Constrained Word Alignment Models for Statistical Machine Translation

Yanjun Ma

B.A., M.A.

A dissertation submitted in fulfilment of the requirements for the award of

Doctor of Philosophy (Ph.D.)

to the

Dublin City University
School of Computing

Supervisor: Prof. Andy Way

September 2009

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Ph.D. is entirely my own work, that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: Yanjun Ma

(Candidate) ID No.: 55156665

Date: September 15th, 2009

Contents

List of Figures

List of Tables

Abstract

Word alignment is a fundamental and crucial component in Statistical Machine Translation (SMT) systems. Despite the enormous progress made in the past two decades, this task remains an active research topic simply because the quality of word alignment is still far from optimal. Most state-of-the-art word alignment models are grounded on statistical learning theory treating word alignment as a general sequence alignment problem, where many linguistically motivated insights are not incorporated. In this thesis, we propose new word alignment models with linguistically motivated constraints in a bid to improve the quality of word alignment for Phrase-Based SMT systems (PB-SMT). We start the exploration with an investigation into segmentation constraints for word alignment by proposing a novel algorithm, namely word packing, which is motivated by the fact that one concept expressed by one word in one language can frequently surface as a compound or collocation in another language. Our algorithm takes advantage of the interaction between segmentation and alignment, starting with some segmentation for both the source and target language and updating the segmentation with respect to the word alignment results using state-of-the-art word alignment models; thereafter a refined word alignment can be obtained based on the updated segmentation. In this process, the updated segmentation acts as a hard constraint on the word alignment models and reduces the complexity of the alignment models by generating more 1-to-1 correspondences through word packing. Experimental results show that this algorithm can lead to statistically significant improvements over the state-of-the-art word alignment models. Given that word packing imposes "hard" segmentation constraints on the word aligner, which is prone to introducing noise, we propose two new word alignment models using syntactic dependencies as soft constraints. The first model is a syntactically enhanced discriminative word alignment model, where we use a set of feature functions to express the syntactic dependency information encoded in both source and target languages. One the one hand, this model enjoys great flexibility in its capacity to incorporate multiple features; on the other hand, this model is designed to facilitate model tuning for different objective functions. Experimental results show that using syntactic constraints can improve the performance of the discriminative word alignment model, which also leads to better PB-SMT performance compared to using state-of-the-art word alignment models. The second model is a syntactically constrained generative word alignment model, where we add in a syntactic coherence model over the target phrases in the context of HMM word-to-phrase alignment. The advantages of our model are that (i) the addition of the syntactic coherence model preserves the efficient parameter estimation procedures; and (ii) the flexibility of the model can be increased so that it can be tuned according to different objective functions. Experimental results show that tuning this model properly leads to a significant gain in MT performance over the state-of-the-art.

Acknowledgments

I would firstly like to express my utmost gratitude to my exemplar supervisor, Prof. Andy Way, for his guidance, suggestions and encouragement throughout my study in DCU. He always encourages me to explore various ideas I am interested in, offers experienced suggestions, creates opportunities for me to develop expertise, and keeps my research on the right track. From general research directions to concrete writing skills, it is his guidance that enables me to build up the ability and confidence to carry out the research work in this thesis as well as work in the future. He is without doubt the most important person that makes this piece of work possible!

Thanks to my first year advisor, Dr. Nicolas Stroppa, for helping me examine every detail of my research topics, having in-depth discussions and answering my questions on probability and machine learning. I learned so much from his strictness in mathematics and creative way of thinking. Many thanks to my second year advisor, Dr. Sylwia Ozdowska, for guiding me in identifying interesting research questions and offering invaluable suggestions.

Special thanks to Dr. Bill Byrne at the University of Cambridge for hosting me, offering meaningful discussions and excellent facilities for carrying out my research. Thanks to Jamie Brunning and Dr. Adrià de Gispert and all other colleagues in Engineering Department of Cambridge University for the kind help and support during my stay in Cambridge.

I give sincere thanks to Prof. Josef van Genabith for his enormous support of my study and various research activities, for his creative comments on my PhD work that inspires me to carry out more explorations, and all the meaningful and enjoyable discussions on various research topics. I would also like to thank Prof. Harold Somers and Dr. Gareth Jones for their careful review on my earlier PhD research and offering critical and crucial suggestions. Thanks to Dr. Haifeng Wang at Toshiba Research & Development Centre (China) for his kind support and guidance on my research direction. I would also like to thank my Master thesis supervisor, Prof.

Ying Liu at Tsinghua University, for her continual encouragement throughout my PhD study.

I owe my thanks to my colleagues at the National Centre for Language Technology and Centre for Next Generation Localisation for all their help and support in the past few years. Particular thanks go to Patrick Lambert for offering experienced suggestions on my research, Yuqing Guo for answering innumerous technical questions, Mary Hearne and Sara Morrissey for their advice on thesis management, and Declan Groves, Hany Hassan and John Tinsley for their meaningful inputs during the course of my research. I must thank Joachim Wagner for always having answers to my technical questions, and the NCLT and CNGL management team for offering all the fantastic facilities for my study.

I would like to thank Science Foundation Ireland for their generous funding of this research which allows me to stand at the forefront of this research area. My sincere gratitude also extends to the Irish people in general for their warm heart during my stay in such a beautiful country. I give a whole-hearted thank you to the friends from my homeland China, for sharing the excitement as well as the disappointment, and making my life here so enjoyable.

My deep gratitude goes to my parents and family for always being there on my side and offering their tremendous support in every aspect of my life. Finally and most importantly, thanks to my beloved wife Yanli Sun for accompanying me wherever I go, sharing my happiness, moments of bitterness and making my life exciting, enjoyable and complete.

Acronyms

MT	Machine Translation
SMT	Statistical Machine Translation
PB-SMT	Phrase-Based Statistical Machine Translation
POS	Part-of-Speech
M1	IBM Model 1
M2	IBM Model 2
M3	IBM Model 3
M4	IBM Model 4
MERT	Minimum Error-Rate Training
HMM	Hidden Markov Model
H	HMM word-to-word alignment model
SH	HMM word-to-phrase alignment model
SSH1	Syntactically constrained HMM word-to-phrase alignment model 1
SSH2	Syntactically constrained HMM word-to-phrase alignment model 2
ITG	Inversion Transduction Grammar
AER	Alignment Error Rate
GDF	Grow-Diag-Final
GALE	Global Autonomous Language Exploitation
IWSLT	International Workshop on Spoken Language Translation
NIST	National Institute of Standards and Technology
Nist	Evaluation metric developed at NIST
MTC	Multiple-Translation Chinese corpus
SVM	Support Vector Machine
CoNLL	Conference on Computational Natural Language Learning
LDC	Linguistic Data Consortium
ICT	Institute of Computing Technology, Chinese Academy of Sciences
CS	Character Segmentation

Notations

Probabilities

P	General probability distribution with (almost) no specific assumptions
$p(\cdot)$	Model-based marginal distribution
$p(\cdot;\cdot)$	Model-based marginal distribution with specified parameters
$p(\cdot\|\cdot)$	Model-based conditional distribution
$p(\cdot\|\cdot;\cdot)$	Model-based conditional distribution with specified parameters
p_θ	Model-based distribution with unknown parameters θ
$\tilde{p}$	Maximum likelihood estimation

Data Representation

$\mathbf{T}$	Bilingual corpus
$\mathbf{f}$	Source sentence
$\mathbf{e}$	Target sentence
$(\mathbf{f},\mathbf{e})$	Bilingual sentence pair
f_1^I	Source sentence containing I tokens
e_1^J	Target sentence containing J tokens
f_i	i^{th} source word
e_j	j^{th} target word
v_k	k^{th} phrase in a sentence
$\bar{f}_k$	k^{th} phrase in a source sentence
$\bar{e}_k$	k^{th} phrase in a target sentence
ϕ_k	Number of words in k^{th} phrase
$v_k[m]$	m^{th} word in phrase v_k

Alignment Representation

(j,i)	Alignment link between target word e_j and source word f_i
(e_j,f_i)	Alignment link between word e_j and f_i
(f_{a_k},v_k)	A set of links connecting source word f_{a_k} and target phrase v_k

$\mathbf{a}$	Word alignment
a_1^K	Word-to-phrase alignment
a_1^J	Target-to-source word-to-word alignment
a_1^I	Source-to-target word-to-word alignment
$\mathcal{G}$	Gold-standard word alignment

Syntactic Dependencies

$T_{\mathbf{f}}$	Source dependency tree
$T_{\mathbf{e}}$	Target dependency tree
r	Dependency type (label)
$R_{\mathbf{f}}$	Set of source dependency types
$R_{\mathbf{e}}$	Set of target dependency types
$\langle f_i, r_{\mathbf{f}}, f_{i'} \rangle$	Dependency between f_i and $f_{i'}$ with dependency label $r_{\mathbf{f}}$
$\langle e_j, r_{\mathbf{e}}, e_{j'} \rangle$	Dependency between e_j and $e_{j'}$ with dependency label $r_{\mathbf{e}}$

Syntactically Enhanced Discriminative Models

Δ	Set of anchor word indices
$\bar{\Delta}$	Set of non-anchor word indices
D	Relative distortion of a link with respect to anchor links
A_{Δ}	Set of anchor word alignments
$A_{\bar{\Delta}}$	Set of non-anchor word alignments
p_d	Relative distortion distribution
p_{Δ}	Anchor alignment distribution
$p_{\bar{\Delta}}$	Non-anchor alignment distribution

HMM Word-to-Phrase Model Components

η	Phrase count parameter
ζ	Syntactic coherence parameter
ε	Kronecker function for word insertion
$\langle j, i, \phi \rangle$	A cell in a source-to-target alignment Forward (Backward) trellis
$\alpha_j(i, \phi, \varepsilon)$	Forward statistics in HMM
$\beta_j(i, \phi, \varepsilon)$	Backward statistics in HMM
$\gamma_j(i, \phi, \varepsilon)$	Posterior emission distribution
$\xi_j(i', \phi', \varepsilon', i, \phi, \varepsilon)$	Posterior transition distribution
p_n	Phrase length model
p_ε	Word insertion distribution
p_a	Transition distribution
p_0	Transition probability into NULL
p_r	Syntactic coherence model
p_v	Phrase translation distribution
p_{t_1}	Unigram lexical translation distribution
p_{t_2}	Bigram lexical translation distribution

Others

$c(\cdot)$	Count
t	Threshold parameter
h	Feature functions in a log-linear framework
λ	Feature weight

Chapter 1

Introduction

Automatic word alignment can be defined as the problem of determining translation correspondences at word level given a parallel corpus of aligned sentences. As a fundamental component, word alignment can be applied to various multilingual Natural Language Processing applications including translation lexicon induction (Melamed, 1996; Lin et al., 2008) and cross-lingual projection of linguistic information (Hwa et al., 2002). Our focus application in this thesis is Machine Translation (MT).

MT is one of the most important tasks in Natural Language Processing. Over the last few decades, MT research and application broadly falls into two paradigms. The first one is referred to as rule-based MT, in which linguistic knowledge is expressed through manually crafted rules. The other one is data-driven MT, where linguistic knowledge is automatically derived from large bilingual corpora annotated on different levels. Data-driven MT is by far the predominant research paradigm; particularly, Statistical Machine Translation (SMT), an example of data-driven MT with well-formed mathematical foundations, has been repeatedly demonstrated in the last twenty years to be an effective solution to the problem of translation. The popularity of SMT can be explained by pointing out several additional features:

- **Speed of deployment**. As opposed to rule-based MT systems, which are time-consuming to build and difficult to maintain on a consistency basis, SMT systems can be produced quickly since their underlying models can be automatically derived from corpora using different machine learning techniques;

- **Adaptability**. SMT models are language-independent, meaning that they can be easily constructed for different language pairs as long as bilingual corpora are available, as opposed to the rule-based systems, which are built upon the specific grammars of the particular languages in question;
- **Low production cost**. SMT systems do not rely on expensive linguistic expertise.

The development of SMT systems started with word-based models (Brown et al., 1993; Germann, 2003), where "words" are the basic translation unit and word ordering is a major problem. Phrase-Based SMT (PB-SMT) came into being by using a sequence of words (a "phrase") as the basic translation unit to partially mitigate the word ordering problem, leading to a major improvement in translation quality. Hierarchical (Wu, 1997; Chiang, 2005) and syntax-based models (Yamada and Knight, 2001; Quirk et al., 2005; Galley et al., 2006; Liu et al., 2006; Marcu et al., 2006) tackle this problem using grammatical structures either automatically induced (hierarchical models) or through syntactic parsers (syntax-based models) to guide translation, and transform a translation problem into a parsing problem. Some efforts are dedicated to the syntactic extensions to PB-SMT systems (Hassan et al., 2007b, 2008), which can be viewed as an intermediate form between PB-SMT and syntax-based models.

Word alignment is a fundamental component in all above-mentioned SMT systems. High-quality word alignment is essential to PB-SMT systems in order to extract a set of reliable phrase pairs; it is also important for hierarchical or syntax-based systems in order to obtain wide-coverage high-quality translation rules. Given this, a large body of research is dedicated to improving word alignment. Besides the quality, another important issue regarding word alignment, which is particularly relevant in the context of SMT, is flexibility. Since we need a set of word alignments which is not only believed to be of high quality by human judges, but also can lead to high SMT performance, the flexibility of the alignment model is crucial so that it can be tuned for different language pairs and/or different types of SMT systems. To summarise, quality together with flexibility are the two measures for judging a word alignment approach.

1.1 Research Questions

The quality of word alignment produced by state-of-the-art alignment methods is still far from optimal. One direction to improve the quality of word alignment is to improve the alignment modelling through the incorporation of linguistically motivated insights.

As we know, most word alignment models normally require that the sentences to be aligned are segmented into sequences of tokens that are meant to be words. The way to segment a sentence can significantly influence the performance of word alignment given that the basic alignment units differ from one segmentation to another, which can lead to dramatic variations in the alignment structure. For example, one of the most difficult parts in word alignment is the case of 1-to-n alignment, i.e. where one word in one language is aligned to n words in another. By uncovering the dependencies[1] between the n words, we can group these words into one "word" so that the alignment process can be simplified.[2] In other words, we can use the bilingually optimal segmentation as a (hard) constraint on the alignment models in a bid to improve the performance of the word alignment process, and subsequently PB-SMT systems. This gives rise to our first research question.

[1] The dependencies between words can be expressed either in the form of frequency co-occurrences in real world data, or linguistically in the form of syntactic dependencies, which can be obtained via dependency parsing.

[2] It is natural to argue that this grouping can increase the vocabulary size and consequently suffer from data sparseness. A critical reader may also raise the point that a group of words that frequently co-occur in one language, e.g. collocations, does not necessarily imply that these words translate into a single word in another language. As answers to these questions, we will present a method that overcomes these problems, and other relative issues will also be discussed in Chapter 3.

(RQ1) Can bilingually motivated word segmentation improve word alignment and PB-SMT?

To discover the optimal segmentation and to directly use them in word alignment imposes hard constraints on the alignment process because once we believe the n words are dependent on each other and correspond to one word in another language, we group the n words as one single word, where noisy groupings can also unavoidably occur. Based on such observations, we can exploit new models to incorporate segmentation information as soft constraints. One form of association between words is expressed by syntactic dependencies. Despite such associations not necessarily implying a clear alignment decision, we can make use of the syntactic dependencies as soft constraints in the alignment learning process and let the model decide to what extent such information can be utilised. In order to investigate the influence of syntactic dependencies on different word alignment models, such as generative and discriminative models, we try to answer the following two research questions:

(RQ2) Can discriminative word alignment models be enhanced by syntactic dependencies?

(RQ3) Can we extend generative word alignment models to incorporate syntactic constraints?

