Syntax-based Hierarchical Rule Extraction

A Thesis

Presented to
The Faculty of the Graduate School of Arts and Sciences
Brandeis University
Computer Science
Nianwen Xue, Advisor

In Partial Fulfillment
of the Requirements for the Degree
Master of Arts

by
Shiman Guo
August, 2015
Acknowledgments

This work was done under the supervision of my thesis advisor Nianwen Xue. He has greatly broadened my horizon in the natural language processing area and led me deep in to the field of machine translation. I appreciate his patience of refining my ideas and drafts as well as his generous support.

I would also like to thank Yuchen Zhang and Yaqin Yang for giving me informative guidance and inspiration and help me walk through the MT process. I’m also grateful to Dun Deng for his linguistic intuition and encouragement.

My thanks must go also to James Pustejovsky and Mark Verhagen for serving on the committee. They showed me the rigorousness that a researcher should have. I would also like to thanks Lotus Goldberg for her support of this excellent master’s program which has been life-changing to me.
Abstract

Syntax-based Hierarchical Rule Extraction

A thesis presented to the Faculty of
the Graduate School of Arts and Sciences of
Brandeis University, Waltham, Massachusetts
by Shiman Guo

Tree-based machine translation has attracted people’s attraction for its outstanding ability of handling long distance reordering and discontinuous expressions. Inferring synchronous context free grammar (SCFG) from syntactic trees also becomes popular since it extracts cleaner linguistic phrases and has the potential of improving the SCFG quality. However, the incompatibility between word alignment and subtree alignment as well as the ambiguity of multiple subtree aligning points hamper the extraction of high quality rules.

In this thesis, we present a new rule extraction approach for Hiero-like models using syntax trees as constraints. We directly take advantage of a hierarchically aligned Chinese-English parallel treebank (HACEPT) as a training resource to infer a minimal but effective SCFG. We first explore how to obtain a cleaner word alignment that prevents the correct subtree pair from being filtered out. Then, we propose an efficient and effective approach to obtain a minimal subtree alignment that yields high quality hierarchical rules. We also analyze the impact of our rule extraction approach on Machine Translation quality using a large scale parallel data set.

Our experiments show that our approach could reduce the size of Hiero’s rule table by almost 90%, with only a slight loss of translation quality as measured by BLEU score.
Contents

Abstract iv

1 Introduction 1

2 Classic Subtree Alignment and Rule Extraction Methods 4

3 The HACEPT Alignment Scheme 7

4 A Cleaner Word Alignment Approach 10
   4.1 GIZA++ Baseline ................................. 10
   4.2 Unsupervised Syntax-Aware Word Alignment .... 11
   4.3 Supervised Word Alignment ........................ 12
   4.4 Evaluation ........................................... 12

5 A Robust Subtree Alignment Framework 13
   5.1 The Ambiguity ................................. 13
   5.2 Binary Classification and Greedy Search .......... 16
   5.3 Score Function and Dynamic Programming ......... 17
   5.4 Evaluation ........................................... 19

6 Experiments 21
   6.1 Data Set ................................................. 21
   6.2 Experiments ............................................. 21
   6.3 Discussion .............................................. 22

7 Conclusion 26

Bibliography 28
Chapter 1

Introduction

Phrase-based machine translation [Koehn et al., 2003] is a simple but powerful model since it expands the translation unit to phrases and is able to capture local reordering. However, it cannot handle long distance reordering and discontinuous phrases [Quirk and Menezes, 2006]. The hierarchical phrase-based model (Hiero) [Chiang, 2005] is a successful framework to address these deficiencies by automatically inferring synchronous context free grammar and decoding with a CKY-like parsing algorithm. Like Hiero, many other tree-based MT models have also been proposed to overcome the shortcomings of phrase-based models, by taking advantages of syntactic features. These models require syntactic phrase structures on either or both sides [Chiang, 2010, Galley et al., 2006, Liu and Gildea, 2008].

A key part that determines the translation quality for tree-based models is the synchronous context free grammar (SCFG). By applying a set of heuristics, Hiero automatically extracts a set of rules without relying on syntactic structures. The extraction heuristics set a couple of hard limits to avoid generating too many “noise” rules. For example, the length of the initial phrases is set to a fixed number (usually between 5 to 10) [Chiang, 2005], therefore in principle it cannot handle long-distance reordering, and the size of rule table still tends to be huge.
Syntax-based models extract rules from syntactic trees, which provide triggers as well as constraints for the rule extraction process. They have better generalization capacities and extract much fewer rules, but the performance of syntax-based systems are usually not as good as Hiero due to the errors in syntactic parse trees and the inability of syntax-based models to handle cross-linguistic structural divergence [Dorr, 1994]. Moreover, classic tree-to-tree models extract rules exponentially from every possible subtree pair, which will create a lot of spurious ambiguities and potentially harm MT [Deng and Xue, 2014].

There is also research on aligning subtrees and extracting a minimal rule set from them. For example, [Lavie et al., 2008] proposed an efficient algorithm that automatically aligns subtrees according to word alignments. [Groves et al., 2004] presented an aligning algorithm that makes use of constituent labels and syntax templates as well. These methods successfully reduced the rule table size a lot without impacting translation quality. Either a word aligned corpus or some heuristics are used to disambiguate the aligning points. Yet there is still a recurring issue that hampers the extraction of high quality rules: the incompatibility between word alignments and phrase structures [DeNero and Klein, 2007]. Improper word alignments sometimes can rule out good hierarchical rules, resulting a rule table too small to generate high quality translations.

