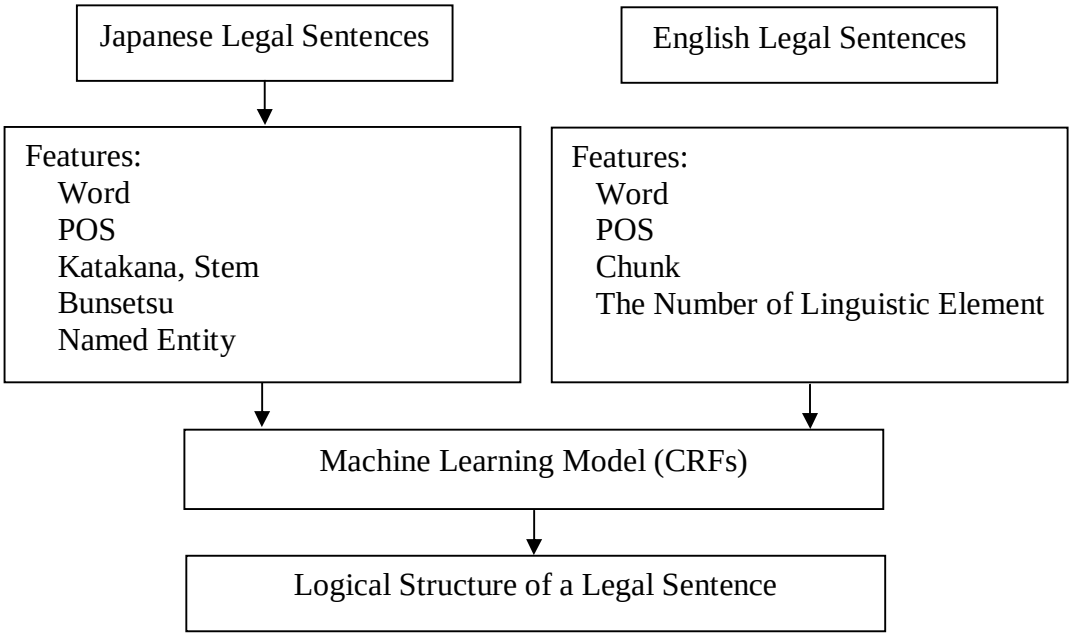


Figure 3.2: The recognition of the logical structure of a legal sentence

To recognize the logical structure of a legal sentence we use sequence learning model described in Kudo (2003). We model the structure recognition task as a sequence labeling problem described in Bach et al. (2010), in which each sentence is a sequence of words. Table 3.1 illustrates an example in IOB notation. In this notation, the first element of a part is tagged by B, the other elements of the part are tagged by I, and an element not included in any part is tagged by O. The sentence in Table 3.1 is tagged with IOB and logical part tags such as a topic part (tag T), an antecedent part (tag A) and a consequent part (tag C).

In the recognition task of the logical structure of a legal sentence, we consider implication types of legal sentences, and five kinds of logical parts, as follows:

- Antecedent part (A)
- Consequent part (C)
- Topic part T1 (correspond to case 1)
- Topic part T2 (correspond to case 2)
- Topic part T3 (correspond to case 3)

In the IOB notation, we will have 11 kinds of tags: B-A, I-A, B-C, I-C, B-T1, I-T1,

Table 3.1: A sentence with IOB notation for the sequence learning model

Sentence	<T> w_1w_2 </T> <A> $w_3w_4w_5$ </A> <C> $w_6w_7w_8$ </C>							
Element	w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8
IOB tags	B	I	B	I	I	B	I	I
IOB with logical part tags	B-T	I-T	B-A	I-A	I-A	B-C	I-C	I-C

B-T2, I-T2, B-T3, I-T3 and O (used for an element not included in any part). For example, an element with tag B-A begins an antecedent part, while an element with tag B-C begins a consequent part.

We use Conditional Random Fields (CRFs) (Lafferty et al., 2001) as a learning method. We provide a brief introduction to Conditional Random Fields. Conditional Random Fields (CRFs) are undirected graphical models, which define the probability of a label sequence *y* given an observation sequence *x* as follows:

$$p(y \mid x, \lambda, \mu) = \frac{1}{Z(x)} \exp\left(\sum_j \lambda_j t_j (y_{i-1}, y_i, x, i) + \sum_k \mu_k s_k (y_i, x, i) \right) \quad (3.1)$$

where $t_j(y_{i-1}, y_i, x, i)$ is a transition feature function (or edge feature), which is defined on the entire observation sequence *x* and the labels at positions *i* and *i*-1 in the label sequence *y*; $s_k(y_i, x, i)$ is a state feature function (or node feature), which is defined on the entire observation sequence *x* and the label at position *i* in the label sequence *y*; and λ_j and μ_k are parameters of the model, which are estimated in the training process, $Z(x)$ is a normalization factor.

Training CRFs is commonly performed by maximizing the likelihood function with respect to the training data using advanced convex optimization techniques like L-BFGS (Byrd 1994). Inference in CRFs, i.e., searching the most likely output label sequence of an input observation sequence, can be done by using Viterbi algorithm.

There are some reasons why we choose CRFs. The first reason comes from the nature of the recognition task of the logical structure of a legal sentence. The recognition task of the logical structure of a legal sentence can be considered as a sequence learning problem, and CRFs is an efficient and powerful framework for sequence learning tasks.

The second reason comes from the advantages of CRFs. CRFs is a discriminative method, it has all the advantages of Maximum Entropy Markov Models (MEMMs) but does not suffer from the label bias problem. The last reason is that CRFs has been applied successfully to many NLP tasks such as POS tagging, chunking, named entity recognition, syntax parsing, information retrieval, information extraction, and so on.

We designed features for Japanese and English differently based on linguistics characteristics of each language. For Japanese, following the features described by Bach et al. (2010), we use five sets of features (using Cabocha tool, Kudo 2003). Each of these feature sets contains one kind of feature. Feature sets are shown in Table 3.2.

Table 3.2: Japanese features

Feature Set	Kind of Features	Window Size	#Features
Set 1	Word	2	12
Set 2	POS	2	12
Set 3	Katakana, Stem	2	24
Set 4	Bunsetsu	2	12
Set 5	Named Entities	2	12

The modern Japanese writing system uses three main scripts:

- *Kanji*, Chinese characters
- *Hiragana*, used along with kanji, for native or naturalised Japanese words, and for grammatical elements
- ***Katakana*,** used for foreign words and names, loanwords, onomatopoeia, scientific names, and sometimes to replace kanji or hiragana for emphasis.

In Japanese, a sentence is divided into some chunks called Bunsetsu. Each Bunsetsu includes one or more content words (noun, verb, adjective, etc.) and may include some function words (case-marker, punctuation, etc.).

In this work, we studied recognizing the logical structure of legal Japanese and English sentences while Bach et al. (2010) studied for Japanese language. For English sentences, we propose some new features to recognize the logical structure of a legal

sentence based on its characteristics and linguistic information and designed a set of features:

- Word form: phonological or orthographic appearance of a word in a sentence.
- Part-of-Speech features: POS tags of the words in a sentence
- Chunking tag: tag of syntactically correlated parts of words in a sentence.
- The number of particular linguistic elements which appear in a sentence as follows:
  + Relative pronouns (e.g, where, who, whom, whose, that)
  + Punctuation marks (. , ; :)
  + Quotes
  + Verb phrase chunks
  + Relative phrase chunks

We parse the individual English sentences by GENIA Tagger (Tsuruoka and Tsujii 2005) and use CRFs tool (Kudo 2003) for sequence learning tasks. Experiments were conducted in the Japanese-English translation corpus. We collected the corpus using Japanese Law Translation Database System[1]. The corpus contains 516 sentences pairs. Table 3.3 shows statistics on the number of logical parts of each type.

From Table 3.3, we see that most types of Japanese logical parts are A(40.1%), C(40.3%), T2(7.7%), T3(11.9%). On the other hand, most types of English logical parts are A (44.1%), C(45.3%) and T3(10.2%), which make up 99.6% than all types.

We divided the corpus into 10 sets, and conducted 10-fold cross-validation tests for recognizing logical structures of the sentences in the corpus. We evaluated the performance of our system by *precision*, *recall*, and F_1 scores as:

$$precision = \frac{\#\ correct\ \ parts}{\#\ predicted\ \ parts} \tag{3.2}$$

$$recall = \frac{\#\ correct\ \ parts}{\#\ actual\ \ parts} \tag{3.3}$$

$$F_1 = \frac{2 * precision\ * recall}{precision\ + recall} \tag{3.4}$$

[1] Available at http://www.japaneselawtranslation.go.jp/

Table 3.3: Statistics on logical parts of the corpus

Japanese Logical Part					
C	A	T1	T2	T3	Total
552	549	0	106	163	1370
English Logical Part					
C	A	T1	T2	T3	Total
576	561	0	4	130	1271

Table 3.4: Experimental results for recognition of the logical structure of a legal sentence

	Precision (%)	Recall (%)	F1 (%)
Japanese			
C	86.95	86.25	86.6
A	86.55	85.29	85.92
T1	0	0	0
T2	83.86	83.02	83.44
T3	75.21	54.56	63.24
Overall	85.10	84.27	84.68
English			
C	84.89	82.85	83.86
A	86.03	85.03	85.56
T1	0	0	0
T2	89.10	85.64	87.34
T3	78.08	56.12	65.30
Overall	84.64	83.24	83.93

Experimental results on the corpus are described in Table 3.4. To investigate the effects of features on the task, we conducted experiments on feature sets. For Japanese language, we conducted on four other feature sets combined with the word features. The experimental results are shown in Table 3.5. In these experiments, Bunsetsu information

was used as features of elements in sequences. Model 1 using only word features is the baseline model. Four other models yielded better results. Model 3 with word and POS features led to an improvement of 0.78% compared with the baseline model. We can see that these features were effective for our recognition task. For English, we conducted on three other feature sets combined with the word features. The experimental results are shown in Table 3.6. Model 1 using only word features is the baseline model. Three other models yielded better results. The POS and Chunk features were effective for our recognition task. From the result we see these features give better result, especially on three main parts, C, A and T2. This means that these features are good to the task of recognition of the logical structure of a legal sentence.

Table 3.5: Experiments with feature sets of Japanese sentences

Model	Feature sets	Precision (%)	Recall (%)	F1 (%)
Model 1	Word	83.75	82.05	82.89
Model 2	Word+Katakana, Stem	83.90	82.15	83.02
Model 3	Word+POS	84.25	83.10	83.67
Model 4	Word+Bunsetsu	83.78	82.06	82.91
Model 5	Word+Named Entity	83.79	82.08	82.93

Table 3.6: Experiments with feature sets of English sentences

Model	Feature sets	Precision (%)	Recall (%)	F1 (%)
Model 1	Word	83.15	82.00	82.57
Model 2	Word+POS	83.85	83.10	83.47
Model 3	Word+Chunk	83.40	82.20	82.80
Model 4	Word+ Number of particular linguistic elements	83.18	82.02	82.60

3.2 Sentence Segmentation

We build a sentence segmentation model based on the logical structure of a legal sentence. According to the logical structure of a legal sentence (Fig. 3.1), a sentence of each case is divided as follows:

- Case 0:
  - Requisite part: [A]
  - Effectuation part: [C]
- Case 1:
  - Requisite part: [T1 A]
  - Effectuation part: [C]
- Case 2:
  - Requisite part: [A]
  - Effectuation part: [T2 C]
- Case 3:
  - Requisite part: [T3 A]
  - Effectuation part: [T3 C]

Example: Sentences of section 3.1.1 are separated as shown in Fig 3.3.

✓ Case 0: <A> 被保険者期間を計算する場合、</A> <C> 月によるものとする。</C> <A> When a period of an insured is calculated, </A> <C> it is based on a month. </C>
The Japanese sentence will be split as: [被保険者期間を計算する場合] [月によるものとする。]
The English sentence will be split as: [When a period of an insured is calculated] [it is based on a month.]

✓ Case 1:

<A> 被保険者の資格を喪失した後、さらにその資格を取得した </A> <T1> 者については、</T1> <C> 前後の被保険者期間を合算する。</C>

<T1> For the person </T1>

<A> who is qualified for the insured after s/he was disqualified, </A>

<C> the terms of the insured are added up together. </C>

The Japanese sentence will be split as:

[者については、被保険者の資格を喪失した後、 さらにその資格を取得した]

[前後の被保険者期間を合算する。]

The English sentence will be split as:

[For the person, who is qualified for the insured after s/he was disqualified]

[the terms of the insured are added up together.]

✓ Case 2:

<T2> この法律による年金の額は、</T2> <A> 国民の生活に水準その地の諸事情に著しい変動が生じた場合には、</A> <C> 変動後の諸事情に応ずる ため、速やかに改定の措置が講ぜられなければならない。</C>

<T2> For the amount of the pension by this law, </T2>

<A> when there is a remarkable change in the living standard of the nation of the other situation, </A>

<C> a revision of the amount of the pension must be taken action promptly to meet the situations. </C>

The Japanese sentence will be split as:

[国民の生活に水準その地の諸事情に著しい変動 が生じた場合には、]

[この法律による年金の額は、変動後の諸事情に応ずるため、速やかに改定の措置が講ぜられなければならない。]

The English sentence will be split as:

[When there is a remarkable change in the living standard of the nation of the other

situation]

[For the amount of the pension by this law, a revision of the amount of the pension must be taken action promptly to meet the situations.]

✓ Case 3:

<T3> 政府は、 </T3> <A> 第―の規定により財政の現況及び見通しを作成したときは、</A> <C> 遅滞 なく、これを公表しなければならない。</C>

<T3> For the Government, </T3>

<A> when it makes a present state and a perspective of the finance, </A>

<C> it must announce it officially without delay. </C>

The Japanese sentence will be split as:

[政府は、第―の規定により財政の現現況及び見通しを作成したときは、]

[政府は遅滞なく、これを公表しなければならない。]

The English sentence will be split as:

[For the Government, when it makes a present state and a perspective of the finance]

[For the Government, it must announce it officially without delay.]

Figure 3.3: Examples of sentence segmentation

Table 3.7: Statistics of the corpus

Corpus		#words	#sentences
Training corpus	English	1,061,044	42,870
	Japanese	1,002,587	
Development corpus	English	45,150	1,400
	Japanese	45,020	
Test corpus	English	17,475	516
	Japanese	17,753	

3.3 Translating Split Sentences with Phrase-based and Tree-based models

We build phrase-based and tree-based models based on two baseline systems: Moses and Moses-chart. Moses is a phrase-based system, it was developed by Kohn et al. (2007). This baseline system was used in the ACL 2007 Second Workshop on Statistical Machine Translation. We used some parameters of the phrase table in Moses as follows:

- a maximum phrase length of 7.
- a lexical reordering model with *msd-bidirectional-fe* parameter.
- a distortion limit of -1 (unlimited).

Because Japanese and English are fairly different in the word order, so the distortion limit is set to unlimited.

Moses-chart is a tree-based system which has become known as *hierarchical phrase-based* and *syntax-based models (Chiang 2005, 2007)*. This system was proposed after Moses by the same author (Kohn et al. 2007). It uses a grammar consisting of SCFG (Synchronous Context-Free Grammar) rules. We used the specific additional parameter *-hierarchical and -glue-grammar* and reduced the number of lexical items in the grammar with *-max-phrase-length 7* in the Moses-chart baseline system.

After recognizing the logical structure of a legal sentence, and applying it to split the sentence to requisition and effectuation parts, we translate the split sentences with Moses and Moses-chart systems.

3.4 Evaluation

We investigated the effects of translating a legal sentence by dividing it based on the logical structure of the sentence. We constructed phrase-based and tree-based SMT systems using Moses and Moses-chart for the English-Japanese and Japanese-English language pairs and evaluated the systems based on the BLEU, NIST (Papineni et al., 2002) and TER scores (Mathew Snover et al., 2006).

3.4.1 Data Preparation

We conducted the experiments on the Japanese-English translation corpus provided by Japanese Law Translation Database System. The training corpus consisted of 42,870

Japanese-English sentence pairs, the development and test set consisted of 1,400 and 516 sentence pairs, respectively. Table 3.7 shows statistics of the corpus. We tested on 516 Japanese-English sentence pairs. Table 3.8 shows statistics of the test corpus. Table 3.9 shows the number of sentences, the statistics of the requisition parts, the effectuation parts and the logical parts after splitting in the test set.