As mentioned earlier, besides the importance of quality, the flexibility of the alignment model is also essential in the context of SMT, where we prefer to design our model in such a manner that it can be tuned for different end tasks. Identifying an interface in alignment models for tuning purposes gives rise to our final research question:

(RQ4) Can we tune the word alignment methods to achieve higher MT performance?

1.2 Thesis Structure

The following chapters in the thesis are dedicated to addressing the four research questions by inclusion of an overview of the previous research on word alignment, development of new algorithms and models and presentation of the experimental results.

Chapter 2 gives a brief overview of state-of-the-art SMT systems and illustrates the crucial role of word alignment in these systems. We then conduct a critical review of the various approaches and models for word alignment and show how our research is motivated with respect to the state-of-the-art. The methodology underlying our research is briefly introduced at the end of this chapter.

Chapter 3 describes an algorithm to perform bilingually motivated word segmentation in a bid to bootstrap the word alignment used for PB-SMT systems. Via a set of experiments we show that a careful balance between vocabulary size and the reduction of 1-to-n alignments can lead to a significant improvement in the performance of a PB-SMT system. We also show that in the scenario where the vocabulary size is limited, monolingual segmentation can be fully abandoned and replaced with our bilingually motivated segmentation approach.

Chapter 4 seeks to improve a discriminative word alignment model by incorporating syntactic dependencies. We take advantage of the dependency structures of both source and target languages and use the bilingual dependency correspondences as soft constraints in word alignment. Experimental results confirm our hypothesis that word alignment can be improved through the incorporation of syntactic dependency information. We also investigate the significance of word alignment tuning and show that tuning word alignment directly according to the translation quality can boost the performance of PB-SMT systems.

Chapter 5 presents our syntactically constrained generative word alignment model, i.e. HMM word-to-phrase alignment model. We extend the standard word-to-phrase alignment model to efficiently incorporate syntactic dependencies into alignment. At the same time, we design the model in a manner such that it can be tuned according to different objective functions.

Chapter 6 continues to investigate the syntactically constrained generative word alignment model through the presentation of the experimental results and an in-depth analysis of various model configurations. Following an examination of the alignment structure, we show the advantages and disadvantages of using this model. Extensive experiments on using our word alignment in PB-SMT systems are also conducted.

Chapter 7 concludes this thesis and points out avenues for further research.

Part of the research presented in the thesis has been published in peer-reviewed international conferences and journals. Ma et al. (2007b) presented our algorithm to bootstrap word alignment through bilingually motivated word packing. Ma and Way (2009a) generalised this algorithm and applied it to direct Chinese word segmentation and domain adaptation. This strand of research was summarised and further extended in Ma and Way (2009b). Ma et al. (2008a) described our primitive model for syntactically enhanced discriminative word alignment. This model was refined in Ma et al. (2009a) and tuning the word alignment for PB-SMT was also discussed. A further investigation into the characteristics of the alignment that benefits the PB-SMT systems in translation quality was conducted in Lambert et al. (2009). These alignment techniques have also been extensively exploited in various MT evaluation campaigns (Hassan et al., 2007a; Ma et al., 2008b; Tinsley et al., 2008; Ma et al., 2009b).

There are a bunch of other papers that do not feature much in this work. These include using word alignment for chunking (Ma et al., 2007a), using word alignment information between source words and MT outputs in Hypothesis Alignment for Combining Outputs from Machine Translation Systems (Du et al., 2009) and incorporating supertags as source-side contexts in PB-SMT systems (Haque et al., 2009).

Chapter 2

Overview of Word Alignment Models and Our Methodology

In this chapter, we first introduce the fundamentals of SMT systems and illustrate the role of word alignment in such systems. State-of-the-art word alignment models and related research concerning the impact of word alignment on SMT systems are subsequently reviewed. Then we elaborate the methodology underlying our research, including the evaluation methods, the data and baseline systems we use throughout this thesis.

2.1 Statistical Machine Translation

Given a source ("Foreign") sentence $f_1^I = f_1, ..., f_j, ...f_I$, which is to be translated into a target ("English") sentence $e_1^J = e_1, ..., e_i, ..., e_J$, among all the possible target sentences, we will choose the sentence with the highest probability as in (2.1):

$$\hat{e}_1^J = \operatorname*{argmax}_{e_1^J} P(e_1^J | f_1^I) \tag{2.1}$$

The argmax operation denotes the search problem, i.e. search for the target sentence that holds the highest probability.

There are two widely used models that decompose the conditional probability shown in (2.1). One is the Source-Channel Model (Brown et al., 1990, 1993), with the other being the log-linear model (Och and Ney, 2002).

2.1.1 Source-Channel Model

According to Bayes' decision rule, we can perform the following maximisation as in (2.2):

$$\hat{e}_1^J = \operatorname*{argmax}_{e_1^J} P(f_1^I|e_1^J) \times P(e_1^J) \tag{2.2}$$

Equation (2.2) is the fundamental equation for SMT. $P(f_1^I|e_1^J)$ in the equation is the sentence translation model, which guarantees the target sentence e_1^J and source sentence f_1^I are translations of each other; $P(e_1^J)$ is the language model of the target language which ensures a fluent target sentence. Typically (2.2) is favoured over the direct translation model of (2.1) by yielding a modular approach, i.e. instead of modelling one probability distribution, we obtain two different knowledge sources that are trained independently. The optimal parameter values in these two models can be obtained by maximising the likelihood of the training data with respect to the model parameters.

Depending on the basic translation units used in translation, SMT has evolved from word-based SMT into Phrase-Based SMT (PB-SMT). Word-based SMT systems (e.g. (Germann, 2003)) learn lexical translation models describing word-to-word mappings between a given language pair. However, words are not the best atomic units of translation because we can have one-to-many mappings between languages. Furthermore, by translating word for word, no contextual information is made use of during the translation process. To attempt to overcome some of these issues, sequences of words can be translated together. By using these sequences of words (so-called "phrases", but not in the linguistic, "constituent" sense), it is possible to avoid many cases of translational ambiguity and better capture instances of local reordering. The set of phrase pairs extracted from the bilingual parallel corpus constitutes the core translation model (phrase table, or t(ranslation)-table) of the PB-SMT system.

Different from word-based models which can directly learn lexical translation from the training data, direct learning of phrase translations turns out to be a task with enormous computational complexity (Marcu and Wong, 2002). Therefore, state-of-the-art PB-SMT systems do not use the more mathematically grounded models to learn phrase translations in training. Instead phrases are extracted with respect to the word alignment based on some heuristics (Och, 2002).

2.1.2 Log-Linear Phrase-Based SMT

With a Markov assumption on the language model, PB-SMT decoding uses the the decision rule as in (2.3):

$$\hat{e}_1^J = \underset{e_1^J}{\operatorname{argmax}} \prod_{k=1}^{K} p_v(\bar{f}_k|\bar{e}_k) p_o(\text{start}_k - \text{end}_{k-1}) \prod_{j=1}^{J} p_{LM}(e_j|e_1 \cdots e_{j-1}) \tag{2.3}$$

where p_v is a phrase translation model indicating the translation probability from target phrase $\bar{e}_k$ to source phrase $\bar{f}_k$, and p_o is a distance-based phrase reordering model. The variable "start$_k$" is defined as the position of the first word of the foreign input phrase which translates to the k^{th} English phrase, and "end$_k$" as the position of the last word of the foreign phrase. Note that the process of segmenting the foreign sentence f_1^I into K phrases is not explicitly modelled, implying any segmentation is equally likely.[1] Note also that the translation of a source phrase does not depend on the translation of the surrounding source phrases, which is an inaccurate assumption.[2]

[1] Ma et al. (2007a) discussed this issue in the presentation and their alignment-guided chunking methods can be viewed as a form of modelling the segmentation process (cf. `http://www.mt-archive.info/TMI-2007-TOC.htm`).

[2] Stroppa et al. (2007), Carpuat and Wu (2007) and Haque et al. (2009) represent some attempts to address this problem.

It is natural to introduce weights to scale the contributions from each component, in which case we arrive at the updated decision rule in (2.4):

$$\hat{e}_1^J = \operatorname*{argmax}_{e_1^J} \prod_{k=1}^{K} p_v(\bar{f}_k|\bar{e}_k)^{\lambda_v} p_o(\text{start}_k - \text{end}_{k-1})^{\lambda_o} \prod_{j=1}^{J} p_{LM}(e_j|e_1 \cdots e_{j-1})^{\lambda_{LM}} \quad (2.4)$$

where λ_v, λ_o and λ_{LM} are the weights for phrase translation, distance-based reordering and language model respectively.

Using a simple logarithm transformation on (2.4), we have (2.5)–(2.7), which is used in state-of-the-art PB-SMT. Besides the distance-based reordering model p_o in (2.6), a more sophisticated lexical reordering model (Koehn, 2009) is normally adopted. As extensions to (2.5), bidirectional phrase translation probabilities are often included into this log-linear framework, i.e. not just $p_v(\bar{f}_k|\bar{e}_k)$ (the translation probability from target phrase to source phrase), but also $p_v(\bar{e}_k|\bar{f}_k)$ (the translation probability from source phrase to target phrase); lexical weighting, which measures the reliability of a phrase pair on lexical level, is also included to smooth the phrase translation probabilities (Koehn et al., 2003).

$$\begin{aligned}
\hat{e}_1^J = \operatorname*{argmax}_{e_1^J} \quad & \lambda_v \sum_{k=1}^{K} \log p_v(\bar{f}_k|\bar{e}_k) && (2.5)\\
+ \quad & \lambda_o \sum_{k=1}^{K} \log p_o(\text{start}_k - \text{end}_{k-1}) && (2.6)\\
+ \quad & \lambda_{LM} \sum_{j=1}^{J} \log p_{LM}(e_j|e_1 \cdots e_{j-1})\} && (2.7)
\end{aligned}$$

In principle, the phrase translation probability can be directly estimated via direct phrase alignment of the bilingual corpus (Marcu and Wong, 2002). However, the complexity of direct alignment of phrases on a bilingual corpus is enormous and exhaustively counting all possible phrase alignments over a corpus of reasonable size is infeasible in practice. Therefore, this strand of research normally constrains the phrase alignment model and the resulting results are not satisfactory (Birch et al., 2006).

One simple yet widely adopted practice is to induce the phrase alignment via word alignment results. The phrase pairs are induced in such a way that each word within a source phrase is aligned to a word in the target phrase and vice versa (Och, 2002). Since the core translation model is actually induced from word alignment, a high-quality word alignment is essential.

2.2 Statistical Word Alignment

There are several classes of methods for creating lexical correspondences given a bilingual sentence pair. The first body of research attempts to directly construct the word alignment on bilingual sentence pairs with notable methods such as the IBM Models (Brown et al., 1993). Another class of approaches generates word alignment in the process of aligning structures or tree representations of the bilingual sentences (Ding et al., 2003; Eisner, 2003; Gildea, 2003; Tinsley et al., 2007). This approach aims to produce an alignment between constituents (or sentence substructures), and word alignment can be viewed as a byproduct of this process. A third group of research is bilingual parsing (Wu, 1997; Alshawi et al., 2000). We regard this approach as an intermediate form of the above-mentioned two classes in that it permits the probabilistic trade-off between lexical correspondences and the amount of information present in the monolingual parses.

Our research roughly falls into the first strand of research. Therefore, we only give a literature review on this strand of research. In this section, we will review three different models for word alignment, namely generative models, discriminative models and association-based models. Throughout this thesis, we use the term **alignment** to indicate the entire structure that connects a sentence pair and the term **link** to denote individual word-to-word connections that make up an alignment. When we talk about different types of links, e.g. a 1-to-1 link referring to the case where one word in the source sentence is connected with exactly one word in the target sentence, and 1-to-n links referring to the case where one word in the source sentence is connected with n words in the target sentence, we (loosely) use the term link and alignment interchangeably, i.e. 1-to-1 alignment means 1-to-1 link, and 1-to-n alignment means 1-to-n links.

2.2.1 Generative Models

The most common approach to word alignment is that of **generative** word alignment models, which view the translation (alignment) process as a sentence in one language generating a sentence in another language. Relating word alignment to the SMT translation model in (2.2), the translation model can be recast as a (statistical) alignment model as in (2.8), which assumes the source sentence f_1^I is generated by the target sentence e_1^J:

$$P(f_1^I|e_1^J) = \sum_{a_1^I} P(f_1^I, a_1^I|e_1^J) \tag{2.8}$$

where a_1^I is the word alignment with each link $i \rightarrow j = a_i$ denoting the association between a source position i and target position $j = a_i$. Normally there is a link $a_i = 0$ to account for the case where a source word is aligned to an empty target word e_0 (or NULL). In principle, word alignment should encode an arbitrary relation between source and target words, i.e. alignment $\mathcal{A}$ should be defined as a subset of the Cartesian product of source and target word postions, as in (2.9):

$$\mathcal{A} \subseteq \{(j, i) : j = 1 \cdots, J; i = 1 \cdots, I\} \tag{2.9}$$

However modelling the alignment to deal with this general representation is hard (Och and Ney, 2003), mainly due to the fact that this representation leads to an exponentially large alignment space, of which an exhaustive exploitation is infeasible in practice (Cherry and Lin, 2006a). Therefore, most models including (2.8) constrain the alignment space in such a way that each source word can only be aligned to exactly one target word. Therefore, using the representation a_1^I, the alignment does not encode a relation between source and target word positions, but only a mapping from source to target word positions (Och and Ney, 2003). For these reasons, these models are also called **asymmetric** models.

For the convenience of understanding, our notation throughout the rest of the thesis assumes the generation of a target language sentence e_1^J from a source sentence f_1^I as opposed to the SMT translation model, which assumes the source sentence f_1^I is generated by target sentence e_1^J.[3] Thereafter, the alignment a_1^J represents a mapping from the target word positions to the source. The transformation process from source to target language covered by the generative process may include word insertion or deletion, word reordering (or **distortion**) indicating the relative position change when generating a target word from a source word, the **fertility** of a source word to account for the one-to-many generation (1-to-n alignments in alignment models), etc (Brown et al., 1990, 1993). Depending on whether fertility is explicitly modelled or not, these generative models can be broadly classified into non-fertility models and fertility-based models.

The most widely used non-fertility models are HMM-based models. Depending on the order of HMM used, there are first-order and zero-order HMMs. IBM Models 1 and 2 (Brown et al., 1993) are zero-order HMM models assuming a generative process as follows: a source position is firstly selected for each position in the target sentence, and a target word is produced as the translation of the selected source word. In IBM Model 1, the source position is selected uniformly, while in IBM Model 2 the selection depends on the target position in question. The first-order HMM model (Vogel et al., 1996) refines the generative process by further assuming that the selection of a source position depends on the previously selected source position.