[Deng and Xue, 2014] builds a hierarchically aligned Chinese-English parallel treebank (HACEPT) with human annotation on both word alignment and phrase alignment, which eliminates the incompatibility and allows the inference of SCFGs that bilingual speakers consider to be intuitive. We design a framework that makes use of the hierarchically aligned corpus to address the ambiguity of subtree alignment to improve the SCFG. Experiments show promising results: Our extraction algorithm generates a significantly smaller (almost 1/10 of Hiero) rule table, but is able to produce comparable translation quality as measured by the BLEU score. The rest of this thesis goes as follows: Chapter 2 describes the classic rule extraction process. Chapter 3 describes the HACEPT alignment scheme. Chapter 4 and 5
describes our word and subtree alignment approach respectively, Chapter 6 demonstrates the
influence of our rule extraction approach on MT as well as our analysis. and we summarize
our conclusions in Chapter 7.

To summarize, the contributions of this thesis are as follows:

- We explored word alignment approaches that could facilitate hierarchical rule extrac-
  tion.

- We present a supervised alignment framework which aims to overcome the incompati-
  bility between word level and phrase level alignments.

- We present a syntactic tree-based rule extraction approach that yields comparable
  translation quality with a rule table that is significantly smaller in size than that of
  Hiero-based systems.
Chapter 2

Classic Subtree Alignment and Rule Extraction Methods

The key part of extracting hierarchical rules from syntactic trees is the alignment of subtrees. In practice, this aligning process is usually based on word alignment so that the candidates could be constrained to a manageable small number. There are also approaches that do not rely on word alignment, like [Xiao and Zhu, 2013], but these models are difficult to scale to larger dataset. The following heuristics are usually applied to infer the subtree alignment candidates:

Given a source tree $S$ and a target tree $T$, for a subtree pair $<\alpha, \beta>$ where $\alpha$ and $\beta$ are subtrees of $S$ and $T$ respectively, if there are at least one word in $\alpha$ that is aligned to a word in $\beta$, and no word in $\alpha$ aligned to words outside of $\beta$, and vice versa, then $<\alpha, \beta>$ is a licensed subtree pair.

Figure 2.1 shows a fragment of parallel texts with parse trees on both sides. With the heuristics above, subtree alignment candidates can be easily inferred from word alignments: $<e_1, f_1>, <e_2, f_3>, <e_3, f_4>, <e_4, f_4>$. 
Figure 2.1: A fragment of Chinese-English bitext with parse trees

<table>
<thead>
<tr>
<th>Alignment</th>
<th>Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;e2, f3&gt;</td>
<td>this &lt;&gt; 它</td>
</tr>
<tr>
<td>&lt;e4, f4&gt;</td>
<td>happening &lt;&gt; 发生</td>
</tr>
<tr>
<td>&lt;e1, f1&gt;</td>
<td>prevent $X_1$ from $X_2$ &lt;&gt; 防止 $X_1$ $X_2$</td>
</tr>
</tbody>
</table>

Table 2.1: Example Subtree Alignment and Rules Extracted 1

It’s obvious that the first two candidates are correct alignments and there are no other possible choices. The last two candidates alignments, however, share the same foreign side node and have some ambiguity. Classic rule extraction methods take all 4 candidates as valid alignments, and other minimal alignment methods take some preferences in terms of node depth. For example, one possibility is to keep the alignment that has the lowest node depth on both sides and discard other candidates. In this example, this would keep the 4th alignment and discard the 3rd.

Hierarchical rules are directly inferred from subtree alignments. Assuming the subtrees are aligned like the following: <e1, f1>, <e2, f3>, <e4, f4>, each subtree pair will result in a translation rule. See Table 2.1.

If the foreign subtree f4 is aligned to English subtree e3 instead of e2, a different rule set will be inferred. See Table 2.2, where rules starting with a star show the difference.

<table>
<thead>
<tr>
<th>Alignment</th>
<th>Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;e2, f3&gt;</td>
<td>this &lt;&gt; 它</td>
</tr>
<tr>
<td>&lt;e3, f4&gt;</td>
<td>*from happening &lt;&gt; 发生</td>
</tr>
<tr>
<td>&lt;e1, f1&gt;</td>
<td>*prevent $X_1$ $X_2$ &lt;&gt; 防止 $X_1$ $X_2$</td>
</tr>
</tbody>
</table>

Table 2.2: Example Subtree Alignment and Rules Extracted 2
Intuitively the rule set from Table 2.1 is preferable over that shown in Table 2.2. From the bottom side, 发生 should be translated into happening, since the prepositional word from plays more of a syntactic role as the head of the PP rather than a semantic counterpart of 发生. The alignment from Table 2.2 will cause more ambiguity when translating 发生 into English. Since 发生 and happening (and its other forms) are a pretty stable translation, extracting a rule that translates 发生 to from happening can make the rule table more sparse in general. From the top side, prevent X₁ from X₂ <-> 防止 X₁ X₂ is also a better rule than prevent X₁ X₂ <-> 防止 X₁ X₂, because the former one tends to be more accurate with the preposition from captured on this level.

As stated at the beginning, these hierarchical rules are all drawn from word alignments. If the word alignment is not good enough, say, 发生 is aligned to both this and from, then alignment <e2, f3> will not be allowed. As a consequence, the top level rule could only be <prevent it from X, 防止 X>, which is less generalizable since the first argument of prevent is not recognized as a variable.

