Rule selection for syntax-based Vietnamese-English statistical machine translation

By Bui Hung Thanh

A thesis submitted to
School of Information Science,
Japan Advanced Institute of Science and Technology,
in partial fulfillment of the requirements
for the degree of
Master of Information Science
Graduate Program in Information Science

Written under the direction of
Professor Akira Shimazu

September, 2010

Rule selection for syntax-based Vietnamese-English statistical machine translation

By Bui Hung Thanh (0810207)

A thesis submitted to
School of Information Science,
Japan Advanced Institute of Science and Technology,
in partial fulfillment of the requirements
for the degree of
Master of Information Science
Graduate Program in Information Science

Written under the direction of
Professor Akira Shimazu

and approved by
Professor Akira Shimazu
Associate Professor Kiyoaki Shirai
Associate Professor Yoshimasa Tsuruoka

August, 2010 (Submitted)

Rule selection for syntax-based Vietnamese-English statistical machine translation

Bui Hung Thanh (0810207)
School of Information Science
Japan Advanced Institute of Science and Technology
August 10, 2010

Keywords: Syntax-based SMT, Hierarchical phrase-based model, Rule selection, Maximum entropy-based rule selection

Abstract

The syntax-based statistical machine translation model uses 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 the correct target-side for a given source side. This is rule selection.

Rule selection is of great importance to syntax-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. However, most of the current syntax-based systems ignore contextual information when they select rules during decoding, especially the information covered by nonterminals. This makes it difficult for the decoder to distinguish rules. Intuitively, information covered by nonterminals as well as contextual information of rules is believed to be helpful for rule selection.

In this work, rule selection for syntax-based Vietnamese-English statistical machine translation, we propose a maximum entropy-based rule selection model for syntax-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 incorporate the maximum entropy-based rule selection model into a state-of-the-art syntax-based Vietnamese-English statistical machine translation model. Experiments show that our approach achieves significant improvements over the baseline system.

This thesis is organized into three main parts. The first chapter presents the introduction and overview of the thesis. The second and the third chapters summarize the related theories by a literature review, giving a detailed exposition of the theory of statistical machine translation and rule selection for syntax-based statistical machine translation. By discussing the experimental output, the last chapter summarizes this thesis and proposes further work.

Acknowledgments

This thesis could have never been completed without the help of several people.

At first, I want to thank my supervisor, Professor Akira Shimazu. Not only he gave an opportunity to study in this exiting natural language processing field, but he has also encouraged and advised me all the time since I entered his Lab.

I'm also thankful to Professor Kiyoaki Shirai, who has been discussing and giving me inspirations.

I would like to thank Dr. 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 should also express my gratitude to colleagues of Shimazu's Lab at Jaist. They gave me invaluable advices and comments, and most importantly cheered me up all the time.

Lastly, I would like to thank my family for sharing my happiness, difficulties all the time and supporting me as always.

Content

List of Figuress

List of Tables

Chapter 1
Introduction

1.1 Introduction

1.1.1 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 early 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. However, 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 of 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 doubted 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 all, 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 according to the level of linguistic analysis (and generation) required by a system to produce translations. Usually, this can be graphically expressed in the machine translation pyramid in the Figure 1-1.

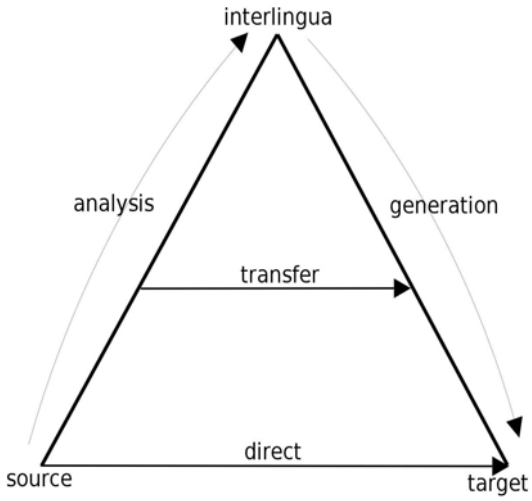

Figure 1-1: The machine translation pyramid

Generally speaking, the bottom of the pyramid represents those systems which do not perform any kind of linguistic analysis of a source sentence in order to produce a target sentence. Moving upwards, 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, each of these approaches is briefly discussed in the following, 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 a 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 made manually, thus involving a great deal of expert human labour and knowledge of comparative grammar of a language pair. Apart from that, when several competing rules can be applied, it is difficult for 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 [Hut92].

Interlingua-based translation

This approach advocates for the deepest analysis of a source sentence, reaching to 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 in 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 system 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 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 systems 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 the statistical approach to machine translation.

1.1.2 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 statistical approach contrasts with the rule-based approaches to machine translation as well as with example-based machine translation.

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. Nowadays it is by far the most widely-studied machine translation method. Mainly, three factors account for this increasing interest:

There is a growing availability of parallel texts (though this applies, in general, only to major languages in terms of presence in internet), coupled with increasing computational power. This enables research on statistical models which, in spite of their huge number of parameters (or probabilities) to estimate, are sufficiently represented in the data.

The statistical methods are more robust to speech disfluencies or grammatical faults. As no deep analysis of a source sentence is done, these systems seek the most probable translation hypothesis given a source sentence, assuming the input sentence is correct.

Last but not least, shortly after their introduction, these methods proved at least as good or even better as rule-based approaches in various evaluation campaigns. A clear example is the German project VerbMobil, which concluded that preliminary statistical approaches outperformed other approaches, on which research had been focused for many years.

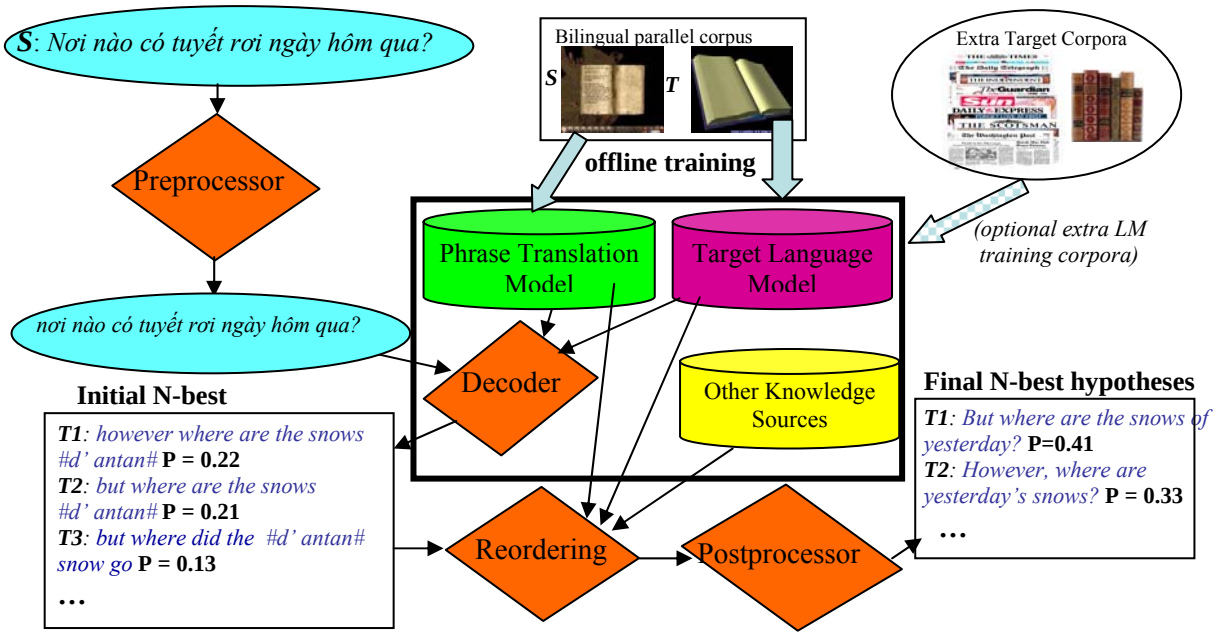

Figure 1-2: Structure of typical statistical machine translation system[34]

1.1.3 Rule selection for syntax-based statistical machine translation

The syntax-based statistical machine translation (SMT) models (Chiang, 2005 [9]; Liu et al., 2006 [26]; Galley et al., 2006 [13]; Huang et al., 2006 [20]) use rules with hierarchical structures (synchronous context free grammar-SCFG) as translation knowledge, which can capture long-distance reorderings. Generally, a translation rule consists of a left-hand side (LHS) and a right-hand side (RHS). The LHS and RHS can be words, phrases, or even syntactic trees, depending on SMT models. Translation rules can be learned automatically from parallel corpus. Usually, an LHS may correspond to multiple RHS in the multiple rules. Therefore, in statistical machine translation, the rule selection task is to select the correct RHS for an LHS during decoding.

1.2 Overview

Chapter 2 provides the detailed background, including the theory of statistical machine translation, hierarchical phrase-based model and rule-selection for syntax based statistical machine translation.

Chapter 3: Introduces the rule selection for syntax-based Vietnamese-English statistical machine translation.

Chapter 4 shows the detail of experiments, including the software used, the corpus texts, the process of a typical experiment and brief introductions of the programs written for the process. This chapter also includes a summary of the research questions corresponding to each experiment.

Chapter 5 discusses the research questions with the experiment results. It draws a conclusion of this thesis and gives suggestion for future work.

1.3 Contribution

Up to now, there is no research about using hierarchical model for Vietnamese-English translation. We select hierarchical model as our baseline model for Vietnamese-English statistical machine translation.

In this work, we propose a novel solution for rule selection for syntax-based Vietnamese-English SMT. We use maximum entropy approach to combine various context features, context words of rules, boundary words of phrases, parts-of-speech information and the information of sub-trees covered by nonterminal in a rule. Therefore, the decoder can use rich context information to perform context-dependent rule selection. We build a maximum entropy-based rule selection (MaxEnt RS) model for each ambiguous hierarchical LHS (Left-hand-side), which contains nonterminals and corresponds to multiple RHS's (Right-hand-side) in multiple rules. We integrate the MaxEnt RS models into the state-of-the-art hierarchical SMT system (Chiang, 2005 [9]). Experiments show that the contextual information can help the decoder to perform a context-

dependent rule selection. Thus, the translation quality of the state-of-the-art SMT system improves, and the improvements are statistically significant.

Our contribution is displayed in two sides:

Research side:

We propose a new solution for Vietnamese-English translation (hierarchical model), use a new kind of feature for rule selection (lexical features around terminal) and propose a new algorithm to extract features for rule selection. In addition, we use nice property of maximum entropy model to combine all features to help rule selection methods.

Technique side:

The Moses-chart is new baseline for syntax-based statistical machine translation. It's developed by many machine translation experts and used in many machine systems. When we used it, we had experiment in installing, using. It's very helpful to develop a new model for machine translation. Beside that, using maximum entropy classifier, coltech-parser, SRILM, vn-tokenizer, vn-tagger also helps us understanding deeply about machine translation system. Basing on research and technique side, we can find the good way for our research.

Chapter 2
Background

2.1 The theory of statistical machine translation

The translation process in statistical machine translation can be formulated as follows: The translation for $f(^{I}_{1})$ is the target string which maximizes probability of $P(e^{I}_{1}| f^{J}_{1})$. Assuming that every target language string $e^{I}_{1} = e_1 \ldots e_I$ is assigned a probability $P(e^{I}_{1})$ of being the language model of the target language and a probability $P(e^{I}_{1}| f^{J}_{1})$ of being an admissible translation for the given source language string $f_1^J = f_1 \ldots f_J$. According to Bayes' decision rule, the optimal translation for f^{J}_{1} is the target string which maximizes the product of the target language model $P(e^{I}_{1})$ and the string translation model $P(f^{I}_{1}| e^{J}_{1})$.

$$e^{I}_{1} = \arg\max \{P(e^{I}_{1}| f^{J}_{1})$$
$$= \arg\max \{P(e^{I}_{1}) P(f^{J}_{1}| e^{I}_{1})$$

Many existing systems for statistical machine translation make use of a special way of structuring the string translation model ([Brown et al. 1993a], [Dagan et al. 1993], [Kay and Roscheisen 1993], [Vogel et al. 1996]): The correspondence between the words in the source and the target string is described by alignments that assign target word positions to each source word position. The probability of a certain target language word to occur in the target string is assumed to depend basically only on the source words aligned to it. The overall architecture of the statistical translation approach is depicted in Figure 2.1. This figure already anticipates the fact that the source strings will be transformed in a certain manner.

Once we specified the Bayes decision rule for statistical machine translation, we have to address three problems (Ney 2001)[14]:

• The modeling problem, i. e. how to structure the dependencies of source and target language sentences;

• The search problem, i. e. how to find the best translation candidate among all possible target language sentences;

• The training problem, i. e. how to estimate the free parameters of the models from the training data.