In a generalised HMM model, the alignment model $P(e_1^J, a_1^J | f_1^I)$ can be written

[3] As a matter of fact, state-of-the-art word alignment (Och and Ney, 2003; Koehn et al., 2003) performs bidirectional word alignment. Therefore, the distinction between source and target does not influence the results of the word alignment process.

as in (2.10) and (2.11):

$$\begin{aligned} P(e_1^J, a_1^J, |f_1^I) &= P(J|f_1^I) \times \prod_{j=1}^{J} P(e_j, a_j | e_1^{j-1}, a_1^{j-1}, f_1^I) && (2.10) \\ &= P(J|f_1^I) \times \prod_{j=1}^{J} P(a_j | e_1^{j-1}, a_1^{j-1}, f_1^I) \times P(e_j | e_1^{j-1}, a_1^J, f_1^I) && (2.11) \end{aligned}$$

where we obtain three different distributions: a length distribution $P(J|f_1^I)$, a transition distribution $P(a_j|e_1^{j-1}, a_1^{j-1}, f_1^I)$ and a translation distribution $P(e_j|e_1^{j-1}, a_1^J, f_1^I)$. Normally, we assume a simplified length distribution, a first-order dependence for the alignment a_j, and that the conditioning of the lexical translation distribution is only the source word at position a_j so that the three above-mentioned models can be re-written as in (2.12)–(2.14).

$$\begin{aligned} P(J|f_1^I) &= p_l(J|I) && (2.12) \\ P(a_j|e_1^{j-1}, a_1^{j-1}, f_1^I) &= p_a(a_j|a_{j-1}, I) && (2.13) \\ P(e_j|e_1^{j-1}, a_1^J, f_1^I) &= p_t(e_j|f_{a_j}) && (2.14) \end{aligned}$$

A standard HMM word alignment model is based on first-order dependencies as in (2.13) for the transition distribution, whereas IBM Models 1 and 2 are based on zero-order dependencies:

- IBM Model 1 has a uniform **reverse distortion** distribution $p_d(a_j|j, I, J) = 1/(I+1)$,[4] which is put together with a simple length distribution and a lexical translation distribution as used in the first-order HMM model. The alignment

[4] This model is called a reverse distortion because it models the relative position change from a target word to a source word. In other words, it is a conditional distribution of source postions a_j conditioned on target positions j, as opposed to the distortion model in IBM Model 3 introduced in the following, which is a conditional distribution of target position j conditioned on source position i. Note also that the conditioning in IBM Model 1 includes the length of both source and target sentences.

model according to IBM Model 1 is shown in (2.15):

$$P(e_1^J, a_1^J|f_1^I) = \frac{p_l(J|I)}{(I+1)^J} \times \prod_{j=1}^{J} p_t(e_j|f_{a_j}) \tag{2.15}$$

- For IBM Model 2, the reverse distortion model $p_d(a_j|j, I, J)$ is estimated from the training data. Therefore, we obtain (2.16):

$$P(e_1^J, a_1^J|f_1^I) = p_l(J|I) \times \prod_{j=1}^{J} [p_d(a_j|j, I, J) p_t(e_j|f_{a_j})] \tag{2.16}$$

However, non-fertility models are generally considered to be relatively weak models, mainly because of the simplicity of the generation process. Some further research has been conducted on improving IBM Model 1 (Moore, 2004) and a particularly large body of research has been carried out to improve the first-order HMM model (Toutanova et al., 2002; Lopez and Resnik, 2005; Liang et al., 2006; Deng and Gao, 2007; Ganchev et al., 2008). This line of research shares the insight that HMM models can be improved by imposing well-motivated constraints on them. Toutanova et al. (2002) and Deng and Gao (2007) introduced some extensions to the original HMM models to better handle the irregularities in the word alignment process; Toutanova et al. (2002) also proposed the addition of "staying" probability to approximately model the "fertility" phenomena. Both Liang et al. (2006) and Ganchev et al. (2008) added constraints into HMM training by enforcing the two asymmetric alignment models to agree, even if the objective function differed. Lopez and Resnik (2005) proposed a syntax-rich transition distribution to replace the standard HMM transition distribution in a resource-scarce scenario, and DeNero and Klein (2007) constrained the HMM training using target language constituent structure in the scenario of translation rule extraction for syntax-based SMT.

Fertility-based alignment models, most notably IBM Models 3 and 4, are more complicated by introducing fertility into the alignment model, which assumes a different generation process. These models first decide how many target words each source word should generate, i.e. determining the source word fertility. For each source word, a specific number of target words will be produced as the translation of the source word according to its fertility. These models then arrange the hypothesised target words to produce a target string according to the **distortion** models, which model the relative position change from a source word to the target words it generates. IBM Model 3 utilises a zero-order distortion model, i.e. each target position is chosen independently for the target words generated by each source word. IBM Model 4 utilises a simplified first-order dependency in positioning the target words.[5] Formally, given a source word f_i which generates ϕ_i target words, we use A_i to denote the positions of the ϕ_i target words. The alignment A between the a source sentence f_1^I and a target sentence e_1^J can be defined as in (2.17):

$$A : i \rightarrow A_i \subset \{1, \cdots, j, \cdots, J\} \tag{2.17}$$

where an important constraint is that all the target positions must be covered exactly once, i.e. A_i have to form a partition of the set $\{1, \cdots, j, \cdots, J\}$. The number of words in A_i is the fertility of source word f_i. The word alignment according to these models use the following decomposition as in (2.18)-(2.20):

$$\begin{aligned} P(e_1^J, a_1^J | f_1^I) &= P(e_1^J, A_0^I | f_1^I) && (2.18) \\ &= P(A_0 | A_1^I) \times \prod_{i=1}^{I} P(A_i | A_1^{i-1}, f_1^I) \times P(e_1^J | A_0^I, f_1^I) && (2.19) \\ &= p(A_0 | A_1^I) \times \prod_{i=1}^{I} p(A_i | A_{i-1}, f_i) \times \prod_{i=0}^{I} \prod_{j \in A_i} p(e_j | f_i) && (2.20) \end{aligned}$$

where A_0 contains the positions of target words that are aligned to the empty (NULL) word f_0, $p(e_j|f_i)$ is a lexical translation distribution and $p(A_i|A_{i-1}, e_i)$ can be decomposed into fertility and distortion distributions with respect to different IBM Models. For example, according to IBM Model 3, $p(A_i|A_{i-1}, f_i)$ is decomposed

[5] Note that the first-order dependencies are built between the target word positions in IBM Model 4 as opposed to source word positions in HMM models.

as in (2.21):

$$p(A_i|A_{i-1}, f_i) \quad = \quad p(\phi_i|f_i) \times \phi_i! \times \prod_{j \in A_i} p(j|i, J) \tag{2.21}$$

where $p(\phi_i|f_i)$ is a fertility distribution and $p(j|i, J)$ is a zero-order distortion distribution.

The Distortion models in both IBM Model 3 and 4 assign probability to invalid target strings in order to achieve a simplified approximation, resulting in the problem of "deficiency". IBM Model 5 is a reformulation of Model 4 with a suitably refined distortion model to avoid deficiency. However, for these models, we are unaware of any efficient training or search algorithm. Consequently, it can only be implemented by approximate, hill-climbing methods and parameter estimation can be very slow, memory-intensive and difficult to parallelise. Given this, Deng and Byrne (2005) proposed an HMM-based word-to-phrase alignment model which explored the desirable features in IBM fertility-based models while keeping the parameter estimation step tractable. This approach will be revisited and extended in Chapter 5.

Moreover, the generative models described above face a degree of criticism over the fact that they make unreasonable assumptions about word alignment structure, namely the 1-to-n assumption, meaning that each source word can be aligned to zero or more target words (or each target word can be aligned to exactly one source word), but not vice versa. Such an asymmetric alignment structure cannot capture the pervasive m-to-n alignments in real world alignment tasks. Consequently, heuristics are needed to "symmetrise" the alignments using bidirectional word alignment in order to produce high-quality phrase pairs for PB-SMT systems, or translation rules for syntax-based SMT. Fraser and Marcu (2007a) attempted to address such a problem by proposing a new generative model capturing m-to-n alignment structures. A consequence of this attempt is that the training process becomes more complicated and more approximations are required.

In general, generative models have been shown to be powerful in their modelling capabilities and they are able to produce high-quality alignments with successful application to various types of SMT systems. A thorough comparison between various generative word alignment models can been found in Och and Ney (2003). Some successful implementations include GIZA++[6] (Och and Ney, 2003), an implementation of HMM models and IBM Model 4, and MTTK[7] (Deng and Byrne, 2006), an implementation of HMM word-to-phrase alignment models (Deng and Byrne, 2005, 2008). The state-of-the-art PB-SMT system MOSES[8] (Koehn et al., 2007) also includes a set of scripts to perform various symmetrisations of the bidirectional word alignments.

Training

The Expectation Maximisation (EM) algorithm (Dempster et al., 1977) can be used to find the maximum likelihood estimates to problems where the value of some variables are not directly observed, providing that the general form of the probability distribution governing these variables are known. For word alignment tasks in the context of SMT, we seek to optimise the unknown parameters θ associated with the particular alignment distributions. Given a parallel corpus $\mathbf{T}$ consisting of $|\mathbf{T}|$ sentence pairs $(\mathbf{f}, \mathbf{e})$, we aim to find the parameters θ that maximise the likelihood of the parallel training corpus, as shown in (2.22)

$$\hat{\theta} = \arg\max_{\theta} \prod_{(\mathbf{f},\mathbf{e})\in\mathbf{T}} p_\theta(\mathbf{e}|\mathbf{f}) = \arg\max_{\theta} \prod_{(\mathbf{f},\mathbf{e})\in\mathbf{T}} \sum_{\mathbf{a}} p_\theta(\mathbf{e},\mathbf{a}|\mathbf{f}) \tag{2.22}$$

For IBM Model 1, the parameter θ only contains parameters in the lexical translation distribution and IBM Model 2 has an additional parameter for the reverse distortion distribution. In the E-step of IBM Model 1, the lexical translation counts

[6] http://www.fjoch.com/GIZA++.html

[7] http://mi.eng.cam.ac.uk/~wjb31/distrib/mttkv1

[8] http://www.statmt.org/moses

$c(f,e;\mathbf{f},\mathbf{e})$ for one sentence pair ($\mathbf{f}$,$\mathbf{e}$) are calculated as in (2.23):

$$c(f,e;\mathbf{f},\mathbf{e}) = \sum_{\mathbf{f},\mathbf{e}} c(\mathbf{f},\mathbf{e}) \sum_{\mathbf{a}} P(\mathbf{a}|\mathbf{f},\mathbf{e}) \sum_{j} \delta(e,e_j)\delta(f,f_{a_j}) \tag{2.23}$$

Here, $c(\mathbf{f},\mathbf{e})$ is the count of the sentence pair ($\mathbf{f}$,$\mathbf{e}$) in the parallel corpus. In the M-step, the lexical translation probability is re-estimated as in (2.24):

$$p_t(e|f) = \frac{\sum_{(\mathbf{f},\mathbf{e})} c(f,e;\mathbf{f},\mathbf{e})}{\sum_{(\mathbf{f},\mathbf{e})} \sum_{f'} c(f',e;\mathbf{f},\mathbf{e})} \tag{2.24}$$

From (2.23), the E-step requires a summation over all $(I+1)^J$ alignments where explicitly enumerating all the alignments is infeasible. Fortunately both model 1 and 2 have a particularly simple mathematical form such that the EM algorithm can be implemented efficiently. For the first-order HMM model, the Baum-Welch algorithm (Baum, 1972), a version of the EM algorithm, can be used. As mentioned earlier, we are unaware of any efficient algorithm for the parameter estimation of fertility-based IBM Models.

Moreover, the more sophisticated IBM Models 3 and 4 are normally trained incrementally by using the parameters of simpler models. One of the most widely adopted training procedures is 5 iterations of Model 1, 5 iterations of HMM, followed by 3 iterations of Model 3, and 3 iterations of Model 4. The first iteration of IBM Model 1 assumes that the component distributions are uniform, and the first iteration of HMM uses the parameters yielded from the fifth iteration of IBM Model 1. Similarly, the first iteration of IBM Model 3 uses the parameters from the fifth iteration of HMM, and so on and so forth.

Search

There are generally two different search methods for finding the best alignment under a particular parameter setting of a particular model. One widely used method is Viterbi search (Viterbi, 1967), with the resulted alignment called **Viterbi align-**

ment. The decision rule is shown in (2.25):

$$\hat{a}_1^J = \arg\max_{a_1^J} p_{\hat{\theta}}(e_1^J, a_1^J | f_1^I) \tag{2.25}$$

For IBM Models 1 and 2, the computation of Viterbi alignment can be accomplished with dynamic algorithms of complexity $O(I \cdot J)$ and for HMM models with complexity $O(I^2 \cdot J)$ (Vogel et al., 1996).

However, for fertility-based models, where the corresponding search problem is NP-complete (Knight, 1999), efficient algorithms for finding the Viterbi alignment do not exist to the best of our knowledge. A greedy search algorithm suggested by Brown et al. (1993) is refined and implemented in GIZA++. The basic idea is to compute the Viterbi alignment of simple models (such as IBM Model 2 or HMM). The alignment is then iteratively improved with respect to the alignment probability of fertility-based models (Och and Ney, 2003).

An alternative to Viterbi alignment search is posterior decoding, where we compute the posterior probability that a source word f_i is aligned to target word e_j under some model. If the posterior probability is above a predefined threshold, we include the link between f_i and e_j into our final alignment. This search method is widely used for HMM word alignment models (Liang et al., 2006; Ganchev et al., 2008).

2.2.2 Discriminative Models

Discriminative word alignment models came into being with the specific intention of overcoming the shortcomings faced by generative models by directly modelling the alignment between source and target sentences. As described in Section 2.2.1, generative alignment approaches model $p(e_1^J, a_1^J | f_1^I)$, where the alignment a_1^J is introduced as a hidden variable in the translation model and the alignment results can be viewed as an "artefact" of the translation process (Cherry and Lin, 2003). Different from generative models, discriminative models are trained by maximising

$p(\mathbf{a}|\mathbf{e},\mathbf{f})$, which corresponds to finding the Viterbi alignment in generative alignment models. Such models normally decompose $p(\mathbf{a}|\mathbf{e},\mathbf{f})$ into a log-linear combination of a set of features, enjoying the flexibility to incorporate various features encoded in the input data. For these models, a certain amount of **manually annotated word alignment** data (cf. Section 2.5.1) is required during training. Formally, the optimal alignment **a** is searched for by maximising a log-linear combination of a set of features, as shown in (2.26):

$$\hat{\mathbf{a}} = \arg\max_{\mathbf{a}} \sum_i \lambda_i h_i(f, \mathbf{a}, e) \tag{2.26}$$

The parameters λ_i can be learned in a supervised manner using various machine learning techniques including perceptron (Moore, 2005), maximum-entropy (Liu et al., 2005; Ittycheriah and Roukos, 2005), Support Vector Machines (SVM) (Taskar et al., 2005; Cherry and Lin, 2006b), Conditional Random Fields (CRF) (Blunsom and Cohn, 2006) etc.

The various discriminative models can be broadly classified into local models where the models discriminate candidate links for each source (target) word in training, and global models where the models discriminate all the possible alignment structures of a sentence pair. All models except Ittycheriah and Roukos (2005) described above are global models.

Despite the flexibility of incorporating various features, the need for a certain amount of annotated word alignment data is often subject to criticism since the annotation of word alignment is a highly subjective task. Moreover, parameters optimised on manually annotated data are not necessarily optimal for MT tasks. Fraser and Marcu (2007b) showed that Alignment Error Rate (AER) (Och and Ney, 2003), a widely used metric to measure word alignment quality by comparing the predicted alignment against manually annotated data, has a weak correlation with MT quality in terms of Bleu (Papineni et al., 2002) in a PB-SMT system. Therefore, some approaches have been proposed to optimise the parameters of discriminative models according to the MT task rather than with respect to some set of annotated data (Lambert et al., 2007). Some semi-supervised approaches have also been used to take advantage of both generative and discriminative approaches (Fraser and Marcu, 2006; Wu et al., 2006). However, we have not seen a discriminative word alignment model that can consistently outperform generative models when used for SMT. One possible reason is that these discriminative models are normally trained to minimise errors on the annotated data, which does not directly reflect an improvement in MT performance.