Classic rule extraction methods extract a rule set that includes rules from both Table 2.2 and Table 2.1, which is larger in size and contains more ambiguity. There are also approaches that attempt to extract a minimal set, that is, only rules from either Table 2.1 or Table 2.2 will be extracted, using some simple heuristics to deal with the ambiguity of multiple subtree aligning points. However, we show that such heuristic approaches are insufficient and a more systematic approach is needed in Chapter 5.
Chapter 3

The HACEPT Alignment Scheme

[Deng and Xue, 2014] construct a hierarchically aligned Chinese-English parallel treebank (HACEPT) with human annotation on both word level and phrase (subtree) level. The innovation of HACEPT alignment is that they leave words without translation counterparts unaligned, and use the subtree alignment to capture them.

The annotation on both levels is done simultaneously in a coordinated manner. On the word level, only words that share the equivalent meaning or grammatical functions are aligned. Words that do not have a translation counterpart are left unaligned. Most of these words are language specific functional words and convey some grammatical relations, such as arguments of a verb, reordering, etc. The unaligned words as well as their grammatical functions are captured by the subtree level alignment.

The subtree alignment is subjective and challenging. When determining whether two subtrees should be aligned, two general principles are applied:

1. Subtree alignment should not sever key dependencies related to the unaligned words.

2. Subtree alignment should be minimal, which means that the subtree alignment should contain only the elements involved in the grammatical relation and nothing more.
The first principle ensures that the grammatical relation is properly captured by the subtree alignment. Take Figure 2.1 as an example, 发生 should not be aligned to from happening because the preposition from indicates a relation between this and happening. The second principle ensures that proper variables (non-terminals) could be found. Usually when there are multiple possible alignments, one of them must be correct, but this has to obey the first principle as well. One example is that the WHNPs are seldomly aligned to each other since they serve as triggers of reordering in questions, and aliging them would make the triggers fail to be captured.

Figure 3.1 shows a complete example sentence pair with HACEPT style word alignment and subtree alignment. By this alignment scheme, HACEPT eliminates both the redundancies and spurious ambiguities, and is a great resource for the goal of inferring a SCFG set that is minimal in size but has high expressiveness capacity.

Given the subtree alignments we can easily extract rules from them. For example, the English PP (on tolls) is aligned to the Chinese PP (对通行费), and NP_{e2} (tolls) on the English side is aligned to NP_{c2} (通行费) on the Chinese side. From these two alignments we can get a hierarchical rule on X <> 对 X, by converting the lower level subtree pair into a variable.

Currently HACEPT contains 9,716 sentence pairs in total, after filtering out incorrect sentence alignments. They are from three genres, online blogs, online discussing forums, and newswire. With the gold subtree alignment, 103,796 rules are extracted in total, and we directly use them as the training set in our subtree alignment model.
Figure 3.1: A hierarchically aligned sentence pair
Chapter 4

A Cleaner Word Alignment Approach

Word alignment is a basic component for a phrase- or syntax-based SMT system. It constrains the spans of phrases in phrase-based models, and the number of subtrees in syntax-based models. The lexical translation probability is also of vital importance to score the phrase table or rule table. We prepared 1,653,140 unlabeled Chinese-English sentence pairs in total (referred as MISC data below)\(^1\), and explored three different approaches to do the word alignment. The first approach is to use GIZA++, which also serves as the baseline. The second approach is an unsupervised word alignment model with syntactic constraints. The last approach is a supervised model that directly learns some of its parameters from the HACEPT annotated data. We will compare their performance by evaluating against a selected test set from the HACEPT data.

4.1 GIZA++ Baseline

GIZA++ is a popular and highly optimized word alignment toolkit that implements IBM models 1-5 [Brown et al., 1993] and HMM word alignment model [Vogel et al., 1996]. To

estimate the word alignment quality of GIZA++ on our annotated HACEPT data, we put together the HACEPT data and the MISC data and obtained the word alignment result with a default GIZA++ configuration in Moses (5 IBM Model 1 iterations, 3 model 3 iterations and 3 model 4 iterations). The alignment is done on both directions and symmetrized by grow-diag-final-and heuristics [Ayan and Dorr, 2006]. No external dictionary is used.

4.2 Unsupervised Syntax-Aware Word Alignment

[DeNero and Klein, 2007] proposed an unsupervised word alignment approach that explicitly makes use of constituency structures based on the HMM model. This approach is reported to be helpful to tree transducer rule extraction systems, and is thus believed to be beneficial to minimizing the misaligned functional words as well.