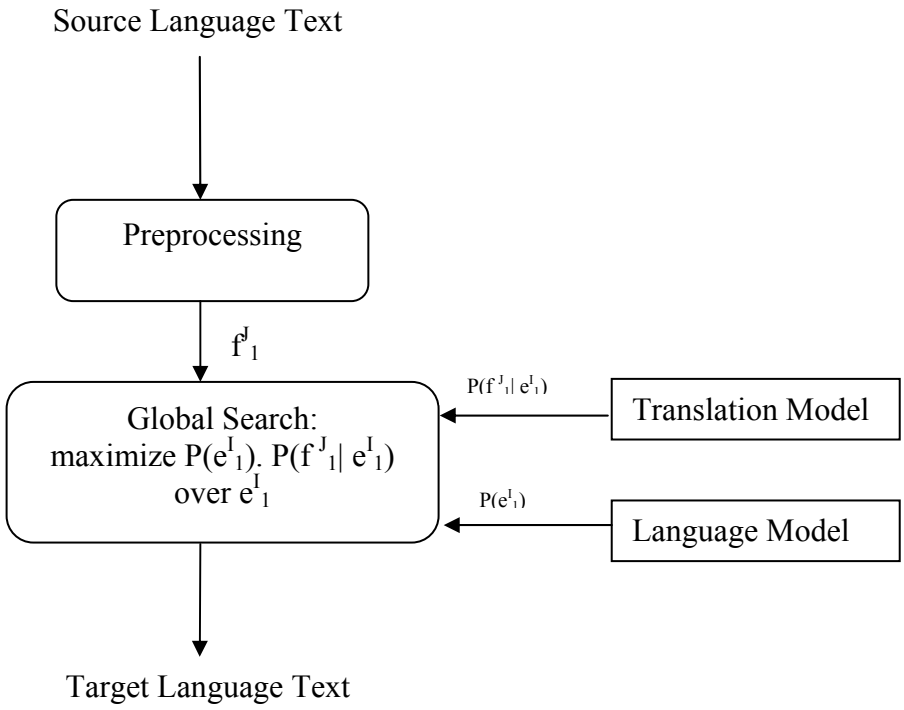

Figure 2-1: Architecture of the statistical machine translation approach based on Bayes' decision rule.

Various researchers have shown better translation quality with the use of phrase translation. The basic idea of phrase-based translation is to segment the given source sentence into phrases, then translate each phrase and finally compose the target sentence from these phrase translations.

Phrase-based MT can be traced back to Och's alignment template model[11], which can be re-framed as a phrase translation system. Other researchers augmented their systems with phrase translation, such as Yamada [43], who used phrase translation in a syntax-based model.

Marcu introduced a joint-probability model for phrase translation. At this point, most competitive statistical machine translation systems use phrase translation, such as the CMU, IBM, ISI, and Google systems, to name just a few. Phrase-based systems came out ahead at a recent international machine translation competition (DARPA TIDES Machine Translation Evaluation 2003-2006 on Chinese-English and Arabic-English).

Of course, there are other ways to do machine translation. Most commercial systems use transfer rules and a rich translation lexicon. Until recently, machine translation researches have been focused on knowledge based systems that use an interlingua representation as an intermediate step between input and output.

There are also other ways to do statistical machine translation. There are some efforts in building syntax-based models that either use real syntax trees generated by syntactic parsers, or tree transfer methods motivated by syntactic reordering patterns.

The phrase-based statistical machine translation model presented here was defined by (Koehn et al. 2003[24]). The alternative phrase-based methods differ in the way the phrase translation table is created, which we discuss in detail below.

2.1.1 Model

The figure below 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 may be reordered.

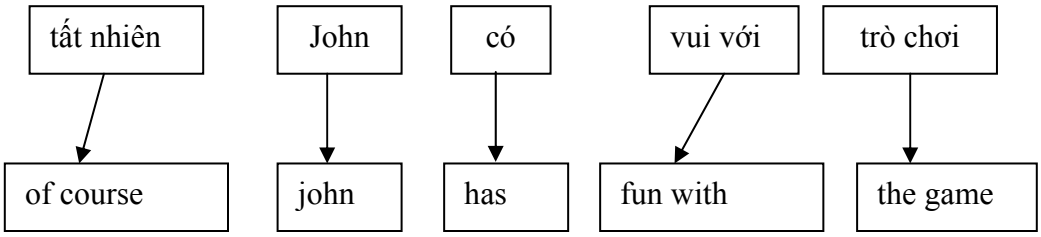

Figure 2-2: The process of phrase-based translation

In this section, we will define the phrase-based machine translation model formally. The phrase translation model is based on the noisy channel model. We use Bayes' rule to reformulate the translation probability for translating a foreign sentence **f** into English **e** as

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

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. We assume 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 ith English phrase, and end_{i-1} denotes the end position of the foreign phrase that was translated into the $(i-1)$th English phrase.

We use 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.

In order to calibrate the output length, we introduce a factor ω (called word cost) 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 our model is

$$e_{best} = argmax\_e\ p(e|f) = argmax_e\ p(f|e)\ p\_LM(e)\ \omega^{length(e)}$$

where $p(f|e)$ is decomposed into

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

2.1.2 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, the problem was focused as a shared task at a recent data driven machine translation workshop (ACL 2005 workshop on building and using parallel texts: data-driven machine translation and beyond, June 29-30, 2005).

Presently, the most common tool to establish a word alignment is the toolkit Giza++ (Och and Ney, 2000[28]). 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.

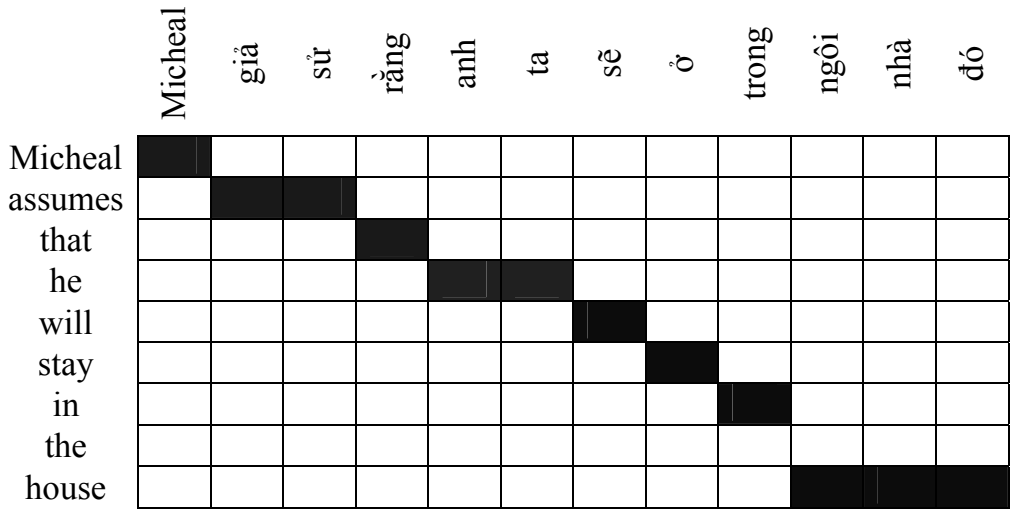

Figure 2-3: Word alignment from English to Vietnamese

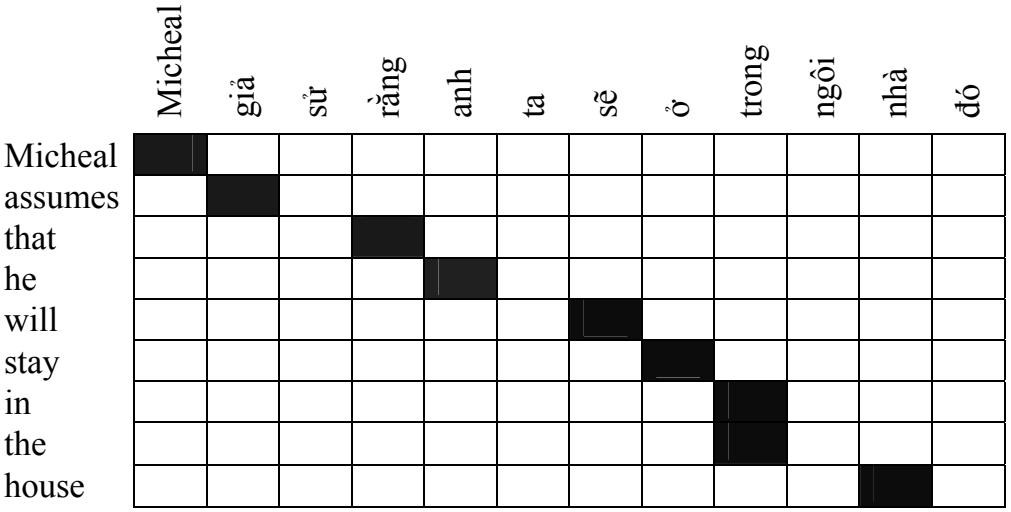

Figure 2-4: Word alignment from Vietnamese to English

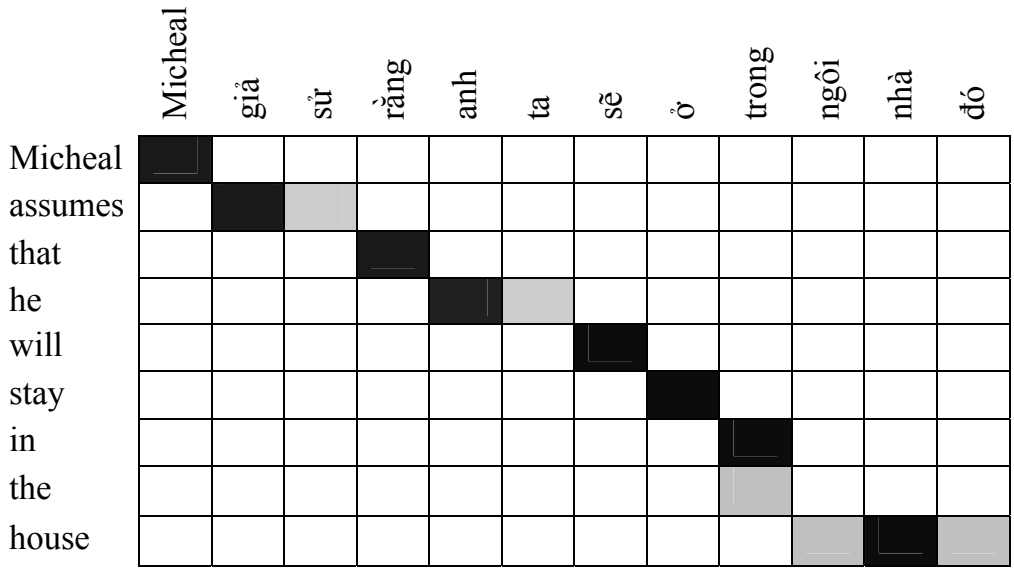

Figure 2-5: Intersection/Union of word alignment

Researchers differ in their methods where to go from here. We describe the details below.

2.1.3 Methods for Learning Phrase Translations

Most of the recently proposed methods use a word alignment to learn a phrase translation table. We discuss three such methods in this section and one exception.

Marcu and Wong

First, the exception: Marcu and Wong (EMNLP, 2002[27]) proposed to establish phrase correspondences directly in a parallel corpus. To learn such correspondences, they introduced a phrase-based joint probability model that simultaneously generates both the source and target sentences in a parallel corpus.

Expectation Maximization learning in Marcu and Wong's framework yields both (i) a joint probability distribution φ(e, f), which reflects the probability that phrases *e* and *f* are translation equivalents; (ii) and a joint

distribution d(i,j), which reflects the probability that a phrase at position i is translated into a phrase at position j.

To use this model in the context of our framework, we simply marginalize the joint probabilities estimated by Marcu and Wong (EMNLP, 2002[27]) to conditional probabilities. Note that this approach is consistent with the approach taken by Marcu and Wong themselves, who use conditional models during decoding.

Och and Ney

Och and Ney (Computational Linguistics, 2003[12]) propose a heuristic approach to refine the alignments obtained from Giza++. At a minimum, all alignment points of the intersection of two alignments are maintained. At a maximum, the points of the union of two alignments are considered. To illustrate this, see the figure below. The intersection points are black, the additional points in the union are shaded grey.