The English sentences are lowercased and tokenized by GENIA tagger tool (Tsuruoka et al., 2005), and the Japanese sentences are segmented by CaboCha tool (Kudo et al., 2003). We run GIZA++ (Och et al., 2000) to obtain word alignment in both translation directions. For the language model, the SRI Language Modeling Toolkit (SRILM) (Stolcke, 2002) was used. We used the data smoothing technique Knerser-Ney and experimented with n=4. The feature weights were optimized by Minimum Error Rate Training (MERT), using the development sentences.

Table 3.8: Statistics of the test corpus

Name of Law	Number of sentences
Act on General Rule for Application of Law	78
Act on Land and Building Leases	120
Administrative Procedure Act	99
Foreign Exchange and Foreign Trade Act	219
Total	516

Table 3.9: Number of requisition part, effectuation part in the test data

	Japanese	English
Sentence	516	516
#of requisition part	433	436
#of effectuation part	516	513
#of segment	949	949

Table 3.10: Translation results in Japanese-English

Model	BLEU	NIST	TER
Moses	0.248	5.08	60.56
our system	0.256	5.24	59.32
N-best system	0.267	5.42	57.42
Moses-chart	0.262	5.32	57.64
our system	0.274	5.50	56.25
N-best system	0.286	5.64	54.73

Table 3.11: Translation results in English-Japanese

Model	BLEU	NIST	TER
Moses	0.298	5.72	57.86
our system	0.310	5.83	56.78
N-best system	0.325	6.01	55.34
Moses-chart	0.318	5.91	56.64
our system	0.331	6.14	54.61
N-best system	0.347	6.26	53.10

3.4.2 Experiment Results

We compared our system with the two baselines, Moses and Moses-chart and evaluated the results using the BLEU, NIST and TER scores. Table 3.10 shows the translation evaluation results in Japanese-English and Table 3.11 shows translation evaluation results in English-Japanese. The Moses and Moses-chart show the results without using split sentences. Our systems using Moses and Moses-chart show the result where the logical structure of a legal sentence is used to split the test sentences in the Moses and Moses-chart, respectively.

We evaluated the effect of the recognition task of the logical structure of a legal sentence to our system by comparing our system with baselines using corpus from n-best output of the recognition of the logical structure of a legal sentence. With each sample, we chose 40-best output as candidates. For each sample, we find one candidate by decoding

phase as follows:

- **For each** output
  - **If** the output is equal with the annotated sentence in the gold corpus (Table 3.3)
  - **Then** the candidate = the output = the annotated sentence in the gold corpus
  - **Else** the candidate **=** the output with the highest score in 40-best output
  - **End If**
- **End For**

When we find the candidate we will split it to requisition and effectuation parts, add the split sentences in the test corpus and translate them with Moses and Moses-chart systems. We call this system as N-best. The result in Table 3.10 and Table 3.11 shows that the N-best system got better results than our models in both Moses and Moses-chart systems. It proves that the translation quality improve if the recognition of the logical structure of a legal sentence is more accurate because there are more candidates to be equal with the annotated sentence in the gold corpus. So the recognition of the logical structure of a legal sentence is important to our model.

From the results, we can see that by using the logical structure of a legal sentence to split a sentence, we can improve the quality in all three metrics comparing with the two baselines. The result in Moses-chart is better than in Moses. Therefore, we show examples of our system using Moses-chart on the test set with the original sentence (O), our split sentence (S), translation sentence by our model (T), translation sentence by baseline (B), and reference sentence (R). We present examples in case where the proposed model works well in Table 3.12. Table 3.13 shows examples in case where the proposed model does not work well.

Table 3.12: Positive translation examples in Moses-chart
(O: original sentence, S: our split sentence, T: translation sentence by our model, B: translation sentence by baseline, and R: reference sentence)

O	<C> A law shall come into effect after the expiration of twenty days following the date of its promulgation; </C> provided, however, that if a different effective date is provided by law, such provision shall prevail.

S	A law shall come into effect after the expiration of twenty days following the date of its promulgation Provided, however, that if a different effective date is provided by law, such provision shall prevail.
T	法律は、公布の日から二十日を経過した日から施行する。ただし、法律でこれと異なる施行期日を定めたときは、そのような規定の下で。
B	法律は、公布の日から二十日を経過した後に施行する。ただし、このような規定の下では逆に、法執行機関によって日付を設定したとき。
R	法律は、公布の日から起算して二十日を経過した日から施行する。ただし、法律でこれと異なる施行期日を定めたときは、その定めによる。

O	<A> 前条の規定による選択がないときは、</A> <C> 法律行為の成立及び効力は、当該法律行為の当時において当該法律行為に最も密接な関係がある地の法による。</C>
S	前条の規定による選択がないときは、 法律行為の成立及び効力は、当該法律行為の当時において当該法律行為に最も密接な関係がある地の法による。
T	In the absence of a choice of law under the preceding Article, the formation and effect of a juridical act shall be governed by the law of the place where the act is most closely connected at the time of the act.
B	In the absence of a choice of law under the preceding Article, establishment of law and effective action, by law of the place where there is most closely related to the legal action at the time of the act in the Act.
R	In the absence of a choice of law under the preceding Article, the formation and effect of a juridical act shall be governed by the law of the place with which the act is most closely connected at the time of the act.

Table 3.13: Negative translation examples in Moses-chart

(O: original sentence, S: our split sentence, T: translation sentence by our model, B: translation sentence by baseline, and R: reference sentence)

O	<T2> 前項の規定は、</T2> <A> 親族法又は相続法の規定によるべき法律行為及び行為地と法を異にする地に在る不動産に関する法律行為については、</A> <C> 適用しない。</C>
S	親族法又は相続法の規定によるべき法律行為及び行為地と法を異にする地に在る不動産に関する法律行為については 前項の規定は、適用しない。
T	The provisions of family law or inheritance law, or a juridical act relating to real property situated in a place governed by a different law from the law of the place where the act was done is not applied to. The provisions of the preceding paragraph shall not apply.
B	The provisions of the preceding paragraph, provisions of family law or inheritance law, or a juridical act relating to real property situated in a place governed by a different law from the law of the place where the act was done should not apply to.
R	The preceding paragraph shall not apply to a juridical act to be governed by the provisions of family law or inheritance law, or a juridical act relating to real property situated in a place governed by a different law from the law of the place where the act was done.
O	<T3> The marital property regime </T3> <A> to which a foreign law should be applied pursuant to the preceding two paragraphs </A> <C> may not be asserted against a third party without knowledge, </C> <A> to the extent that it is related to any juridical act done in Japan or any property situated in Japan. </A>
S	The marital property regime to which a foreign law should be applied pursuant to the preceding two paragraphs to the extent that it is related to any juridical act done in Japan or any property situated in Japan. The marital property regime may not be asserted against a third party without knowledge, to the extent that it is related to any juridical act done in Japan or any property situated in Japan.

T	外国法を適用することは、日本においてされた法律行為及び日本に在る財産していることをあるのは、前二項の規定により適用されるべきであるに夫婦財産制度。夫婦財産制は、日本においてされた法律行為及び日本に在る財産していることをある、善意の第三者に対抗することができない。
B	夫婦財産制度は、外国法を適用すべき、前二項の規定を適用を善意の第三者に対抗することができない、日本においてされた法律行為及び日本に在る財産していることをある。
R	前二項の規定により外国法を適用すべき夫婦財産制は、日本においてされた法律行為及び日本に在る財産については、善意の第三者に対抗することができない。

We performed a manual evaluation where we asked eighteen judges to assign fluency and adequacy ratings for both English translations of Japanese sentences (eight judges) and Japanese translations of English sentences (ten judges), using two systems: baseline and our system (in Moses).

We selected randomly 1/20 of translation output of the legal translation systems in Japanese-English and English-Japanese translations. We evaluated translation output based on three questions as follows:

- ✓ Question 1: How is the quality of overall translation?
- ✓ Question 2: How is the quality of lexical translation?
- ✓ Question 3: How is the readability from the viewpoint of understanding a law article?

We defined scores for each question. Question 1 is from one to five, for Question 2 from one to three, and for Question 3 among three. For Question 1, we used the flowchart of the Patent Machine Translation task at NTCIR-9[2]. Fig. 3.4, Fig. 3.5 and Fig. 3.6 define scores for each question, respectively. Table 3.14 shows statistics about people in human evaluation.

[2] http://ntcir.nii.ac.jp/PatentMT/

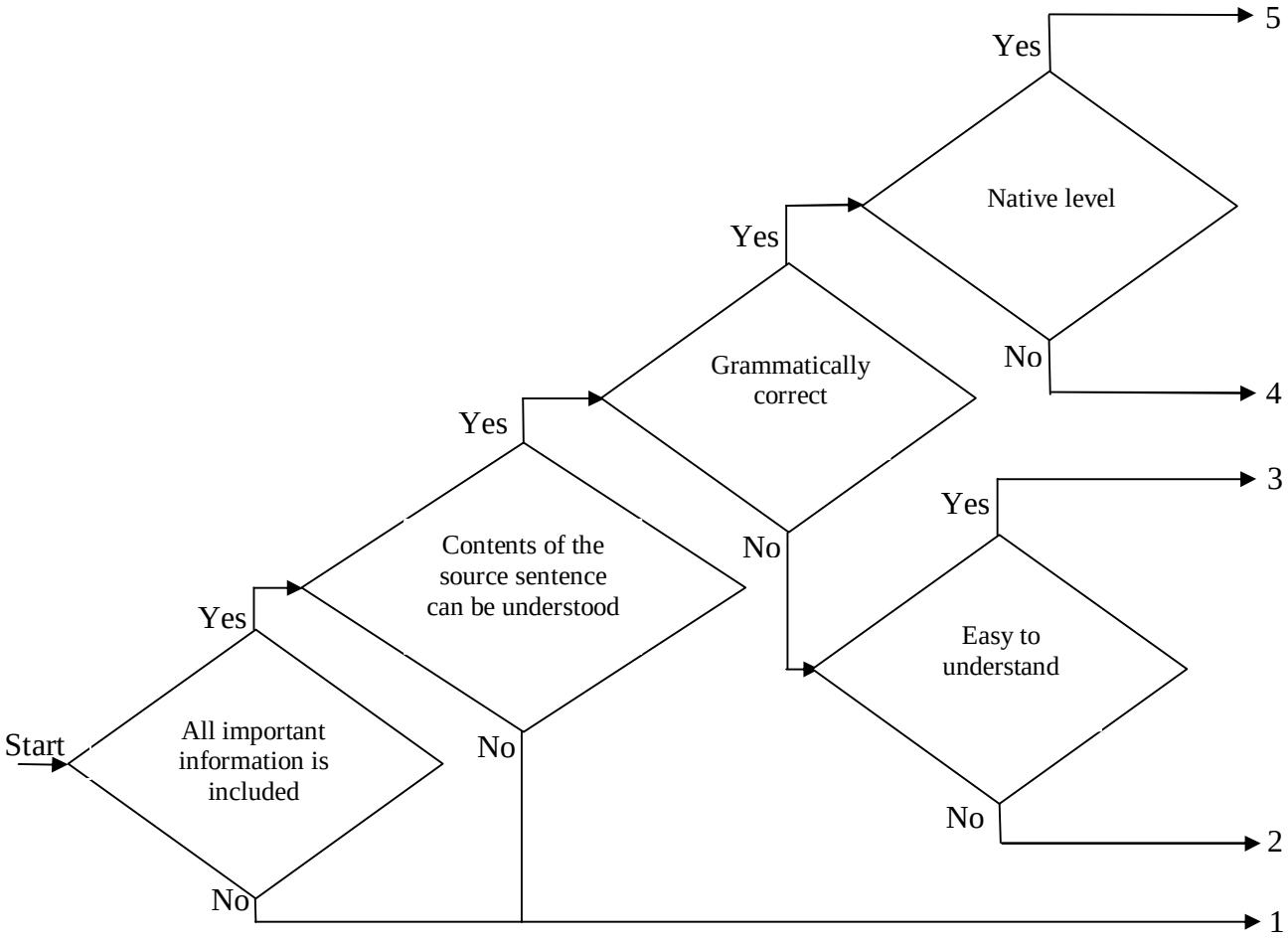


Figure 3.4: Score for Question 1
(How is the quality of overall translation?)

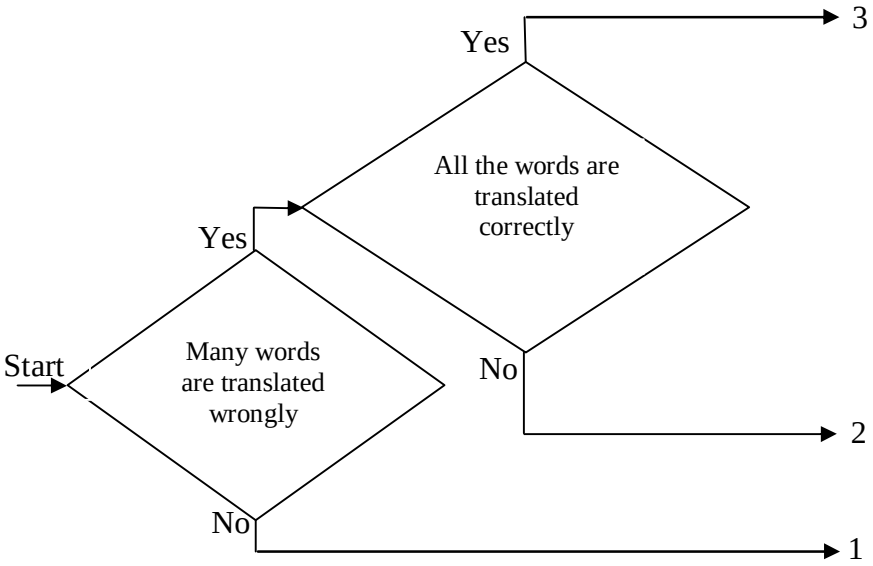


Figure 3.5: Score for Question 2 (How is the quality of lexical translation?)

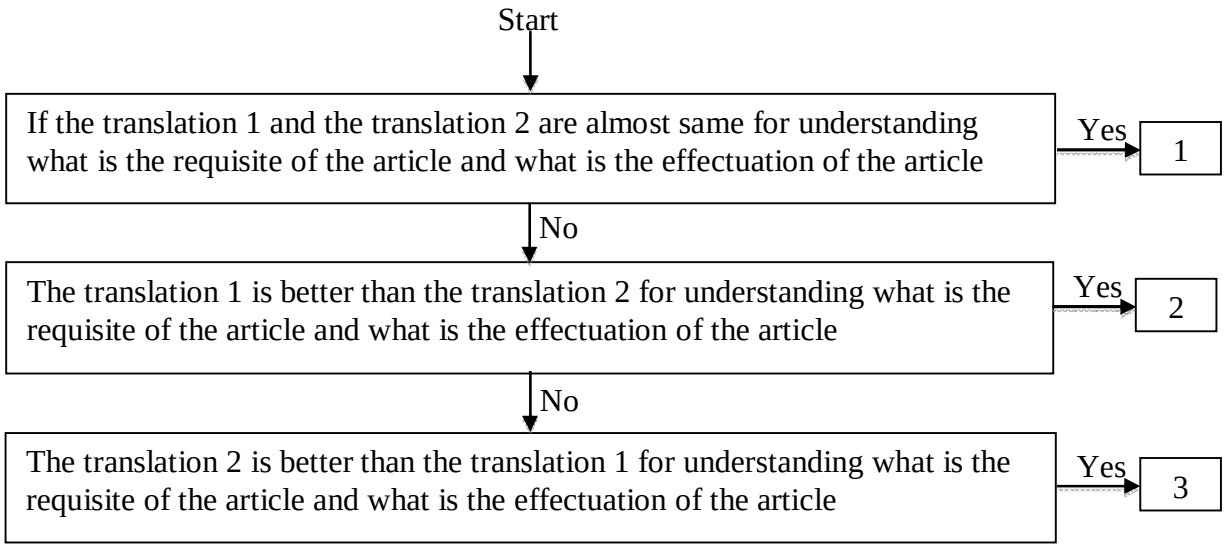


Figure 3.6: Score for Question 3
(How is the readability from the viewpoint of understanding a law article?)

Table 3.14: Statistics about people in human evaluation

Translation	Age Average	Nationality	Knowledge/ Experience of Law	
			Yes	No
English-Japanese	39	Japan: 10	3	7
Japanese-English	33	Canada:1, USA:6, Australia:1	4	4

Table 3.15: Human evaluation result for English-Japanese and Japanese-English translation

Comparing Our system & Baseline	Question 1			Question 2			Question 3		
	better	equal	worse	better	equal	worse	better	equal	worse
English-Japanese	45	122	33	23	156	21	73	93	34
Japanese-English	34	109	17	31	109	20	59	76	25

We got a score for each question and made a statistics by comparing the scores of our system with the baseline shown in Table 3.15.