Berkeley Aligner provides an efficient implementation of the syntax-aware word alignment approach (referred as syntactic HMM below) with a joint training algorithm proposed by [Liang et al., 2006]. As opposed to using standard grow-diag-final-and heuristics to combine outputs from different models, the joint training maximizes a combination of data likelihood and agreement between them using an EM-like algorithm. We parsed the English sentences with the Berkeley Parser, and set 5 iterations of IBM Model 1 and syntactic HMM models on the forward (source-to-target) direction. On the reverse direction, we set another 5 iterations of IBM Model 1 and the normal HMM model. We didn’t use the syntactic HMM model on the reverse direction since it’s not stable yet in the current version. The posterior decoding threshold is set to the default value 0.5. The training data is the same as that used with GIZA++. We also used a small Chinese-English dictionary that contains 44,107 entries.
<table>
<thead>
<tr>
<th>Approach</th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>GIZA++</td>
<td>0.5992</td>
<td>0.8040</td>
<td>0.6866</td>
</tr>
<tr>
<td>Syntactic HMM</td>
<td>0.7357</td>
<td>0.7414</td>
<td>0.7385</td>
</tr>
<tr>
<td>Supervised ITG</td>
<td>0.7461</td>
<td>0.7900</td>
<td>0.7674</td>
</tr>
</tbody>
</table>

Table 4.1: Evaluation of different word alignment approaches

4.3 Supervised Word Alignment

A potentially better way to get the HACEPT scheme word alignment is using the annotated data as a training set. [Haghighi et al., 2009] proposed a supervised word alignment model that uses the inversion transduction grammar (ITG) as a constraint. The BerkeleyAligner also provides an implementation of this model. To properly evaluate the alignment quality, we randomly selected 7,772 sentence pairs from the HACEPT data as the labeled training set, plus the MISC data as the unlabeled training set, and leave the rest 1,944 sentence pairs for testing. On the forward direction we set 5 iterations of joint HMM and IBM Model 1, and 5 iterations of independent HMM and Model 1 on the reverse direction.

4.4 Evaluation

Table 4.1 shows the evaluation results of the three approaches mentioned, using precision, recall and F-measure scores. The quality is tested against the 1,944 sentence pairs we split out from the HACEPT data in the supervised word alignment model. As shown in the table, while GIZA++ result has a much higher recall, the precision is much lower than the other two approaches. Adding syntactic constraints in the unsupervised alignment model helped a lot in improving the precision. Although the recall is decreased a lot, the general F-measure score of the syntactic HMM model is better than GIZA++. The best result is from the supervised ITG model, as expected, but in practice the decoding process of a trained ITG model is significantly slower than the unsupervised models.
Chapter 5

A Robust Subtree Alignment Framework

5.1 The Ambiguity

While a HACEPT style word alignment prevents good hierarchical rules from being filtered out, it creates lots of ambiguous aligning points. As shown in Figure 2.1, bad subtree alignment not only causes improper phrase translations, but also weakens the expressiveness of the SCFG.

In our research, we use the heuristics described in Chapter 2 to find possible target side subtrees \( \beta_1, \beta_2, \ldots \) for each source side subtree \( \alpha \); similarly, for each target side subtree \( \beta \), there may also be multiple source side subtrees that can be aligned to it. Subtrees that share the same counterpart set will be merged together. It is possible that both the source side and the target side have more than one aligning points, like \(<\{\alpha_1, \alpha_2\}, \{\beta_1, \beta_2, \beta_3\}>\). We call the resulting merged set a Frame. Figure 5.1 shows a complete parallel tree. Rectangles with the same background color are the frames. Possible alignment points are surrounded by a solid box. Note that the whole sentence pair (covering the root node and all terminal
Figure 5.1: Example of Frames
<table>
<thead>
<tr>
<th># of Candidates</th>
<th># of Frames</th>
<th>Percentage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20829</td>
<td>63.84</td>
</tr>
<tr>
<td>2-4</td>
<td>10902</td>
<td>33.42</td>
</tr>
<tr>
<td>4+</td>
<td>894</td>
<td>2.74</td>
</tr>
</tbody>
</table>

Table 5.1: Distribution of Aligning Points in Frames

words) is also a valid frame.

Frames can be deterministically inferred from word alignment, but there is a lot of uncertainty in how the subtrees should be aligned within each frame. A frame is composed of at least one subtree on the English side and at least one subtree on the foreign side. The subtrees on both sides follow the heuristics described in Chapter 2, and each subtree on one side could be a translation counterpart of any subtree on the other side. Therefore, one frame could produce $m \times n$ subtree alignments, where $m$ and $n$ are the number of subtrees on the two sides. In the HACEPT data set, it is estimated that over 36.16% of all frames have multiple aligning points. Table 5.1 illustrates the distribution of possible alignments in all frames extracted from the training part of the HACEPT data (7,772 sentence pairs).

Like the syntax tree, frames are also hierarchical. It can have several sub-frames. We call the frame that is not dominated by any other frames the **root frame**. One parallel tree has only one root frame, which contains the root of both trees. The alignment of the source root and the target root is considered to be the only and correct alignment in the root frame, and is called **root alignment**. Meanwhile, we call frames that do not have any sub-frames **terminal frames**. In Figure 5.1, the invisible box that covers everything from the root to the terminals is the root frame. The pink, blue, and green areas are all terminal frames.

To obtain a translation rule, we first need to select an alignment $a$ in a frame, and use this subtree pair as the **base**. If it is in a terminal frame, then the base itself is a phrase translation rule, otherwise, if its frame has sub-frames, select an alignment $b$ for each sub-frame as the **variables**. Both **base** and **variable** are subtree pairs. The **base** determines the span of a rule while the **variables** determine the non-terminals in the span. We use the notation
Rule(\(a, \{b_1, b_2, ..., b_i\}\)) to refer the rule that is extracted from the base alignment \(a\) and the set of variable alignments \(\{b_1, b_2, ..., b_i\}\). For example, in the parallel tree shown in Figure 5.1, let alignments \(u = \langle NP_{e3}, NP_{f4} \rangle\), \(v = \langle VP_{e5}, VP_{f5} \rangle\), and \(w = \langle VP_{e1}, VP_{f1} \rangle\), then Rule(\(w, \{u, v\}\)) means the rule extracted from base \(w\) with variables \(u\) and \(v\), namely, has prevented \(X_1\) from \(X_2\) \(<\>) 防止了 \(X_1 X_2\).