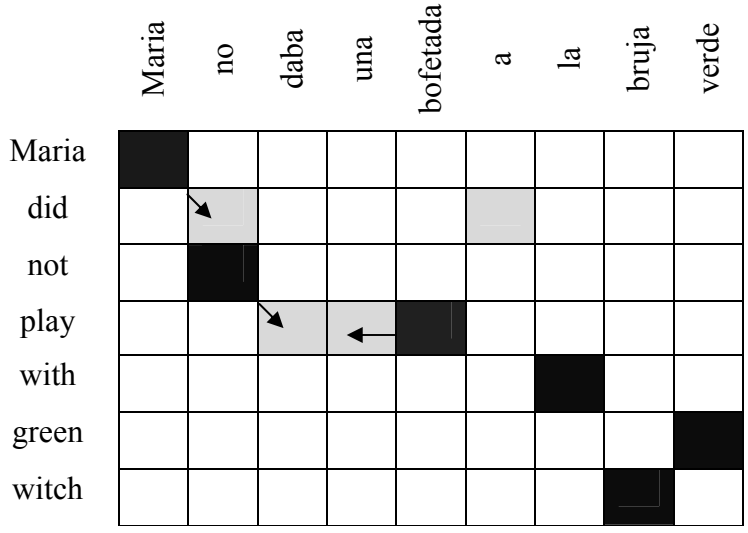

Figure 2-6: Methods for learning phrase translations of Och and Ney

Och and Ney explore the space between intersection and union with expansion heuristics that start with the intersection and add additional alignment points. The decision which points to add may depend on a number of criteria:

- In which alignment does the potential alignment point exist? Foreign-English or English-foreign?

- Does the potential point neighbor already established points?

- Does neighboring mean directly adjacent (block-distance), or also diagonally adjacent?

- Is the English or the foreign word that the potential point connects unaligned so far? Are both unaligned?

- What is the lexical probability for the potential point?

Och and Ney (Computational Linguistics, 2003[12]) are ambiguous in their description about which alignment points are added in their refined method. This method is reimplemented for Moses-chart (the software chosen in this thesis as baseline system) as following:

The heuristic proceeds as follows: We start with intersection of the two word alignments. We only add new alignment points that exist in the union of two word alignments. We also always require that a new alignment point connects at least one previously unaligned word.

Firstly, we expand to only directly adjacent alignment points. We check for potential points starting from the top right corner of the alignment matrix, and alignment points for the first English word, then continue with alignment points for the second English word, and so on.

This is done iteratively until no alignment point can be added anymore. In the final step, we add non-adjacent alignment points, with otherwise the same requirements.

We collect all aligned phrase pairs that are consistent with the word alignment: The words in a legal phrase pair are only aligned to each other, and not to words outside. The set of bilingual phrases BP can be defined formally (Zens, KI 2002[33]) as:

$BP(f_1^J, e_1^J, A) = \{ (f_j^{j+m}, e_i^{i+n}) \}$:

forall (i', j') in A :

j <= j' <= j+m <-> i <= i' <= i+n

The figure below displays all the phrase pairs that are collected according to this definition for the alignment from our running example.

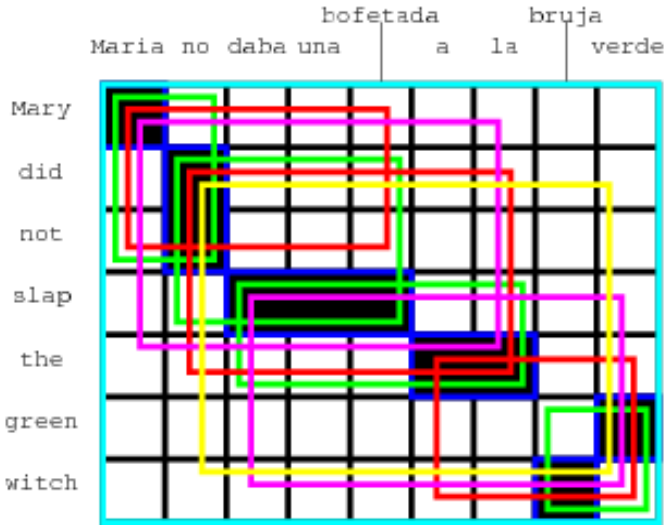

Figure 2-7: An example for learning phrase translations of Och and Ney applied for Moses-chart [31].

(Maria, Mary), (no, did not), (slap, daba una bofetada), (a la, the), (bruja, witch), (verde, green), (Maria no, Mary did not), (no daba una bofetada, did not slap), (daba una bofetada a la, slap the), (bruja verde, green witch), (Maria no daba una bofetada, Mary did not slap), (no daba una bofetada a la, did not slap the), (a la bruja verde, the green witch) (Maria no daba una bofetada a la, Mary did not slap the), (daba una bofetada a la bruja verde, slap the green witch), (no daba una bofetada a la bruja verde, did not slap the green witch), (Maria no daba una bofetada a la bruja verde, Mary did not slap the green witch)

Given the collected phrase pairs, we estimate the phrase translation probability distribution by relative frequency:

$\varphi(f|e) = \text{count}(f,e) \; \Sigma_f \, \text{count}(f,e)$

No smoothing is performed, although lexical weighting addresses the problem of sparse data. More details are explained in the paper on phrase-based translation (Koehn at al, HLT-NAACL 2003[24]).

Tillmann

Tillmann (EMNLP, 2003[4]) proposes a variation of this method. He starts with phrase alignments based on the intersection of the two Giza alignments and uses points of the union to expand these.

Venugopal, Zhang, and Vogel

Venugopal et al. (ACL 2003[1]) allows also for a collection of phrase pairs that are violated by word alignment. They introduce a number of scoring methods which take consistency with the word alignment, lexical translation probabilities, phrase length, etc. into account.

Zhang et al. 2003[43] proposes a phrase alignment method that is based on word alignments and tries to find a unique segmentation of a sentence pair, as it is done by Marcu and Wong directly. This enables them to estimate joint probability distributions, which can be marginalized into conditional probability distributions.

Vogel et al. (2003[35]) review these two methods and shows that the combining phrase tables generated by different methods improves results.

2.1.4 The evaluation of machine translation

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[30]) 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:

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

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:

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

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 in 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,

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

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[10]) 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:

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

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

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

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:

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

The F-measure

The F-measure (Turian et al., 2003[21]) 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:

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

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

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

In the simplest case, the F-measure for a MT translation candidate can be based on unigram precision and recall. See Figure 2-1 for an illustration of this method.

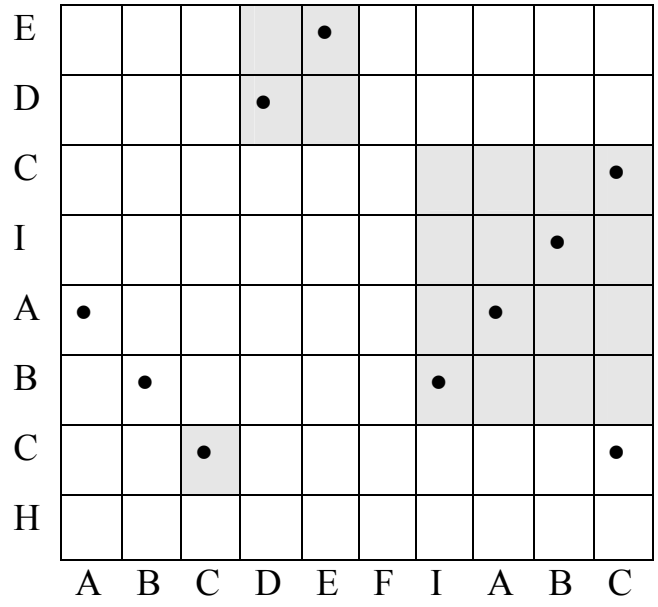

Figure 2-8: Unigram matches, quoted from (Turian et al., 2003[21]).

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. Figure 2-8 shows a maximum matching with dark background.

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

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

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

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 Figure 2-1 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:

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

In the above equation, *e* is the weighing factor which favors 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.

The three automatic methods (Bleu metrics, NIST metric and F-measure) are currently the most commonly used for MT evaluation. In the experiments of this thesis, we applied with the BLEU metric, best known and best adopted machine evaluation for (machine) translation.

2.2 Hierarchical phrase-based model

The hierarchical model (Chiang, 2005[9]; Chiang, 2007[8]) 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 consists of terminal and nonterminal, γ (RHS right-hand side) is the translation of α, ~ defines a one-to-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)

(Because sometime α, γ contains commas (,), so that we use symbol "|||" to separate the source side (left-hand side) and the target side (right-hand side))

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 hierarchical 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[29]) to combine various features. The weight of a derivation D is computed by:

$$w(D) = \prod_i \phi_i(D)^{\lambda_i}$$

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 independent of context information, while the left neighboring *n-1* target words are used for computing n-gram language model.

An example about partial derivation of a synchronous CFG is as follows:

We have a rule:

$$có\ X_1\ với\ X_2 \,|||\, have\ X_2\ with\ X_1$$

Alignment phrases are:

[*Úc*] [*là*] [*một*] [*trong số ít nước*] [*có*] [*quan hệ ngoại giao*] [*với*] [*Triều Tiên*]
[*Australia*] [*is*] [*one of the few countries*] [*that have*] [*diplomatic relations*] [*with*] [*North Korea*]

Some extracted rules are as follows:

$X \rightarrow$ (có X_1 với X_2 ||| have X_2 with X_1) (1)

$X \rightarrow$ (X_1 với X_2 ||| the X_2 that X_1) (2)

$X \rightarrow$ (một trong số X_1 ||| one of X_1) (3)

$X \rightarrow$ (Úc ||| Australia) (4)

$X \rightarrow$ (Triều Tiên ||| North Korea) (5)

$X \rightarrow$ (là ||| is) (6)

$X \rightarrow$(quan hệ ngoại giao ||| diplomatic relations) (7)

$X \rightarrow$(một trong số ít nước ||| one of the few contrries) (8)

$X \rightarrow$(S_1 X_2 ||| S_1 X_2) (9)

$S \rightarrow$(X_1 ||| X_1) (10)

We can get the derivation as:

< S_1 , S_1>

(9) $\rightarrow$ <S_2 X_3 ||| S_2 X_3>

(9) $\rightarrow$ <S_4 X_5 X_3 ||| S_4 X_5 X_3>

(10) $\rightarrow$ <X_6 X_5 X_3 ||| X_6 X_5 X_3>

(4) $\rightarrow$ <Úc X_5 X_3 ||| Australia X_5 X_3>

(6) $\rightarrow$ <Úc là X3 ||| Australia is X_3>

…..

$\rightarrow$ <Úc là một trong số ít nước có quan hệ ngoại giao với Triều Tiên |||

Australia is one of the few countries that have diplomatic relations with North Korea>

2.3 Rule selection for syntax-based statistical machine translation

Rule selection is of great importance to syntax-based SMT systems. Comparing with word selection in word-based SMT and phrase selection in

phrase-based SMT, rule selection is more generic and important. This is because that a rule contains not only terminals (words or phrases), but also noterminals and structural information. Terminals indicate lexical translations, while nonterminal and structural information can capture short or long distance reordering. Consider the following rules for Vietnamese-to-English translation:

(1) $X \rightarrow$ <với X_1 của X_2 ||| X_2 of X_1>

(2) $X \rightarrow$ <với X_1 của X_2 ||| at X_1 's X_2>

(3) $X \rightarrow$ <với X_1 của X_2 ||| with X_2 of X_1>

We can see syntactic structures of the same source-side in different rules in figure 2-9.

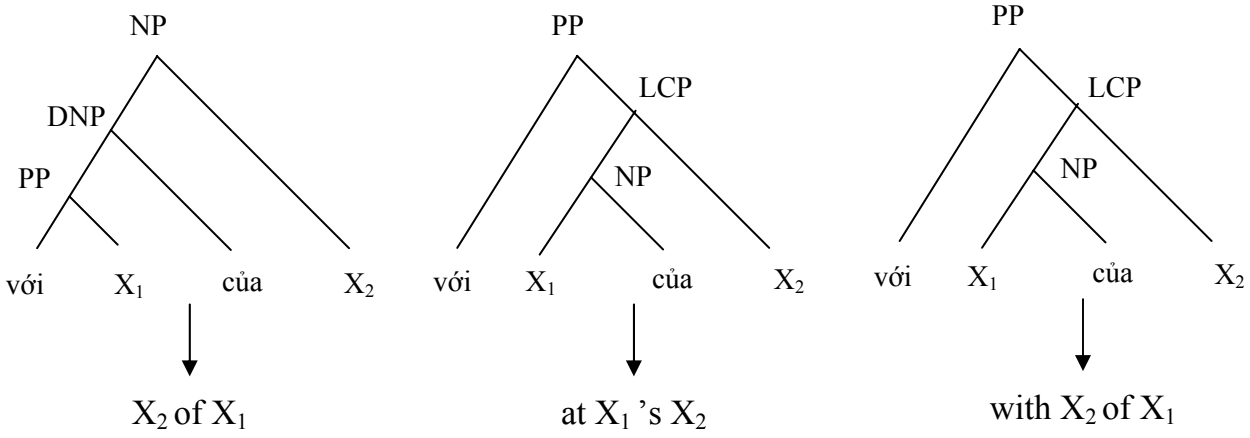

Figure 2-9: Syntactic structures of the same source-side in different rules

These rules have the same source side. However, on the target side, either the translation for terminals or the phrase reoderings for nonterminals are quite different. During decoding, when a rule is selected and applied to a source side, both lexical translation (for terminals) and reorderings (for nonterminals) are determined. Therefore, rule selection affects both lexical translation and phrase reordering.