Training

As mentioned earlier, discriminative models can be trained using a variety of supervised machine learning techniques. Depending on the particular technique deployed, the training procedure may differ from one to another. For example, during the model training, the averaged perceptron and SVM algorithms only require one single best output to be inferred under current models. However, for maximum-entropy models (Jelinek, 1977), all possible alignments under current model are required, which is infeasible without making certain assumptions (e.g. a first-order Markov assumption over the alignment sequence, constraining that each target word can only be aligned to one source word) over the alignment structure (Blunsom and Cohn, 2006), and thereafter approximations of the whole alignment space are needed (Liu et al., 2005).

Search

Without making assumptions on the alignment structure, finding the optimal alignment according to many of the discriminative alignment models is non-trivial. Another factor that complicates the search is the non-decomposable feature functions deployed in the models (Moore et al., 2006). Among the above-described discriminative models, only the CRF model in Blunsom and Cohn (2006) enables efficient training and search using dynamic programming. This is achieved by balancing the complexity of the model and the use of different features. Other models use either greedy search (Liu et al., 2005) or beam search (Moore, 2005).

2.2.3 Association-Based Models

Another class of approaches to word alignment is that of **association-based models**, which obtain word alignments by using similarity functions to determine the association between source and target words (Smadja et al., 1996; Ker and Chang, 1997; Melamed, 2000). These models also directly model the alignments through a combination of a set of features as the discriminative alignment models do. However, they differ from discriminative alignment models in that the various features deployed for alignment are combined based on heuristics rather than a discriminative learning procedure. In this sense, association-based models are also called heuristics-based models. Richer syntactic information, including POS tags, chunk labels (Tiedemann, 2003; Ren et al., 2007) and dependency trees (Cherry and Lin, 2003) is deployed in recent development. A heuristics-based algorithm (Melamed, 2000; Tiedemann, 2003) or a greedy algorithm (Cherry and Lin, 2003; Ren et al., 2007) is often applied during the word alignment search process. The advantage of such approaches is their simplicity; however, the use of a similarity function would appear to be arbitrary and the performance of such methods is often inferior compared to the statistical approach of Och and Ney (2003).

2.3 Alignment Space

An alignment space determines the set of all possible alignments that can exist for a given sentence pair (Cherry and Lin, 2006a). Given a source sentence containing I words and the corresponding target sentence containing J words, the largest alignment space for this sentence pair has $2^{I \times J}$ possible alignments, which can be described as the case where each of the $I \times J$ potential links can be either on or off without restrictions. This space is too large for exhaustive exploitation for a sentence pair of reasonable lengths.

Therefore, most alignment models make further assumptions on the alignment structures in a bid to limit the alignment space. Melamed (2000) introduced an algorithm called "competitive linking" which enforces a 1-to-1 alignment constraint,[9] under which each token in the sentence pair can only participate in one link. Using this algorithm, the 1-to-1 constraint can be imposed by allowing each token in the source sentence to pick up a token from the target sentence to link to, which is then removed from the competition. By taking the NULL links (1-to-0) into account, the actual number of possible alignments lies between $J!(J \leq I)$ (or $\frac{J!}{(J-I)!}(J > I)$) and $(J+1)^I$. This space is called the **Permutation Space** according to Cherry and Lin (2006a).

Most of the models we described in Section 2.2 fall into the Permutation Space. For example, the asymmetric IBM Models search a version of permutation space with 1-to-n constraints, i.e. one target word can only be aligned to one source words. All the new methods developed in the following chapters of this thesis also fall into this alignment space.

2.4 Word Alignment and Phrase-Based SMT

As mentioned at the end of Section 2.1.2, PB-SMT systems normally induce phrase pairs based on established word alignments. In this section, we describe how word alignment is closely related to PB-SMT systems.

2.4.1 Heuristics for Symmetric Word Alignment

Given that the most widely used word alignment models, namely the generative models, are mostly asymmetric, i.e. these models assume that each target word can be aligned to exactly one source word, these models can produce 1-to-1 and 1-to-n links, but not n-to-1 links (cf. Section 2.2.1). Figure 2.1 shows examples of Chinese–

[9] Another recent work that imposes this constraint is Taskar et al. (2005).

English and English–Chinese word alignments represented as alignment matrices, where one Chinese word can be aligned to multiple English words in Chinese–English direction and one English word can be aligned to multiple Chinese words in English–Chinese direction.[10]

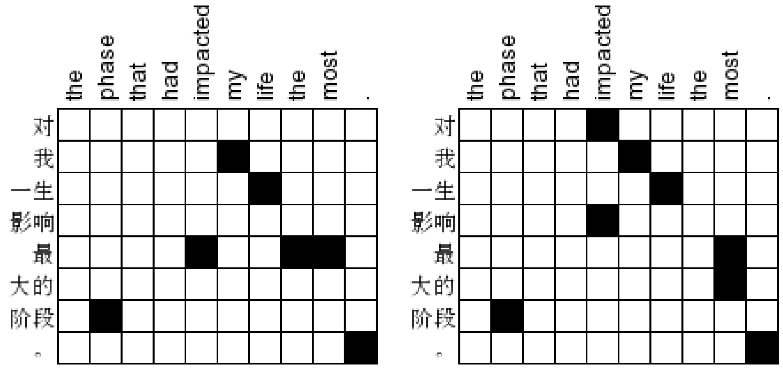


Figure 2.1: An example of asymmetric Chinese–English (left) and English–Chinese alignments (right)

Och and Ney (2003) were the first to introduce heuristics for symmetric word alignment by heuristically select links from the union of links produced by source-to-target and target-to-source word alignment. A set of heuristics were presented including union, intersection and refined methods, of which refined methods systematically produce better SMT results in their experiments. Given source-to-target alignment $A_{\mathbf{f}\rightarrow\mathbf{e}}$ and target-to-source alignment $A_{\mathbf{e}\rightarrow\mathbf{f}}$, the alignment intersection $A_{\cap}$ and union $A_{\cup}$ are defined as follows:

- Intersection: $A_{\cap} = A_{\mathbf{f}\rightarrow\mathbf{e}} \cap A_{\mathbf{e}\rightarrow\mathbf{f}}$
- Union: $A_{\cup} = A_{\mathbf{f}\rightarrow\mathbf{e}} \cup A_{\mathbf{e}\rightarrow\mathbf{f}}$

Given this definition, intersection links are a subset of union links $A_{\cap} \subseteq A_{\cup}$. Figure 2.2 shows the intersection and union links of the example in Figure 2.1. We use black squares to denote intersection links and grey ones to indicate the links in the union but not in the intersection.

[10] In this thesis, the glosses for the Chinese examples are deliberately ignored; instead, colors and lines are used to indicate the correspondences between Chinese and English words.

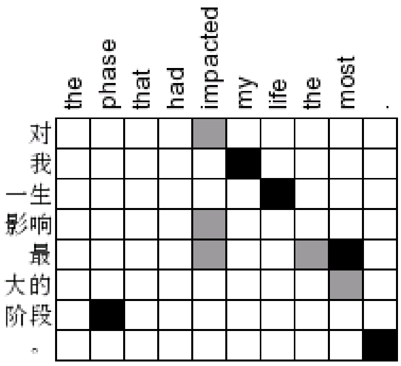


Figure 2.2: An example of alignment intersection and union

Alignment intersection normally has a higher precision and union yields a higher recall. However, neither of them are most suitable for PB-SMT systems. Intersection contains too few links and results in a large number of phrase pairs in the phrase extraction stage because the phrases that are consistent with word alignment increase substantially when a large number of unaligned words exist, which causes the phrase extraction to be not properly constrained. Union normally contains a large number of incorrect links which can prohibit the extraction of useful phrases.[11] Therefore, the refined method described in Och and Ney (2003) is widely adopted. In the Moses PB-SMT system, a similar method is implemented as "Grow-Diag-Final" heuristics, which includes three separate processes, name "Grow", "Diag" and "Final", to expand the links in the intersection using the links in the union. In all three steps, it is required that a new added link should connect at least one unaligned word. Here, the "Grow" and "Diag" steps include the adjacent neighbourhood link points, and the "Final" step adds in the non-adjacent alignment points. The adjacent neighbouring link points of an intersection link point (the black square) can be classified into horizontal (H), vertical (V) and diagonal (D) links as shown in Figure 2.3.

The "Grow" heuristic only includes horizontal and vertical adjacent link points, "Grow-Diag" heuristics further add in the diagonal neighbourhood links and the "Fi-

[11] Depending on the characteristics of the data, the advantages of the refined methods over other heuristics may not always hold; however, it is widely recognised that refined methods can achieve consistently good results under different data settings (Och and Ney, 2003).

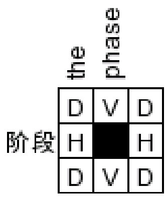


Figure 2.3: An example of neighbourhood links

nal" step adds in the non-adjacent link points. Each of these three steps can expand the intersetion links and the recall of the alignment can be improved. Figure 2.4 shows the resulting alignment using "Grow" and "Grow-Diag" heuristics on top of the intersection links in Figure 2.2. The black squares denotes the intersection links and grey ones denote the expanded links using "Grow" or "Grow-Diag" heuristics.

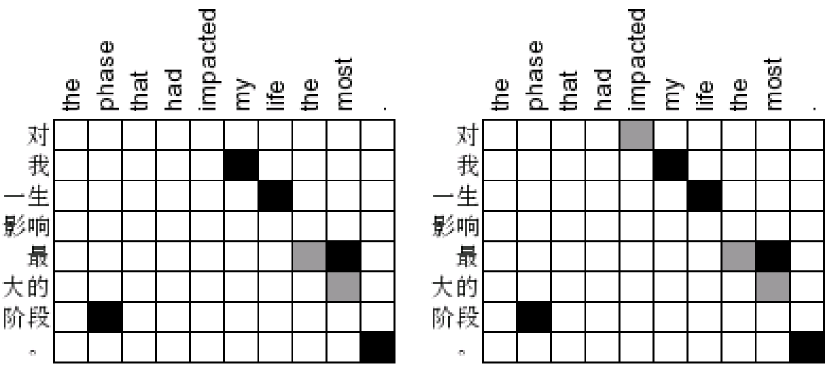


Figure 2.4: An example of alignment using "Grow" (left) and "Grow-Diag" heuristics (right)

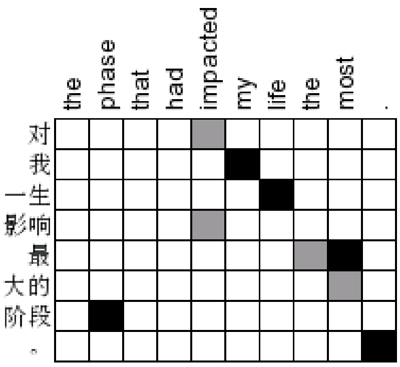


Figure 2.5: An example of alignment using "Grow-Diag-Final" heuristics

Figure 2.5 is an example of alignment using "Grow-Diag-Final" heuristics. Compared with Figure 2.2, most links in the union can be included using these heuristics.

Some other methods for symmetrisation have also been proposed. Matusov et al. (2004) proposed an algorithm which considers the alignment problem as a task of finding the edge cover with minimal costs in a bipartite graph, where the parameters of IBM Models and first-order HMM word-to-word alignment models are used to determine the costs of aligning a specific target word to a source word. Fraser and Marcu (2007a) presented a new generative model allowing the production of m-to-n links; however, this model substantially increases the complexity of the alignment process.

2.4.2 Alignment Quality and Translation Quality

The intrinsic alignment quality is normally measured against a manually annotated word alignment data. In the context of MT, the impact of word alignment on the final translation quality is considered to be more important. However, the correlation between **intrinsic** word alignment quality (e.g. precision, recall and F-score) and **extrinsic** translation quality of PB-SMT systems (e.g. Bleu) is quite complicated. Despite current intensive investigations into the impact of word alignment quality on SMT, no conclusive agreement can be reached given that different studies used different data and systems. However, there is a widespread recognition within the community that an improvement in intrinsic word alignment quality (measured using AER for example) does not necessarily imply an improvement in translation quality (normally measured with Bleu) (Liang et al., 2006; Ma et al., 2008a), and vice-versa (Vilar et al., 2006). Fraser and Marcu (2007b) and Ma et al. (2009a) also showed that the correlation is weak when the intrinsic quality is measured with F-score.

Besides general measures like F-score and AER, various studies have investigated the effect of balancing precision and recall on MT performance. While Ayan and Dorr (2006) and Chen and Federico (2006) observed that higher precision alignments are more useful in a PB-SMT system, Mariño et al. (2006) observed that a high recall alignment improved the performance of an N-gram-based SMT system. Fraser and Marcu (2007b) compared the performance of PB-SMT using the word alignment obtained via the intersection, union and refined symmetrisation of IBM Model 4 source-to-target and target-to-source alignments. The word aligner was trained with different amounts of data so that the quality of word alignment varied. Their results on large corpora do not confirm the hypothesis that higher precision alignments are more beneficial to PB-SMT systems than higher recall alignments. From their experiments, increasing the alignment precision (for example, by taking the intersection of source-to-target and target-to-source alignments) improves PB-SMT systems only when the training data is small. With larger corpora, higher recall alignments (like union or refined methods) are better.

Vilar et al. (2006) improved the translation quality of a German–English phrase-based SMT system by deleting links between the English verb and the German particle part of the verb, which is situated far from the main part of the verb and produces a long-distance link. Note that these long-distance links are nevertheless correct from the point of view of alignment quality.

2.5 Methodology

In this thesis, we develop new algorithms and models for word alignment and compare our approach against the state-of-the-art word alignment models, notably IBM Model 4 and HMM word-to-phrase alignment model in terms of both intrinsic and extrinsic quality. We use the Giza++ (Och and Ney, 2003) implementation of IBM Model 4 and the MTTK (Deng and Byrne, 2006) implementation of HMM word-to-phrase alignment model. These two word alignment toolkits are chosen not only because of their widespread use within the community, but also due to the fact that they are open-source software, which is essential to facilitate the reproducibility of our results. In this section, our evaluation methods for word alignment are described along with the data sets and the baseline system configurations used in our experiments.

2.5.1 Evaluation

As mentioned in Section 2.4.2, the intrinsic word alignment quality refers to the quality of word alignment itself according to linguistic experts. Therefore, the judgement of the intrinsic quality is normally conducted by comparing the resulting word alignment against a manually annotated word alignment data, i.e. the **gold-standard**, which consists of bilingual sentence pairs and all the lexical correpondences between source and target words established by linguistic experts. The most widely used metrics for the measurement of intrinsic alignment quality are F-score and AER. The calculation of AER requires the links in the gold-standard $\mathcal{G}$ to be classified into sure links ($\mathcal{S}$) and possible links ($\mathcal{P}$) where $\mathcal{S} \subseteq \mathcal{P}$, reflecting the annotator's confidence of creating each link. The distinction between sure links and possible links does not necessarily hold in many available gold-standard word alignment data. We hereafter use F-score instead of AER to measure the alignment quality.

F-score and AER are both high-level quality measures which produce an overall score for a given alignment result. Given that the alignment models described in this thesis assume 1-to-n alignment structure, we can further examine the quality of each type of alignment, e.g. 1-to-1 alignment and 1-to-2 alignment. The quality of each type of alignment is an informative indicator of the capability of different word alignment models, e.g. IBM Model 4 equipped with fertility models is expected to perform better in 1-to-2 alignment than HMM word-to-word alignment models since the 1-to-n alignment is directly modelled using fertility models. Based on these observations, we not only conduct a "macro-evaluation" through calculating the F-score of the alignment, but also a "micro-evaluation" to measure the quality of each type of alignment.