The ambiguity comes from the fact that there are potentially multiple aligning points in all frames. In each non-terminal frame, there are multiple possible bases, and for each sub-frame in it, we have potentially many variables, and we need to select the correct aligning point for each frame. Taking the red frame in Figure 5.1 as example, to obtain a hierarchical rule, we first need to select an alignment for subtree VP_{f1}, either VP_{e1} or VP_{e2}, so that we can have the outer span. Then we need to select an alignment for each of the sub-frames. The blue frame has only one candidate, which will be chosen automatically. The choice of alignment in the green frame will affect how the variable will look like. If we choose VP_{f5} aligned to VP_{e5}, then we get a rule prevented \(X_1\) from \(X_2\) \(<\>) 防止了 \(X_1 X_2\). Otherwise, we will have another rule, namely prevented \(X_1 X_2\) \(<\>) 防止了 \(X_1 X_2\).

We made the following two assumptions to help solve the ambiguity:

- The correct alignment in the root frame is the subtree pair that has the deepest subtree on both sides. Since we assume that the two sentences are parallel.
- Each frame has one and only one correct alignment.

### 5.2 Binary Classification and Greedy Search

As proposed by [Sun et al., 2010], we can use a discriminative binary classification model like MaxEnt and greedy search to address this problem. Given a frame \(<s_1, s_2, ... s_i, t_1, t_2, ..., t_j>\), the probability of a subtree alignment \(a = \langle s_i, t_j \rangle\) is defined as the following:
\[ P(a = 1 \mid \theta) = \frac{\exp\left[ \sum_{m=1}^{M} \lambda_m h_m(a = 1, \theta) \right]}{\sum_{a' \in \{0,1\}} \exp\left[ \sum_{m=1}^{M} \lambda_m h_m(a', \theta) \right]} \]

where

\( a = 1 \) means that subtree \( s_i \) and \( t_j \) should be aligned. Feature functions are defined as \( h_m(a, \theta) \), where \( \theta \) is an additional variable for its contextual information such as its parent tree.

The greedy search algorithm starts from the root frame. The best alignment in the root frame is considered to be the deepest node pair, since two sentences are assumed to be parallel. We then proceed to frames in the next level, and propose the alignment candidate with highest confidence in the sub-frames. We continue this process until all frames have been assigned a best alignment.

[Sun et al., 2010] also proposed a set of useful features, including the translation probabilities of two phrases, word alignment probabilities, difference of tree spans, number of descendants and tree depth, as well as constituency category features.

In this subtree alignment model, the probability of each alignment depends on its local features (mostly on semantic equivalence), and the greedy algorithm is essentially a local optimum, yet good performance is reported in their research. We will show another approach that directly takes the rule quality into account and uses dynamic programming (DP) to reach a global optimum.

### 5.3 Score Function and Dynamic Programming

Instead of thinking about “what is the probability that alignment \( a \) is a good subtree alignment” described in the previous section, we start from “what kind of alignments will produce a rule set that is the most consistent with the HACEPT scheme”. There are two differences
between these two perspectives. On the one hand, apart from the semantic equivalence features that are used in the previous section, we also need to know whether this alignment could yield a good rule before we finally decide to align them or not. On the other hand, we aim to extract a globally optimal rule set from a parallel tree, as opposed to local correctness.

Given a frame $F = < s_1, s_2, ..., s_i, t_1, t_2, ..., t_j >$, an alignment $a = < s_i, t_j >$, we define the base alignment selected from F’s parent frame as $a'$, and the probability of $a$ is given by:

$$P(a) = \begin{cases} 1, & \text{if } a \text{ is the root alignment} \\ P(a')\phi(a, a'), & \text{otherwise} \end{cases}$$

In this equation, the probability of a non-root alignment $a$ is given by the product of the probability of its base $a'$ and the score of the extracted rule $\phi(a, a')$, defined as:

$$\phi(a, a') = \begin{cases} \theta(s_i|t_j)\theta(t_j|s_i), & \text{if } a \text{ is in a terminal frame} \\ \frac{\delta(r',\text{Rule}(a',\{a\}))}{|\pi|}, & \text{otherwise} \end{cases}$$

where $\pi$ is the gold rule set extracted from HACEPT.

As shown above, there are two cases for scoring a rule. If $\text{Rule}(a', \{a\})$ is a hierarchical rule (a rule that contains non-terminal symbols), then the rule is scored by its prior probability in the gold rule set $\pi$. If $\text{Rule}(a', \{a\})$ is a terminal rule that doesn’t contain any non-terminal symbols, then it is scored by the forward phrase translation probabilities $\theta(s_i \mid t_j)$ and the backward probability $\theta(t_j \mid s_i)$, defined as the following equations:

$$\theta(s_i \mid t_j) = (\prod_{v \in t_j} \sum_{u \in s_i} P(u|v))^{1/|t_j|}$$

$$\theta(t_j \mid s_i) = (\prod_{u \in s_i} \sum_{v \in t_j} P(v|u))^{1/|s_i|}$$

The phrase translation probability $\theta(s_i \mid t_j)$ is calculated as follows: For each lexical item in subtree $t_j$, we take the summation of its lexical translation probability to each lexical item
in subtree $s_i$, and then take the production of all summations normalized by the number of lexical items in $t_j$. We calculate the reverse phrase translation probability in the same way.