However, most of current syntax-based systems ignore contextual information when they select rules during decoding, especially the information of terminals and sub-trees covered by nonterminals. For example, the information of X_1 and X_2 is not recorded when rules are extracted from the training. This makes the decoder hardly distinguish the two rules. Intuitively, information of terminals and sub-trees covered by nonterminals as well as contextual information of rules are believed be helpful for rule selection.

Recent researchers showed that rich context information can help SMT systems select rules and improves translation quality (Galley et al. 2004, Huang and Chiang 2008, Liu et al. 2009, Marton and Resnik 2008, Chiang et al. 2009, Shen et al. 2009).

The discriminative phrasal reordering models (Xiong et all., 2006[39]; Zens and Ney, 2006[42]) provided a lexicalized method for phrase reordering. In these model, LHS and RHS can be considered as phrases and reordering types, respectively. Therefore the selection task is to select a reordering type for phrases. They use a MaxEnt model (Zhang, Le. 2004) to combine context features and distinguished two kinds of reordering between two adjacent phrases: monotone or swap. However, our method is more generic, we use the maximum entropy approach to combine rich contextual information around a rule and the information of sub-trees covered by nonterminals in a rule. In our model, the rules with hierarchical structures can handle reorderings of non-adjacent phrases. Furthermore, the rule selection can be considered as a multi-class classification task, while the phrase reordering between two adjacent phrases is a two-class classification task.

Recently, word sense disambiguation (WSD) techniques improved the performance of SMT systems by helping the decoder perform lexical selection. Carpuat and Wu (2007b)[6] integrated a WSD system into a phrase-based SMT

system, Pharaoh (Koehn, 2004a)[25]. Furthermore, they extended WSD to phrase sense disambiguation (PSD) (Carpuat and Wu 2007a)[5]. Either the WSD or PSD system combines rich contextual information to resolve ambiguity problem for words or phrases. Their experiments showed stable improvements of translation quality. These are different from our work. On the one hand, they focus on solving the lexical ambiguity problem, and use a WSD or PSD system to predict translations for phrases which only consist of words. However, we put emphasis on rule selection, and predict translations for hierarchical LHS's which consist of both words and nonterminals. On the other hand, they incorporated a WSD or PSD system into a phrased-based SMT system with a weak distortion model for phrase reordering. While we incorporate MaxEnt RS moels into the state-of-the-art syntax-based SMT system, which can capture phrase reordering by using a hierarchical model.

Chan et al (2007)[7] incorporated a WSD system into the hierarchical SMT system, Hiero (Chiang, 2005)[9], and reported statistically significant improvement. However they only focus on solving ambiguity for terminals of translation rules, and limited the length of terminals up to 2. Different from their work, we consider a translation rule as a whole, which contains both nonterminals and terminals. Moreover, they explored features for the WSD only on the source-side while we define context features for the MaxEnt RS models on both the source-side and target-side and use information of sub-trees covered by nonterminals in a rule.

He et al., (2008)[44] integrating a MERS model (Zhang, Le. 2004) into a formally syntax-based SMT model, the hierarchical phrase-based model (Chiang 2005)[9]. In another work, they incorporate the MERS model into a state-of-the-art linguistically syntax-based SMT model, the tree-to-string alignment template (TAT) model (Liu et al., 2006[40]).

In this work, we incorporate the MERS model into Moses-chart (Hieu et al. 2009)([18],[31]) and use our algorithm to extract features and rules. The basic differences are:

- We re-implement [He et al 2008][44] for rule selection in hierarchical statistical methods on Vietnamese-English translation

- He et al., (2008) used method suggested in (Chiang, 2005) to extract translation rules, and they used the tree-to-string alignment template (TAT) as translation rules while we use Moses-chart (Hieu et al., 2009) to extract translation rules. So that the rules we used are different from their rule as well as their linguistic and contextual information they used.

- They didn't use linguistic and contextual information around nonterminal. They only use linguistic and contextual informatyion for nonterminal while we use linguistic and contextual information for both nonterminal and terminals around nonterminal.

- We applied the nice property of maximum entropy model to combine all features to help rule selection methods better.

- We incorporate the maximum entropy-based rule selection into a state-of-the-art syntax-based SMT model, the Moses-chart (Hieu et al., 2009). This model is developed by many machine translation experts and used in many machine translation systems, while they used the hierarchical phrase-based model (Chiang, 2005) and tree-to-string alignment template (TAT) model.

In the section 2.2, we presented about hierarchical phrase-based model and SCFG rule. An SCFG rule has the form: $X \rightarrow (\alpha, \gamma, \sim)$

Where X is nonterminal, α is an LHS (left-hand side) string consists of terminal and nonterminal, γ (RHS: right-hand side) is the translation of α, $\sim$ defines a one-to-one correspondence between nonterminals in α and γ.

Next, we discuss more detail about translation rules used in our work.

Definitions about translation rules

***Definition 1*:**

Given a word-aligned sentence pair $< f, e, \sim >$, let f_i^j stand for the substring of f from position i to position j inclusive, and similarly for e_i^j. Then a rule $< f_i^j , e_i^j >$ is an initial phrase pair of $< f, e, \sim >$ iff:

1. $f_k \sim e_{k'}$ for some $k \in [i, j]$ and $k' \in [i', j']$;
2. $f_k \nsim e_{k'}$ for all $k \in [i, j]$ and $k' \notin [i', j']$;
3. $f_k \nsim e_{k'}$ for some $k \notin [i, j]$ and $k' \in [i', j']$

In order to obtain rules from phrases, we look for phrases that contain other phrases and replace the subphrases with nonterminal symbols. For example, given the initial phrases shown in Figure 2-11, we could form the rule:

$X \rightarrow <X_1$ năm qua trong X_2 ||| X_2 over the last X_1 years>

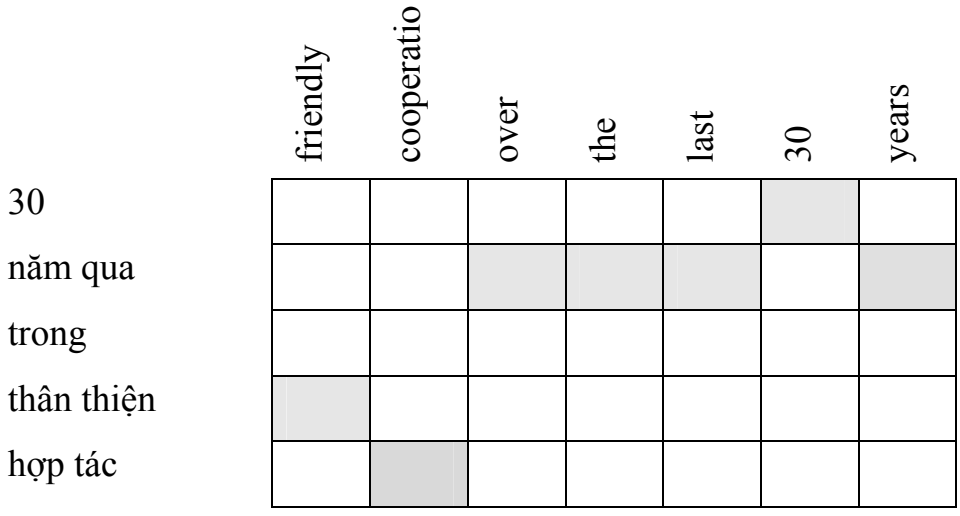

Figure 2-10: Grammar extraction example - Input word alignment.

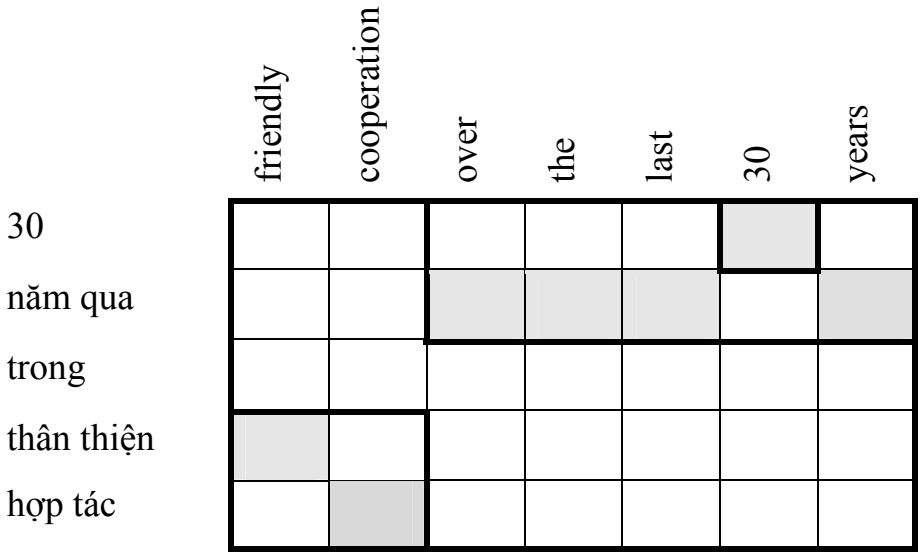

Figure 2-11: Grammar extraction example - Initial phrases.

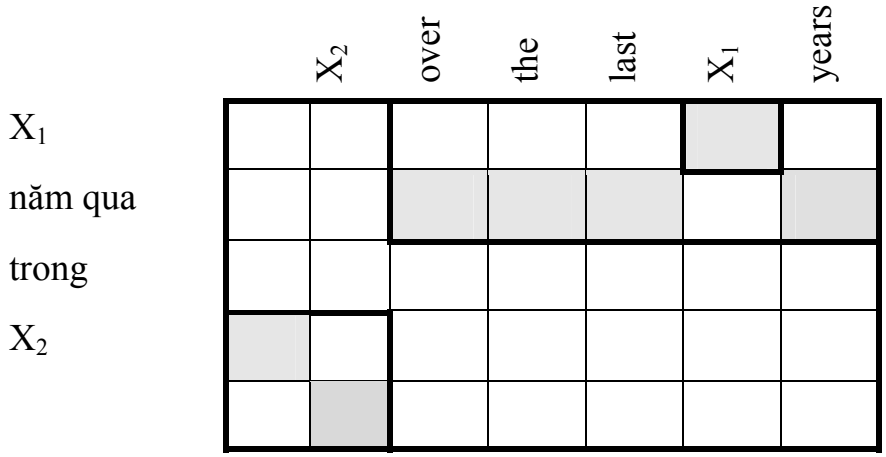

Figure 2-12: Grammar extraction example - Example rule.

Definition 2

The set of rules of $<f, e, \sim>$ is the smallest set satisfying the following:

1. If $< f_i^j , e_i^{j'} >$ is an initial phrase pair, then

$$X \rightarrow <f_i^j, e_i^{j'}>$$ is a rule of $<f, e, \sim>$.

2. If $(X \rightarrow <\gamma, \alpha>)$ is a rule of $<f, e, \sim>$ and $<f_i^j \; e_i^{j'}>$ is an initial phrase pair such that $\gamma = \gamma_1 f_i^j \gamma_2$ and $\alpha = \alpha_1 e_i^{j'} \alpha_2$, then

$$X \rightarrow <\gamma_1 X_k \gamma_2, \alpha_1 X_k \alpha_2>$$

where k is an index not used in γ and α, is a rule of $<f, e, \sim>$

This scheme generates a very large number of rules, which is undesirable not only because it makes training and decoding very slow, but also because it creates spurious ambiguity - a situation where the decoder produces many derivations that are distinct yet have the same model feature vectors and give the same translation. This can result in k-best lists with very few different translations or feature vectors, which is problematic for the minimum-error-rate training algorithm (Och 2003). To avoid this, Moses-chart filter it's grammar according to the following constraints, chosen to balance grammar size and performance on the development set:

1. If there are multiple initial phrase pairs containing the same set of alignments, only the smallest is kept. That is, unaligned words are not allowed at the edges of phrases.

2. Initial phrases are limited to a length of 10 words on either side.

3. Rules are limited to two nonterminals plus terminals on the Vietnamese side.

4. Rules can have at most two nonterminals, which simplifies the decoder implementation. This also makes our grammar weakly equivalent to an inversion transduction grammar (Wu 1997)[38], although the conversion would create a very large number of new nonterminal symbols.

5. It is prohibited for nonterminals to be adjacent on the Vietnamese side, a major cause of spurious ambiguity.

6. A rule must have at least one pair of aligned words, so that translation decisions are always based on some lexical evidence.

Other Rules

Glue rules.