Intrinsic Macro-evaluation

We evaluate the intrinsic quality of the predicted alignment $\mathcal{A}$ against the gold-standard $\mathcal{G}$ with Precision, Recall and the balanced F-score with $\alpha = 0.5$.

$$\text{Precision} = \frac{|\mathcal{A} \cap \mathcal{G}|}{|\mathcal{A}|} \quad \text{Recall} = \frac{|\mathcal{A} \cap \mathcal{G}|}{|\mathcal{G}|}$$

$$\text{F-score}(\mathcal{A}, \mathcal{G}, \alpha) = \frac{1}{\frac{\alpha}{Precision(\mathcal{A},\mathcal{G})} + \frac{1-\alpha}{Recall(\mathcal{A},\mathcal{G})}}$$

Intrinsic Micro-Evaluation

Given a source sentence f_1^I, target sentence e_1^J and the correct word alignment $\mathbf{a}$ between f_1^I and e_1^J according to $\mathcal{G}$, we evaluate the source-to-target word alignment, i.e. how the source sentence generates the target sentence. Using the asymmetric generative word alignment models, the alignment structure can only contain 1-to-n links, i.e. each target word can be aligned to exactly one source word and each source word can be aligned to multiple target words. One way to peek into the details of how each source word is aligned is to classify the source words into groups according to their alignment types, such as 1-to-1 alignment or 1-to-2 alignment in the gold-standard, and evaluate each type separately.

To do this, we have to firstly determine the alignment type that each source word falls into according to the gold-standard alignment $\mathcal{G}$ so that we can compare the predicted links for the relevant source words against the gold-standard, i.e. source words belonging to 1-to-1 type are compared against the 1-to-1 gold-standard and so on and so forth.[12] We use $\mathcal{G}_i = \{j_1, \cdots, j_m | a_{j_1} = a_{j_2} = \cdots = a_{j_m} = i\}$ to denote the links involving each source word f_i according to the gold-standard.

1-to-1 alignment can be evaluated by classifying the predicted links into three different quality levels. Formally, the source words that are involved in 1-to-1 alignment in the gold-standard are defined as such words that $\mathcal{G}_i = \{j | a_j = i; \forall j' \neq j, a_{j'} \neq i\}$. A set of predicted alignment links for word f_i is denoted as $\mathcal{A}_i = \{j_1 \cdots j_m\}$, which is considered to be:

- **correct** iff $m = 1$ and $a_{j_1} = i$, indicating that the 1-to-1 alignment is com-

[12] The alignment type of each source word is determined according to the gold-standard; therefore, depending on the quality of the predicted links, a source word which should be aligned to exactly one target word can possibly be aligned to multiple target words.

pletely correct.

- **redundant** iff $m \geq 2$ and $\exists j_m \in \mathcal{A}_i, j_m \in \mathcal{G}_i$ and $\exists j_m \in \mathcal{A}_i, j_m \notin \mathcal{G}_i$, indicating that besides the correct link, one or more redundant links have been predicted.

- **incorrect** iff $\forall j_m \in \mathcal{A}_i, j_m \notin \mathcal{G}_i$, indicating that all the predicted links are incorrect.

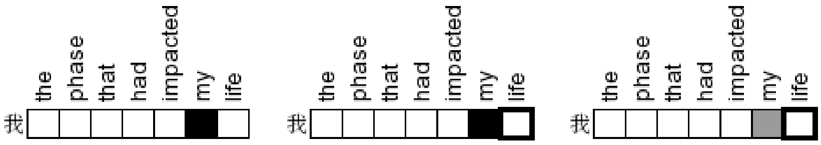


Figure 2.6: Examples of 1-to-1 correct (left), redundant (middle) and incorrect links (right)

Figure 2.6 shows examples of correct, redundant and incorrect links for a source word which should be involved in a 1-to-1 alignment according to the gold-standard, where the black squares indicate the correctly predicted link, white square with bold black borders indicating the incorrectly predicted link and grey squares indicating the links according to the gold-standard.

We evaluate 1-to-2 alignment by classifying the predicted alignment into four quality levels. The source words that are involved in 1-to-2 alignment in the gold-standard can be defined as such source words f_i that $\mathcal{G}_i = \{j_1, j_2 | a_{j_1} = i; a_{j_2} = i; \nexists j' \notin \{j_1, j_2\}, a_{j'} = i\}$. A set of predicted alignment links for word f_i is denoted as $\mathcal{A}_i = \{j_1 \cdots j_m\}$, which is considered to be:

- **correct** iff $m = 2$ and $a_{j_1} = a_{j_2} = i$, indicating that the 1-to-2 alignment is *exactly* correct.

- **incomplete-link missing** iff $m = 1$ and $a_{j_1} = i$, indicating that the only one link has been predicted, which is correct.

- **incomplete-link redundant** iff $m \geq 2$ and $\exists j_m \in \mathcal{A}_i, j_m \in \mathcal{G}_i$ and $\exists j_m \in \mathcal{A}_i, j_m \notin \mathcal{G}_i$, indicating that of the predicted links, one link was correct, the others are incorrect.

- **redundant** iff $m > 2$ and $\forall j \in \mathcal{G}_i : j \in \mathcal{A}_i$ and $\exists j_m \in \mathcal{A}_i, j_m \notin \mathcal{G}_i$, indicating that besides the two correct links, one or more redundant links have been predicted.
- **incorrect** iif. $\forall j_m \in \mathcal{A}_i, j_m \notin \mathcal{G}_i$, indicating that all the predicted links are incorrect.

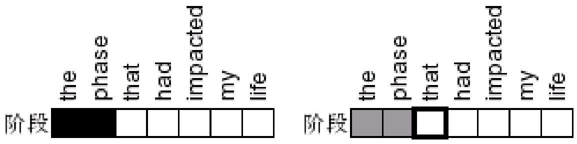


Figure 2.7: Examples of 1-to-2 correct (left) and incorrect links (right)

Figure 2.7 shows examples of correct and incorrect 1-to-2 alignment links with the same symbols indicating the same meaning as in Figure 2.6.

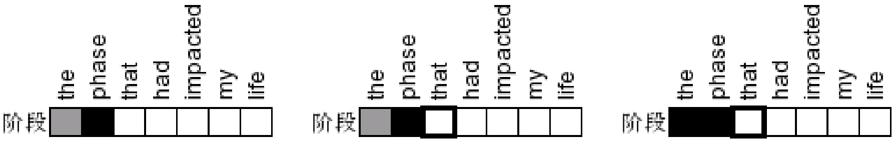


Figure 2.8: Examples of 1-to-2 incomplete-missing (left), incomplete-redundant (middle) and redundant links (right)

Figure 2.8 are examples of 1-to-2 incomplete-missing, incomplete-redundant and redundant alignment links, with the same symbols indicating the same meaning as in Figure 2.6.

For each type of alignment, we calculate the percentage of each quality level. For example, for source words involved in 1-to-1 alignment, both a higher ratio of correct links and lower ratio of incorrect links imply more correct links for the source words and a better alignment quality. This evaluation method is primarily used in Chapter 6, where we discuss the alignment structures produced by different alignment models, using a relatively large amount of gold-standard data. We currently do not apply this evaluation method in Chapter 3 and 4, in which the intrinsic quality of the alignment is not our main concern.

Gold-Standard Annotation

	Chinese–English		English–Chinese	
	con.	n.c.	con.	n.c.
1-to-0	3.09		2.32	
1-to-1	59.32		58.22	
1-to-2	8.12	2.73	9.08	1.71
1-to-3	1.63	0.34	0.61	0.06
1-to-n ($n > 3$)	0.20	0.00	0.08	0.06
2-to-1	18.50	3.49	15.95	5.35
3-to-1	1.86	0.17	4.80	1.00
n-to-1 ($n > 3$)	0.33	0.22	0.77	0.00
m-to-n	0.96	5.76	1.71	5.02

Table 2.1: Distribution of alignment types for manually aligned IWSLT Chinese–English corpus (%)

Two annotators were employed to annotate 502 sentence pairs from IWSLT data in the dialogue domain using the annotation tool proposed in Nichols and Hwa (2005). The annotation process follows the GALE[13] Chinese–English word alignment guidelines v3.0. Two annotators performed annotation independently and a discussion over conflicting annotation results was held. Given that the annotation guidelines are very detailed and annotated sentences are relatively short, very few conflicts were encountered. Table 2.1 shows the distribution of alignment types on IWSLT data. The 1-to-n (($n \geq 2$)) and n-to-1 (($n \geq 2$)) alignments are classified into consecutive (con.) and non-consecutive (n.c.) groups depdending on whether the n words are consecutive or not.

In addition to IWSLT data, we use another data set in the news domain created for the GALE program.[14] Table 2.2 shows the distribution of alignment types for Chinese–English and English–Chinese word alignment respectively. From the Table, we can see that there is a higher ratio of 1-to-n ($n \geq 2$) alignments in the Chinese–English direction than the English–Chinese direction, implying a larger proportion of Chinese words generating (aligned to) multiple English words.

[13] `http://projects.ldc.upenn.edu/gale/`

[14] We have access to this data set following a collaboration with the Cambridge University Engineering department, a participant in the GALE program.

	Chinese–English		English–Chinese	
	con.	n.c.	con.	n.c.
1-to-0	12.26		8.12	
1-to-1	49.26		39.10	
1-to-2	11.07	4.01	3.62	0.57
1-to-3	3.95	1.25	0.53	0.08
1-to-n ($n > 3$)	1.54	0.51	0.12	0.02
2-to-1	10.93	1.62	18.25	6.68
3-to-1	2.35	0.35	9.89	3.42
n-to-1 ($n > 3$)	0.78	0.10	6.39	3.21
m-to-n	6.47	17.01	3.78	18.16

Table 2.2: Distribution of alignment types for manually aligned GALE Chinese–English corpus (%)

Extrinsic Evaluation

While the intrinsic measures can give a direct evaluation of the quality of the word alignment, it is faced with several limitations. First of all, it is really difficult to build a reliable and objective gold-standard. Second, research has shown that an increase in AER does not necessarily imply an improvement in translation quality (Liang et al., 2006) and vice-versa (Vilar et al., 2006). It has also been shown that F-score has a very weak correlation with SMT translation quality in terms of Bleu score (Zhang et al., 2008). Consequently, we also extrinsically evaluate the performance of our approach on the Chinese–English translation task, i.e. we measure the influence of the word alignment process on the final translation output. The quality of the translation output is mainly evaluated using Bleu, with Nist (Doddington, 2002) and Meteor (Banerjee and Lavie, 2005) as complementary metrics. Both Bleu and Meteor metrics produce a score ranging from 0 to 1; in this thesis, we use percentage for these scores, ranging from 0 to 100.

We perform significance testing on the improvement in the translation quality in terms of Bleu using approximate randomisation (Noreen, 1989), which is shown to be more appropriate for this task than booststrap resampling (Koehn, 2004) by Collins et al. (2005) and Riezler and Maxwell (2005).

2.5.2 Data

		Chinese	English
IWSLT	Sentences	502	
	Running words	3796	3868
	Vocabulary size	922	898
GALE	Sentences	12,172	
	Running words	275,669	341,625
	Vocabulary size	16,784	14,633

Table 2.3: Data Set 1 and 2: statistics for the gold-standard data

We provide the various statistics of the data sets used in our experiments. Data Set 1 and 2 are the gold-standard corpus in the dialogue domain (IWSLT data) and news domain (GALE data) as shown in Table 2.3.

Throughout the thesis, we use the Chinese–English Data Set 3, which is provided within the IWSLT 2006 and 2007 evaluation campaigns. This multilingual speech corpus contains sentences similar to those that are usually found in phrase-books for tourists going abroad (Takezawa et al., 2002). Specifically, we use the standard training data, to which we add devset1 and devset2. Devset4 is used to tune the parameters and the performance of the system is tested on IWSLT 2006 and 2007 test sets. We use both test sets because they are quite different in terms of sentence length and vocabulary size. Based on the original manual segmentation for Chinese, the various statistics for the IWSLT corpora are shown in Table 2.4. Despite the size of the data being small, this data set has a small vocabulary, which simplifies the translation task and results in MT systems with reasonable performance. To

		Chinese	English
Train	Sentences	40,958	
	Running words	357,968	385,065
	Vocabulary size	11,362	9,718
Dev.	Sentences	489 (7 ref.)	
	Running words	5,717	46,904
	Vocabulary size	1,143	1,786
Eval.	Sentences	489 (7 ref.)/489 (6 ref.)	
	Running words	6,066/3,166	51,500/23,181
	Vocabulary size	1,339/862	2,016/1,339

Table 2.4: Data Set 3: statistics for the IWSLT data

test the scalability of our approach, we add in the HIT corpus[15] containing 120K sentence pairs, which was made available for the IWSLT 2008 evaluation campaign.

		Chinese	**English**
UN	Sentences	40,000	
	Running words	881,861	956,023
	Vocabulary size	16,100	20,068
GALE	Sentences	90,603	
	Running words	2,616,938	2,529,311
	Vocabulary size	56,452	51,624

Table 2.5: Data Set 4 and 5: statistics for the UN and GALE training data

In Chapter 3, in order to test the performance of our segmenter across different domains, we additionally use data from parliamentary documents, i.e. a portion of UN data for the NIST[16] 2006 evaluation campaign (Data Set 4) for MT training. This large data set (over 3 million sentence pairs) facilitates the testing of scalability of our approach.

In Chapter 6, an additional GALE data set (Data Set 5) containing financial news created within the GALE program (catalogue number LDC2006E26) is used. The various statistics of Data Set 4 and 5 are listed in Table 2.5, where the Chinese data is segmented using the LDC segmenter.[17] Note that in the Table we merely list the statistics of a very small portion (40K sentence pairs) of the whole UN data used in a preliminary test of our approach.

		Chinese	**English**
Dev.	Sentences	993 (9 ref.)	
	Running words	26,735	267,222
	Vocabulary size	4,738	10,665
Eval.	Sentences	878/935/919 (4 ref.)	
	Running words	25,354/27,922/26,748	105,530/112,729/113,781
	Vocabulary size	4,273/4,755/4,998	7,388/7,110/7,875

Table 2.6: Data Set 6: statistics for the MTC development and test data

The MT systems trained on Data Set 4 or 5 are developed and tested on Data Set

[15] `http://mitlab.hit.edu.cn/index.php/resources/29-the-resource/111-share-bilingual-corpus.html`

[16] In this thesis, we use NIST to represent the National Institute of Standards and Technology and Nist to denote the MT evaluation metric.

[17] `http://www.ldc.upenn.edu/Projects/Chinese`

6, i.e. using the LDC Multiple-Translation Chinese (MTC) Corpus for development and MTC parts 2, 3 and 4 for testing. The various statistics for Data Set 6 segmented using the LDC segmenter are shown in Table 2.6.

2.5.3 Baseline System

We build the baseline word alignment and PB-SMT systems using existing open-source toolkits for the purpose of fair comparison. Unless specifically mentioned, all the Chinese data in the IWSLT data set (Data Set 1 and 3) is manually segmented (Paul, 2006), and that in UN (Data Set 4), GALE (Data Set 2 and 5) and MTC data sets (Data Set 6) is segmented using the LDC word segmenter, which is basically a dictionary-based segmenter with word frequency information for disambiguation. When Part-of-Speech (POS) tags are required, we use the maximum-entropy-based POS tagger Mxpost (Ratnaparkhi, 1996) trained on the English Penn Treebank (PTB) (Marcus et al., 1993) and Penn Chinese Treebank (CTB) (Xue et al., 2005) respectively to tag English and Chinese texts. The English and Chinese POS tag sets can be found in Appendix A and B respectively.