The computation of the probability of an alignment requires calculating the probability of the base alignment recursively. To obtain the best alignments, we start from the root alignment in the root frame, which is considered to be always 1.0. With this alignment as the base, we calculate the probability of each variable alignment. We move on to sub-frames in the next level and obtain the probability of the alignments in them, so on and so forth until we reach the terminal frames. For each alignment, a backpointer is kept to trace its best base alignment, so that the final alignments can be recovered using these backpointers.

Intuitively this process could also be understood as a Markov Chain. A state is a candidate alignment in its frame, a transition is to select alignments in its sub-frames. The initial state is the root alignment, and the terminal state is a state where there are no more sub-frames. The transition probabilities are calculated by scoring the rules which are obtained by two levels’ alignments, and the task is to find the best path from the initial state and the terminal state.

## 5.4 Evaluation

To evaluate the quality of our DP subtree alignment model, we used the same 1,944 parallel sentences with gold trees mentioned in Chapter 4. Table 5.2 shows us the evaluation results using different word alignment approaches.

<table>
<thead>
<tr>
<th>Word Alignment</th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>GIZA++</td>
<td>0.7444</td>
<td>0.5148</td>
<td>0.6087</td>
</tr>
<tr>
<td>Syntactic HMM</td>
<td>0.7149</td>
<td>0.6842</td>
<td>0.6992</td>
</tr>
<tr>
<td>Supervised ITG</td>
<td>0.7627</td>
<td>0.7388</td>
<td>0.7567</td>
</tr>
</tbody>
</table>

Table 5.2: Evaluation of subtree alignment

With GIZA++ word alignment, our final F1 score of the subtree alignment is much lower
than the rest two results, which are around or more than 0.70. This result empirically shows that improper word alignment would significantly harm rule extraction.
Chapter 6

Experiments

6.1 Data Set

We conducted MT experiments on large parallel dataset to show the impact of our rule extraction approach. As described in Chapter 4, we collected 1,653,140 Chinese-English parallel sentences in total. We used Stanford NLP toolkits [Manning et al., 2014] for English tokenization and Chinese segmentation, and used the Berkeley Parser to obtain the parse trees on both sides. We also used NiuTrans [Xiao et al., 2012] preprocessing tools to remove sentence pairs that contains meaningless markups and characters to avoid unexpected problems.

6.2 Experiments

We obtained two versions of word alignment for this large dataset, by using GIZA++ and the Berkeley Aligner with unsupervised syntactic HMM models. We also tried using Berkeley supervised model since it shows highest accuracy on the HACEPT data as well as great compatibility with hierarchical rule extraction. However the decoding speed is prohibitively
slow and we ended up using the unsupervised syntactic HMM word alignment results.

We trained a trigram language model using SRILM [Stolcke et al., 2002] on the English side of the Gigaword corpus. Good-Turing discounting is also performed to refine the probability distributions.

We conducted three experiments with Moses [Koehn et al., 2007]:

- **Hiero**: a classic Hiero system, with GIZA++ word alignment and hierarchical phrase rules extracted from it.

- **HACEPT-GIZA**: a customized Hiero system with GIZA++ word alignment and our own rule extraction approach.

- **HACEPT-HMM**: a customized Hiero system with syntactic HMM word alignment and our own rule extraction approach.

All of them share the same default decoding parameters in Moses hierarchical mode. We used NIST03 Chinese-English parallel data for tuning and NIST05 for testing.

Table 6.1 showed the extracted rule table size and the BLEU score of these three systems. While Hiero still outperforms the other two systems, the difference between it and HACEPT-HMM is pretty small, only 1.8% relative reduction. However, the rule table size of HACEPT-HMM is significantly smaller than that of Hiero (a 89.46% relative reduction).

<table>
<thead>
<tr>
<th>System</th>
<th>Rule Size</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hiero</td>
<td>2,173M</td>
<td>0.3177</td>
</tr>
<tr>
<td>HACEPT-GIZA</td>
<td>234M</td>
<td>0.3054</td>
</tr>
<tr>
<td>HACEPT-HMM</td>
<td>229M</td>
<td>0.3120</td>
</tr>
</tbody>
</table>

Table 6.1: Results of MT experiments
6.3 Discussion

We found that although HACEPT-HMM’s translation tends to be more grammatical, it suffers a lot from over-reordering. We believe that it is due to the lack of syntactic features during decoding. Since the left hand side of the SCFG is always decoded as an universal ‘X’, rules with few terminal words (e.g. $X_1, X_2 \leftrightarrow X_2, X_1$) are easily abused during decoding. While it is believed that taking into account all syntactic information usually harms tree-based MT, it is possible that a selected set of them could show more potentials of syntactically inferred SCFGs.

Another obstacle that hinders the improvement of translation quality is the insufficient structures. Unreliable parse trees cause good non-terminals to be missing and result in rules with too many terminals. As a consequence, the SCFG will be more sparse and have a weaker generation ability.