Having extracted rules from the training data, let X be the grammar's start symbol and translate new sentences using only the extracted rules. For

robustness and for continuity with phrase-based translation models, we allow the grammar to divide a Vietnamese sentence into a sequence of chunks and translate one chunk at a time. We formalize this inside a SCFG using rules as follows, which we call the glue rules:

$$S \rightarrow \langle S_1\ X_2 \mid\mid\mid S_1\ X_2 \rangle$$

$$S \rightarrow \langle X_1 \mid\mid\mid\ X_1 \rangle$$

These rules rewrite an S (the start symbol) as a sequence of X_s which are translated without reordering. Note that if we restricted our grammar to comprise only the glue rules and conventional phrase pairs (that is, rules without nonterminal symbols on the right-hand side), the model would reduce to a phrase-based model with monotone translation (no phrase reordering).

Entity rules.

Finally, for each sentence to be translated, we run some specialized translation modules to translate numbers and dates in a sentence, and insert these translations into the grammar as new rules. Such modules are often used by phrase-based systems as well, but here their translations can plug into hierarchical phrases. For example, the rule:

$$X \rightarrow \langle X_1\ \textit{năm qua} \mid\mid\mid \textit{over the last}\ X_1\ \textit{years} \rangle$$

allows to generalize the number of years.

Chapter 3

Rule selection for syntax-based Vietnamese-English statistical machine translation

The syntax-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 called 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 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 syntax-based systems ignore contextual information when they select rules during decoding, especially the information covered by nonterminals. This makes the decoder hardly distinguish rules. Intuitively, information covered by nonterminals as well as contextual information of rules is believed to be helpful for rule selection.

In this work, we propose a maximum entropy-based rule selection model for syntax-based Vietnamese-English 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, the nice properties of maximum entropy model (lexical and

syntactic information for rule selection) are helpful for rule selection methods. Our model allows the decoder to perform context-dependent rule selection during decoding. We incorporate the maximum entropy-based rule selection model into a state-of-the-art syntax-based Vietnamese-English statistical machine translation model. Experiments show that our approach archives significant improvements over the baseline system.

Our works are described as follows:

Firstly, we determine a baseline system to translate lower-cased and tokenized Vietnamese sentences into lower-cased and tokenized English sentences.

Secondly, we extract rules from aligned words of Vietnamese-English parallel corpus.

Thirdly, we extract features from rules, parse trees and tagged sentences of Vietnamese-English parallel corpus.

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

Next, we evaluate and analyze experimental results.

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

The diagram of rule selection for syntax-based Vietnamese-English SMT is shown in Figure 3-1.

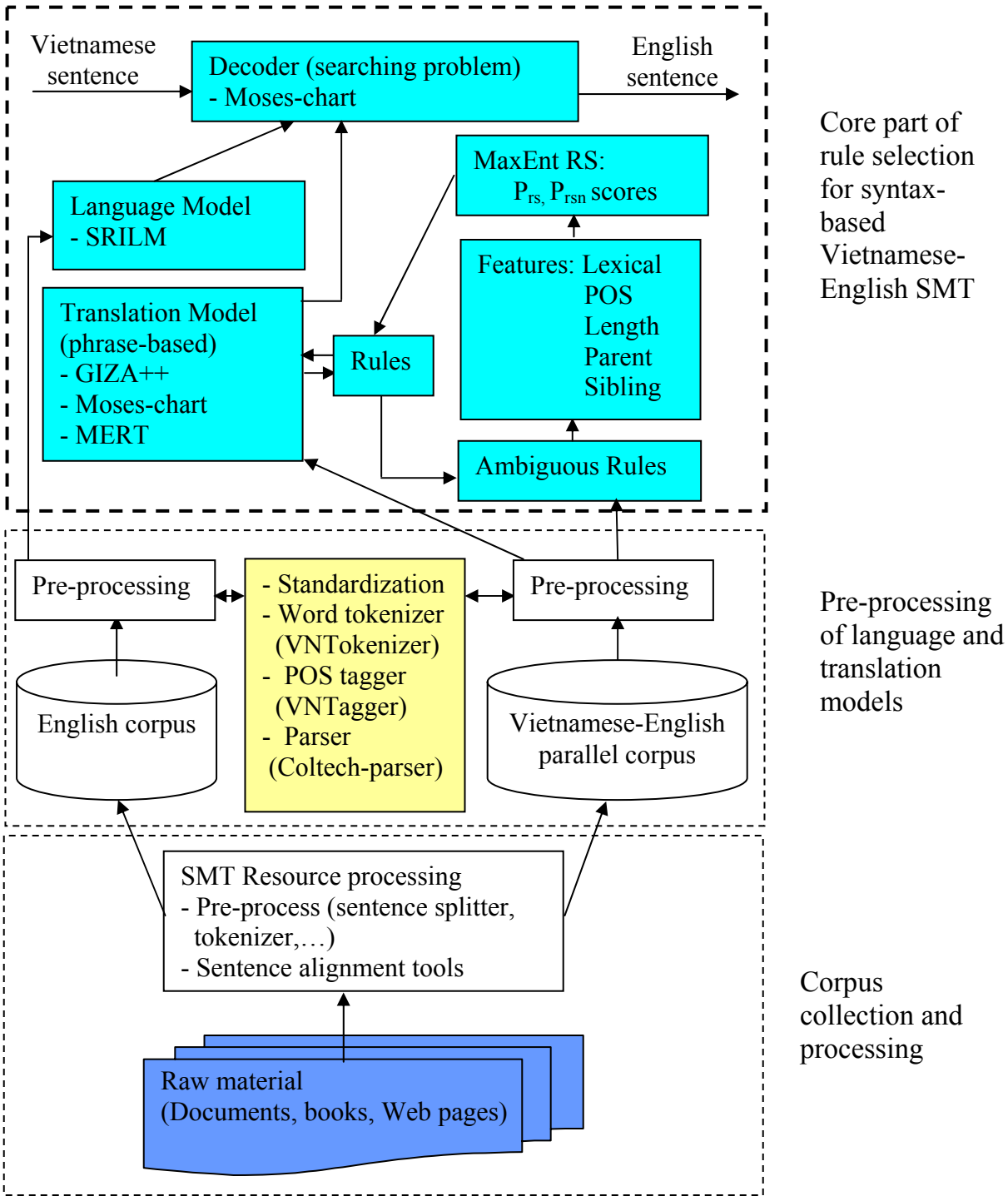

Figure 3-1: Rule selection for syntax-based Vietnamese-English statistical machine translation

This chapter describes about maximum entropy based rule selection model.

In this chapter, we describe some characteristics of Vietnamese, maximum entropy-based rule selection model (MaxEnt RS model) for Vietnamese-English statistical machine translation, features of MaxEnt RS model, the way to extract features and method to integrate the MaxEnt RS model into the translation model (hierarchical phrase-based model).

3.1 Vietnamese language and machine translation in Vietnam

3.1.1 Vietnamese language

Vietnamese is spoken by about 80 millions people around the world, yet very few concrete works on this language have been noticed in natural language processing until now.

To begin with some characteristics of Vietnamese, we remind some important specificities of Vietnamese (T. B. Nguyen et al., 2004[37]).

Vocabulary

Vietnamese has a special unit called "tiếng" that corresponds at the same time to a syllable with respect to phonology, a morpheme with respect to morpho-syntax, and a word with respect to sentence constituent creation. For convenience, we call these "tiếng" syllables. The Vietnamese vocabulary contains:

-Simple words, which are monosyllabic.

-Reduplicated words composed by phonetic reduplication (e.g. trắng/white - trăng trắng/whitish).

-Compound words composed by semantic coordination (e.g. quần/trousers, áo/shirt - quần áo/clothes).

-Compound words composed by semantic subordination (e.g. xe/vehicle, đạp/pedal - xe đạp/bicycle).

-Some compound words whose syllable combination is not recognizable (bồ nông/pelican).

-Complex words phonetically transcribed from foreign languages (cà phê/ coffee).

Grammar

As with other isolating languages, the most important syntactic information source in Vietnamese is word order. The basic word order is subject-verb-object. The other syntactic means are tool words, the reduplication, and the intonation.

Vietnamese belongs to the class of topic-prominent languages (Charles N. Li & Sandra A. Thompson, 1976). In these languages, topics are coded in the surface structure and they tend to control co-referentiality (cf. *Cây đó lá to nên tôi không thích / Tree that leaves big so I not like*, which means *This tree, its leaves are big, so I don't like it*); the topic-oriented "double subject" construction is a basic sentence type (cf. *Tôi tên là Nam, sinh ở Hà Nội / I name be Nam, born in Hanoi*, which means *My name is Nam, I was born in Hanoi*), while a subject-oriented construction as the passive and "dummy" subject sentences are rare (cf. *There is a cat in the garden* should be translated in *Có một con mèo trong vườn / exist one <animal-classifier> cat in garden*).

3.1.2 Machine translation in Vietnam

In Vietnam, with the booming of Internet, the demand for translation from popular foreign languages such as English, French, etc. into Vietnamese sharply increases. Building machine translation systems is the most concerned topic in Vietnamese's natural language processing research circle. There are four main machine translation groups with different approaches:

- National Center for Technology Progress: Rule-based approach to English-Vietnamese MT systems. These are the only MT commercial systems in Vietnam (EVTRAN3.0, VETRAN3.0)

- Univ. of Natural Sciences, VNU HCM: Transfer-based MT using BTL (Bitext Transfer Learning) for English-Vietnamese MT systems. They have experience in building dictionary and bilingual corpus.

- HCM Univ. of Technology, VNU HCM: Since 1989 they have various trails. Statistical approach to Vietnamese-English translation (since 2002) and phrase-based approach to English-Vietnamese translation and phrase extraction from Penn Tree-bank (since 2003)

- The research group at JAIST: Rule-based approach to English-Vietnamese MT system. The system is completed but still not published. Now, they focus on statistical MT, and improve the rule-based MT system using statistical techniques.

Works on machine translation in Vietnam are in top layers but less basic work at lower layers. They lack of common itinerary. Works are done in isolation, not having inheritance. People have to do their work from the scratch without sharing and collaboration.

This research, rule selection for syntax-based Vietnamese English statistical machine translation is a new solution for Vietnamese-English machine translation. We use maximum entropy-based rule selection model for our work. This is helpful to combines local contextual information around rules and information of sub-trees covered by variables in rules. Therefore, the nice properties of maximum entropy model (use of lexical and syntactic features of Vietnamese language for rule selection) are helpful for rule selection methods. This is useful for better phrase reordering as well as better lexical translation, so that our approach archives significant improvements over the baseline system.

3.2 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 all, 1996)[3] 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)[9], 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-to-one correspondence between nonterminals in source and target sides. Let us use $f(X_k)$ to represent the source text covered by X_k and $e(X_k)$ to represent the translation of $f(X_k)$. Let $C(\alpha)$ be the context information of the source text matched by α and $C(\gamma)$ be the context information of target text matched by γ. Under the MaxEnt model, we have:

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

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 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.

3.3 Lexical and syntactic features for rule selection

3.3.1 Lexical features of nonterminals

Each hierarchical rule has nonterminals. Features of nonterminals consist of Lexical features, Parts-of-speech features and Length features:

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

Parts-of-speech (POS) features, POS tags of source words defined in lexical features;

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

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

Table 3-1 Lexical features of nonterminals

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

Rule

X →(, X_1 bị mời ra X_2 ||| , X_1 shown X_2)

X_1 anh_ta đã X_2 khỏi X_1 he was X_2 the

Source Phrase

, anh_ta đã bị mời ra khỏi

, he was shown the

Source sentence

/e	/v	/n	/e	/p	/,	/n	/r	/v	/v	/v	/v	/n
Sau_ khi	lăng_mạ	chủ_ nhà	của	anh_ấy	,	anh_ta	đã	bị	mời	ra	khỏi	cửa
After	having	insulted	his	host	,	he	was	shown	the	door		

Features of this example are shown as Table 3-2

Type	Features
Lexical Features	$W_{\alpha-1}$ = anh_ấy $W_{\alpha+1}$ = cửa
	$WL_{f(X_1)}$ = anh_ta $WR_{f(X_1)}$ = đã $WL_{f(X_2)}$ = khỏi $WR_{f(X2)}$ = khỏi
	$WL_{e(X_1)}$ = he $WR_{e(X_1)}$ = was $WL_{e(X_2)}$ = the $WR_{e(X_2)}$ = the
POS Features	$P_{\alpha-1}$ = p $P_{\alpha+1}$ = n
	$PL_{f(X_1)}$ = n $PR_{f(X_1)}$ = r $PL_{f(X_2)}$ = v $PR_{f(X_2)}$ = v
Length Features	$LEN_{f(X_1)}$ = 2 $LEN_{f(X_2)}$ = 1 $LEN_{e(X_1)}$ = 2 $LEN_{e(X_2)}$ = 1

Table 3- 2: Lexical features of nonterminals of the example

3.3.2 Lexical features around terminals

Lexical features around terminals have the same meaning with features of nonterminals

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

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

Side	Type	Name	Description
Source-side	Lexical feature	$W_{f(X_k)-1}$	The first word adjacent preceding the left of $f(X_k)$
		$W_{f(X_k)+1}$	The first word adjacent following the right of $f(X_k)$
	POS Features	$P_{f(X_k)-1}$	POS of $W_{f(X_k)-1}$
		$P_{f(X_k)+1}$	POS of $W_{f(X_k)+1}$
Target-side	Lexical features	$W_{e(X_k)-1}$	The first word of $e(X_k)$-*1*
		$W_{e(X_k)+1}$	The last word of $e(X_k)$+*1*

Table 3-3 Lexical features around terminals

Example: with a rule

$X \rightarrow$(*anh_ta* X_1 *bị* X_2 *lần đầu_tiên* ||| *he* X_1 *was* X_2 *on the first time*)

We have lexical features around terminals shown as Table 3-4

	Type	Features
Source-side	Lexical feature	$W_{f(X_1)-1}$ = anh_ta, $W_{f(X_2)-1}$ = bị
		$W_{f(X_1)+1}$ = bị , $W_{f(X_2)+1}$ = *lần*
	POS Features	$P_{f(X_1)-1}$ = n, $P_{f(X_2)-1}$ = v
		$P_{f(X_1)+1}$ = v, $P_{f(X_2)+1}$ = r
Target-side	Lexical features	$W_{e(X_1)-1}$ = he, $W_{e(X_2)-1}$ = was
		$W_{e(X_1)+1}$ = was, $W_{e(X_2)+1}$ = on

Table 3-4: Lexical features around terminals of the example

3.3.3 Syntactic features

Let $R \rightarrow <\alpha, \gamma, \sim>$ is a translation rule and $f(\alpha)$ is a 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 a 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.