The syntactic dependencies for both English and Chinese are obtained using a state-of-the-art dependency parser, Maltparser,[18] which achieved 84% and 88% labelled attachment scores for Chinese and English respectively (Nivre et al., 2007). The English and Chinese dependency labels that occurred in the training data are listed in Appendix C and D respectively. The English model is pre-trained[19] using PTB. The constituent structures in PTB are converted into dependency structures using pennconverter[20] (Johansson and Nugues, 2007), which primarily uses a head percolation table (Magerman, 1995) proposed in Yamada and Matsumoto (2003) (see Appendix E). The dependency types are derived from a set of hand-crafted rules (Johansson and Nugues, 2007) (see Appendix G).

[18] http://maltparser.org/

[19] The pre-trained model is available here: http://w3.msi.vxu.se/users/jha/maltparser/mco/english_parser/engmalt.html

[20] http://nlp.cs.lth.se/pennconverter/

The constituent structures in CTB v5.1 are converted into dependency structures using Penn2Malt v0.2[21] with a head percolation table (see Appendix F) compiled by Yuan Ding for the purpose of Machine Translation. The dependency types are derived using a set of hand-crafted rules (Hall, 2006) (see Appendix H).

Word Alignment

The Giza++ implementation of IBM Model 4 is used as the baseline for word alignment, and the "Grow-Diag-Final" heuristic described in Koehn et al. (2003) to derive the refined alignment from bidirectional alignments. Model 4 is incrementally trained by performing 5 iterations of Model 1, 5 iterations of HMM, 3 iterations of Model 3, and 3 iterations of Model 4. In some comparative experiments, we also use the MTTK implementation of HMM word-to-phrase alignment model. The model training includes 10 iterations of Model 1, 5 iterations of Model 2, 5 iterations of HMM word-to-word alignment, 20 iterations (5 iterations respectively for phrase length 2, 3, 4 with unigram translation probability, and phrase length 4 with bigram translation probability) of HMM word-to-phrase alignment for Chinese–English alignment and 5 iterations (5 iterations for phrase length 2 with uniform translation probability) of HMM word-to-phrase alignment for English–Chinese. The "Grow-Diag-Final" heuristic is used by default to derive alignment from bidirectional alignments.

Machine Translation

The baseline in our experiments is a standard log-linear PB-SMT system. With the word alignment obtained using the above-mentioned method, we perform phrase-extraction using heuristics described in (Koehn et al., 2003), Minimum Error-Rate Training (MERT) (Och, 2003) optimising the Bleu metric, a 5-gram language model with Kneser-Ney smoothing (Kneser and Ney, 1995) trained with SRILM[22] (Stolcke,

[21] `http://w3.msi.vxu.se/~nivre/research/Penn2Malt.html`

[22] Specifically, we used SRILM release 1.4.6.

2002) on the English side of the training data, and Moses[23] for decoding.

2.6 Summary

In this chapter, we reviewed state-of-the-art log-linear PB-SMT systems and pointed out the important role of word alignment in these systems. We then presented a critical review of existing word alignment models that produce word alignments between strings of source and target languages, including generative, discriminative and association-based alignment models. Finally, the methodology of this thesis was explained by including the evaluation methods, data, baseline systems for both word alignment and MT.

In the next chapter, we will exploit the segmentation constraints for word alignment with a novel algorithm that interactively performs word segmentation and alignment, namely word packing.

[23] Specifically, we used revision 1881 checked out from the Moses repository.

Chapter 3

Bootstrapping Word Alignment via Word Packing

In this chapter, segmentation as a constraint in word alignment is presented and discussed. Specifically, a new algorithm, namely word packing, is proposed to bootstrap word alignment through the optimisation of word segmentation. As a generalisation of the word packing algorithm, this approach is directly used to perform Chinese word segmentation without relying on any existing word segmenters. The motivation and procedures in the algorithm will be detailed in the following sections, and the effectiveness of the algorithm will be tested through extensive experiments.[1]

3.1 Introduction

State-of-the-art Statistical Machine Translation (SMT) requires a certain amount of bilingual corpora as training data in order to achieve competitive results. The only assumption behind most current statistical models (Brown et al., 1993; Vogel et al., 1996; Deng and Byrne, 2005) is that the aligned sentences in such corpora should be segmented into sequences of tokens that are meant to be words. Therefore, for languages where word boundaries are not orthographically marked, tools which

[1] The material in this chapter has been published, albeit in a different form, in Ma et al. (2007b), Ma and Way (2009a) and Ma and Way (2009b).

segment a sentence into words are required. Even for a language like English, where spaces can offer an easy approximation to the minimal content-bearing units, an optimal segmentation is still required when analysing multi-word units, especially non-compositional compounds such as "kick the bucket" and "hot dog" (Melamed, 1997).

However, this segmentation is often performed in a **monolingual** context without any **bilingual** consideration, i.e. the segmentation of the source (target) language is performed regardless of the corresponding target (source) language at hand, which makes the word alignment task more difficult since different languages may realise the same concept using varying numbers of words (cf. Wu (1997)). This can generate a great deal of complexity for (bilingual) word alignment models if the corresponding texts are inappropriately segmented. Moreover, most segmenters are usually trained on a manually segmented domain-specific corpus. Therefore, such a segmentation tends to be sensitive to the domain of the data and may not produce consistently good results when used across different domains.

A substantial amount of research has been carried out to address the problems of word segmentation in the context of PB-SMT. Some statistical alignment models allow for 1-to-n word alignments for those reasons; however, they rarely question the monolingual tokenisation and the basic unit of the alignment process.[2] Moreover, statistical alignment models assume a first-order dependency between alignment decisions in order to make the alignment process efficient. Some middle- or long-range dependencies cannot be captured under such models. Some more recent research focuses on combining various segmenters either in SMT training (Zhang et al., 2008) or decoding (Dyer et al., 2008). One important yet often neglected fact is that the optimal segmentation of the source (target) language is dependent on the target (source) language itself, its domain and its genre. Segmentation considered to be "good" from a monolingual point of view may be unadapted for training alignment

[2] Interestingly, this is actually even the case for approaches that directly model alignments between phrases (Marcu and Wong, 2002; Birch et al., 2006).

models or PB-SMT decoding (Ma et al., 2007b). The resulting segmentation will consequently influence the performance of a PB-SMT system, and a bilingually motivated segmentation is highly desirable for PB-SMT tasks.

In summary, we focus on optimising the segmentation with the goals of (i) simplifying the task of automatic word aligners by packing several consecutive words together when we believe they correspond to a single word in the opposite language (**word packing**). By identifying enough such cases, we reduce the number of 1-to-n alignments, thus making the task of word alignment both easier and more natural; from an information-theoretic perspective, such a process reduces the predictive power of translation models (Melamed, 1997); and (ii) capturing long-distance dependencies between alignment decisions in an incremental manner, i.e. we bootstrap the word packing and subsequently optimise the word segmentation based on its influence on SMT performance. We then generalise this method to produce a bilingually motivated automatically domain-adapted word segmentation approach for PB-SMT without relying on any existing word segmenters. We first utilise a small bilingual corpus with the relevant language segmented into basic writing units (e.g. characters for Chinese), and then cast the segmentation problem into an alignment problem. Various issues regarding scalability related to such a process is also investigated.

3.2 Interaction Between Word Segmentation and Alignment

In this section, we first detail a pilot study of the influence of word segmentation on the performance of PB-SMT. Then we show that the pervasive 1-to-n alignments in Chinese–English word alignment motivate us to take advantage of the interaction between word segmentation and alignment in order to simplify the alignment task.

3.2.1 The Influence of Word Segmentation on PB-SMT

The monolingual word segmentation step in traditional SMT systems has a substantial impact on the performance of such systems. A considerable amount of recent research has focused on the influence of word segmentation on SMT (Ma et al., 2007b; Chang et al., 2008; Zhang et al., 2008). However, most explorations have focused on the impact of various segmentation guidelines and the mechanisms of the segmenters themselves. Our research also concerns the consistency of performance across different domains. From our experiments, we show that monolingual segmenters cannot produce consistently good results when applied to a new domain.

Our pilot investigation into the influence of word segmentation on SMT involves three off-the-shelf Chinese word segmenters, including ICTCLAS (ICT) Olympic version,[3] LDC segmenter and Stanford segmenter version 2006-05-11.[4] Both ICTCLAS and Stanford segmenters utilise machine learning techniques, with Hidden Markov Models for ICT (Zhang et al., 2003) and Conditional Random Fields for the Stanford segmenter (Tseng et al., 2005). Both segmentation models are trained on news domain data with named entity recognition functionality. The LDC segmenter is dictionary-based with word frequency information to help disambiguation, both of which are collected from data in the news domain. We use Chinese character-based and manual segmentations as points of contrast. Table 3.1 shows the pairwise F-measure of the automatic segmenters. On the IWSLT data set in the dialogue domain, we can observe the strongest agreement between the LDC and ICT segmenters, which is even stronger than for Stanford and ICT segmenters. On UN data, as expected, the Stanford and ICT segmenters agree more. On both data sets, the LDC and Stanford segmenters show the greatest discrepancies.

We conduct MT experiments on a range of different-sized amounts of the above-mentioned data using Moses. The performance of the PB-SMT system is measured via Bleu score (Papineni et al., 2002). We first measure the influence of word seg-

[3] http://ictclas.org/index.html

[4] http://nlp.stanford.edu/software/segmenter.shtml

		ICT	LDC	Stanford
IWSLT	ICT	100	94.45	93.80
	LDC	**94.45**	100	90.13
	Stanford	93.80	90.13	100
UN	ICT	100	95.18	96.44
	LDC	95.18	100	93.38
	Stanford	**96.44**	93.38	100

Table 3.1: Pairwise F-measure between segmenters (%)

mentation on in-domain data with respect to the three above-mentioned segmenters, namely UN data from the NIST 2006 evaluation campaign.[5] As can be seen from Table 3.2, using monolingual segmenters achieves consistently better SMT performance than character-based segmentation (CS) on different data sizes, which means that character-based segmentation is not good enough for this domain where the vocabulary tends to be large. We can also observe that the ICT and Stanford segmenters consistently outperform the LDC segmenter. Even using 3M sentence pairs for training, the differences between the Stanford and LDC segmenters are still statistically significant (p<0.05).

	40K	160K	640K	3M
CS	8.33	12.47	14.40	17.80
ICT	10.17	14.85	**17.20**	20.50
LDC	9.37	13.88	15.86	19.59
Stanford	**10.45**	**15.26**	16.94	**20.64**

Table 3.2: Impact of word segmentation on translation quality using UN training data (Bleu)

However, when tested on out-of-domain data, i.e. IWSLT data in the dialogue domain, the results seem to be more difficult to predict. We trained the system on different amounts of data and evaluated the system on two test sets: IWSLT 2006 and 2007. From Table 3.3, we can see that on the IWSLT 2006 test set, LDC achieves consistently good results and the Stanford segmenter is the worst.[6] Furthermore,

[5] Note that the UN data containing parliamentary documents is not exactly "in-domain"; however, it is more similar to the news domain compared to dialogues. We chose this corpora simply because of its availability and its relatively large size that enable us to test our approach in terms of scalability. We expect a better performance on "strictly" in-domain data using the ICT and Stanford segmenters.

[6] Interestingly, the developers themselves also note the sensitivity of the Stanford segmenter and

character-based segmentation also achieves competitive results. On IWSLT 2007 test set, all monolingual segmenters outperform character-based segmentation and the LDC segmenter is only slightly better than the other segmenters.

		40K	160K
IWSLT06	CS	19.31	23.06
	Manual	19.94	-
	ICT	20.34	23.36
	LDC	**20.37**	**24.34**
	Stanford	18.25	21.40
IWSLT07	CS	29.59	30.25
	Manual	**33.85**	-
	ICT	31.18	33.38
	LDC	31.74	**33.44**
	Stanford	30.97	33.41

Table 3.3: Impact of word segmentation on translation quality using IWSLT data (Bleu)

From the experiments reported above, we can come to the following conclusions. First of all, character-based segmentation cannot achieve state-of-the-art results in most experimental settings. This also motivates the necessity to work on better segmentation strategies. Second, monolingual segmenters cannot achieve consistently good results when used in another domain. In the following sections, we propose a bilingually motivated segmentation approach which can be automatically derived from a small representative data set, and the experiments show that we can consistently obtain state-of-the-art results in different domains. Using this approach, we can either enhance the existing monolingual segmenter or directly perform word segmentation without relying on any monolingual segmenters.

3.2.2 The Case of 1-to-n Word Alignment

The same concept can be expressed in different languages using varying numbers of words; for example, a single Chinese word may frequently surface as a compound or a collocation in English given the great differences between the two languages. To quickly (and approximately) evaluate this phenomenon, we trained the statistical

incorporate external lexical information to address such problems (Chang et al., 2008).

IBM word-alignment model 4 (Brown et al., 1993)[7] using GIZA++ for the following language pairs: Chinese–English (ZH–EN), Italian–English (IT–EN), and German–English (DE–EN), using the IWSLT 2006 corpus (Takezawa et al., 2002; Paul, 2006) for the first two language pairs, and the Europarl corpus (Koehn, 2005) for the last one. These asymmetric models produce alignments between one word and several words in both directions. Word segmentation was performed totally independently of the bilingual alignment process, i.e. it was done in a monolingual context. For European languages, we applied the maximum-entropy-based tokeniser of OpenNLP;[8] the Chinese sentences were manually segmented (Paul, 2006).

Table 3.4 reports the frequencies of the different types of alignments for the various languages and directions. We also differentiate consecutive (con.) and non-consecutive (n.c.) target words. As expected, the number of 1-to-n alignments with $n \neq 1$ is high for Chinese–English ($\simeq$ 40%), and significantly higher than for European languages. The case of 1-to-n alignments is, therefore, obviously an important issue when dealing with Chinese–English word alignment.[9] We can also observe that for all three language pairs, most of the n words involved in 1-to-n alignments are consecutive.

	1-to-0	1-to-1	1-to-2		1-to-3		1-to-n ($n > 3$)	
			con.	n.c.	con.	n.c.	con.	n.c.
ZH–EN	22.19	59.60	9.92	1.69	3.06	1.24	1.02	1.28
EN–ZH	28.48	57.08	9.27	1.81	1.28	0.83	0.42	0.83
IT–EN	16.96	64.77	11.85	1.12	3.98	0.49	0.50	0.34
EN–IT	25.34	62.15	8.75	1.00	1.35	0.47	0.50	0.45
DE–EN	22.05	65.34	5.86	1.92	1.10	1.26	0.31	2.16
EN–DE	24.39	65.38	4.86	2.28	0.5	1.08	0.10	1.40

Table 3.4: Distributions of alignment types for different language pairs (%)

These findings are also confirmed by the statistics obtained from the IWSLT gold-standard Chinese–English data as shown in Table 2.1,[10] where a similar distribution

[7] More specifically, we performed 5 iterations of Model 1, 5 iterations of HMM, 5 iterations of Model 3, and 5 iterations of Model 4.

[8] `http://opennlp.sourceforge.net/`.

[9] Note that a 1-to-0 alignment may denote a failure to capture a 1-to-n alignment with $n > 1$.

[10] Note that the gold-standard data also contain n-to-1 and m-to-n alignments, while using GIZA++ can only produce 1-to-n alignments due to its asymmetric nature.

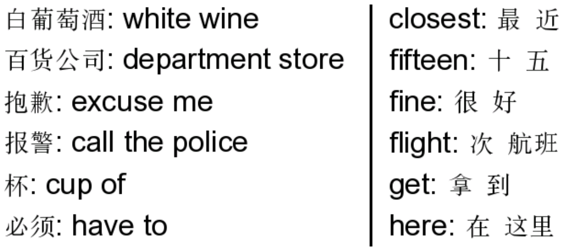
for 1-to-n ($n > 1$) alignments is observed. The main difference is that the automatic aligners tend to produce more 1-to-n alignments, while human annotators tend to generate more m-to-n alignments.