Table 6.2 and Table 6.3 show top frequent hierarchical rules (rules with non-terminals) extracted from HACEPT-HMM and the Hiero baseline respectively. There are two things that we can notice from the comparison: First, rules from HACEPT-HMM tends to be more generalized, not only because intuitively these rules are of higher level, the frequencies of top HACEPT-HMM rules are also larger than that from top Hiero baseline rules. Second, rules from HACEPT-HMM have better integrity while rules from Hiero baseline tend to be more fragmented. For example, HACEPT-HMM extracts $(X) \leftrightarrow (X)$ as a single rule while Hiero baseline extracts three different rules (rank 4, 5 and 6). This shows that our rule extraction approach that follows the HACEPT scheme is successful in terms of expressiveness.
<table>
<thead>
<tr>
<th>Rank</th>
<th>Chinese</th>
<th>English</th>
<th>Alignment</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X X</td>
<td>X X</td>
<td>0-0 1-1</td>
<td>149034</td>
</tr>
<tr>
<td>2</td>
<td>X X .</td>
<td>X X .</td>
<td>0-0 1-1 2-2</td>
<td>118126</td>
</tr>
<tr>
<td>3</td>
<td>( X )</td>
<td>( X )</td>
<td>0-0 1-1 2-2</td>
<td>79642</td>
</tr>
<tr>
<td>4</td>
<td>X X</td>
<td>X X</td>
<td>0-1 1-0</td>
<td>61930</td>
</tr>
<tr>
<td>5</td>
<td>in X</td>
<td>in X</td>
<td>0-0 1-1</td>
<td>39154</td>
</tr>
<tr>
<td>6</td>
<td>X X</td>
<td>X of X</td>
<td>0-2 1-0</td>
<td>31137</td>
</tr>
<tr>
<td>7</td>
<td>X , X X .</td>
<td>X , X X .</td>
<td>0-0 1-1 2-2 3-3 4-4</td>
<td>24091</td>
</tr>
<tr>
<td>8</td>
<td>is X</td>
<td>is X</td>
<td>0-0 1-1</td>
<td>21543</td>
</tr>
<tr>
<td>9</td>
<td>of X</td>
<td>of X</td>
<td>0-1 1-0</td>
<td>18325</td>
</tr>
<tr>
<td>10</td>
<td>X X</td>
<td>X X .</td>
<td>0-0 1-1</td>
<td>12603</td>
</tr>
<tr>
<td>11</td>
<td>with X</td>
<td>with X</td>
<td>0-0 1-1</td>
<td>12384</td>
</tr>
<tr>
<td>12</td>
<td>to X</td>
<td>to X</td>
<td>0-0 1-1</td>
<td>8215</td>
</tr>
<tr>
<td>13</td>
<td>X and X</td>
<td>X and X</td>
<td>0-0 1-1 2-2</td>
<td>8067</td>
</tr>
<tr>
<td>14</td>
<td>in X</td>
<td>at X</td>
<td>0-0 1-1</td>
<td>7359</td>
</tr>
<tr>
<td>15</td>
<td>to X</td>
<td>to X</td>
<td>0-0 1-1</td>
<td>7143</td>
</tr>
<tr>
<td>16</td>
<td>by X</td>
<td>by X</td>
<td>0-0 1-1</td>
<td>7031</td>
</tr>
<tr>
<td>17</td>
<td>will X</td>
<td>will X</td>
<td>0-0 1-1</td>
<td>7026</td>
</tr>
<tr>
<td>18</td>
<td>to X</td>
<td>to X</td>
<td>0-0 1-1</td>
<td>6910</td>
</tr>
<tr>
<td>19</td>
<td>in X</td>
<td>in X</td>
<td>0-0 1-1</td>
<td>6711</td>
</tr>
<tr>
<td>20</td>
<td>to X</td>
<td>to X</td>
<td>0-0 1-1</td>
<td>6448</td>
</tr>
<tr>
<td>21</td>
<td>for X</td>
<td>for X</td>
<td>0-0 1-1</td>
<td>6334</td>
</tr>
<tr>
<td>22</td>
<td>will X</td>
<td>will X</td>
<td>0-0 1-1</td>
<td>6180</td>
</tr>
<tr>
<td>23</td>
<td>in X</td>
<td>in X</td>
<td>0-0 1-1</td>
<td>5835</td>
</tr>
<tr>
<td>24</td>
<td>X and X</td>
<td>X and X</td>
<td>0-0 1-1 2-2</td>
<td>5696</td>
</tr>
<tr>
<td>25</td>
<td>from X</td>
<td>from X</td>
<td>0-0 1-1</td>
<td>5481</td>
</tr>
</tbody>
</table>