The result in Table 3.15 prove that our system gets the better results than baseline in three questions. The results of question 1, 2, 3 in human evaluation prove that dividing and

translating a legal sentence according to the requisite-effectuation structure get the better translation quality, and that showing the requisites and effectuation separately increase readability. The advantage of the proposed method arises from the translation model based on the logical structure of a legal sentence where the decoder searches over shortened inputs. Because we use the logical structure of a legal sentence to split sentence, the split sentence reserves its structure and the average length of split sentence is much smaller than those of no split sentence. They are expected to help realize an efficient statistical machine translation search.

We compared the separate translation of requisite and effectuation parts of a sentence with translation of clauses of the sentence.

For an English sentence, we used the discourse segmentation method (Bach et al., 2012) to split the sentence. For a Japanese sentence, we haven't found any available system splitting a sentence into clauses. So, we manually split a sentence into clauses. Each split segment is translated independently. We compare each translation of 1/20 test sentences in three aspects:

- Comparison of the translation result.

For the sentence in Case 0 of the logical structure of a legal sentence, our proposed method has similar affecting in translation results of the clauses. However, we get the better results than the translation results of the clauses when we apply for the sentences in Case 1, Case 2 and Case 3. Because in Case 0, the requisite part of the logical structure of a legal sentence has antecedent part (A) and effectuation part has consequence part (A), the order of each logical structure does not change, so split sentence of our method is translated in order. In Case 1, Case 2 and Case 3, the topic part of the logical structure has the different position in the split sentence. For the clause splitting, such a position does not change and the translations are different between the requisite and effectuation splitting and the clause splitting. Table 3.16 and Table 3.17 show the translation results of the manual split method and our method using Moses baseline in Japanese-English and English-Japanese translation respectively. The improvements in BLEU percentages scores are +0.4/+0.5, NIST scores are +0.6/+0.8, and TER percentage values are -1.4/-2.2.

- Comparison of phrase reordering of the translation result

When we analyze phrase reordering in translation results, our translation also gets better results than the translation results of clause splitting. Our method first splits a sentence into topic part, an antecedent part and consequent part. Then it composes a requisite and an effectuation parts using the topic, the antecedent and the consequent parts. It will help decoder produce better phrase reordering. Table 18 shows translation examples of test sentences in Case 3 in the manual split method and our method. The translation of requisite and effectuation parts gets better result than the translation of split clauses phrase reordering.

- Comparison of lexical translation.

We compared the translation of requisite and effectuation parts with the translation results of split clauses and confirmed that the lexical translation in the translation of requisite and effectuation parts is better than in the translation results of split clauses. So, we think that when the decoder produces a better phrase reordering, in most cases it determines the better translations of the split sentence source. The translation examples of requisite and effectuation parts get better result than translation of split clauses in lexical translation shown in the underlined parts of Table 3.18.

Table 3.16. Translation results in Japanese-English

Type	BLEU	NIST	TER
Clause splitting	0.254	5.22	59.38
Requisite and effectuation splitting	0.258	5.28	59.24

Table 3.17. Translation results in English-Japanese

Model	BLEU	NIST	TER
Clause splitting	0.307	5.80	57.02
Requisite and effectuation splitting	0.312	5.88	56.70

Table 3.18: Translation examples of test sentences in Case 3

The Japanese sentence in Japanese-English translation is the original sentence. The English sentence in English-Japanese translation is the reference translation in the government web page

Japanese-English translation	
	Clause splitting
Sentence	前項の規定にかかわらず、為地法に適合する　方式は、有効とする。
Split Sentence	前項の規定にかかわらず、為地法に適合する　方式は、有効とする。
Translation	Notwithstanding the provisions of the preceding paragraph, system that conforms to the ground for the law, as an active.
	Requisite and effectuation splitting
Sentence	<C>前項の規定にかかわらず、</C><A> 行為地法に適合する</A> <T3>方式は、</T3> <C> 有効とする。</C>
Split Sentence	方式は、前項の規定にかかわらず、有効とする。 方式は、行為地法に適合する
Translation	Notwithstanding the provisions of the preceding paragraph, the formalities is effective The formalities is adapted with the law of the place
English-Japanese translation	
	Clause splitting
Sentence	Notwithstanding the preceding paragraph, the formalities that comply with the law of the place where said act was done shall be valid.
Split Sentence	Notwithstanding the preceding paragraph, the formalities that comply with the law of the place where said act was done shall be valid.
Translation	前項の規定にかかわらず、行為が行われていたと述べた場所の法律を遵守手続きは有効でなければならない。
	Requisite and effectuation splitting
Sentence	<C> Notwithstanding the preceding paragraph, </C> <T3> the formalities </T3> <A> that comply with the law of the place where said act was done </A> <C> shall be valid. </C>
Split Sentence	the formalities notwithstanding the preceding paragraph, shall be valid. the formalities that comply with the law of the place where said act was done
Translation	方式は、前項の規定にかかわらず、有効でなければならない 方式は、当該行為が行われた場所の法律の遵守します。

3.5 Conclusion

In this chapter a novel approach was presented to improve legal translation by dividing and translating a legal sentence using its logical structure. How to split a complex legal sentence based on its logical structure was showed. The approach gave better result for translating legal sentences. The experiment results for the translation between Japanese and English showed some improvements in the translation quality measured by the BLEU, NIST and TER scores. The subjective evaluation also shows better results. The subjective evaluation also shows better results.

There are still some important issues to be considered in future. Based on our observation, there are cases that a translation of a sentence differs semantically from a translation of the split sentence and the current model performs well depending on the recognition of the logical structure of a legal sentence. In the future, integrating split sentences into training and more sophisticated features will be investigated to improve the recognition of the logical structure of a legal sentence and apply it to enhance the model.

4 Rule Selection for Tree-Based Statistical Machine Translation

The tree-based statistical machine translation is a model using rules with hierarchical structures as translation knowledge, which can capture long-distance reorderings. Typically, a translation rule consists of a source side and a target side. However, the source side of a rule usually corresponds to multiple target-sides in multiple rules. Therefore, during decoding, the decoder should select correct target-side for a source side. This is rule selection.

Rule selection is important to tree-based statistical machine translation systems. This is because that a rule contains not only terminals (words or phrases), but also nonterminals and structural information. During decoding, when a rule is selected and applied to a source text, both lexical translations (for terminals) and reorderings (for nonterminals) are determined. Therefore, rule selection affects both lexical translation and phrase reorderings. However, most of the current tree-based systems ignore contextual information when they select rules during decoding, especially the information covered by nonterminals. This makes the decoder hardly to distinguish rules. Intuitively, information covered by nonterminals as well as contextual information of rules is believed to be helpful for rule selection.

Linguistic and contextual information have been widely used to improve translation performance. It is helpful to reduce ambiguity, thus guiding the decoder to choose correct translation for a source text on phrase reordering. Carpuat and Wu (2007) integrated word-sense-disambiguation (WSD) and phrase-sense-disambiguation (PSD) into a phrased-based SMT system to solve the lexical ambiguity problem. Chan et al. (2007) incorporated a WSD system into the hierarchical SMT system, focusing on solving ambiguity for terminals of translation rules. He et al., extended WSD like the approach proposed in to hierarchical decoders and incorporated the MERS model into a state-of-the-art syntax-based SMT model, the tree-to-string alignment template model. Chiang et al. (2009) used 11,001 features for statistical machine translation

In our research, we integrate dividing a legal sentence based on its logical structure into the first step of the rule selection. We propose a maximum entropy-based rule selection model for tree-based English-Japanese statistical machine translation in legal domain. The

maximum entropy-based rule selection model combines local contextual information around rules and information of sub-trees covered by variables in rules. Therefore, the nice properties of maximum entropy model (lexical and syntax for rule selection) are helpful for rule selection methods better.

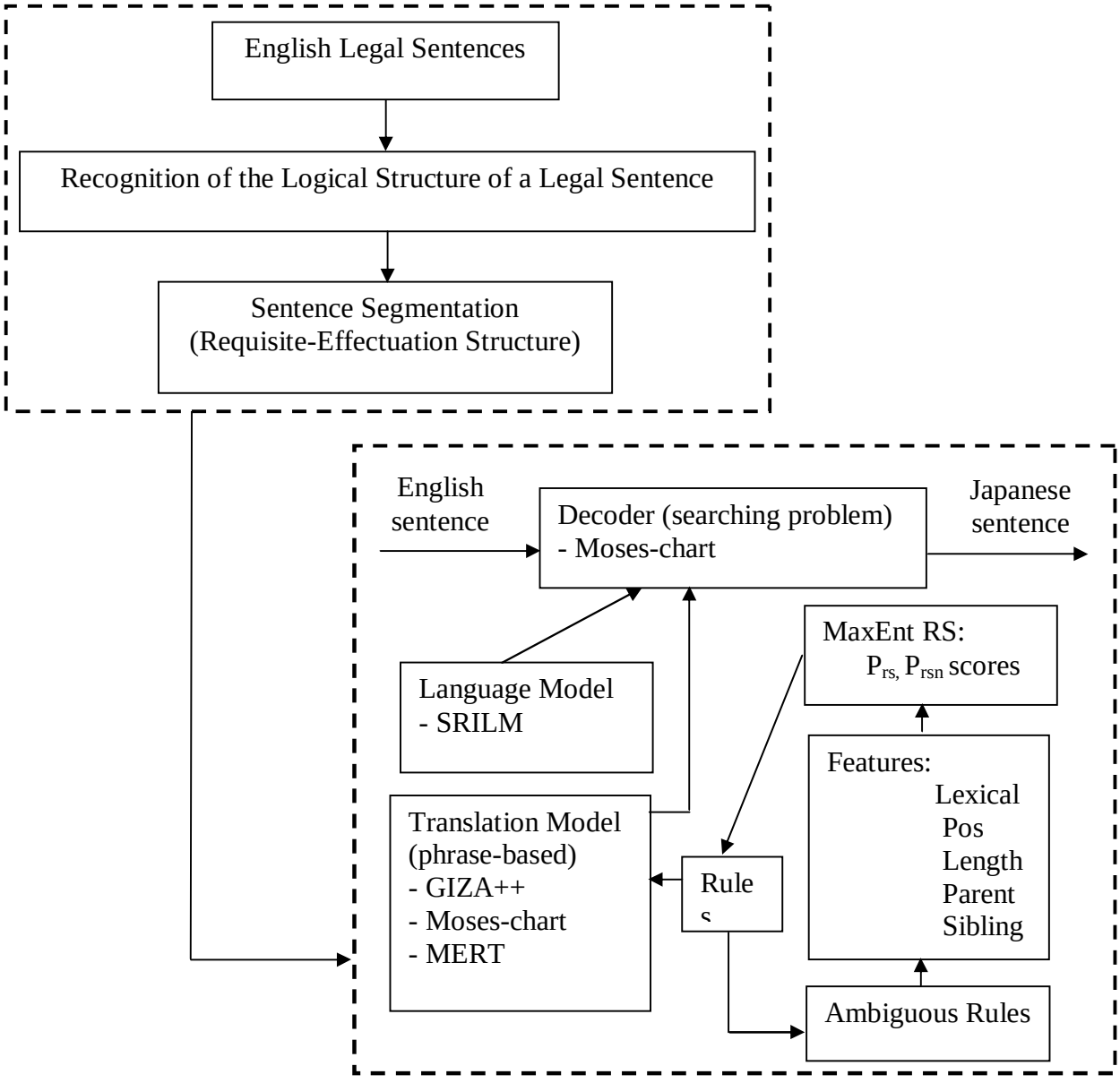


Figure 4.1. The diagram of our proposed method

Our model allows the decoder to perform context-dependent rule selection during decoding. We integrate dividing the legal sentence based on its logical structure as the first step and

incorporate the maximum entropy based rule selection model into a state-of-the-art linguistically tree-based English-Japanese statistical machine translation model. Experiments show that our approach archives significant improvements over the baseline system Moses and Moses-chart.

Our works are described as following:

Firstly, we divide the sentences into the logical structures.

Secondly, we determine baseline system to translate lowercased and tokenized English legal sentences into lowercased and tokenized Japanese legal sentences.

Thirdly, we extract rules from aligned words and English-Japanese legal parallel corpus.

Fourthly, we extract features from rules, parse trees and tagged sentence of English-Japanese legal parallel corpus.

Then, we integrate the features into maximum entropy-based rule selection (MaxEnt RS model); after that integrate the score features into tree-based model.

Next, we evaluate and analyze the results.

Lastly, we test the performance of the model on the large scale corpus.

The diagram of Rule selection for syntax-based English-Japanese SMT shows in Figure 4.1

This chapter describes maximum entropy based-rule selection model (MaxEnt RS model) for English-Japanese statistical machine translation, features of MaxEnt RS model, the way to extract features and method to integrate MaxEnt RS model into the tree-based translation model.

4.1 Maximum Entropy based Rule Selection Model (MaxEnt RS model)

The rule selection task can be considered as a multi-class classification task. For a source-side, each corresponding target-side is a label. The maximum entropy approach (Berger et al., 1996) is known to be well suited to solve the classification problem. Therefore, we build a maximum entropy-based rule selection (MaxEnt RS) model for each ambiguous hierarchical LHS (left-hand side).

Following (Chiang, 2005), we use (α, γ) to represent a SCFG rule extracted from the training corpus, where α and γ are source and target strings, respectively. The nonterminal in α and γ are represented by X_k , where k is an index indicating one-one correspondence between nonterminal in source and target sides. Let us use $e(X_k)$ to represent the source text covered by X_k and $f(X_k)$ to represent the translation of $e(X_k)$. Let $C(\alpha)$ be the context information of source text matched by α and $C(\alpha)$ be the context information of source text matched by γ. Under the MaxEnt model, we have:

$$P_{rs}(\gamma \mid \alpha, e(X_k), f(X_k)) = \frac{\exp[\sum_i \lambda_i h_i(C(\gamma), C(\alpha), e(X_k), f(X_k))]}{\sum_{\gamma'} \exp[\sum_i \lambda_i h_i(C(\gamma'), C(\alpha), e(X_k), f(X_k))]} \quad (4.1)$$

Where h_i a binary feature function, λ_i the feature weight of h_i. The MaxEnt RS model combines rich context information of grammar rules, as well as information of the subphrases which will be reduced to nonterminal X during decoding. However, these information is ignored by Chiang's hierarchical model.

We design five kinds of features for a rule (α, γ): Lexical, Parts-of-speech (POS), Length, Parent and Sibling features.

4.2 Linguistic and Contextual Information for Rule Selection

4.2.1 Lexical Features of Nonterminal

In the each hierarchical rules, there are nonterminals. Features of nonterminal consist of Lexical features, Parts-of-speech features and Length features:

Lexical features, which are the words immediately to the left and right of α, and boundary words of subphrase $e(X_k)$ and $f(X_k)$;

Parts-of-speech (POS) features, which are POS tags of the source words defined in lexical features.

Length features, which are the length of subphrases $e(X_k)$ and $f(X_k)$.