3.3 Bootstrapping Word Alignment via Word Packing

Our approach (cf. (Ma et al., 2007b)) consists of packing consecutive words together when we believe they correspond to a single word in the other language. This bilingually motivated packing of words changes the basic unit of the alignment process, and simplifies the task of automatic word alignment. We thus minimise the number of 1-to-n alignments in order to obtain more comparable segmentations in the two languages. In this section, we present an automatic method that builds upon the output from an existing automatic word aligner. More specifically, we (i) use a word aligner to obtain 1-to-n alignments, (ii) extract candidates for word packing, (iii) estimate the reliability of these candidates, (iv) replace the groups of words to pack by a single token in the parallel corpus, and (v) re-iterate the alignment process using the updated corpus. The first three steps are performed in both directions, and produce two bilingual dictionaries (source-to-target and target-to-source) of groups of words to pack.

3.3.1 Candidate Extraction

In the following, we assume the availability of an automatic word aligner that can output alignments $A_{\mathbf{f}\rightarrow\mathbf{e}}$ and $A_{\mathbf{e}\rightarrow\mathbf{f}}$ for any sentence pair (f_1^I, e_1^J) in a parallel corpus. We also assume that $A_{\mathbf{f}\rightarrow\mathbf{e}}$ and $A_{\mathbf{e}\rightarrow\mathbf{f}}$ contain 1-to-n alignments. Our method for repacking words is very simple: whenever a single word is aligned with several consecutive words, they are considered as candidates for repacking. Formally, given an alignment $A_{\mathbf{f}\rightarrow\mathbf{e}}$ between f_1^I and e_1^J, if the alignment between a sequence of target words v_k and one single source word f_i is denoted as $\{j_1, \cdots, j_\phi\} \rightarrow i = a_k$, with $v_k = \{e_{j_1}, \ldots, e_{j_\phi}\}$ and $\forall m \in [\![1, \phi - 1]\!] (\phi \geq 2)$, $j_{m+1} - j_m = 1$, then the alignment a_k between f_{a_k} and the sequence of words v_k is considered a candidate for word repacking. The same goes for $A_{\mathbf{e} \rightarrow \mathbf{f}}$. Some examples of such 1-to-n alignments between Chinese and English (in both directions) that we can derive automatically are displayed in Figure 3.1.



白葡萄酒: white wine	closest: 最 近
百货公司: department store	fifteen: 十 五
抱歉: excuse me	fine: 很 好
报警: call the police	flight: 次 航班
杯: cup of	get: 拿 到
必须: have to	here: 在 这里

Figure 3.1: Examples of 1-to-n word alignments between Chinese and English

3.3.2 Candidate Reliability Estimation

Of course, the process described above is error-prone and if we want to change the input to the word aligner, we need to make sure that we are not making harmful modifications.[11] We thus additionally evaluate the reliability of the candidates we extract and filter them before inclusion into our bilingual dictionary. To perform this filtering, we use two simple statistical measures. In the following parts, a_k or (f_{a_k}, v_k) denotes a candidate.

The first measure we consider is co-occurrence frequency ($COOC(f_{a_k}, v_k)$), i.e. the number of times f_{a_k} and v_k co-occur in the bilingual corpus. This very simple measure is frequently used in association-based approaches (Melamed, 1997; Tiedemann, 2003). The second measure is the alignment confidence, defined as

$$AC(a_k) = \frac{c(a_k)}{COOC(f_{a_k}, v_k)}, \tag{3.1}$$

[11] Consequently, if we compare our approach to the problem of collocation identification, we may say that we are more interested in precision than recall (Smadja et al., 1996). However, note that our goal is not recognising specific sequences of words such as compounds or collocations; rather it is making (bilingually motivated) changes that simplify the alignment process.

where $c(a_k)$ denotes the number of alignments proposed by the word aligner that are identical to a_k. In other words, $AC(a_k)$ measures how often the aligner aligns f_{a_k} and v_k when they co-occur. We also impose that $|v_k| = \phi \leq n$, where n is a fixed integer that may depend on the language pair (between 3 and 5 in practice). The rationale behind this is that it is very rare to obtain a reliable alignment between one word and n consecutive words when n is high.

The candidates are included in our bilingual dictionary if and only if their measures are above some fixed thresholds t_{COOC} and t_{AC}, which allow for the control of the size of the dictionary and the quality of its contents. Some other measures including the Dice coefficient (van Rijsbergen, 1979) could be considered; however, it has to be noted that we are more interested here in the filtering than in the discovery of alignments, since our method builds upon existing aligners. Moreover, we will see that even these simple measures can lead to an improvement in the alignment process in an MT context (cf. Section 3.6).

3.3.3 Bootstrapped Word Repacking

Once the candidates are extracted, we repack the words in the bilingual dictionaries constructed using the method described above; this provides us with an updated training corpus, in which some word sequences have been replaced by a single token. This update is totally naive; if an entry (f_{a_k}, v_k) is present in the dictionary and matches one sentence pair (f_1^I, e_1^J) (i.e. f_{a_k} and v_k are respectively contained in f_1^I and e_1^J), then we replace the sequence of words v_k with a single token which becomes a new lexical unit.[12] Note that this replacement occurs even if no alignment is found between f_{a_k} and v_k for the pair (f_1^I, e_1^J). This is motivated by the fact that the filtering described above is quite conservative; we trust the entry (f_{a_k}, v_k) to be correct. This update is performed in both directions. It is then possible to run the word aligner using the updated (simplified) parallel corpus, in order to obtain new

[12] In case of overlap between several groups of words to replace, we select the one with the highest confidence (according to t_{AC}).

alignments. By performing a deterministic word packing, we simplify the estimation of the fertility parameters associated with fertility-based models.

Word packing can be applied several times; once we have grouped some words together, they become the new basic unit to consider, and we can re-run the same method to get additional groupings. However, we have not seen in practice much benefit from running it more than twice (few new candidates are extracted after two iterations).

It is also important to note that this process is bilingually motivated and strongly depends on the language pair. For example, *white wine*, *excuse me*, *call the police*, and *cup of* (cf. Figure 3.1) translate respectively as *vin blanc*, *excusez-moi*, *appellez la police*, and *tasse de* in French. Those groupings would not be found for a language pair such as French–English, which is consistent with the fact that they are less useful for French–English than for Chinese–English in an MT perspective.

3.3.4 Word Unpacking and Phrase-Based SMT Decoding

The bidirectional grouping approach can improve the quality of alignment and correspondingly improve the quality of phrase extraction and the estimation of related parameters. In the decoding stage, given that the input is not packed and the language model is also trained on unpacked word segmentations, we need to undertake "**word unpacking**" before estimating the parameters. The unpacking process in PB-SMT is performed following the phrase extraction process. Specifically, in a log-linear PB-SMT system, the phrase translation probabilities and lexical reordering models are re-estimated based on relative frequencies; the lexical weighting probabilities are calculated based on the lexical translation distribution with word packing.

The unpacking step is particularly necessary in the context of bilingual word packing, i.e. both source and target sentences are packed, given that the language models are trained on texts without word packing. If we constrain the word packing process by only packing the source language, the word unpacking step could be avoided and word-lattice decoding could be utilised instead (cf. Section 3.4.3).

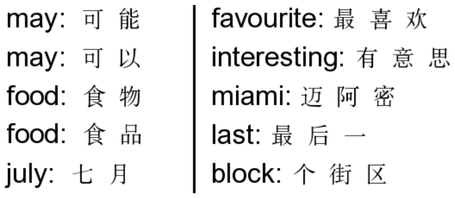
3.4 Bilingually Motivated Word Segmentation

3.4.1 Word Segmentation as an Alignment Problem

The approach proposed in Section 3.3 can be applied to word segmentation by only packing the source language. The only assumption is that the sentence to be segmented can be split into basic writing units (e.g. characters for Chinese and kana for Japanese). The notation in Section 3.3 can be easily adapted for this task. Given a Chinese sentence f_1^I consisting of I characters $\{f_1, \ldots, f_I\}$ and an English sentence e_1^J consisting of J words $\{e_1, \ldots, e_J\}$, $A_{\mathbf{e}\rightarrow\mathbf{f}}$ will denote a English–Chinese word-to-character alignment between e_1^J and f_1^I. Since we are primarily interested in 1-to-n alignments, $A_{\mathbf{e}\rightarrow\mathbf{f}}$ can be represented as a set of links a_1^K connecting one single English word e_{a_k} and a few Chinese characters v_k. The set v_k is empty if the word e_j is not aligned to any character in f_1^I.

3.4.2 Bootstrapped Word Segmentation

We use the same approach proposed in Section 3.3.1 to extract candidate words. Our method for Chinese word segmentation is as follows: whenever a single English word is aligned with several consecutive Chinese characters, they are considered candidates for grouping. Some examples of such 1-to-n alignments between Chinese characters and English words derived automatically are displayed in Figure 3.2.



may:	可 能	favourite:	最 喜 欢
may:	可 以	interesting:	有 意 思
food:	食 物	miami:	迈 阿 密
food:	食 品	last:	最 后 一
july:	七 月	block:	个 街 区

Figure 3.2: Examples of 1-to-n word-to-character alignments between English words and Chinese characters

We can use the same measures proposed in Section 3.3.2 to estimate the reliability of the candidate words and apply the boostrapping approach in Section 3.3.3 to derive better word segmentation.

3.4.3 Word Lattice Decoding

Casting word segmentation as an alignment problem implies that word segmentation of a sentence depends not only on the current sentence to segment but also on the corresponding target language. In such a context, the word lattice representation is particularly suitable in the decoding stage which aims to search for the most likely target sentence.

Word Lattices

In the decoding stage, the various segmentation alternatives can be encoded into a compact representation of word lattices. A **word lattice** $G = \langle V, E \rangle$ is a directed acyclic graph that formally is a weighted finite state automaton. In the case of word segmentation, each edge is a candidate word associated with its weights. A straightforward estimation of the weights is to distribute the probability mass for each node uniformly to each outgoing edge.[13] The single node having no outgoing edges is designated as the "end node". An example of a word lattice for a Chinese sentence is shown in Figure 3.3.

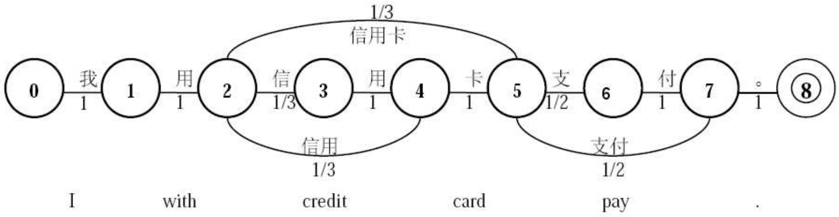


Figure 3.3: An example of a word lattice for a Chinese sentence

[13] We can also use language models to assign probabilities to each edge as in Xu et al. (2005). In this case, however, we have to rely on some segmented data to train the language model.

Word Lattice Generation

Previous research on generating word lattices relies on multiple monolingual segmenters (Xu et al., 2005; Dyer et al., 2008). One advantage of our approach is that the bilingually motivated segmentation process facilitates word lattice generation without relying on other segmenters. As described in Section 3.4.2, the update of the training corpus based on the constructed bilingual dictionary requires that the sentence pair meets the bilingual constraints. Such a segmentation process in the training stage facilitates the utilisation of word lattice decoding.

Phrase-Based Word Lattice Decoding

Given a Chinese input sentence f_1^I consisting of I characters, the traditional approach is to first determine the best word segmentation and perform decoding afterwards. In such a case, we first seek a single best segmentation, as in (3.2):

$$\hat{v}_1^K = \arg\max_{v_1^K, K}\{P(v_1^K|f_1^I)\} \tag{3.2}$$

Then in the decoding stage, we seek the translation of the most likely source segmentation, as in (3.3):

$$\hat{e}_1^J = \arg\max_{e_1^J, J}\{P(e_1^J|\hat{v}_1^K)\} \tag{3.3}$$

In such a scenario, some segmentations which are potentially optimal for translation may be lost. This motivates the need for word lattice decoding. The decision rules (3.2) and (3.3) can be rewritten as in (3.4)–(3.6):

$$\begin{aligned}
\hat{e}_1^J &= \arg\max_{e_1^J, J}\{\max_{v_1^K, K} P(e_1^J, v_1^K|f_1^I)\} && (3.4)\\
&= \arg\max_{e_1^J, J}\{\max_{v_1^K, K} P(e_1^J)P(v_1^K|e_1^J, f_1^I)\} && (3.5)\\
&\simeq \arg\max_{e_1^J, J}\{\max_{v_1^K, K} p(e_1^J)p(v_1^K|f_1^I)p(v_1^K|e_1^J)\} && (3.6)
\end{aligned}$$

where $p(e_1^J)$ is the language model, $p(v_1^K|f_1^I)$ is the word segmentation model and $p(v_1^K|e_1^J)$ is the translation model. Compared to the decision rule of the standard source-channel model for SMT (cf. Equation (2.2)), (3.6) has an additional segmentation model.[14]

Given the fact that the number of segmentations K grows exponentially with respect to the number of characters J, it is impractical to firstly enumerate all possible v_1^K and then to decode. However, it is possible to enumerate all the alternative segmentations for a substring of f_1^I which contains a very limited number of characters, making the utilisation of word lattices tractable in PB-SMT.

3.5 Experimental Setup

The MT experiments were primarily carried out using Data Set 3, i.e. the IWSLT 2007 Chinese–English dataset. Detailed corpus statistics are shown in Table 2.4. To test the adaptability of our algorithm, MT experiments were also conducted using Data Set 4 and 6, of which the detailed statistics are shown in Tables 2.5 and 2.6.

Given that our algorithm is directly optimised according to MT performance and we are primarily interested in the impact of the refined alignment with segmentation constraints on translation quality, we do not conduct an intrinsic evaluation.

3.6 Experiments

In this section, we present the experimental results applying our algorithm for bootstrapping the word alignment for both segmented data using monolingual segmenters (word packing) and data without segmentation on the source side (word segmentation).

[14] Although in (3.6) we use the approximation rather than the equality sign, it is appropriate to mention explicitly that inferring (3.6) directly from (3.5) is invalid in strict mathematical terms. Nonetheless, this approximation is necessary for reasons of tractability, and perhaps surprisingly, tends to work well in practice.

3.6.1 Word Packing

Results

The initial word alignments are obtained using the manually segmented IWSLT data and baseline word alignment configuration described in Section 2.5.3. From these, we build two bilingual 1-to-n dictionaries (one for each direction), and the training corpus is updated by repacking the words in the dictionaries, using the method presented in Section 3.3. As previously mentioned, this process can be repeated several times; at each step, we can choose to exploit only one of the two available dictionaries, if so desired. We then extract aligned phrases using the same procedure as for the baseline system, and the only difference is the basic unit we are considering. Once the phrases are extracted, we perform the estimation of the features of the log-linear model and unpack the grouped words to recover the initial words. Finally, MERT (Och, 2003) and decoding (Koehn et al., 2007) are performed.