In Figure 3.2 shows that the Parent feature of a subtree covering X_1 is *NP*.

Sibling feature (SBF)

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

In Figure 3.3 shows that the Sibling feature of a subtree covering X_1 is *N* .

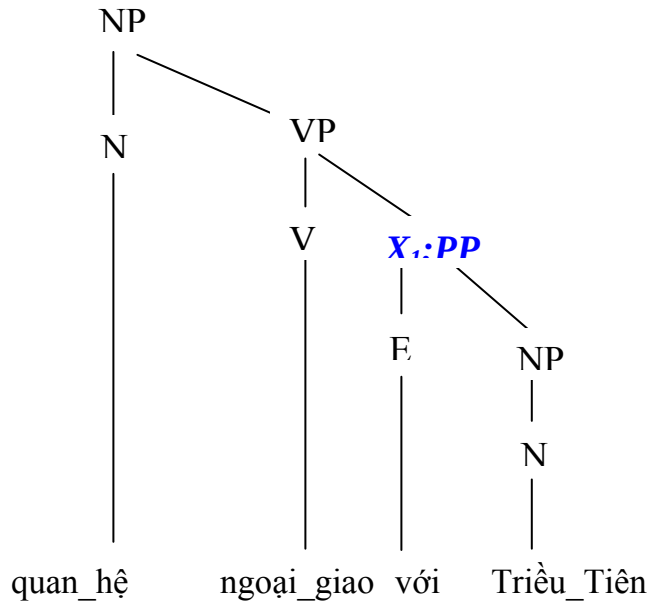

Figure 3-2: Sub-tree covers nonterminal X_1:*PP*

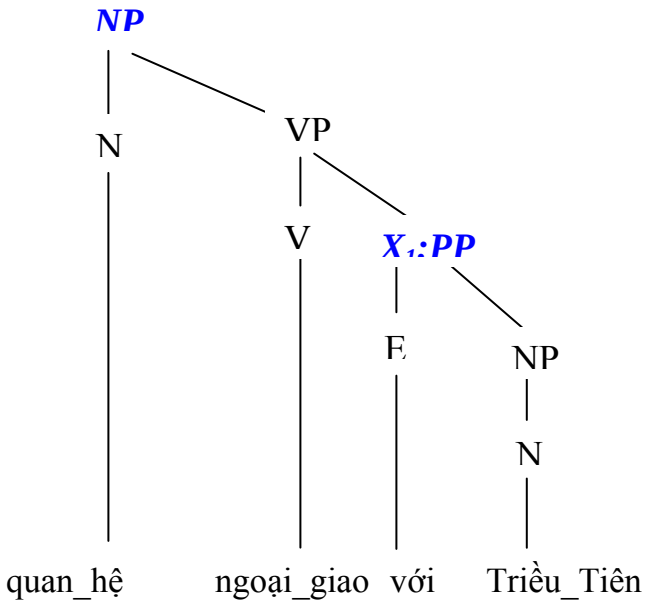

Figure 3-3: NP - Parent feature of a sub-tree covers nonterminal X_1:*PP*

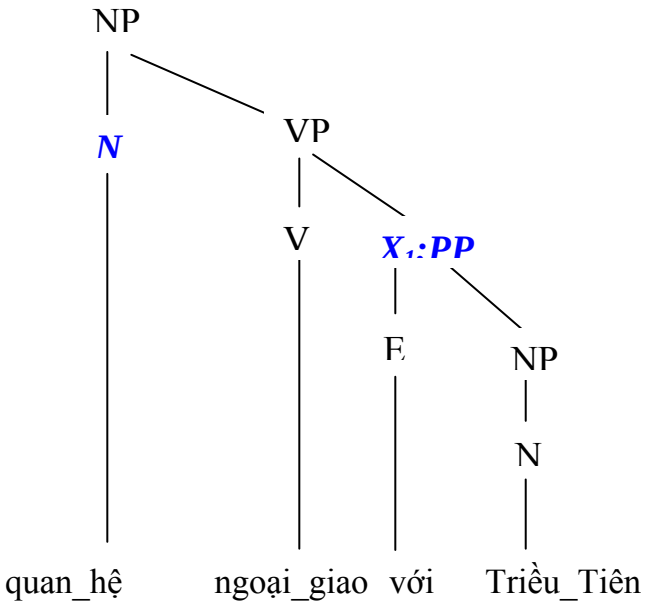

Figure 3-4: N - Sibling feature of a sub-tree covers nonterminal X_1:*PP*

Those features: Lexical feature, Parts-of-speech features, Length features, Parent features and Sibling features make use of rich 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)[9]. We use Moses-chart to extract phrases

and rules. Using the Toolkit of Le Hong Phuong [16] to tag, tokenize Vietnamese source sentence, Coltech-parser of Hanoi University of Engineering and Technology to parse Vietnamese a source sentence, after that we use following algorithm to extract features:

R={R_i}={Hirerarchical rules}, P={P_j,P'_j}={Vietnamese-English phrase alignment}, S={S_l,E_l}={sentence pair}, S'={S'_l}={tagged Vietnamese sentence}, S''= {S''_l}={parsed Vietnamese sentence}

Input: Hirerarchical rules, Vietname-English phrase alignment, sentence pair, Tagged Vietnamese sentence, Parsed Vietnamese sentence

Output: Features of nonterminals; Features around terminals and Syntactic features

for $i \in \{1..\ n\}$ **do**
  X_k = *Nonterminal of LHS of* R_i
  X'_k = *Nonterminal of RHS of* R_i
  Y = *LHS of* R_i
  Z= *RHS of* R_i
  for $j \in \{1..\ m\}$ **do**
    if $Y \in P_j, Z \in P'_j$ **then**
      X_k = *phrase*
      X'_k = *phrase'*
      for $l \in \{1..\ v\}$ **do**
        features of nonterminal:
          Lexical features
            $W_{\alpha-1}$ = *word adjacent to the left of* P_j *in* S_l
            $W_{\alpha+1}$= *word adjacent to the right of* P_j *in* S_l
            $WL_{f(X_k)}$ = *word to the left of* P_j
            $WR_{f(X_k)}$ = *word to the right of* P_j
            $WL_{e(X_k)}$= *word to the left of* P'_j
            $WR_{e(X_k)}$= *word to the left of* P'_j
          Parts-of-speech features
            $P_{\alpha-1}$= *POS of* $W_{\alpha-1}$ *in* S'_l
            $P_{\alpha+1}$= *POS of* $W_{\alpha+1}$ *in* S'_l
            $PL_{f(X_k)}$ = *POS of* $WL_{f(X_k)}$ *in* S'_l
            $PR_{f(X_k)}$ = *POS of* $WR_{f(X_k)}$ *in* S'_l
          Lenght features
            $LEN_{f(X_k)}$ = *Length of source subphrase* X_k

$LEN_{e(X_k)}$ = *Length of target subphrase* X'_k

8 *features around terminal:*

Lexical features

$W_{f(X_k)-1}$ = *word adjacent to the left of* X_k *in* P_j

$W_{f(X_k)+1}$ = *word adjacent to the right of* X_k *in* P_j

$W_{e(X_k)-1}$ = *word adjacent to the left of* X'_k *in* P'_j

$W_{e(X_k)+1}$ = *word adjacent to the right of* X'_k *in* P'_j

Parts-of-speech features

$P_{f(X_k)-1}$ = *POS of* $W_{f(X_k)-1}$ *in* S'_l

$P_{f(X_k)+1}$ = *POS of* $W_{f(X_k)+1}$ *in* S'_l

9 *Syntax features: (*$T(X_k)$ *is sub-tree covering* X_k*. in parsed* S''_l*)*

Parent features = parent of $T(X_k)$

Sibling features = sibling features of root of $T(X_k)$

10 **enfor**

11 **enif**

12 **endfor**

13 **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 Yoshimasa Tsuruoka, Tsujii laboratory, Department of Computer Science, University of Tokyo (2006)[19] to train a MaxEnt RS model for each ambiguous hierarchical LHS.

The flowchart of algorithm to extract features shows in figure 3-5

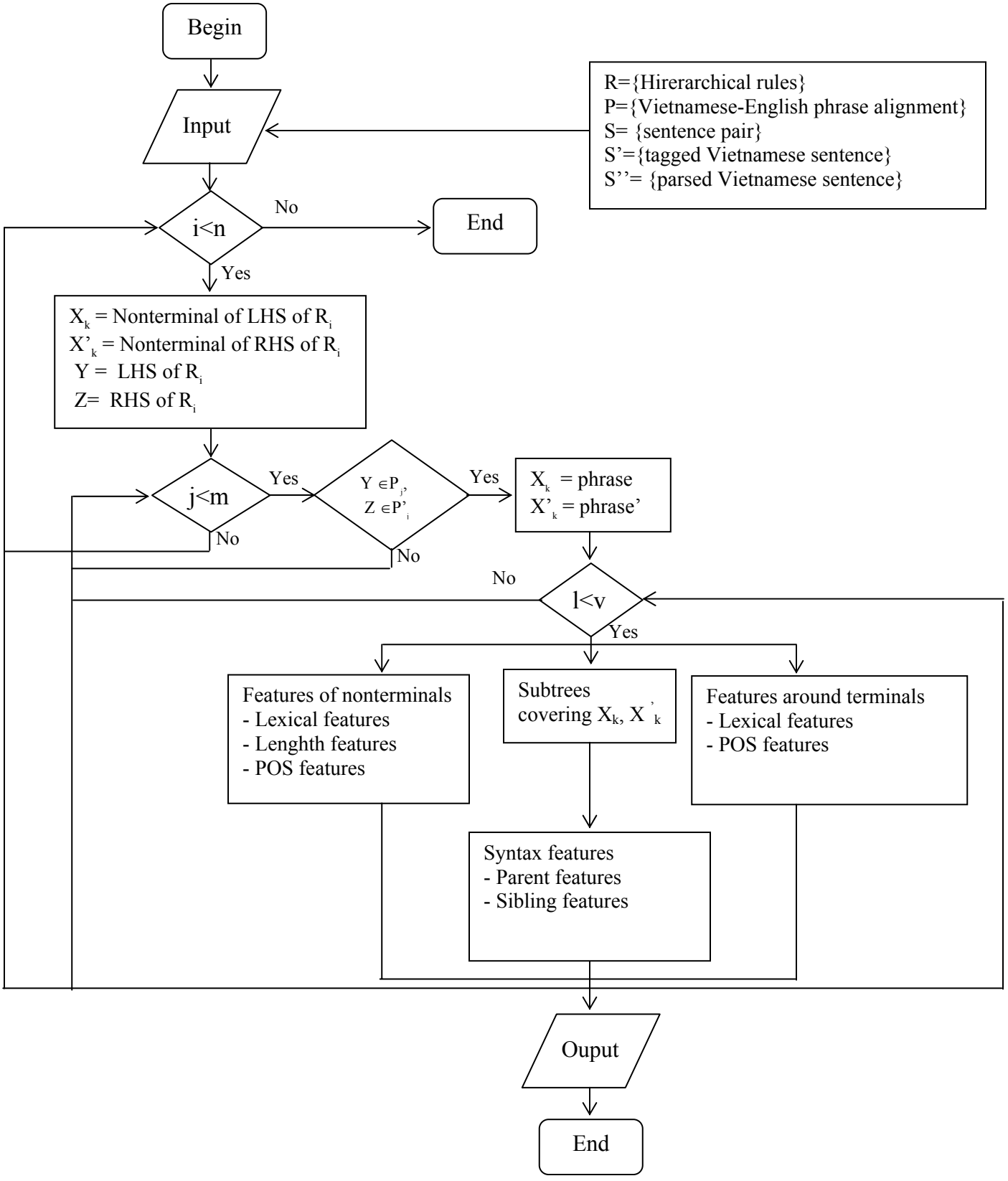

Figure 3-5: The flowchart of algorithm to extract features

3.4 Integrating MaxEnt RS model into the translation model

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

In (Chiang, 2005)[9], 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) $P_{rs}(\gamma \mid \alpha, f(X_k), e(X_k))$.