Table 4.1: Lexical features of nonterminals

Side	Type	Name	Description
Source-side	Lexical features	$W_{\alpha -1}$	The source word immediately to the left of α
		$W_{\alpha +1}$	The source word immediately to the right of α
		$WL_{e(X_k)}$	The first word of $e(X_k)$
		$WR_{e(X_k)}$	The last word of $e(X_k)$
	Pos features	$P_{\alpha -1}$	POS of $W_{\alpha -1}$
		$P_{\alpha +1}$	POS of $W_{\alpha +1}$
		$PL_{e(X_k)}$	POS of $WL_{e(X_k)}$
		$PR_{e(X_k)}$	POS of $WR_{e(X_k)}$
	Length feature	$LEN_{e(X_k)}$	Length of source subphrase $e(X_k)$
Target-side	Lexical features	$WL_{f(X_k)}$	The first word of $f(X_k)$
		$WR_{f(X_k)}$	The last word of $f(X_k)$
	Length feature	$LEN_{f(X_k)}$	Length of target subphrase $f(X_k)$

For example, we have a rule, source phrase and source sentence as following:

Rule

$X \rightarrow (X_1$ it officially $X_2 | X_2$ これを $X_1)$

Source Phrase

must announce it officially without delay

遅滞なくこれを公表しなければならない

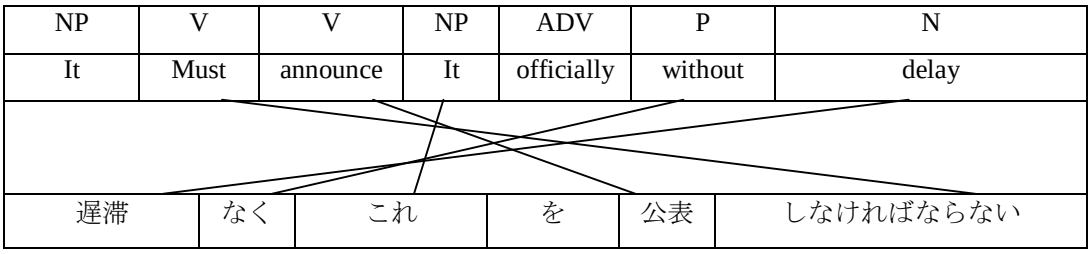
X_1 must announce
X_2 without delay
X_1 公表しなければならない
X_2 遅滞なく

Source sentence

It must announce it officially without delay

遅滞なくこれを公表しなければならない

***Alignment of English-Japanese sentence pair*:**



Features of this example are shown in Table 4.2

Table 4.2: Lexical features of nonterminal of the example

Type	Features
Lexical Features	$W_{\alpha-1}$ = it $WL_{e(X_1)}$ = must $WR_{e(X_1)}$ = announce $WL_{e(X_2)}$ = without $WR_{e(X2)}$ = delay $WL_{f(X_1)}$ =公表 $WR_{f(X_1)}$ = しなければならない $WL_{f(X_2)}$ = 遅滞 $WR_{f(X_2)}$ = なく
POS Features	$P_{\alpha-1}$ = NP $PL_{e(X_1)}$ = V $PR_{e(X_1)}$ = V $PL_{e(X_2)}$ = P $PR_{e(X_2)}$ = N
Length Features	$LEN_{e(X_1)}=2$ $LEN_{e(X_2)}=2$ $LEN_{f(X_1)}=2$ $LEN_{f(X_2)}=2$

4.2.2 Lexical Features around Nonterminals

These features are same meaning as features of nonterminal

Lexical features, which are the words immediately to the left and right of subphrase $e(X_k)$ and $f(X_k)$;

Parts-of-speech (POS) features, which are POS tags of the source words defined in lexical features.

Table 4.3: Lexical features around nonterminal

Side	Type	Name	Description
Source-side	Lexical feature	$WL_{e(X_k)-1}$	The first word immediately to the left of $e(X_k)$
		$WR_{e(X_k)+1}$	The first word immediately to the right of $e(X_k)$
	POS Features	$PL_{e(X_k)-1}$	POS of $WL_{e(X_k)-1}$
		$PR_{e(X_k)+1}$	POS of $WR_{e(X_k)+1}$
Target-side	Lexical features	$WL_{f(X_k)-1}$	The first word of $f(X_k)-1$
		$WR_{f(X_k)+1}$	The last word of $f(X_k)+1$

Table 4.4: Lexical features around nonterminal of the example

	Type	Features
Source-side	Lexical feature	$WL_{e(X_1)-1}$ = remarkable $WR_{e(X_1)+1}$ = in
		$WL_{e(X_2)-1}$ = standard $WR_{e(X_2)+1}$ = the
	POS Features	$PL_{e(X_1)-1}$ = ADJ $PR_{e(X_1)+1}$ = P
		$PL_{e(X_2)-1}$ = N $PR_{e(X_2)+1}$ = DET
Target-side	Lexical features	$WL_{f(X_1)-1}$ =著しい
		$WL_{f(X_2)-1}$ = 国民 $WR_{f(X_2)+1}$ =生活

Example: with a rule:

X →(a remarkable X_1 in the living standard X_2 the nation |
国民 X_2 生活に水準の諸事情に著しい X_1)

We have lexical features around nonterminal as Table 4.4

4.2.3 Syntax Features

Let $R \rightarrow <\alpha, \gamma\sim>$ is a translation rule and $e(\alpha)$ is source phrase covered by α X_k is nonterminal in α, $T(X_k)$ is sub-tree covering X_k.

Parent feature (PF):

The parent node of $T(X_k)$ in the parse tree of source sentence. The same sub-tree may have different parent nodes in different training examples. Therefore, this feature may provide information for distinguishing source sub-trees

Sibling feature (SBF)

The sibling features of the root of $T(X_k)$. This feature considers neighboring nodes which share the same parent node.

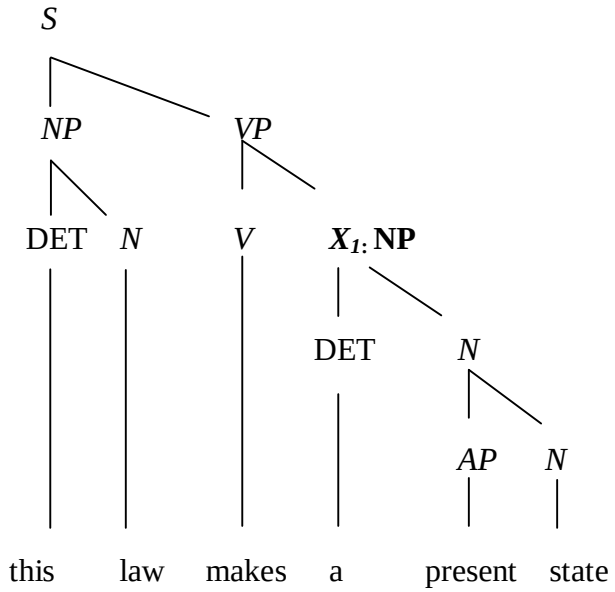


Figure 4.2: Sub-tree covers nonterminal X_1

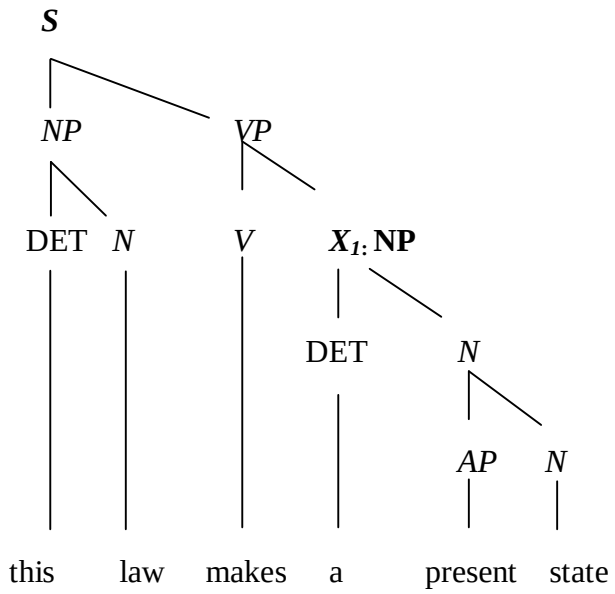


Figure 4.3: Parent feature of sub-tree covers nonterminal X_1: S

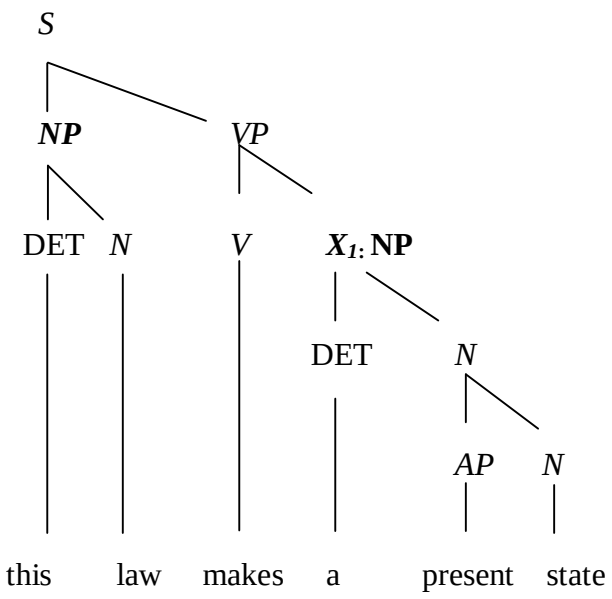


Figure 4.4: Sibling feature of sub-tree covers nonterminal X_1: *NP*

Figure 4.2 shows sub-tree covers nonterminal X_1, Figure 4.3 shows *S* node is the Parent feature of subtree covering X_1 and *NP* node is the Sibling feature shown in Figure 4.4.

Those features: Lexical feature, Parts-of-speech features, Length features, Parent features and Sibling features make use rich of information around a rule, including the contextual information of a rule and the information of sub-trees covered by nonterminals. These features can be gathered according to Chiang's rule extraction method (Chiang, 2005). We use Moses-chart to extract phrases and rules, Cabocha and Stanford Tagger toolkits to tag, tokenize Japanese and English source sentence, Stanford parser to parse English source sentence, after that we use following algorithm to extract features:

for Rule=1 to n
find nonterminal in Rule
for phrase =1 to m
find phrase equals with nonterminal
for source, tag, parse =1 to v
find features for nonterminal:
Lexical features
Parts-of-speech features
Lenght features

find features around nonterminal:

Lexical features

Parts-of-speech features

find syntax features:

Parent features

Sibling features

enfor

endfor

endfor

In Moses-chart, the number of nonterminal of a rule are limited up to 2. Thus a rule may have 36 features at most. After extracting features from training corpus, we use the toolkit implemented by Tsuruoka, Tsujii laboratory, Department of Computer Science, University of Tokyo (2006) to train a MaxEnt RS model for each ambiguous hierarchical LHS.

4.3 Integrating MaxEnt RS Model into Tree-based Model

We integrate the MaxEnt RS model into the tree-based model during the translation of each source sentence. Thus the MaxEnt RS models can help the decoder perform context-dependent rule selection during decoding.

In (Chiang, 2005), the log-linear model combines 8 features: the translation probabilities $P(\gamma \mid \alpha)$ and $P(\alpha \mid \gamma)$, the lexical weights $P_w(\gamma \mid \alpha)$ and $P_w(\alpha \mid \gamma)$, the language model, the word penalty, the phrase penalty, and the glue rule penalty. For integration, we add two new features:

(1) $P_{rs}(\gamma \mid \alpha, e(X_k), f(X_k))$.

This feature is computed by the MaxEnt RS model, which gives a probability that the model selecting a target-side γ given an ambiguous source-side α, considering context information.

(2) $P_{rsn} = exp(1)$.

This feature is similar to phrase penalty feature. In our experiment, we find that some source-sides are not ambiguous, and correspond to only one target-side. However, if

a source-side α' is not ambiguous, the first features P_{rs} will be set to *1.0*. In fact, these rules are not reliable since they usually occur only once in the training corpus. Therefore, we use this feature to reward the ambiguous source-side. During decoding, if an LHS has multiple translations, this feature is set to *exp(1)*, otherwise it is set to *exp(0)*.

The advantage of our integration is that we need not change the main decoding algorithm of a tree-based system. Furthermore, the weights of the new features can be trained together with other features of the translation model.

Chiang (2007) used the CKY (Cocke-Kasami-Younger) algorithm with a cube pruning method for decoding. This method can significantly reduce the search space by efficiently computing the top-n items rather than all possible items at a node, using the k-best algorithms of Huang and Chiang (2005) to speed up the computation. In cube pruning, the translation model is treated as the monotonic backbone of the search space, while the language model score is a non-monotonic cost that distorts the search space. Similarly, in the MaxEnt RS model, source-side features form a monotonic score while target-side features constitute a non-monotonic cost that can be seen as part of the language model.

For translating a source sentence E_I^J, the decoder adopts a bottom-up strategy. All derivations are stored in a chart structure. Each cell *c[i,j]* of the chart contains all partial derivations which correspond to the source phrase e_i^j. For translating a source-side span *[i,j]*, we first select all possible rules from the rule table. Meanwhile, we can obtain features of the MaxEnt RS model which are defined on the source-side since they are fixed before decoding. During decoding, for a source phrase e_i^j, suppose the rule

$X \rightarrow (e_i^k X_1 e_t^j, f_{i'}^{k'} X_1 f_{t'}^{j'})$

is selected by the decoder, where $i \leq k < t \leq j$ and $k+1 < t$, then we can gather features which are defined on the target-side of the subphrase X_1 from the ancestor chart cell *c[k+1, t-1]* since the span *[k+1, t-1]* has already been covered. Then the new feature scores P_{rs} and P_{rsn} can be computed. Therefore, the cost of derivation can be obtained. Finally, the decoding is completed when the whole sentence is covered, and the best derivation of the source sentence E_I^J is the item with the lowest cost in cell *c[I,J]*.

4.4 The Detail of Experiments

The above theory is applied to English-Japanese SMT in legal domain by this thesis with large-scale experiment. This chapter records the details of the experiment, including the software systems, the training and testing corpora, and the typical process that is used by all the experiment of this thesis.

The system for the experiments is built upon existing pieces of software. The engineering work includes the choosing and compiling of the software systems and libraries, the selecting and formatting of corpora, the code analysis in accordance with the theory of the last two chapters, the software development work to combine and coordinate different software systems, and the application of automatic MT evaluation methods. One of the challenges of the experiments is training the system with significantly large amounts of data within a reasonable time frame; the techniques used include filtering dispensable time consuming data, running tasks in parallel, and doing experiments incrementally.

4.4.1 Software

a) Baseline

Moses, a beam-search decoder for factored phrase-based statistical machine translation models, is a statistical machine translation system that allows you to automatically train translation models for any language pair.

Moses supports models that have become known as *phrase-based models* and *tree-based models*. Moses-chart is a main branch of Moses referred as tree-based system.

Moses-chart is strong for language pairs. Moses-chart implements a CKY+ algorithm for an arbitrary number of non-terminals per rule and an arbitrary number of types of non-terminals in the grammar.

The baseline system (Moses-chart) translates lowercased and tokenized source sentences into lowercased and tokenized target sentences.

We chose Moses-chart as a baseline system because the source of Moses-chart is open. It is developed by many experts and also used in many machine translation systems.

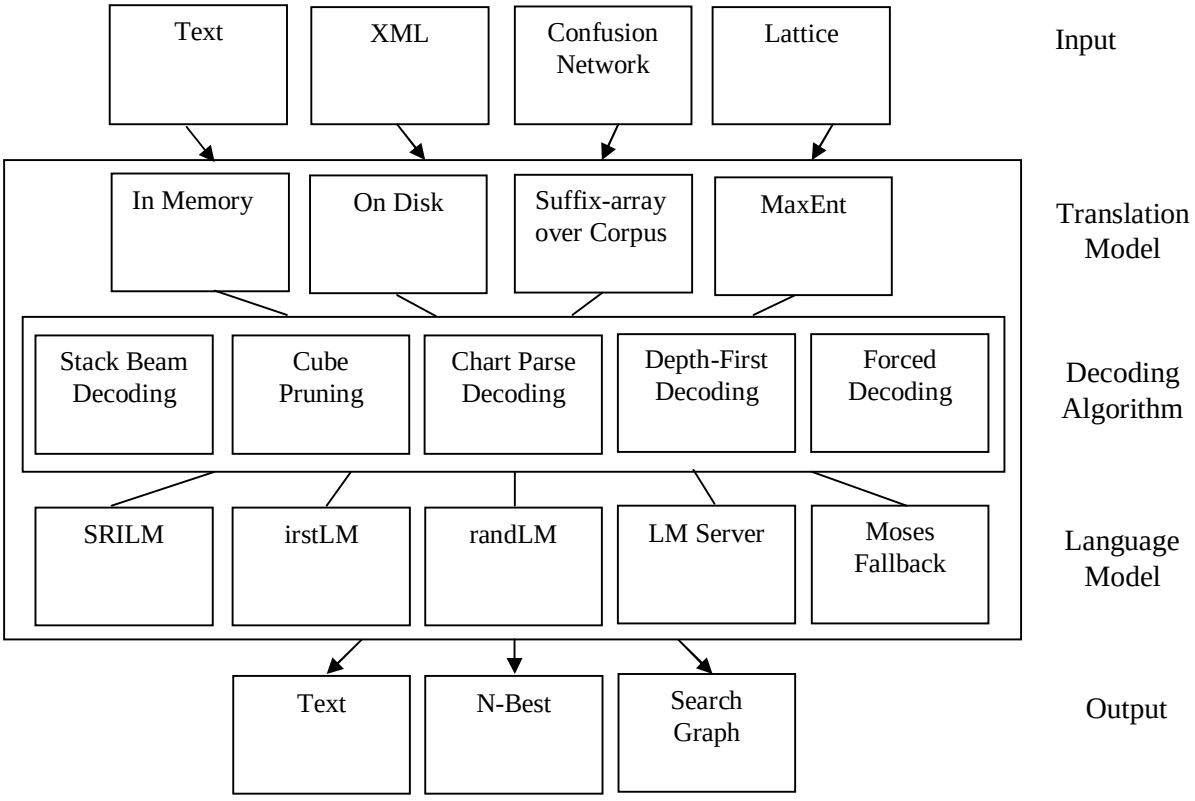


Figure 4.5: The model of Moses-chart

b) Giza++

GIZA++ (Och and Ney, 2000) is a general word alignment tool. It is used by this thesis to obtain word-to-word translation probabilities between Vietnamese and English. It is based on the word alignment models, and it incorporates many features. GIZA++ is written in C++.

c) SRILM

SRILM is a collection of C++ libraries, executable programs, and helper scripts designed to allow both production of and experimentation with statistical language models for speech recognition and other applications. SRILM is freely available for noncommercial purposes. The toolkit supports creation and evaluation of a variety of language model types based on N-gram statistics, as well as several related tasks, such as statistical tagging and manipulation of N-best lists and word lattices.

d) Tokenizer, Tagger

We use CaboCha toolkits (Kudo, 2003) for tokenization of Japanese sentences and Stanford POS Tagger for English sentences.

e) Parser

Stanford parser was mainly written by Dan Klein, with support code and linguistic grammar development by Christopher Manning. This parser is implemented in Java. We use this toolkit to parse English sentences.

f) Maximum Entropy Classification

We chose maximum entropy classification toolkit developed by Tsuruoka, Tsujii laboratory, Department of Computer Science, University of Tokyo (2006). It's also freely distributed under the GNU/GPL license and available online on.