Table 6.2: Frequent Rules in HACEPT-HMM
<table>
<thead>
<tr>
<th>Rank</th>
<th>Chinese</th>
<th>English</th>
<th>Alignment</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X .</td>
<td>X .</td>
<td>1-1 0-0</td>
<td>51743</td>
</tr>
<tr>
<td>2</td>
<td>X ,</td>
<td>X ,</td>
<td>1-1 0-0</td>
<td>44759</td>
</tr>
<tr>
<td>3</td>
<td>, X</td>
<td>, X</td>
<td>0-0 1-1</td>
<td>42810</td>
</tr>
<tr>
<td>4</td>
<td>X )</td>
<td>X )</td>
<td>1-1 0-0</td>
<td>35374</td>
</tr>
<tr>
<td>5</td>
<td>( X</td>
<td>( X</td>
<td>0-0 1-1</td>
<td>28169</td>
</tr>
<tr>
<td>6</td>
<td>( X )</td>
<td>( X )</td>
<td>0-0 2-2 1-1</td>
<td>24508</td>
</tr>
<tr>
<td>7</td>
<td>and X</td>
<td>和 X</td>
<td>0-0 1-1</td>
<td>20366</td>
</tr>
<tr>
<td>8</td>
<td>X and</td>
<td>X 和</td>
<td>1-1 0-0</td>
<td>20196</td>
</tr>
<tr>
<td>9</td>
<td>the X</td>
<td>的 X</td>
<td>0-0 1-1</td>
<td>19218</td>
</tr>
<tr>
<td>10</td>
<td>and X</td>
<td>及 X</td>
<td>0-0 1-1</td>
<td>17791</td>
</tr>
<tr>
<td>11</td>
<td>today X</td>
<td>今日 X</td>
<td>0-0 1-1</td>
<td>15756</td>
</tr>
<tr>
<td>12</td>
<td>X and</td>
<td>X 及</td>
<td>1-1 0-0</td>
<td>15744</td>
</tr>
<tr>
<td>13</td>
<td>today (X</td>
<td>今日 (X</td>
<td>0-0 1-1 2-2</td>
<td>15701</td>
</tr>
<tr>
<td>14</td>
<td>X and X</td>
<td>X 和 X</td>
<td>1-1 0-0 2-2</td>
<td>15258</td>
</tr>
<tr>
<td>15</td>
<td>today X</td>
<td>今日 X</td>
<td>0-0 2-2 1-1</td>
<td>14324</td>
</tr>
<tr>
<td>16</td>
<td>today (X</td>
<td>今日 (X</td>
<td>0-0 1-1 3-3 2-2</td>
<td>14320</td>
</tr>
<tr>
<td>17</td>
<td>, X</td>
<td>, X</td>
<td>0-0 1-1</td>
<td>13963</td>
</tr>
<tr>
<td>18</td>
<td>in X</td>
<td>在 X</td>
<td>0-0 1-1</td>
<td>13456</td>
</tr>
<tr>
<td>19</td>
<td>the X of</td>
<td>的 X</td>
<td>0-0 1-1</td>
<td>13376</td>
</tr>
<tr>
<td>20</td>
<td>the X</td>
<td>X 的</td>
<td>0-1 1-0</td>
<td>13308</td>
</tr>
<tr>
<td>21</td>
<td>X , X</td>
<td>X , X</td>
<td>1-1 0-0 2-2</td>
<td>12715</td>
</tr>
<tr>
<td>22</td>
<td>X and X</td>
<td>X 及 X</td>
<td>1-1 0-0 2-2</td>
<td>12387</td>
</tr>
<tr>
<td>23</td>
<td>X ( X</td>
<td>X ( X</td>
<td>1-1 0-0 2-2</td>
<td>12001</td>
</tr>
<tr>
<td>24</td>
<td>’ X</td>
<td>「 X</td>
<td>0-0 1-1</td>
<td>11900</td>
</tr>
<tr>
<td>25</td>
<td>X ,</td>
<td>X ,</td>
<td>1-1 0-0</td>
<td>11654</td>
</tr>
</tbody>
</table>

Table 6.3: Frequent Rules in Hiero Baseline
Chapter 7

Conclusion

Tree-based translation models have become very popular in the MT community. Their ability to handle long distance reordering and discontinuous expressions demonstrate great potential. Non-syntactically tree-based models (e.g. Hiero) do not require parse tree annotation and could produce state-of-the-art translation quality; however, they have disadvantages in global reordering of very long phrases, and suffer from having an enormous rule table. Syntactically tree-based models tend to be overly constrained and faces the problem of incompatibility between word alignment and subtree alignment, as well as the ambiguity of multiple subtree alignment points.

With the help of HACEPT, we systematically explored the issues in building a tree-based model that has harmonized alignments at the word level and optimized alignment at the subtree level, in order to improve the SCFG. We presented an automated rule extraction approach that makes use of constituency structures for both word alignment and subtree alignment. MT experiments on large data set show that our rule extraction approach gets close to the state-of-the-art system in translation accuracy, while significantly reducing the rule table size by an order of magnitude. Analysis of the extracted rules also shows its clarity and expressiveness.
Our future work includes more refined subtree alignment models, recovery of missing structures, as well as more selected syntactic features for decoding. We believe that these efforts will help fully exploit the potentials of tree-based MT models.
Ayan, N. F. and B. J. Dorr


Brown, P. F., V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer


Chiang, D.


Chiang, D.


DeNero, J. and D. Klein

Deng, D. and N. Xue


Dorr, B. J.


Galley, M., J. Graehl, K. Knight, D. Marcu, S. DeNeefe, W. Wang, and I. Thayer


Groves, D., M. Hearne, and A. Way


Haghighi, A., J. Blitzer, J. DeNero, and D. Klein


Koehn, P., H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan,

Koehn, P., F. J. Och, and D. Marcu

Lavie, A., A. Parlikar, and V. Ambati

Liang, P., B. Taskar, and D. Klein

Liu, D. and D. Gildea

2014. The stanford corenlp natural language processing toolkit.

Quirk, C. and A. Menezes
2006. Do we need phrases?: challenging the conventional wisdom in statistical machine

Stolcke, A. et al.


Sun, J., M. Zhang, and C. L. Tan


Vogel, S., H. Ney, and C. Tillmann


Xiao, T. and J. Zhu


Xiao, T., J. Zhu, H. Zhang, and Q. Li