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

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

This feature is similar to the 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.

Chiang (2007)[8] uses 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)[9] 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 F_1^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 f_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 f_i^j, suppose the rule

$$X \rightarrow (f_i^k X_1 f_t^j \,|||\, e_{i'}^{k'} X_1 e_{t'}^{j'})$$

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 F_1^J is the item with the lowest cost in cell $c[I,J]$.

Chapter 4
The detail of experiments

We applied the above theory to Vietnamese-English SMT 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 the experiments. The output and research questions are discussed in chapter 5.

The system for the experiments is built upon existing pieces of software. The engineering work includes choosing and compiling of the software systems and libraries, selecting and formatting of corpora, code analysis in accordance with the theory of the last two chapters, software development work to combine and coordinate different software systems, and 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.

4.1 Software

4.1.1 Baseline

Moses[31], 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.

• Beam-search: an efficient search algorithm finds quickly the highest probability translation among the exponential number of choices

• Phrase-based: the state-of-the-art method in statistical machine translation allows translation of short text chunks.

• Factored: words may have factored representation (surface forms, lemma, part-of-speech, morphology, word classes...).

The decoder was mainly developed by Hieu Hoang and Philipp Koehn at the University of Edinburgh and extended during a Johns Hopkins University Summer Workshop and further developed under EuroMa-trix and GALE project funding. The Moses decoder was supported by the European Framework 6 projects EuroMatrix, TC-Star, the European Framework 7 project EuroMatrixPlus, and the DARPA GALE project, as well as several universities such as the University of Edinburgh, the University of Maryland, ITC-irst, Massachusetts Institute of Technology, and others.

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

Traditional phrase-based models have as an atomic translation step the mapping from an input phrase to an output phrase. Tree-based models operate on so-called grammar rules, which include variables in the mapping rules.

X_1 *không phải* $\rightarrow$ *not* X_1 (Vietnamese-English)

ate X_1 $\rightarrow$ *habe* X_1 *gegessen* (English-German)

X_1 *of the* X_2 $\rightarrow$ *le* X_2 X_1 (English-French)

The variables in these grammar rules are called non-terminals, since their occurrence indicates that the process has not yet terminated to produce the final words (terminals). Besides a generic non-terminal *X*, linguistically motivated non-terminals such as *NP* (noun phrase) or *VP* (verb phrase) may be used as well in a grammar (or translation rule set).

Phrase-based decoding generates a sentence from left to right, by adding phrases to the end of a partial translation. Tree-based decoding builds a chart, which consists of partial translation for all possible spans over the input sentence. 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 lower-cased and tokenized source sentences into lower-cased 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.

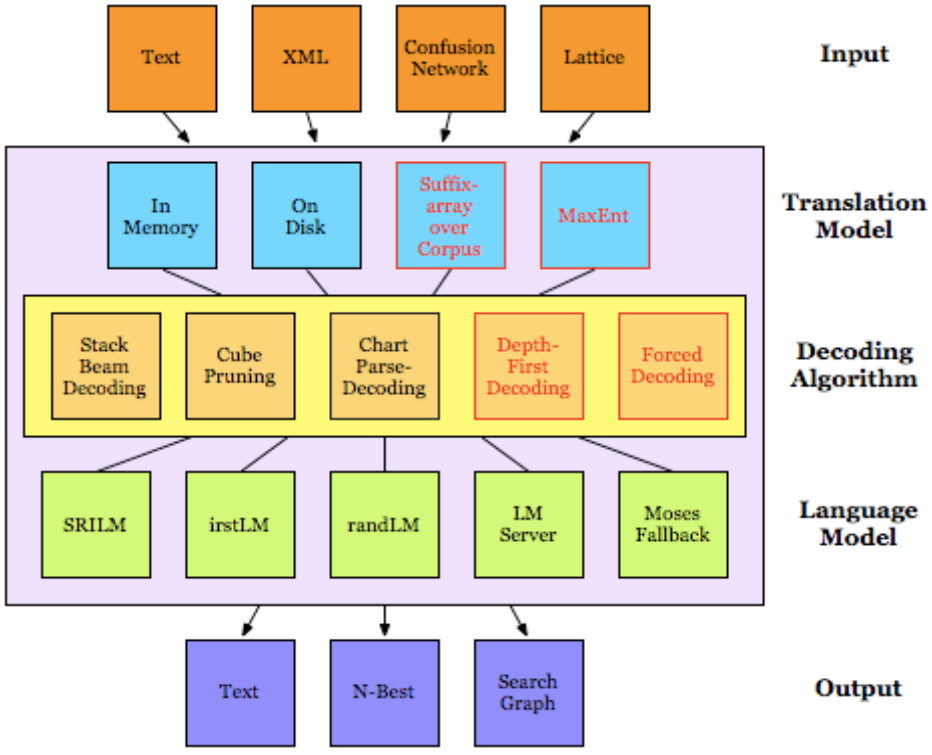

Figure 4-1: The model of Moses-chart [18]

4.1.2 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++ software [15] is written in C++.

4.1.3 SRILM

SRILM[17] 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.

Fuliang Weng wrote the initial version of the lattice rescoring tool in SRILM; Dimitra Vergyri developed the score combination optimizer based on simplex search; Anand Venkataraman contributed N-best decoding and other enhancements to the statistical tagging tools. Development of SRILM has benefited greatly from its use and constructive criticism by many colleagues at SRI, the Johns Hopkins summer workshops, and the larger research community.

4.1.4 Tokenizer

VNTokenizer[16] is an automatic tokenizer for tokenization of Vietnamese texts developed by Le Hong Phuong (INRIA Lorraine, the French National Institute for Computer Science and Automation.)[16].

The program is developed in the Java programming language and is platform-independent, the program gives very good result precision and recall ratios are in the range of 96%-98%.

Example:

bạn không thể ngăn cản anh ấy tiêu tiền của chính anh ấy

The result:

bạn không_thể ngăn_cản anh_ấy tiêu_tiền của chính anh_ấy

4.1.5 Tagging

VNTagger[16] is also developed by Le Hong Phuong. This is an automatic tagger for tagging Vietnamese texts with high accuracy around 95%. The software is written in Java programming language and is platform-independent. The development of the software has been greatly facilitated thanks to open source implementation of a maximum entropy part-of-speech tagger of the Stanford Natural Language Processing Group. This software implements a maximum entropy classifier which uses a conjugate gradient procedure and a Gaussian prior to maximize the data likelihood (Toutanova et al., 2003). It's also freely distributed under the GNU/GPL license and available online.

Example:

vợ của anh ấy vẫn nhận được tin tức của anh ấy thường xuyên

The result:

vợ/N của/E anh_ấy/P vẫn/R nhận/V được/R tin_tức/N của/E anh_ấy/P thường_xuyên/A

4.1.6 Parser

Coltech-parser is developed by Hanoi University of Engineering and Technology. Coltech-parser is a mono-lingual parser that supports different types of statistical parsing models. It is used in the experiments to produce Vietnamese mono-lingual grammars. This parser is implemented in Java.

The Coltech-parser requires tokenization of information for each sentence in the input sentence. Therefore, before using the parser, Tokenization (VNTokenizer - Le Hong Phuong, 2009[16]) is required, which tokenizes input words.

Example:

(anh_ấy uống cà_phê vừa_xong thì anh_ấy bắt_đầu thấy ngái_ngủ)

The result:

((S (S[-H] (NP[-H] (NP[-H] (P[-H] anh_ấy))) (VP (V[-H] uống) (NP (NP[-H] (N[-H] cà_phê) (N vừa_xong))))) (C thì) (S (NP[-H] (NP[-H] (P[-H] anh_ấy))) (VP (V[-H] bắt_đầu) (VP (V[-H] thấy) (AP (A[-H] ngái_ngủ)))))))

4.1.7 Maximum entropy classification

We chose maximum entropy classification toolkit developed by Yoshimasa Tsuruoka, Tsujii laboratory, Department of Computer Science, University of Tokyo (2006)[19]. It's also freely distributed under the GNU/GPL license and available online .

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)[2], smoothing with Gaussian priors (Chen and Rosenfeld, 1999)[36], modeling with inequality constraints (Kazama and Tsujii, 2003[23]), 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.*

4.2 Corpus

We carry out experiment on the corpus includes 16,397 Vietnamese-English sentence pairs which were collected from some grammar books (named "Conversation") (Nguyen et al., 2007[32]) with 853k Vietnamese and 637k English words as our training. The English part is used to train a trigram language model. We use the corpus with 672 sentence pairs as the test set.

Name	Type	Vietnamese	English
Train corpus	Total	16,397	16,397
	Minimum	1	1
	Maximum	40	40
	Average	7.72	8.92
Test corpus	Total	672	672
	Minimum	1	1
	Maximum	31	28
	Average	7.53	8.28

Table 4-1: Statistical table of train and test corpus

4.3 Training

To train the translation model, we first run GIZA++ (Och and Ney, 2000[28]) to obtain word alignment in both translation directions. Then we use Moses-chart to extract SCFG grammar rules. We use toolkits of Le Hong Phuong to tag, pos and parse Vietnamese source sentence. Meanwhile, we gather lexical and syntactic features for training the MaxEnt RS models. The maximum initial phrase length is set to 10 and the maximum rule lengh of the source side is set to 5.

We use SRI Language modeling toolkit (Stocke, 2002[17]) to train language models for both tasks. We use minimum error rate training (Och, 2003[12]) 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[30]), as calculated by mteval-v11b.pl with case-insensitive matching of n-grams, where n=4.

4.4 Baseline + MaxentRS

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

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

This feature is computed by the MaxEnt RS model, which gives a probability that the model selects 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 baseline system.

After using Moses-chart to extract rules, we have rule-table (table which contains rules) and moses.ini (a file which consists of variables for decoding).

Each hierarchical rule in rule-table has a form as following:

, [*X*][*X*] *trước* [*X*] ||| *,* [*X*][*X*] *before* [*X*] ||| *1-1* ||| *0.0975805 0.135281 0.0584394 0.108758 2.718* ||| *0.539366 0.900618*

In this form

[*X*][*X*]: Nonterminal;

, [X][X] trước [X]: Left-hand side

, [X][X] before [X]: Right-hand side

0.0975805 0.135281 0.0584394 0.108758 2.71: scores of rule

We insert two scores into the rule, change variables of moses.ini file and adjust codes of Moses-chart, we get the result in Table 4.2

System	Vietnamese-English corpus
Moses	24.60
Moses-chart	27.03
+ MaxEnt RS	
Lexical features of nonterminal (Lex+POS+Len)	27.69
Lexical features around nonterminal (Pos+Lex)	27.17
Syntax features (Parent and sibling)	27.67
Lexical features of nonterminal + Syntax features	27.78
All features	28.02

Table 4-2: BLEU-4 scores (case-insensitive) on Vietnamese-English corpus.

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

Using all features to train the MaxEnt RS models, the BLEU-4 score is 28.02, with an absolute improvement 0.99 over the baseline.

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 nonterminals and syntactic features are the most useful features since they can generalize over all training examples. Moreover, Lexical features around terminals also yields improvement. However, these features are never used in the baseline.

Chapter 5

The results and conclusions

5.1 The result and discussion

Using Moses-chart to extract rule, with 16,397 Vietnamese-English sentences in the training sets, we extracted 1,090,670 rules. There are 887,487 rules contains nonterminals and 203,189 rules does not contain nonterminals. The number of glue grammar rules are 3 and the number of rules which match in the test are 9,051 rules. The result is shown in Table 5-1.

Name	Number
The number of rules	1,090,670
The number of rules contain nonterminals	887,478
The number of rules don't contain nonterminals	203,189
The number of glue grammar rules	3
The number of rules which match in the test	9,051

Table 5-1: Statistical table of rules

When we use Baseline and Beaseline + MaxEnt RS (all features), the number of hierarchical rules and ambiguous hierarchical rules change. The result shows in Table 5-2.

	Rule	Number of Hierarchical-LHS	Number of Ambiguous hierarchical-LHS
Baseline	9,051	5,261	2,155
+MaxEnt RS (All features)	9,051	6,021	3,953

Table 5-2: Number of possible source-sides of SCFG rule for Vietnamese-English corpus and number of source-sides of the best translation.