This toolkit is a C++ class library for maximum entropy classification. The main features of this library are: fast parameter estimation using the BLMVM algorithm (Benson and More, 2001), smoothing with Gaussian priors (Chen and Rosenfeld, 1999), modelling with inequality constraints (Kazama and Tsujii, 2003), support for real-valued features, saving/loading a model to/from a file and allowing integrating model data into source code.

We used it with e*xtracted features of ambiguous rules* as input and output as *scores of ambiguous rules.*

4.4.2 Corpus

We conducted the experiments on the English-Japanese translation corpus provided by Japanese Law Translation Database System. The training corpus consisted of 40,000 English-Japanese original sentence pairs, the development and test set consisted of 1,400 and 516 sentence pairs, respectively. The statistics of the corpus is shown in Table 4.5. We tested on 516 English- Japanese sentence pairs. Table 4.6 shows statistics of the test corpus. The test set is recognized and divided by the method described in chapter three. Table 4.7 shows the number of sentences, the statistics of the requisite parts, the effectuation parts and the logical parts after splitting in the test set. Then, we applied rule selection for the split sentence in the test set.

Table 4.5: Statistical table of train and test corpus

Corpus		#words	#sentences
Training corpus	English	990,011	40,000
	Japanese	935,467	
Development corpus	English	45,150	1,400
	Japanese	45,020	
Test corpus	English	17,475	516
	Japanese	17,753	

Table 4.6 Statistics of the test corpus

Name of Law	Number of sentences
Act on General Rule for Application of Law	78
Act on Land and Building Leases	120
Administrative Procedure Act	99
Foreign Exchange and Foreign Trade Act	219
Total	516

Table 4.7 Number of requisite part, effectuation part in the test data

Sentence	516
#of requisite part	436
#of effectuation part	513
#of segment	949

4.4.3 Training

To train the translation model, we first run GIZA++ (Och and Ney, 2000) to obtain word alignment in both translation directions. Then we use Moses-chart to extract SCFG grammar rules. We use Cabocha toolkits to token and tag Japanese sentences; Stanford parser toolkits to tag, pos and parse English source sentence. Meanwhile, we gather lexical and syntax features for training the MaxEnt RS models. The maximum initial phrase length is set to 7 and the maximum rule length of the source side is set to 5.

We use SRI Language modeling toolkit (Stocke, 2002) to train language models. We use minimum error rate training (Och, 2003) integrated in Moses-chart to tune the feature weights for the log-linear model.

The translation quality is evaluated by BLEU metric (Papineni et al.., 2002), as calculated by mteval-v12.pl with case-insensitive matching of n-grams, where n=4.

4.4.4 Baseline + MaxentRS

As we described, we add two new features to integrate the Maxent RS models into the Moses-chart.

(1) $P_{rs}(\gamma \mid \alpha, e(X_k), f(X_k))$.

This feature is computed by the MaxEnt RS model, which gives a probability that the model selecting a target-side γ given an ambiguous source-side α, considering context information.

(2) $P_{rsn} = exp(1)$.

This feature is similar to phrase penalty feature. In our experiment, we find that some source-sides are not ambiguous, and correspond to only one target-side. However, if a source-side α' is not ambiguous, the first features P_{rs} will be set to *1.0*. In fact, these rules are not reliable since they usually occur only once in the training corpus. Therefore, we use this feature to reward the ambiguous source-side. During decoding, if an LHS has multiple translations, this feature is set to *exp(1)*, otherwise it is set to *exp(0)*.

The advantage of our integration is that we need not change the main decoding algorithm of a SMT system. Furthermore, the weights of the new features can be trained together with other features of the translation model.

To run decoder, we share the same pruning setting with the Moses, Moses-chart baseline systems.

We use BLEU metric (Papineni et al., 2002) as calculated by mteval-v12.pl with case-insensitive matching of n-grams, where n=4 and we get the result in Table 4.8

We evaluate both original test sentence and split test sentence with Maxent RS model. We compare the results of four systems: Moses using original test sentence (MM), Moses-chart using original test sentence (MC), Moses-chart using split test sentence (MS)

and Moses-chart applying rule selection or our system (MR). The results are shown in Table 4.8. In Table 4.8, Moses system using original test sentence (MM) got 0.287 BLEU scores, Moses-chart system using original test sentence (MC) got 0.306 BLEU scores, Moses-chart system using split sentence (MS) got 0.318 BLEU scores, using all features defined to train the MaxEnt RS models for Moses-chart using split test sentence our system got 0.329 BLEU scores, with an absolute improvement 4.2 over MM system, 2.3 over MC system and 1.1 over MS system.

In order to explore the utility of the context features, we train the MaxEnt RS models on different features sets. We find that lexical features of nonterminal and syntax features are the most useful features since they can generalize over all training examples. Moreover, Lexical features around nonterminal also yields improvement. However, these features are never used in the baseline.

Table 4.8: BLEU-4 scores (case-insensitive) on English-Japanese corpus.

Lex= Lexical Features, POS= POS Features, Len= Length Feature, Parent= Parent Features, Sibling = Sibling Features.

System	BLEU
MM	0.287
MC	0.306
MS	0.318
MR (MaxEnt RS)	
Lexical features of nonterminal (Lex+POS+Len)	
Lexical features around nonterminal (Pos+Lex)	0.320
Syntax features (Parent and sibling)	0.325
Lexical features of nonterminal + syntax features	0.327
All features	0.329
MM	0.287

4.4.5 The Results and Discussion

When we used MS system to extract rule, we got the rules as Table 4.9:

Table 4.9: Statistical table of rules

Name	Number
The number of rules	1,480,741
The number of rules contain nonterminal	1,126,440
The number of rules don't contain nonterminal	354,298
The number of glue grammar rules	3
The number of rules match test	12,148

Table 4.10: Number of possible source-sides of SCFG rule for English-Japanese corpus and number of source-sides of the best translation.

H-LHS = Hierarchical LHS, AH-LHS = Ambiguous hierarchical LHS

	Rule	NO of H-LHS	NO of AH-LHS
MS	12,148	6,541	3,416
Our system (MR, all features)	12,148	7,741	5,214

Table 4.10 shows the number of source-sides of SCFG rules for English-Japanese corpus. After extracting grammar rules from the training corpus, there are 12,148 source-sides match the test corpus, they are hierarchical LHS's (H-LHS, the LHS which contains nonterminals). For the hierarchical LHS's, 52.22% are ambiguous (AH-LHS, the H-LHS which has multiple translations). This indicates that the decoder will face serious rule selection problem during decoding. We also note the number of the source-sides of the best translation for the test corpus. However, by incorporating MaxEnt RS models, that proportion increases to 67.36%, since the number of AH-LHS increases. The reason is that, we use the feature Prsn to reward ambiguous hierarchical LHS's. This has some advantages. On one hand, H-LHS can capture phrase reorderings. On the other hand, AH-LHS is more reliable than non-ambiguous LHS, since most non-ambiguous LHS's occur only once in

the training corpus. In order to know how the MaxEnt RS models improve the performance of the SMT system, we study the best translation of MS and our system. We find that the MaxEnt RS models improve translation quality in 2 ways:

Better Phrase reordering

Since the SCFG rules which contain nonterminals can capture reordering of phrases, better rule selection will produce better phrase reordering.

Table 4.11 shows translation examples of test sentences in Case 3 in MS and our systems (MR, all features), our system gets better result than the MS system in phrase reordering.

Table 4.11: Translation examples of test sentences in Case 3 in MS and our systems (MR, all features).
The Japanese sentence in Japanese-English translation is the original sentence. The English sentence in English-Japanese translation is the reference translation in the government web page

Sentence	<C> Notwithstanding the preceding paragraph, </C> <T3> the formalities </T3> <A> that comply with the law of the place where said act was done </A> <C> shall be valid. </C>
Split Sentence	the formalities notwithstanding the preceding paragraph, shall be valid. the formalities that comply with the law of the place where said act was done
MS	同項の規定にかかわらず、手続きは、有効なものでなければならない。 行為が行われていたと述べた場所の法律を遵守手続き
Our System (MR, all features)	前項の規定にかかわらず、方式は、有効でなければならない 方式は、当該行為が行われた場所の法律の遵守します。

Better Lexical Translation

The MaxEnt RS models can also help the decoder perform better lexical translation than the baseline. This is because the SCFG rules contain terminals. When the decoder selects a rule for a source-side, it also determines the translations of the source terminals.

The examples of our system get better result than the MS system in lexical translation shown in the underlined parts of Table 4.11.

4.5 Conclusion

Like human translation, machine translation has two essential factors – unit element (unbreakable word or phrase that carries meaning) translation and target sentence organization. The simplest models for SMT are word-based, where the unit elements are words and sentence organization modeled by comparatively simple mechanisms such as word reordering. One of the main improvements of phrase-based models over the word-based models is on the definition of unit elements, which includes phrases. Tree-based models further improved the target sentence organization. The models have improved translation accuracy by evolving towards a higher level of abstraction, while word alignment often serves as the basis for more complex models.

Rule selection is of great importance to tree-based statistical machine translation systems. This is because that a rule contains not only terminals (words or phrases), but also nonterminals and structural information. During decoding, when a rule is selected and applied to a source text, both lexical translations (for terminals) and reorderings (for nonterminals) are determined. Therefore, rule selection affects both lexical translation and phrase reorderings.

In this chapter, we propose a generic lexical and syntax approach for rule selection. We build maximum entropy based-rule selection models for each ambiguous hierarchical source-side of translation rules. The MaxEnt RS models combine rich context information, which can help the decoder perform context-dependent rule selection during decoding. We use dividing a legal sentence based on its logical structure as the first step of rule selection and integrate the MaxEnt RS models into the tree-based SMT model by adding two new features. Experiments show that the lexical and syntax approach for rule selection achieves statistically significant improvements over the state-of-the-art tree-based SMT system-Moses-chart and Moses systems.

5 Sentence Paraphrasing and Named Entity for Legal Translation

This chapter presents how to improve legal translation quality in two ways: sentence paraphrasing and named entity.

Statistical machine translation (SMT) systems learn how to translate by analyzing bilingual parallel corpora. Generally speaking, high-quality translations can be produced when training data is available. However, because of low density of legal language pairs that do not have large-scale parallel corpora, limited amount of training data usually leads to a problem of low coverage in that many phrases encountered at run-time have not been observed in the training data. This problem becomes more serious for higher-order n-grams, and for morphologically richer languages. To overcome the coverage problem of SMT we investigate using sentence paraphrasing approach. We apply a monolingual sentence paraphrasing method for augmenting the training data for statistical machine translation systems by creating it from data that is already available.

The terms (name phrases) for legal texts are difficult to translate as well as to understand, so we apply named entity to improve translation quality. We split the long sentence into several block areas (recognized by NER) that could be translates independently. Firstly, we generate NER training data automatically from a bilingual parallel corpus, employ an existing high-performance English NER system to recognized NEs at the English side, and then project the labels to the Japanese side according to the word alignment. We integrate dividing a legal sentence based on its logical structure into the first step of sentence paraphrasing and named entity. Our proposed method improves the translation quality.

5.1 Sentence Paraphrasing

Using paraphrasing has been proven useful for improving SMT quality. The studies can be classified into two categories by the target of paraphrasing: (1) paraphrasing the input source sentences; (2) paraphrasing the training corpus. In the first category, the proposed approaches mainly focus on handling n-grams that are unknown to the SMT model. Callison-Burch et al. (2006) and Marton et al. (2009) paraphrase unknown terms in the input sentences using phrasal paraphrases extracted from bilingual and monolingual

corpora. Mirkin et al. (2009) rewrite unknown terms with entailments and paraphrases acquired from WordNet. Onishi et al. (2010) and Du et al. (2010) build paraphrase lattices for input sentences and select the best translations using a lattice-based SMT decoder. In the second category of paraphrasing training corpus, Bond et al. (2008) and Nakov (2008) paraphrase the source side of training corpus using hand-crafted rules, He et al. (2011) enriches SMT training data using a statistical paraphrase generating model.

We apply a monolingual sentence paraphrasing method by creating it from data that is already available rather than having to create more aligned data. In particular, we apply sentence-level paraphrasing on the source-language side. The proposed approach augments the training corpus with paraphrases of the original sentences, thus augmenting the training bi-text without increasing the number of training translation pairs needed. It is also monolingual; other related approaches map from the source language to other languages in order to obtain paraphrases.

5.1.1 Method

Given a sentence from the source (English) side of the training corpus, we generate conservative meaning-preserving syntactic paraphrases of that sentence. Each paraphrase is paired with the foreign (Japanese) translation that is associated with the original source sentence in the training bi-text. This augmented training corpus is then used to train an SMT system.

We paraphrase by parsing a sentence to an abstract semantic representation using the English Resource Grammar then generating from the resultant semantic representation using the same grammar. The semantic representation used is Minimal Recursion Semantics (MRS: Copestake et al., 2005). We give an example of the paraphrasing process in Figure 5.2 that shows three kinds of paraphrasing. The input sentence is "*For the Government, it must announce it officially without delay*" It is paraphrased to the MRS shown in Figure 5.1.

From that, five sentences are generated. The paraphrased sentences show two changes. Firstly, the adverb *officially* appears in three positions (pre-verb, post-verb, post-verb-phrase). Lastly, comas appear after adverb. We consider the changes in lexical paraphrases and in syntactic paraphrases. Of course, for most sentences there is a

combination of lexical and syntactic paraphrases. “Score” in Figure 5.2 gives a maximum entropy based likelihood estimate to each of the paraphrases. Note that the highest ranked paraphrase is not in this case the original sentence. The paraphrase is quite conservative: sentence initial *officially* is not generated, as that is given a different semantics (it is treated as focused). There are no open class paraphrases like *film* ≡ *movie*. Only a handful of closed class words are substituted, typically those that get decomposed semantically, (e.g., *everybody* ≡ *every(x),person(x)*).

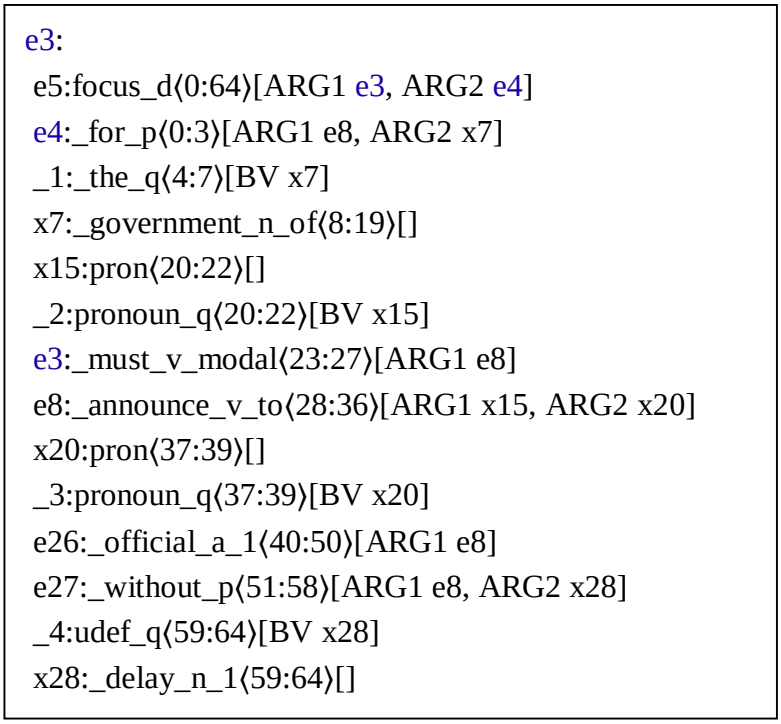
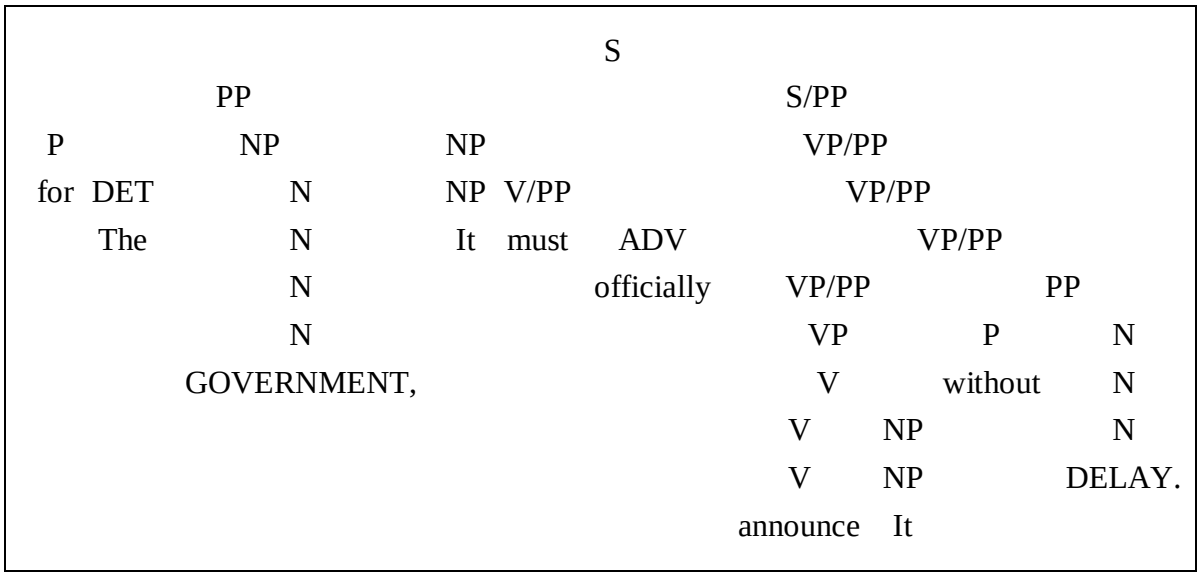
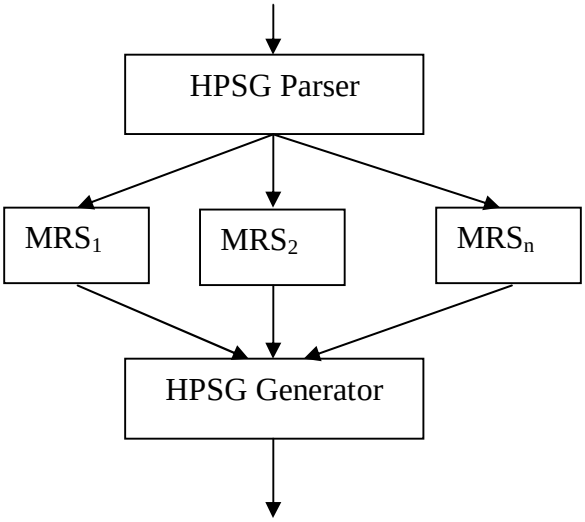


Figure 5. 1: Semantic Representation of
“For the Government, it must announce it officially without delay”



"For the Government, it must announce it officially without delay"



(0) For the government, it must officially announce it without delay. [2.3]
(1) For the government, it must announce it officially without delay. [0.9]
(2) For the government, it must announce it officially, without delay. [0.3]
(3) For the government, it must officially announce it without delay. [0.3]
(4) For the government, it must announce it officially, without delay. [0.1]

Figure 5.2 HPSG parser and paraphrase process for sentence "*For the Government, it must announce it officially without delay"*

5.1.2 Experiment

The LinGO English Resource Grammar (ERG; Flickinger, 2000) is a broad-coverage, linguistically precise HPSG-based grammar of English that has been under development at the Center for the Study of Language and Information (CSLI) at Stanford University since 1993. The ERG was originally developed within the Verb*mobil* machine translation effort, but over the past few years has been ported to additional domains and significantly extended. The grammar includes a hand-built lexicon of around 43,000 lexemes. We are using the development release LinGO (Apr-08). Parsing was done with the efficient, unification-based chart parser, PET (Callmeier, 2002), and generation with the Linguistic Knowledge Base (Copestake, 2002). The ERG and associated parsers and generators are freely available from the Deep Linguistic Processing with HPSG Initiative.

For the most part, we use the default settings and the language models trained in the LOGON project both for parsing and generation (Velldal and Oepen, 2006). However, we set the root condition, which controls which sentences are treated as grammatical, to be robust for parsing and strict for generation. This means that robust rules (e.g. a rule that allows verbs to not agree in number with their subject) will apply in parsing but not in generation. The grammar will thus parse *The dog bark* or *The dog barks* but only generate *The dog barks*.

We attempted to parse all sentences of the legal corpus (Japanese-English translation corpus provided by Japanese Law Translation Database System) with the ERG and the PET parser. However most the sentences in the legal corpus are very long and complex, ERG and the PET parser do not work well on this corpus. So we do experiment on two corpuses. The first is split sentences obtained from dividing a legal sentence based on its logical structure in test corpus, 949 split sentences (516 sentences). The second is original corpus. We filter the second corpus with limitation of the length of the sentence to 30 and applied into this corpus. We got one or more well-formed semantic representation for 90% and 60% of the sentences from the first and second corpus, respectively (the remainder were rejected as ungrammatical). We selected the top ranked representation and attempted to generate from it by using the ERG and the LKB generator. We were able to generate one or more realizations for 95% and 72% of the original sentences in the first and the second corpus. However, many of these gave only one realization and it was identical to the input sentence. For the first corpus, 40% of the sentences had at least one distinct

paraphrase; 30% had two, 25% had three, dropping down to only 0.7% with ten distinct paraphrases. The ratio in the second corpus are 45%, 25%, 15% and 0.5%, respectively.

We use baseline Moses in the ACL 2007 Second Workshop on Statistical Machine Translation with a 5-gram language model. We use morphological analyzers to tokenize our data. We used the Stanford Tagger (Kristina and Christopher, 1994) for English and MeCab (Kudo et al., 2004) for Japanese. Part-of-speech information was discarded after tokenization.

We used the MERT implementation distributed with Moses. We evaluated the effects of adding paraphrases to various initial training data sizes using BLEU and NIST scores. We compared a baseline of no-paraphrases-added to systems with progressively larger numbers of new paraphrased sentence pairs added to each training data size. Table 5.1 shows the statistic of the first corpus before and after adding new paraphrased sentence pairs. The statistic of the second corpus before and after adding new paraphrased sentence pairs is shown in Table 5.2. The first corpus is small, so we use the second corpus for machine translation experiment. Translation result shows in Table 5.3

Overall, we show significant consistent improvements on the legal corpus. Paraphrased SMT systems show statistically significant improvements over the baseline for the majority of the data sizes tested. As is to be expected from the BLEU and NIST scores, the system with paraphrases often gives the better translation.

Table 5.1 Statistic of the first corpus

Name	Japanese	English
Split sentence	949	949
Split sentence + paraphrases		1,271 (+324)

Table 5.2 Statistic of the second corpus

Name	Japanese	English
Training corpus	27,029	27,029
Test corpus	300	300
Train corpus + paraphrases		30,749 (+3720)

Table 5.3 Translation result

	BLEU	NIST
Baseline	0.248	5.08
+ paraphrases	0.253	5.15

We theorize that the additional data provided by our paraphrases results in better phrasal alignments, which, in turn, improves lexical selection and allows the language model to produce more natural-sounding translations.

Compared to Callison-Burch et al. (2006), Madnani et al. (2007), or Nakov (2008) we get a slightly lower improvement in quality. Our paraphrasing does not extract effectively because of two reasons. Firstly, we apply in the legal domain and almost legal sentences are long and complex. All of previous researches do experiment on split sentence. Lastly, our HPSG parser's generation model does not work well on this corpus. Table 5.1 shows that when we apply dividing a legal sentence based on its logical structure before paraphrasing, our training corpus increase 34%. Therefore, we should split sentence before paraphrasing and retrain our HPSG parser's generation model to effectively rank the new lexical paraphrases.

5.2 Named Entity

Because the terms (name phrases) for legal texts are difficult to translate as well as to understand, so we apply splitting the long sentence into several block areas that could be translates independently. We propose named entity to solve this problem.

We present a method to generate Japanese NER training data from a bilingual corpus automatically. Our method trades off manual effort to annotate named entities in legal text for effort to identify pairs of parallel documents, which is easier than NE manual annotation.

In this section we describe our approach of generating NER training data from a parallel corpus. The framework of our system consists of five components as follows:

- **Sentence Segmentation**: Dividing legal sentence based on its logical structures.

- **Alignment:** Word alignment is performed on a discourse- level aligned bilingual corpus.

- **English NER:** We identify NEs on the English side of the parallel corpus, making use of an existing high performance English NER system.

- **NE Candidates Generation:** Based on the result of the word alignment, we project the English NE labels to the Japanese side and generate training data candidates.

- **Integrating Named Entity into SMT.** We split the long sentence into several named entities and integrate them into SMT

5.2.1 Sentence Segmentation

We divide legal sentence based on its logical structure as described in the chapter two: dividing and translating legal sentence based on its logical structure.

5.2.2 Alignment and Automatic English NER

We use GIZA++ toolkit for word alignment. This toolkit can generate one-to-many word alignments in a certain direction (Japanese to English or English to Japanese). However, we need many-to-many alignments. Hence, we need GIZA++ to run on the bilingual corpus in both directions and merge the results, as shown in Figure 5.3.

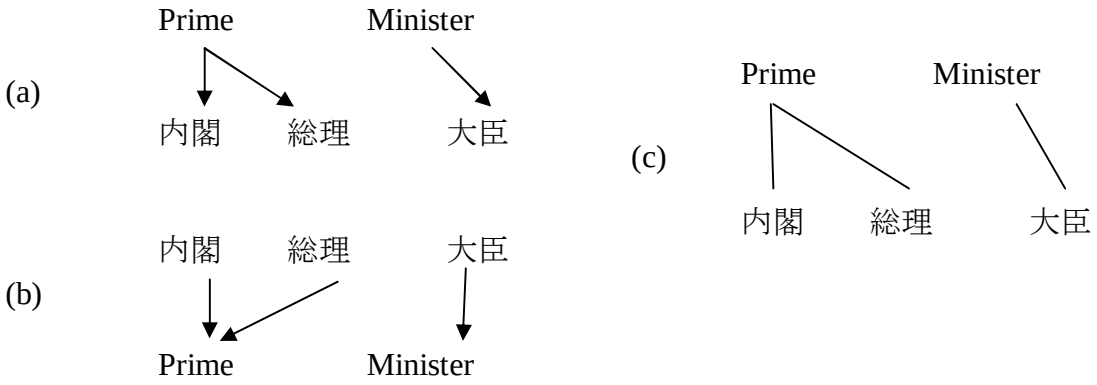


Figure 5.3. (a) Word Alignment from English to Japanese. (b) Word Alignment from Japanese to English. (c) The Merged Result of Both Directions.

English NER is easier than Japanese because of the capitalization information and the needlessness of word segmentation. So the performance of English NER systems is usually higher than the Japanese ones on average. Hence a widely used open-source NER system, Stanford Named Entity Recognizer is employed to label NEs on the English side of the parallel corpus. The system is based on linear chain Conditional Random Field (CRF)

(J.Lafferty et al., 2001) sequence models and can recognize three kinds of named entities (PERSON, LOCATION and ORGANIZATION).

5.2.3 Japanese NE Candidates Generation

After the English NER, we map the English NE labels to the Japanese side to discover Japanese NEs candidates, according to the result of word alignment. We consider all related alignment pairs of every word within an English NE. There are also some English words connecting with NULL at Japanese side. We ignore these word alignment pairs. According to the alignment, we project the NE labels from English to Japanese and generate the named entity candidates on the Japanese side.

5.2.4 Training Data Selection

However, the generated NER training data candidates are noisy because of the errors in English NER or word alignment. In this section, we present the strategies of selecting training data.

a. Filtering Based on Rules

As the common definition, a named entity is a continuous string, whether it is in English or in Japanese. So we assume that every named entity alignment pair is a closed alignment pair of two continuous strings, as shown in Fig. 5.3 (a).

Based on this assumption, we make two alternative rules to filter the training data candidates. One is a soft filtering rule to retain training instances as many as possible. Another is a hard filtering rule to guarantee the quality of the generated corpus. These two rules are shown as follows:

- **Rule 1 (the soft rule):** Label a Japanese NE candidate as a non-NE, if a word within it has an alignment pair with an English word out of the corresponding English NE, such as Fig. 5.3 (b).

- **Rule 2 (the hard rule):** Discard the whole sentence where there is a case satisfying Rule 1.

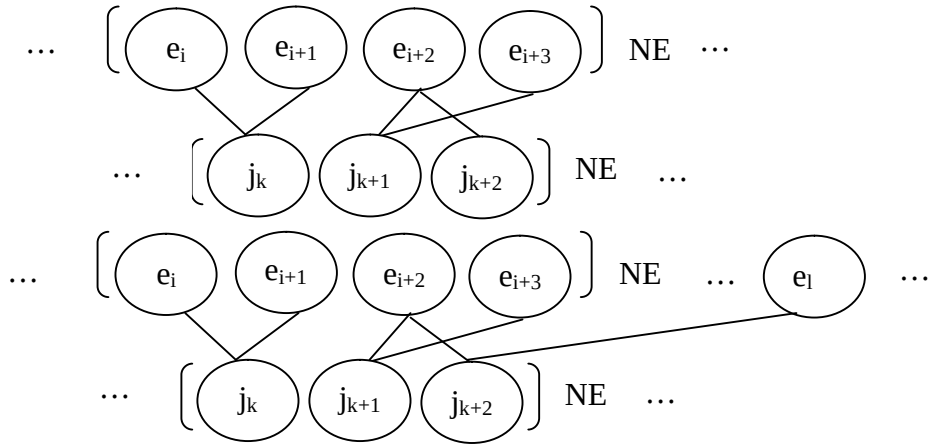


Figure 5.4: (a) An eligible case; (b) An ineligible case. In (b), the word alignment pair e_i – j_k is against the rule, while $l > i+3$ or $l < i$.

Rule 1 prefers to keep training instances as many as possible. But it may make some NEs be labeled as non-NE mistakingly on the Japanese side for incorrect word alignments, which are the noises in the generated training data. Rule 2 prefers to guarantee the quality of the generated data but may make useful training instances be discarded and the data scale shrinking. Based on the rules, we can filter lots of ill conditioned named entity candidates, such as overlapped entities, nested entities and so on.

b. Filtering Based on Scores

Although many ill conditioned candidates are filtered out by the rules, the remaining data is still noisy because of the incorrect labeling of the English NER and the incorrect NE alignment. In fact, the accuracy of NE alignment is only affected by the boundary alignment of English and Japanese NEs. In other words, we do not care about how to align within or without the NEs. Hence, we score Japanese named entity candidates by formula 5.1 (followed by Fu et al., 2011).

$$score(N_j) = \varphi(N_e) \prod_{w \in B(N_j)} \left(\frac{1}{|A(w)|} \sum_{<e,w> \in A(w)} p(<e,w>) \right) \qquad (5.1)$$

Here, $\varphi(N_e)$ denotes the confidence of the English named entity N_e, which is derived from Stanford NER system. $B(N_j)$ denotes the boundaries of the Japanese named entity N_j, which are actually the left-most and the right-most word within. *e* denotes an English word, and *w* denotes a Japanese word. A(w) denotes all related alignment pairs of word *w* in current Japanese named entity N_j. p(<e, w>) denotes the probability of alignment (e, w), which is obtained from GIZA++.