Table 5-2 shows the number of source-sides of SCFG rules for Vietnamese-English corpus. After extracting grammar rules from the training corpus, there are 9,051 source-sides which match the test corpus, they are hierarchical LHS's (H-LHS, the LHS which contains nonterminals). For the hierarchical LHS's, 40.96% 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 65.65%, 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 baseline and baseline+MaxEnt RS. 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. For example, the source sentence:

"... năm thành viên thường trực của hội đồng bảo an Liên hợp quốc ... "

is translated as follows:

Reference:

... the five permanent members of the UN Security Council ...

Baseline:

... the [United Nations Security Council]$_1$ [five permanent members]$_2$ ...

+MaxEnt RS:

...[the five permanent members]$_2$ of [the UN Security Council]$_1$ ...

The source sentence is translated incorrectly by the baseline system, which selects the rule

$X \rightarrow \langle X_1$ của $X_2 \mid\mid\mid$ the X_1 $X_2 \rangle$

and produces a monotone translation. In contrast, by considering information of the subphrases X_1 and X_2 , the MaxEnt RS model chooses the rule

$X \rightarrow \langle X_1$ của $X_2 \mid\mid\mid X_2$ of $X_1 \rangle$

and obtains a correct translation by swapping X_1 and X_2 on the target-side.

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. For example, the translations of the source sentence:

"Tôi ngại rằng chuyến bay này đầy"

are as follows:

Reference:

I'm afraid this flight is full.

Baseline:

I'm afraid already booked for this flight.

+MaxEnt RS:

I'm afraid this flight is full.

Here, the baseline translates the Vietnamese phrase

này đầy ” into “booked” by using the rule:

X→<X_1 này đầy ||| X_1 booked>

The meaning is not fully expressed since the Vietnamese word “này” is not translated. However, the MaxEnt RS model obtains a correct translation by using the rule:

X→<X_1 này đầy ||| X_1 full>

However, we also find that some results produced by the MaxEnt RS models seem to decrease the BLEU score. An interesting example is the translation of the source sentence:

“Tên của con đường này là gì”:

Reference1:

What is the name of this street?

Reference2:

What is this street called?

Baseline:

What is the name of this street?

+MaxEnt RS:

What’s this street called?

In fact, both translations are correct. But the translation of the baseline fully matches *Reference1*. Although the translation produced by the MaxEnt RS model is almost the same as *Reference2*, as the BLEU metric is based on n-gram matching, the translation “What’s” cannot match “What is” in *Reference2*. Therefore, the MaxEnt RS model achieves a lower BLEU score.

anh_ấy nỗ_lực đứng_dậy	Moses-chart	he attempt to stood up
	Moses-chart + features	he made an effort to stand up.
	Reference	he attempts to stand up
Chúng_tôi đến vào buổi_sáng của ngày thứ_sáu	Moses-chart	We come into the morning of day Friday
	Moses-chart + features	We arrived in the morning of the sixth
	Reference	We arrived on the morning of Friday
như_vậy sẽ là 5 đêm , thưa ông .	Moses-chart	so will is five night , sir .
	Moses-chart + features	such will be the fifth night, sir
	Reference	such will be the fifth night, sir
anh_ấy uống cà_phê vừa_xong thì anh_ấy bắt_đầu thấy ngái_ngủ	Moses-chart	he has just finished their coffee began to see him sleeping
	Moses-chart + features	he drinks coffee than when he began to feel drowsy
	Reference	no sooner had he drunk the coffee than he began to feel drowsy

Table 5-3:

Some output sentences of Moses-chart, Moses-chart + features and Reference

5.2 Summary

Like human translation, machine translation has two essential factors – unit element (unbreakable word or phrase that carries meaning) translation and target sentence organisation. The simplest models for SMT are word-based, where the unit elements are words and sentence organisation 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. Hierarchical phrase based and tree-based models further improved the target sentence organisation. 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 syntax-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 work, we propose a generic lexical and syntactic 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 integrate the MaxEnt RS models into the hierarchical SMT model by adding two new features. Experiments show that the lexical and syntactic approach for rule selection achieves statistically significant improvements over the state-of-the-art syntax-based SMT system.

5.3 Future work

Our approach can be used for the formally syntax-based statistical machine translation systems and also can be applied to the linguistically syntax-based statistical machine translation systems. For future work, we will explore more sophisticated features for the Maximum entropy-based rule selection models and test the performance of the Maximum entropy-based rule selection model on large scale corpus.

Each of languages has different structure and vocabulary, but also uses the same method. We want apply this model for another language and compare the result to find a good way as well as a good result.

Clearly, phrase-based systems are very good at predicting content words, but are less accurate in producing function words, or producing output that correctly encodes grammatical relations between content words. Syntax-based can help to solve this problem, so that syntax-based is good way to approach for statistical machine translation.

We see that using deep syntactic structures yielded by an HPSG parser can improve syntax-based translation. We want to study and evaluation of other method to improve syntax-based statistical machine translation such as: Maximum entropy-based rule selection (in this thesis), Head-driven phrase structure grammar (HPSG), Deep syntactic structures and other existing approaches to construct a sufficient model for syntax-based statistical machine translation.

We will evaluation of the model obtained, applying the model to different languages (English-Japanese, English-Vietnamese,..) in different size of corpus, improving of our model to achieve a sufficient model for syntax-based statistical machine translation, which:

- Integrates linguistic information in our model
- Uses deep syntactic structures in our model.
- Integrates into several languages
- Promises result

Reference

1. Ashish Venugopal, Stephan Vogel, Alex Waibel, 2003 "Effective Phrase Extraction from Alignment Models", In the Proceedings of ACL 2003, Sapporo, Japan
2. Benson and Moré, 2001 Benson, S.J., Moré, J., 2001. A limited memory variable-metric algorithm for bound-constrained minimization. Technical Report ANL/MSC-P909-O901, Mathematics and Computer Science Division, Argonne National Laboratory
3. Berger, A. L., S. A. Della Pietra, and V. J. Della. 1996. A maximum entropy approach to natural language processing. Computational Linguistics, page 22(1):39-72.
4. C.Tillmann, F. Xia: A Phrase-based Unigram Model for Statistical Machine Translation. Proc. Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics Annual Meeting (HLT-NAACL), Companion Volume: Short Papers, pp. 106–108, Edmonton, Canada, May/June 200
5. Carpuat, Marine and Dekai Wu. 2007a. How phrase sense disambiguation outperforms word sense disambiguation for statistical machine translation. In 11th Conference on Theoretical and Methodological Issues in Machine Translation, pages 43–52.
6. Carpuat, Marine and Dekai Wu. 2007b. Improving statistical machine translation using word sense disambiguation. In Proceedings of EMNLP-CoNLL 2007, pages 61–72.
7. Chan, Yee Seng, Hwee Tou Ng, and David Chiang. 2007. Word sense disambiguation improves statistical machine translation. In Proceedings of the

45th Annual Meeting of the Association for Computational Linguistics, pages 33–40.

8. David Chiang. 2007. Hierarchical phrase-based translation. Computational Linguistics, pages 33(2):201–228.
9. David Chiang. 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.
10. Doddington, G. (2002). Automatic Evaluation of Machine Translation Quality using N-gram Cooccurrence Statistics. Proceedings of the Second Human Language Technologies Conference (HLT) (pp.138-145). Morgan Kaufmann. San Diego, USA.
11. F.J. Och, H. Ney: The Alignment Template Approach to Statistical Machine Translation. Computational Linguistics, Vol. 30, No. 4, pp. 417–449, December 2004.
12. Franz Josef Och, Hermann Ney. "A Systematic Comparison of Various Statistical Alignment Models", Computational Linguistics, volume 29, number 1, pp. 19-51 March 2003.
13. Galley, Michel, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, and Ignacio Thayer. 2006. Scalable inference and training of context-rich syntactic translation models. In Proceedings of COLING-ACL 2006, pages 961–968.
14. H. Ney: Stochastic Modelling: from Pattern Classification to Language Translation. Proc. 39th Annual Meeting of the Assoc. for Computational Linguistics (ACL): Workshop on Data-Driven Machine Translation, pp. 1–5, Morristown, NJ, July 2001.
15. http://www.fjoch.com/GIZA++.html

16. http://www.loria.fr/~lehong/softwares.php
17. http://www.speech.sri.com/projects/srilm
18. http://www.statmt.org/moses
19. http://www-tsujii.is.s.u-tokyo.ac.jp/~tsuruoka/maxent
20. Huang, Liang, Kevin Knight, and Aravind Joshi. 2006. Statistical syntax-directed translation with extended domain of locality. In Proceedings of the 7th Biennial Conference of the Association for Machine Translation in the Americas.
21. Joseph Turian , Luke Shen , I. Dan Melamed, 2003. Evaluation of Machine Translation and its Evaluation, In Proceedings of MT Summit IX , New Orleans, LA.
22. K. Yamada, K. Knight: A Syntax-Based Statistical Translation Model. Proc. 39th Annual Meeting of the Assoc. for Computational Linguistics (ACL), pp. 523–530, Toulouse, France, July 2001.
23. Kazama, Jun'ichi and Jun'ichi Tsujii. Evaluation and Extension of Maximum Entropy Models with Inequality Constraints. In the Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing (EMNLP 2003). pp. 137-144, 2003.
24. Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In Proceedings of HLT-NAACL 2003, pages 127–133, Edmonton, Canada.
25. Koehn, Philipp. 2004a. Pharaoh: a beam search decoder for phrase-based statisticalmachine translation models. In Proceedings of the Sixth Conference of the Association forMachine Translation in the Americas, pages 115–124.
26. Liu, Yang, Qun Liu, and Shouxun Lin. 2006. Tree-to-string alignment template for statistical machine translation. In Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics, pages 609–616

27. Marcu, Daniel and William Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In Proceedings of EMNLP 2002, pages 133–139, Philadelphia, PA.
28. Och, Franz Josef and Hermann Ney. 2000. Improved statistical alignment models. In Proceedings of the 38th Annual Meeting of the ACL, pages 440–447, Hong Kong.
29. Och, Franz Josef and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In Proceedings of the 40th Annual Meeting of the ACL, pages 295–302, Philadelphia, PA.
30. Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU:a method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting of the ACL, pages 311–318, Philadelphia, PA.
31. Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, Evan Herbst, Moses: Open Source Toolkit for Statistical Machine Translation, Annual Meeting of the Association for Computational Linguistics (ACL), demonstration session, Prague, Czech Republic, June 2007.
32. Phuong Thai Nguyen, Akira Shimazu, Le-Minh Nguyen, and Van-Vinh Nguyen. 2007. A syntactic transformation model for statistical machine translation. International Journal of Computer Processing of Oriental Languages (IJCPOL), 20(2):1–20.
33. Richard Zens, Franz Josef Och and Hermann Ney. Phrase-Based Statistical Machine Translation. In: M. Jarke, J. Koehler, G. Lakemeyer (Eds.) : KI - 2002: Advances in Artificial Intelligence. 25. Annual German Conference on AI, KI 2002, Vol. LNAI 2479, pp. 18-32, Springer Verlag, September 2002.

34. Roland Kuhn, George Foster, Nicola Ueffing: The State of the Art in Phrase-Based Statistical Machine Translation (SMT), Institute for Information Technology, Canada, February 2007
35. S. Vogel: SMT Decoder Dissected: Word Reordering. Proc. Int. Conf. on Natural Language Processing and Knowledge Engineering (NLP-KE), pp. 561–566, Beijing, China, October 2003.
36. Stanley F. Chen , Ronald Rosenfeld. A Gaussian Prior for Smoothing Maximum Entropy Models (1999)
37. T. B. Nguyen, T. M. H. Nguyen, L. Romary, and X. L. Vu. 2004b. Lexical descriptions for vietnamese language processing. In ALR–04, Workshop on Asian Language Resources, Hainan, China.
38. Wu, Dekai. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. Computational Linguistics, 23:377–404.
39. Xiong, Deyi, Qun Liu, and Shouxun Lin. 2006. Maximum entropy based phrase reordering model for statistical machine translation. In Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics, pages 521–528.
40. Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-string alignment template for statistical machine translation. In Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics, pages pages 115–124.
41. Yuval Marton and Philip Resnik. “Soft Syntactic Constraints for Hierarchical Phrased-Based Translation”. The 46th Annual Meeting of the Association for Computational Linguistics. Columbus, Ohio, June 16-18, 2008.
42. Zens, Richard and Hermann Ney. 2006. Discriminative reordering models for statistical machine translation. In Proceedings of the Workshop on Statistical Machine Translation, pages 55–63.

43. Zhang 2003 Competitive grouping in integrated phrase segmentation and alignment model. ACL Workshops, Proceedings of the ACL Workshop on Building and Using Parallel Texts Ann Arbor, Michigan
44. Zhongjun He, Qun Liu, and Shouxun Lin. 2008. Improving statistical machine translation using lexicalized rule selection. Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pages 321-328, Manchester, August 2008.