As mentioned before, Stanford NER is based on CRF. The inference of CRF is that given an observable sequence $\vec{x}$, we want to find the most likely set of labels $\vec{y}$ for $\vec{x}$. The probability of $\vec{y}$ given $\vec{x}$ is calculated followed by Lafferty et al., 2001:

$$p(\vec{y}|\vec{x}) = \frac{1}{Z(\vec{x})}\prod_{j=1}^{n}\delta_j(\vec{x},\vec{y}) \tag{5.2}$$

$$Z(\vec{x}) = \sum_{\vec{y}'}\prod_{j=1}^{n}\delta_j(\vec{x},\vec{y}') \tag{5.3}$$

$$\delta_j(\vec{x},\vec{y}) = \exp\left(\sum_{i=1}^{m}\lambda_i f_i\left(y_{j-1}, y_j, \vec{x}, j\right)\right) \tag{5.4}$$

In formulae 5.2, 5.3 and 5.4, *j* denotes the index of the *j*th word in sequence $\vec{x}$. *n* denotes the length of $\vec{x}$. *m* denotes the number of the features. Now the substring $x_k x_{k+1}\ldots x_{k+l}$ in $\vec{x}$ is labeled as an NE N_e. The label sequence of N_e is $y^*_k\, y^*_{k+1}\ldots\, y^*_{k+l}$ which is denoted as $\overrightarrow{y^*_{N_e}}$.

We compute the marginal probability $\varphi(N_e)$ as follows:

$$\varphi(N_e) = \frac{Z(N_e,\vec{x})}{Z(\vec{x})} \tag{5.5}$$

$$Z(N_e, \vec{x}) = \sum_{\vec{y}' : y'_{k \ldots k+l} = \overrightarrow{y^* N_e}} \prod_{j=1}^{n} \delta_j(\vec{x}, \vec{y}') \tag{5.6}$$

The factor $\varphi(N_e)$ of every English NE is used to measure the confidence of NER. We apply the forward-backward algorithm to compute them. For p(<e,w>) we use the probabilities of alignment pairs which are computed by GIZA++. GIZA++ outputs the probability p(t|s) of translating source word *s* as target word *t*. There are two kinds of probabilities of alignment in two directions. Since our alignment is bidirectional, we merge the probabilities in two directions to come up with formula.

$$p(< e, w >) = \max\left\{p(e|w), p(w|e)\right\} \tag{5.7}$$

Particularly we set zero while the translation pair "s → t" does not exist in the translation table given by GIZA++. We set experiential thresholds for every category to filter the Japanese NE candidates.

5.2.5 Integrating Named Entity into SMT.

After aligning and filtering the named entities in the sentence pair, we get the Japanese named entities. We locate the area of these named entities in the sentences and translate them with the baseline.

We use constrains proposed by Kohn and Haddow (2009) in Moses decoder to insert named entities for open and close brackets.

The zone constraints introduced by Koehn and Haddow (2009) are compatible for our work.

- Zones: Words within zone have to be translated without reordering with outside material.

Moses decoder uses zone constraints with the following restrictions:

> If a <zone> tag is detected, then a block is identified until a </zone> tag is found. The text between tags <zone> and </zone> is identified and translated as a block.

We apply the zone constraint as the block area marker. Let us consider the example shown below.

Japanese

借地権の存続期間が満了する場合において、借地権者が契約の更新を請求したときは、建物がある場合に限り、前条の規定によるもののほか、従前の契約と同一の条件で契約を更新したものとみなす。ただし、借地権設定者が遅滞なく異議を述べたときは、この限りでない。

English

In cases where the Land Lease Right Holder requests the renewal of the contract in cases where the duration of the Land Lease Right expires , limited to cases where there is a building , in addition to cases pursuant to the provisions of the preceding Article, the contract shall be deemed to have been renewed with the same conditions as those of the prior contract ; provided , however , that this shall not apply when the Lessor makes an objection without delay.

The Stanford NER system recognizes English sentence as follows:

In cases where the <ORGANIZATION> Land Lease Right Holder </ORGANIZATION> requests the renewal of the contract in cases where the duration of the <ORGANIZATION> Land Lease Right </ORGANIZATION> expires , limited to cases where there is a building , in addition to cases pursuant to the provisions of the preceding Article, the contract shall be deemed to have been renewed with the same conditions as those of the prior contract ; provided , however , that this shall not apply when the <PERSON> Lessor </PERSON> makes an objection without delay.

Using aligning and filtering for English-Japanese sentence pair, we have two named entity alignments, after that we add <zone> markers into Japanese sentence, the sentence looks like this:

借地 権 の 存続 期間 が 満了 する 場合 において 、<zone> 借地 権 者 </zone> が 契約 の 更新 を 請求 し た とき は 、 建物 が ある 場合 に 限り 、 前条 の 規定 による もの の ほか 、 従前 の 契約 と 同一 の 条件 で 契約 を

更新 した ものと みなす。 ただし、<zone> 借地 権 設定 者 </zone> が 遅滞 なく 異議 を 述べた ときは、 この 限り でない。

5.2.6 Experiment

We conducted the experiments on the Japanese-English translation corpus provided by Japanese Law Translation Database System. The training corpus consisted of 42,870 Japanese-English sentence pairs, the development and test set consisted of 1,400 and 516 sentence pairs, respectively. Table 5.4 shows statistics of the corpus.

Table 5.4: Statistics of the corpus

Corpus	#words		#sentences
Training corpus	English	1,061,044	42,870
	Japanese	1,002,587	
Development corpus	English	45,150	1,400
	Japanese	45,020	
Test corpus	English	17,475	516
	Japanese	17,753	

We used Moses as a decoder in the experiments. Moses used a phrase table with a maximum phrase length of 7, a lexical reordering model with msd-bidirectional-fe, and a distortion-limit of -1 (unlimited). For the language model, the SRI Language Modeling Toolkit (SRILM) is used. We used the data smoothing technique Knerser-Ney and experiment with n=5. The feature weights were optimized for BLEU by Minimum Error Rate Training (MERT), using the development sentences.

We use the Stanford NER system to recognize English sentence. After that, the aligning and filtering is used for Japanese sentence. Table 5.5 shows the statistics of the #LOCATION, #ORGANIZATION, #PERSON in the English test data recognized by Stanford NER system and the number of zone after filtering in Japanese side.

We evaluated the translation result using BLEU and NIST score. Table 5.6 shows the results. The baseline shows the results without using any reordering constraints “zone”.

Table 5.5: The statistics of the number of zones in the test data

	Type	English	Japanese
Stanford NER	#LOCATION	82	
	#ORGANIZATION	980	
	#PERSON	2	
Filtering	#LOCATION		70
	#ORGANIZATION		940
	#PERSON		2
	#zone		1012

Table 5.6: Translation results

Method	BLEU	NIST
Baseline	0.248	5.08
Original sentence + zone	0.252	5.14
Split sentence + zone	0.256	5.20

From the result in Table 5.6, we see that by adding zone constraints, the translation quality improves in two metrics NIST and BLEU score. When we integrate sentence segmentation with named entity, we get the better results.

Comparing with other research as Kim and Ehara (1994) proposed using a rule-based method to split long sentences into multiple sentences; Xu et al. (2005) proposed to separate a sentence pair into sub-pairs based on a modified IBM Model 1; Sudoh et al. (2010) proposed dividing the source sentence into small clauses using a syntactic parser; Xiong et al. (2010) used Maximum Entropy Markov Models to learn the translation boundaries based on word alignments in hierarchical trees our method is simple and does not require complicated processes. We use same constraints as Chooi-Ling Goh et al. (2011). However, Chooi-Ling Goh et al. (2011) used rule-based method to look for continuous sequence of words that fall into some predefined POS tag and we use automatic statistical approach basing on NER and alignment.

5.3 Conclusion

In this chapter we investigate using sentence paraphrasing and named entity to improve translation quality.

We propose a monolingual sentence paraphrasing method for augmenting the training data for statistical machine translation systems by creating it from data that is already available.

We generate NER training data automatically from a bilingual parallel corpus, employ an existing high-performance English NER system to recognized NEs at the English side, and then project the labels to the Japanese side according to the word alignment. We apply splitting the long sentence into several block areas that could be translates independently.

We integrate dividing a legal sentence based on its logical structure into sentence paraphrasing and named entity as the first step.

Our experiment shows that the proposed method improves the translation quality over the baseline system.

6. Conclusion and Future Works

In this chapter, we summaries the main results and contribution of this thesis's research, and we discuss directions for future work.

6.1 Summary of the Thesis

In this thesis, we focus on improvement of translation quality in legal domain. Among the six chapters of the thesis, the main chapters are 3, 4, and 5. The main contributions of the thesis can be summarized as follows:

We propose three methods to deal with three mentioned problems of legal translation.

Firstly, to solve the first problem: sentences in legal texts are usually long and complicate, we propose a novel method for translating a legal sentence by dividing it based on the logical structure of a legal sentence. We first recognize the logical structure of a legal sentence using statistical learning model with linguistic information. Then we segment a legal sentence into parts of its structure and translate them with statistic machine translation models. In this study, we applied into the phrased-based and the tree-based models separately and evaluated them with baseline models. With this method, our experiments on Japanese-to-English and English-to-Japanese translations show that the method achieves better translations on measuring by the BLEU, NIST and TER scores. The subjective evaluation also shows better results.

Secondly, solving the problem in several language pairs such as English-Japanese the target phrase order differs significantly from the source phrase order, selecting appropriate synchronous context-free grammars translation rule (SCFG) to improve phrase-reordering is especially hard in the tree-based model, we propose using rich linguistic and contextual information for rule selection specifically:

- We divide the sentence into the logical structures.
- We use rich linguistic and contextual information for both non-terminals and terminals. Linguistic and contextual information around terminals have never

been used before, we see that these new features are very useful for selecting appropriate translation rules if we integrate them with the features of non-terminals.

- We propose a simple and sufficient algorithm for extracting features in rule selection.
- We use Moses-chart to extract translation rules with rich linguistic and contextual information. Moses-chart system is a tree-based model developed by many machine translation experts and used in many systems, so that, our model is more generic.
- We use a simple way to classify features by using maximum entropy-based rule selection model and incorporate this model into a state-of-the-art syntax-based SMT model, the tree-based model (Moses-chart). We obtain substantial improvements over the Moses-chart and Moses system.

Lastly, with the problem the terms (name phrases) for legal texts are difficult to translate as well as to understand, we propose sentence paraphrasing and named entity approaches. We apply a monolingual sentence paraphrasing method for augmenting the training data for statistical machine translation systems by creating it from data that is already available. We generate NER training data automatically from a bilingual parallel corpus, employ an existing high-performance English NER system to recognized NEs at the English side, and then project the labels to the Japanese side according to the word alignment. We apply splitting the long sentence into several block areas that could be translate independently. We integrate dividing a legal sentence based on its logical structures into the first step of sentence paraphrasing and named entity. Our proposed method achieves better translation quality.

6.2 Future Work

In the future work, we will focus on the remaining and related issues in this thesis. When we apply dividing and translating legal text basing on the logical structure of a legal sentence, one of limitations of our model in chapter 3 is there are cases that a translation of a sentence differs semantically from a translation of the split sentence, the current model performs well depending on the recognition of the logical structure of a legal sentence, and

our model is just applied into test phase. We will study to integrate split sentences into training, investigate more sophisticated features to improve the recognition of the logical structure of a legal sentence.

In chapter 4 we presented about rule selection for tree-based model, we also intend exploring more sophisticated features for the maximum entropy-based rule selection models, and test the performance of the maximum entropy-based rule selection model on a large scale corpus as well as on the other models.

With our sentence paraphrasing and named entity in chapter 5, one of effective is dividing a long legal sentence to smaller segments, consequently we also plan to integrate split sentence by our method in chapter 3 to this model in the training and explorer more linguistic and contextual information of the sentence to improve translation quality.

Publications

Journal

[1] Bui Thanh Hung, Nguyen Le Minh and Akira Shimazu. (2013). “Dividing and Translating Legal Text based on the Logical Structure of a Legal Sentence”, submitted to Journal of Natural Language Processing (revised).

[2] Bui Thanh Hung, Nguyen Le Minh and Akira Shimazu. (2013). “Translating Legal Sentence by Segmentation and Rule Selection". International Journal on Natural Language Computing, Volume 2, Number 4, pp. 35-54, August 2013.

Referred International Conference

[3] Bui Thanh Hung, Nguyen Le Minh and Akira Shimazu. (2011). “Using Rich Linguistic and Contextual Information for Tree-Based Statistical Machine Translation”. In Proceedings of International Conference on Asian Language Processing, pp. 189-192.

[4] Bui Thanh Hung, Nguyen Le Minh and Akira Shimazu. (2012). “Sentence Splitting for Vietnamese-English Machine Translation”. In Proceedings of the Fourth International Conference on Knowledge and Systems Engineering, pp. 156-160.

[5] Bui Thanh Hung, Nguyen Le Minh and Akira Shimazu. (2012). “Divide and Translate Legal Text Sentence by Using its Logical Structure”. In proceedings of 7th International Conference on Knowledge, Information and Creativity Support Systems, pp. 18-23, 2012.

Bibliography

[Ambati et al., 2009] Ambati, V., Lavie, A., and Carbonell, J. (2009). “Extraction of syntactic translation models from parallel data using syntax from source and target languages”. In MT Summit.

[Atefeh and Guy, 2008] Atefeh F. and Guy L. (2008). “Automatic Translation of Court Judgments”. In Proceedings of the Eighth Conference of the Association for Machine Translation in the Americas.

[Atefeh and Guy, 2009] Atefeh F. and Guy L. (2009). “Machine Translation of Legal Information and Its Evaluation”. In Proceedings of the 22nd Canadian Conference on Artificial Intelligence: Advances in Artificial Intelligence, pp 64-73.

[Bach et al., 2010] Bach, N., X., Minh, N., L., and Shimazu, A. (2010). “PRE Task: The Task of Recognition of Requisite Part and Effectuation Part in Law Sentences”, in International Journal of Computer Processing of Languages (IJCPOL), Volume 23, Number 2.

[Banerjee and Lavie, 2005] Banerjee, S. and Lavie, A. (2005). “METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments.” In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, pp. 65–72 Ann Arbor, Michigan. Association for Computational Linguistics.

[Blum and Mitchell, 1998] Blum .A, Mitchell T. (1998). “Combining labeled and unlabeled data with co-training”. In Proceedings of the Workshop on Computational Learning Theory.

[Bond et al., 2008b] Bond, F., Nichols, E., Appling, D. S., and Paul, M. (2008b). “Improving Statistical Machine Translation by Paraphrasing the Training Data.” In Proceedings of IWSLT 2008, pp. 150–157 Hawaii. Bond, F., Oepen, S., Siegel, M., Copestake, A., and Flickinger, D. (2005). “Open Source Machine Translation with DELPH-IN.” In Open-Source MT: Workshop at MT Summit X, pp. 15–22 Phuket.

[Byrd et al., 1994] Byrd, R. H., Nocedal, J., and Schnabel, R. B. (1994). "Representations of quasi-Newton matrices and their use in limited memory methods". Math. Prog. 63, 4, pp. 129–156.

[Callison-Burch et al., 2006] Callison-Burch, C., Koehn, P., and Osborne, M. (2006). "Improved Statistical Machine Translation Using Paraphrases." In Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, pp. 17–24.

[Carpuat and Dekai, 2007] Carpuat, M. and Dekai W. (2007). "Improving statistical machine translation using word sense disambiguation". In Proceedings of EMNLP-CoNLL, pp. 61–72.

[Chan et al., 2007] Chan, Yee S., Hwee T. Ng., and Chiang D. (2007). "Word sense disambiguation improves statistical machine translation". In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, pp. 33-40.

[Chiang et al., 2009] Chiang D., Knight K., and Wang W. (2009). "11,001 new features for statistical machine translation". In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics.

[Chiang, 2005] Chiang D. (2005). "A hierarchical phrase-based model for statistical machine translation". In Processing of the 43rd Annual Meeting of the Association for Computational Linguistics, page 263-270.

[Chiang, 2005] Chiang D. (2007). "Hierarchical phrase-based translation". Computational Linguistics, pages 33(2):201–228.

[Colin and Chris, 2005] Colin B. and Chris C. (2005). "Paraphrasing with bilingual parallel corpora". In Proceedings of ACL.

[Collins et al., 2005] Collins, M., Koehn, P., and Kucerova, I. (2005). "Clause Restructuring for Statistical Machine Translation." In Proceedings of the 43rd Annual Meeting of the ACL, pp. 531–540.

[Copestake et al., 2005] Copestake, A., Flickinger, D., Pollard, C., and Sag, I. A. (2005). "Minimal Recursion Semantics. An Introduction." Research on Language and Computation, 3 (4), pp. 281–332.

[Copestake, 2002] Copestake, A. (2002). “Implementing Typed Feature Structure Grammars”. CSLI Publications.

[Doi and Eiichiro, 2003] Doi, T. and Eiichiro S. (2003). “Input Sentence Splitting and Translatiing”. In Proceedings of the HLT/NAACL: Workshop on Building and Using Parallel Texts.

[Erik and Hervé, 2001] Erik, T., K., S., and Hervé D. (2001). “Introduction to the CoNLL-2001 Shared Task: Clause Identification”. In Proceedings of CoNLL, pp.53-57.

[Erik, 2002] Erik, T., K., S. (2002). “Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition”. In Proceedings of CoNLL, pp.1-4.

[Fanz, 2003] Fanz, J., O. (2003). “Minimum Error Rate Training in Statistical Machine Translation”. In Proceedings of ACL, pages 160-167.

[Flickinger, 2000] Flickinger, D. (2000). “On Building a More Efficient Grammar by Exploiting Types.” Natural Language Engineering, 6 (1), pp. 15–28. (Special Issue on Efficient Processing with HPSG).

[Franz and Hermann, 2000] Franz, J., O., and Hermann, N. (2000). “Improved Statistical Alignment Models”. Proc. of the 38th Annual Meeting of the Association for Computational Linguistics, Hongkong, China, pp. 440-447, October.

[Furuse et al., 2001] Furuse, O., Yamada, S., and Yamamoto, K. (2001). “Splitting Ill-formed Input for Robust Spoken-Language Translation” Transactions of IPSJ, vol.42 No.45.

[Galley et al., 2004] Galley, M., Hopkins, M., Knight, K., and Marcu, D. (2004). “What’s in a translation rule”. In Proc. of HLT/NAACL-04.

[Galley et al., 2006] Galley, M., Jonathan, G., Knight, K., Marcu, D., DeNeefe, S., Wei, W., and Thayer, I. (2006). “Scalable Inference and Training of Context-Rich Syntactic Translation Models”. In Proceedings of COLING-ACL, pages 961–968.

[Goh and Eiichiro, 2011] Goh, C., and Eiichiro, S. (2011). “Spliting Long Input Sentences for Phrase-based Statistical Machine Translation”, In Proceedings of the 17th Annual Meeting of Association for Natural Language Processing, pp. 802-805.

[Goodman and Bond, 2009] Goodman, M., W., and Bond, F. (2009). “Using Generation for Grammar Analysis and Error Detection.” In Joint conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, pp. 109–112 Singapore.

[Guzman and Garrido, 2007] Guzman, H., F. and Garrido L. (2007). “Using Translation Paraphrases from Trilingual Corpora to Improve Phrase-Based Statistical Machine Translation: A Preliminary Report.” In Proceedings of the Mexican International Conference on Artificial Intelligence, pp. 163–172 Los Alamitos, CA, USA. IEEE Computer Society.

[Haghighi and Klein (2009)] Haghighi, A., Klein, D. (2009). “Simple Coreference Resolution with Rich Syntactic and Semantic Features”. In Proceedings of EMNLP, pp.1152-1161.

[He et al., 2008] He, Z., Liu, Q., and Lin, S. (2008). “Improving statistical machine translation using lexicalized rule selection”. Proceedings of the 22nd International Conference on Computational Linguistics, August 2008, pp. 321-328.

[Jinhua et al., 2010] Jinhua, D., Jie, J., and Andy, W. (2010). “Facilitating Translation Using Source Language Paraphrase Lattices”. In Proceedings of EMNLP, pages 420-429.

[Katayama, 2007] Katayama, T. (2007). “COE Research Monograph Series”, vol.2, JAIST press.

[Katayama, 2007] Katayama, T. (2007). “Legal Engineering-An Engineering Approach to Laws in E-society Age”, in Proc. of the 1st International Workshop on JURISIN.

[Katrin et al., 2007] Katrin, T., Joachim, W., and Udo, H. (2007). “Sentence and Token Splitting Based on Conditional Random Fields”, in PACLING 2007 - Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics, pp. 49-57. Melbourne, Australia.

[Kim and Ehara, 1994] Kim, Y., and Terusama, E. (1994). “A Method for Partitioning of Long Japanese Sentences with Subject Resolution in J/E Machine Translation”, In Proceedings of International Conference on Computer Processing of Oriental

Languages, pp. 467-473.

[Kimura et al., 2008] Kimura, Y., Nakamura, M., and Shimazu, A. (2008). “Treatment of Legal Sentences Including Itemized and Referential Expressions - Towards Translation into Logical Forms”. In New Frontiers in Artifical Intelligence, volume 5447 of LNAI, pp.242-253.

[Koehn and Haddow, 2009] Koehn, P., and Haddow, B. (2009). “Endinburgh’s Submission to all Track of the WMT2009 Shared Task with Reordering and Speed Improvements to Moses”. In Proceedings of the Fourth Workshop on Statistical Machine Translation, pp. 160-164.

[Koehn et al., 2003] Koehn P., Och, F., J., and Marcu, D. (2003). “Statistical Phrase-Based Translation”. In Proceedings of HLT/NAACL, pages 48–54

[Koehn et al., 2007] Koehn, P., Shen, W., Federico, M., Bertoldi, N., Callison-Burch, C., Cowan, B., Dyer, C., Hoang, H., Bojar, O., Zens, R., Constantin, A., Herbst, E., Moran, C., and Birch, A. (2007). “Moses: Open Source Toolkit for Statistical Machine Translation.” In Proceedings of the ACL 2007 Demo and Poster Sessions, pp. 177–180 Prague.

[Koehn, 2004] Koehn, P. (2004). “Pharaoh: a beam search decoder for phrase-based statistical machine translation models”. In Proceedings of the Sixth Conference of the Association for Machine Translation in the Americas, pages 115–124.

[Koehn, 2010] Koehn, P. (2010). “Statistical machine translation”, 488 pages, Cambridge press.

[Komachi et al., 2006] Komachi, M., Matsumoto, Y., and Nagata, M. (2006). “Phrase Reordering for Statistical Machine Translation Based on Predicate-Argument Structure.” In Proceedings of IWSLT 2006.

[Kudo et al., 2004] Kudo, T., Yamamoto, K., and Matsumoto, Y. (2004). “Applying Conditional Random Fields to Japanese Morphological Analysis.” In EMNLP 2004 Proceedings, pp. 230–237 Barcelona, Spain. Association for Computational Linguistics.

[Kudo, 2003] Kudo, T. (2003). “Yet Another Japanese Dependency Structure Analyzer”, http://chasen.org/taku/software/cabocha/

[Lafferty et al., 2001] Lafferty, J., McCallum, A. and Pereira, F. (2001). “Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data”, in Proceedings of ICML, pp. 282–289.

[Li et al., 2005] Li, X., Zong, C., and Hu, R. (2005). “Hierarchical Parsing Approach with Punctuation Processing for Long Chinese Sentences”, In Proceedings of the Second International Joint Conference on Natural Language Processing, Jeju, Republic of Korea, pp. 7-12.

[Liang, 2005] Liang, P. (2005). “Semi-Supervised Learning for Natural Language”. Master's thesis, Massachusetts Institute of Technology.

[Lopex, 2008] Lopex, A. (2008). “Statistical Machine Translation”. In ACM Computing Surveys 40(3): Article 8, pages 1–49, August.

[Madnani et al., 2007] Madnani, N., Ayan, N., F., Resnik, P., and Dorr, B. (2007). “Using Paraphrases for Parameter Tuning in Statistical Machine Translation.” In Proceedings of the Second Workshop on Statistical Machine Translation, pp. 120–127 Prague, Czech Republic. Association for Computational Linguistics.

[Marton et al., 2009] Marton, Y., Callison-Burch, C., and Resnik, P. (2009). “Improved Statistical Machine Translation Using Monolingually-Derived Paraphrases.” In EMNLP 2009 Proceedings, pp. 381–390 Singapore. Association for Computational Linguistics.

[Marton et al., 2009] Marton, Y., Callison-Burch, C., and Resnik, P. (2009). Improved Statistical Machine Translation Using Monolingually-Dervied Paraphrases. In Proceedings of EMNLP, pages 381-390.

[Matthew et al., 2006] Matthew, S., Dorr, B., Schwartz, R., Micciulla, L., and John, M. (2006). “A Study of Translation Edit Rate with Targeted Human Annotation”, Proceedings of Association for Machine Translation in the Americas.

[McCallum et al., 2000] McCallum, A., Freitag, D., Pereira, F. (2000). “Maximum entropy Markov models for information extraction and segmentation”. In Proceedings of ICML, pp.591-598.

[Melamed, 2004] Melamed, I., D. (2004). “Algorithms for Syntax-Aware Statistical Machine Translation”. In Proceedings of the Conference on Theoretical and Methodological Issues in Machine Translation (TMI), Baltimore.

[Mi et al., 2008] Mi, H., Huang, L., and Liu, Q. (2008). “Forest-based translation”. In Proceedings of ACL-08: HLT, pages 192–199, Columbus, Ohio. Association for Computational Linguistics.

[Murata et al., 2000] Murata, M., Uchimoto, K., Ma, Q., and Isahara, H. (2000). “Bunsetsu Identification Using Categoryexclusive Rules”. In Proceedings of COLING, pp.565-571.

[Nakamura et al., 2007] Nakamura, M., Nobuoka, S., and Shimazu, A. (2007). “Towards Translation of Legal Sentences into Logical Forms”, in Proceedings of the 1st International Workshop on JURISIN.

[Nakov, 2008] Nakov, P. (2008). “Improved Statistical Machine Translation Using Monolingual Paraphrases.” In Proceedings of the European Conference on Artificial Intelligence (ECAI’08) Patras, Greece.

[Nguyen et al., 2007] Nguyen, P., T., Shimazu, A., Minh, N., L., and Vinh, N., V. (2007). “A syntactic transformation model for statistical machine translation”. International Journal of Computer Processing of Oriental Languages (IJCPOL), 20(2):1–20.

[Nguyen et al., 2011] Nguyen, B., Qin, G., Stephan, V. and Alex, W. (2011). “TriS: A Statistical Sentence Simplifier with Log-linear Models and Margin-based Discriminative Training”, In Proceedings of the 5th International Joint Conference on Natural Language Processing-JCNLP.

[Och and Ney, 2003] Och, F. J. and Ney, H. (2003). “A Systematic Comparison of Various Statistical Alignment Models.” Computational Linguistics, 29 (1), pp. 19–51.

[Orasan, 2000] Orasan, C. (2000). “A Hybrid Method for Clause Splitting in Unrestricted English Texts”, in Proceedings of ACIDCA, Monastir, Tunisia.

[Papineni et al., 2002] Papineni, Kishore, Roukos, S., Ward, T., and Zhu, W. (2002). “BLEU: A Method for Automatic Evaluation of Machine Translation”, in Proceedings of the 40th Annual Meeting of the ACL, pp.311-318.

[Sha and Pereira, 2003] Sha, F., and Pereira, F. (2003). Shallow parsing with conditional random fields. In Proceedings of NAACL, pp.213-220.

[Shiqi et al., 2009] Shiqi, Z., Xiang, L., Ting, L., and Sheng, L. (2009). Application-driven Statistical Paraphrase Generation. In Proceedings of ACL, pages 834-842.

[Shirai et al., 1993] Shirai, S., Ikehara, S., and Kawaoka, T. (1993). "Effects of Automatic Rewriting of Source Language within a Japanese to English MT System." In Proceedings of the Fifth International Conference on Theoretical and Methodological Issues in Machine Translation, pp. 226–239 Kyoto, Japan.

[Stolcke, 2002] Stolcke. (2002). "SRILM-An Extensible Language Modeling Toolkit", in Proceedings of International Conference on Spoken Language Processing, vol. 2, (Denver, CO), pp. 901-904.

[Sudoh et al., 2010] Sudoh, K., Kevin, D., Tsukada, H., Hirao, T., and Nagata, M. (2010). "Divide and Translate: Improving Long Distance Reordering in Statistical Machine Translation", in Proceedings of the Joint 5th Workshop on SMT and MEtricsMATR, pp 418-427.

[Sutton et al., 2006] Sutton, C., and McCallum, A. (2006). An Introduction to Conditional Random Fields for Relational Learning. In Introduction to Statistical Relational Learning, Chapter 4, MIT Press.

[Takano et al., 2010] Takano, K., Nakamura, M., Oyama, Y., and Shimazu, A. (2010). Semantic Analysis of Paragraphs Consisting of Multiple Sentences - Towards Development of a Logical Formulation System. In Proceedings of JURIX, pp. 117-126.

[Takashi et al., 2010] Takashi, O., Utiyama, M., and Sumita, E. (2010). Paraphrase Lattice for Statistical Machine Translation. In Proceedings of ACL, pages 1-5.

[Tanaka et al., 1993] Tanaka, K., Kawazoe, I., and Narita, H. (1993). "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.

[Tsuruoka and Tsujii, 2005] Tsuruoka, Y. and Tsujii, J. (2005). "Bidirectional Inference with the Easiest-First Strategy for Tagging Sequence Data", Proceedings of HLT/EMNLP, pp. 467-474.

[Tsuruoka, 2011] Tsuruoka, Y. (2011). “A Simple C++ Library for Maximum Entropy Classification”. http://www-tsujii.is.s.u-tokyo.ac.jp/ tsuruoka/maxent/.

[Turian et al., 2010] Turian, J., Ratinov, L., and Bengio, Y. (2010). “Word Representations: A Simple and General Method for Semi-Supervised Learning”. In Proceedings of ACL, pp.384-394.

[Vincent, 2010] Vincent, Ng. (2010). “Supervised Noun Phrase Coreference Research: The First Fifteen Years”. In Proceedings of ACL, pp. 1396-1411, 2010.

[Wang and Waibel, 1997] Wang, Y. and Waibel, A. (1997). “Decoding Algorithm In Statistical Machine Translation”**.** Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics, Madrid, Spain, July.

[Watanabe et al., 2002] Watanabe, T., Shimohata, M., and Sumita, E. (2002). “Statistical Machine Translation on Paraphrased Corpora.” In Proceedings of the Third International Conference on Language Resources and Evaluation, pp. 2074–2081 Las Palmas, Spain.

[Xiong et al., 2009] Xiong, H., Xu, W., Mi, H., Liu, Y., Lu, Q. (2009). “Sub-Sentence Division for Tree-Based Machine Translation”, in Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP, Short Papers, Singapore, pp. 137-140.

[Xu et al., 2005] Xu, Jia, Zens, R., and Ney, H. (2005). “Sentence Segmentation Using IBM Model Alignment Model 1”, in Proceedings of the EMNLP, pp. 280-287.

[Yamada and Knight, 2001] Yamada, K. and Knight, K. (2001). “A syntax-based statistical translation model”. In Proc. of ACL, pages 523–530.

[Yamada and Knight, 2002] Yamada, K. and Knight, K. (2002). A Decoder for Syntax-Based Statistical MT". Proc. of the Conference of the Association for Computational Linguistics (ACL).

[Yamamoto, 2001] Yamamoto, K. (2001). “Paraphrasing Spoken Japanese for Untangling Bilingual Transfer.” In Proceedings of Natural Language Processing Pacific Rim Symposium 2001, pp. 203–210.

[Zhu et al., 2010] Zhu, Z., Bernhard, D. and Gurevych, I. (2010). “A Monolingual Tree-based Translation Model for Sentence Simplification”, In Proceedings of the 23rd International Conference on Computational Linguistics, Coling.

[Zollmann and Venugopal, 2006] Zollmann, A., and Venugopal, A. (2006). “Syntax augmented machine translation via chart parsing”. In Proceedings on the Workshop on Statistical Machine Translation, pages 138–141, New York City. Association for Computational Linguistics.