	Bleu	Nist	Meteor
Baseline	**33.85**	6.3837	54.85
m=1. with C-E dict.	**35.02**	6.5145	55.55
m=1. with E-C dict.	34.83	6.4638	56.06
m=2. with C-E dict.	34.42	6.5553	55.74
m=2. with E-C dict.	**35.69**	6.6294	57.23

Table 3.5: Influence of word packing on translation quality of IWSLT 2007 test set

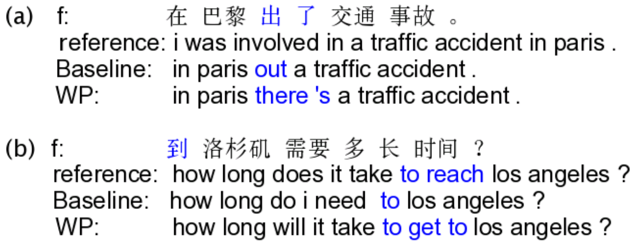
The various parameters of the method (n, t_{COOC}, t_{AC}, cf. Section 3.3.2) were optimised on the development set. We found out that it was enough to perform two iterations of repacking: the optimal set of values was found to be $n = 3$, $t_{AC} = 0.9$, $t_{COOC} = 20$ for packing English words and $t_{AC} = 0.3$, $t_{COOC} = 10$ for packing Chinese words in the first iteration, and $t_{AC} = 0.9$, $t_{COOC} = 8$ for packing English words and $t_{AC} = 0.7$, $t_{COOC} = 15$ for packing Chinese words in the second iteration.[15] In Table 3.5, we report the results obtained on the IWSLT 2007

[15] The parameters n, t_{AC}, and t_{COOC} are optimised for each step, and the alignment obtained using the best set of parameters for a given step is used as input for the following step.

test set, where m denotes the iteration. For each iteration, we first considered the inclusion of only the Chinese–English dictionary, and then only the English–Chinese dictionary.[16]

After the first step, we can already see an improvement over the baseline when considering one of the two dictionaries. More gain can be obtained by packing English words, leading to an increase of 1.17 absolute Bleu points (3.46% relative). The improvement is also confirmed by Nist and Meteor evaluation metrics. However, in the second step (m=2), the inclusion of the Chinese–English dictionary is harmful, probably because 1-to-n alignments have been captured during the first step. By including the English–Chinese dictionary only, we can achieve an increase of 1.84 absolute Bleu points (5.44% relative) over the initial baseline, which is statistically significant (p<0.01).[17]

The improvement in performance can be attributed to better word alignment after simplifying the alignment task after word packing, and subsequently higher quality phrasal translations for PB-SMT systems. Figure 3.4 gives two examples of better translation after word packing (WP). Phrases such as "there 's" and "get to" are packed words in the C-E bilingual dictionary so that valid phrase pairs can be included in the phrase table. Moreover, the probability of these valid phrase pairs can be boosted after word packing so that the correct hypothesis can survive in the decoding stage.



(a) f: 在 巴黎 出 了 交通 事故 。
reference: i was involved in a traffic accident in paris .
Baseline: in paris out a traffic accident .
WP: in paris there 's a traffic accident .

(b) f: 到 洛杉矶 需要 多 长 时间 ？
reference: how long does it take to reach los angeles ?
Baseline: how long do i need to los angeles ?
WP: how long will it take to get to los angeles ?

Figure 3.4: Translation examples using word packing

[16] We intend to consider including both Chinese–English and English–Chinese dictionaries in future work. However, in this case the parameter optimisation is more complicated as we need to jointly optimise the parameters for both directions.

[17] Note that this setting (using only Chinese dictionary for the first step and only the English dictionary for the second step) is also the best setting on the development set.

Quality of the Dictionaries To assess the quality of the extraction procedure, we simply manually evaluated the ratio of incorrect entries in the dictionaries. After one step of word packing, the Chinese–English and the English–Chinese dictionaries contain 13.6% and 8.6% incorrect entries respectively. After two steps of packing, they only contain 7.7% and 7.2% incorrect entries. More interestingly, some errors committed in the first step can be corrected in the second step, leading to a dictionary of higher quality. Some cases generally considered to be difficult such as m-to-n non-compositional phrasal alignments can also be identified in the second step.

Alignment Types

Intuitively, the word alignments obtained after word packing are more likely to be 1: 1 than before. Indeed, the word sequences in one language that usually align to one single word in the other language have been grouped together to form one single token. Table 3.6 shows the detail of the distribution of alignment types after one and two steps of automatic repacking.

		1: 0	1: 1	1: 2	1: 3	1: n ($n > 3$)
ZH-EN	Base.	28.48	**57.08**	11.08	2.11	1.25
	n=1	27.30	57.68	11.49	2.19	1.32
	n=2	17.45	**65.28**	10.68	4.12	2.48
EN-ZH	Base.	22.19	**59.60**	11.61	4.30	2.30
	n=1	21.27	62.91	9.82	3.74	2.25
	n=2	26.76	**63.78**	6.25	1.94	1.25

Table 3.6: Distribution of alignment types after word packing (%)

In particular, we can observe that the 1: 1 alignments are more frequent after the application of repacking: the ratio of this type of alignment has increased by 8.2% for Chinese–English and 4.18% for English–Chinese.

Influence of Word Segmentation Approach

To test the influence of the initial word segmentation on the process of word packing, we considered an additional segmentation configuration, based on the LDC segmenter.

	BLEU
Original segmentation	33.85
Original segmentation + Word packing	**35.02**
Automatic segmentation	31.74
Automatic segmentation + Word packing	**32.58**

Table 3.7: Influence of different Chinese segmentation on the performance of word packing (IWSLT 2007 data)

The results obtained are displayed in Table 3.7. The automatic segmenter leads to lower results than the human-corrected segmentation. However, the proposed method seems to be beneficial irrespective of the choice of segmentation. Indeed, we can also observe an improvement in the new setting: 0.84 points absolute increase in BLEU (2.65% relative), which is statistically significant (p<0.05). The experimental results of word packing reported so far are based on either manual segmentation or automatic segmentation using monolingual segmenters. In the next section, we show the results of directly using the word packing approach to perform word segmentation.

3.6.2 Word Segmentation

Results

The initial word alignments are obtained using the baseline configuration by segmenting the Chinese sentences into characters. From these we build a bilingual 1-to-n dictionary, and the training corpus is updated by grouping the characters in the dictionaries into a single word. To optimise the weights for the features of the log-linear PB-SMT system using MERT, we segment the Chinese sentences in the development set using a simple dictionary-based maximum matching algorithm to obtain a single best segmentation.[18] Finally, in the decoding stage, we use the same segmentation algorithm to obtain the single best segmentation on the test set,

[18] In order to save computing time, we used the same set of parameters obtained above to decode both the single-best segmentation and the word lattice. Recent work has been done on lattice-based MERT (Macherey et al., 2008).

and word lattices can also be generated using the bilingual dictionary. The various parameters of the method (n, t_{COOC}, t_{AC}, cf. Section 3.4.2) were optimised on the development set. One iteration of character grouping on the UN task was found to be enough; the optimal set of values was found to be $n = 3$, $t_{AC} = 0.0$ and $t_{COOC} = 0$, meaning that all the entries in the bilingually dictionary are kept. On the IWSLT data, we found that two iterations of character grouping were needed: the optimal set of values was found to be $n = 3$, $t_{AC} = 0.3$, $t_{COOC} = 8$ for the first iteration, and $t_{AC} = 0.2$, $t_{COOC} = 15$ for the second.

	Bleu	Nist	Meteor
CS	8.43	4.6272	37.78
Stanford	**10.45**	**5.0675**	36.99
Stanford-WordLattice	8.61	4.5456	37.15
BS-SingleBest	7.98	4.4374	35.10
BS-WordLattice	9.04	4.6667	**38.34**

Table 3.8: Bilingually motivated word segmentation on the UN task

As can be seen from Table 3.8, our bilingually motivated segmenter achieved statistically significantly ($p<0.03$) better results than character-based segmentation when enhanced with word lattice decoding.[19] Compared to the best in-domain segmenter, namely the Stanford segmenter on this particular task, our approach is inferior according to Bleu and Nist. We firstly attribute this to the small amount of training data, from which we are unable to obtain a high quality bilingual dictionary due to data sparseness problems. We also attribute this to the vast amount of named entity terms in the test sets, which is extremely difficult for our approach.[20] We expect to see better results when a larger amount of data is used and the segmenter is enhanced with a named entity recogniser.

On the IWSLT data (cf. Tables 3.9 and 3.10), the improvements over character-based segmentation are both statistically significant ($p<0.03$ for IWSLT 2006 test set

[19] Note that the Bleu scores are lower due to the number of references used (4 references, compared to 6 references for the IWSLT data), in addition to the small amount of training data available.

[20] As we previously point out, both ICT and Stanford segmenters are equipped with named entity recognition functionality. This may risk causing data sparseness problems on small training data. However, this is beneficial in the translation process compared to character-based segmentation.

	Bleu	Nist	Meteor
CS	19.31	6.1816	49.98
LDC	20.37	6.2089	49.84
LDC-WordLattice	20.15	6.2876	50.51
BS-SingleBest	18.65	5.7816	46.02
BS-WordLattice	**20.41**	**6.2874**	**51.24**

Table 3.9: Bilingually motivated word segmentation on the IWSLT 2006 task

	Bleu	Nist	Meteor
CS	29.59	6.1216	52.16
LDC	31.74	6.2464	54.03
LDC-WordLattice	**31.94**	6.2884	55.74
BS-SingleBest	30.23	6.0476	51.25
BS-WordLattice	31.71	**6.3518**	**56.03**

Table 3.10: Bilingually motivated word segmentation on the IWSLT 2007 task

and $p<0.01$ for IWSLT 2007 test set respectively). Compared to the best in-domain segmenter, the LDC segmenter, our approach yields a consistently good performance on both translation tasks. Moreover, the good performance is confirmed by all three evaluation measures. Note also that the MT system using automatic segmenters yields inferior translation results compared to that using manual segmentation (cf. Table 3.7 and 3.10).

From the experiments, we observe that adding in a word lattice mechanism into the PB-SMT system trained on monolingually segmented data does not help or even harms the system due to the mismatch between PB-SMT training and decoding. Previous research has already shown that combining phrase tables using different segmentations is necessary for word lattice decoding (Dyer et al., 2008). This confirms the advantage of our bilingually motivated segmentation, which facilitates word lattice decoding because it can generate different segmentations for the same Chinese sentence given different target English translations.

Parameter Search Graph

The reliability estimation process is computationally intensive. However, this can easily be parallelised. From our experiments, we observed that the translation results

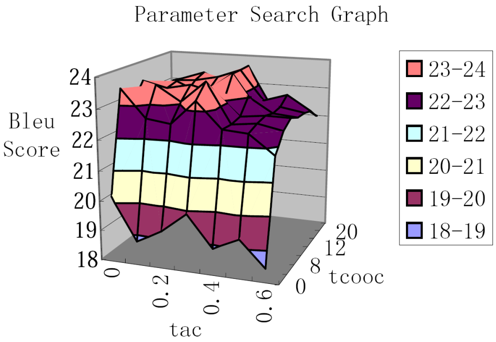


Figure 3.5: The search graph on the development set in the IWSLT task

are very sensitive to the parameters and this search process is essential to achieve good results. Figure 3.5 shows the search graph on the IWSLT data set in the first iteration step. From this graph, we can see that filtering of the bilingual dictionary is essential in order to achieve better performance.

Vocabulary Size

	voc.	char. voc	run. words
CS	6,057	6,057	1,412,395
ICT	16,775	1,703	870,181
LDC	16,100	2,106	881,861
Stanford	22,433	1,701	880,301
BS	**18,111**	**2,803**	**927,182**

Table 3.11: Chinese vocabulary size of the UN task (40K)

	voc.	char. voc	run. words
CS	2,742	2,742	488,303
ICT	11,441	1,629	358,504
LDC	9,293	1,963	364,253
Stanford	18,676	981	348,251
BS	**3,828**	**2,740**	**402,845**

Table 3.12: Vocabulary size of the IWSLT task (40K)

Our bilingually motivated segmentation approach has to overcome another challenge in order to produce competitive results, i.e. data sparseness. Given that our segmentation is based on bilingual dictionaries, the segmentation process can significantly increase the size of the vocabulary, which could potentially lead to a data sparseness problem when the size of the training data is small. Tables 3.11 and 3.12 list the statistics of the Chinese side of the training data, including the total vocabulary (voc), character vocabulary (char. voc, referring to the number of automatically generated "words" by word packing which contain only one single character) in voc, and the running words (run. words) when different word segmentations were used. From Table 3.11, we can see that our approach suffered from data sparseness on the UN task, i.e. a large vocabulary was generated, of which a considerable amount of characters still remain as separate words (15.48%). On the IWSLT task, since the dictionary generation process is more conservative, we maintained a reasonable vocabulary size, which contributed to the final good performance.

Scalability

The experimental results reported above are based on a small training corpus containing roughly 40,000 sentence pairs. We are particularly interested in the performance of our segmentation approach when it is scaled up to larger amounts of data. Given that the optimisation of the bilingual dictionary is computationally intensive, it is impractical to directly extract candidate words and estimate their reliability. As an alternative, we can use the obtained bilingual dictionary optimised on the small corpus to perform segmentation on the larger corpus. We expect competitive results when the small corpus is a representative sample of the larger corpus and large enough to produce reliable bilingual dictionaries without suffering severely from data sparseness.

	IWSLT06	IWSLT07
CS	23.06	30.25
ICT	23.36	33.38
LDC	**24.34**	**33.44**
Stanford	21.40	33.41
BS-SingleBest	22.45	30.76
BS-WordLattice	24.18	32.99

Table 3.13: Scaling up to 160K on IWSLT data sets (Bleu)

As we can see from Table 3.13, our segmentation approach achieved consistent

	160K	640K
CS	12.47	14.40
ICT	14.85	17.20
LDC	13.88	15.86
Stanford	15.26	16.94
BS-SingleBest	12.58	14.11
BS-WordLattice	**13.74**	**15.33**

Table 3.14: Scalability of bilingually motivated word segmentation on the UN task (Bleu)

results on both the IWSLT 2006 and 2007 test sets. On the UN task (cf. Table 3.14), our approach outperforms the basic character-based segmentation; however, it is still inferior compared to the other in-domain monolingual segmenters due to the low quality of the bilingual dictionary induced (cf. the **Results** reported at the beginning of this section).

Using Different Word Aligners

The above experiments rely on Giza++ to perform word alignment. We next show that our approach is not dependent on the word aligner given that we have a conservative reliability estimation procedure. Table 3.15 shows the results obtained on the IWSLT data set using the MTTK alignment tool (Deng and Byrne, 2005, 2006).

	IWSLT06	IWSLT07
CS	21.04	31.41
ICT	20.48	31.11
LDC	20.79	30.51
Stanford	17.84	29.35
BS-SingleBest	19.22	29.75
BS-WordLattice	**21.76**	**31.75**

Table 3.15: Bilingually motivated word segmentation on IWSLT data sets using MTTK (Bleu)

3.7 Related Work

Fertility-based models such as IBM Models 3, 4, and 5 allow for alignments between one word and several words in order to capture the pervasive 1-to-n correspondences. They can be seen as extensions of the simpler IBM Model 1 and 2 (Brown et al., 1993). Similarly, Deng and Byrne (2005) proposed an HMM framework with special attention to dealing with 1-to-n alignment, which is an extension of the original model of Vogel et al. (1996). However, as mentioned above, these models rarely question the monolingual tokenisation, i.e. the basic unit of the alignment process is the word. One alternative to extending the expressivity of one model (and usually its complexity) is to focus on the **input representation**; in particular, we argue that the alignment process can benefit from a simplification of the input, which consists of trying to reduce the number of 1-to-n alignments to consider. Note that the need to consider segmentation and alignment at the same time is also mentioned in Tiedemann (2003), and related issues are reported in Wu (1997).

Xu et al. (2004) were the first to question the use of word segmentation in SMT and showed that the segmentation proposed by word alignments can be used in PB-SMT to achieve competitive results compared to using monolingual segmenters. However, Xu et al. (2004) used word aligners to reconstruct a (monolingual) Chinese dictionary and reused this dictionary to segment Chinese sentences as other monolingual segmenters do. Our approach features the use of a bilingual dictionary and conducts segmentation based on the bilingual dictionary. In addition, we add a process which optimises the bilingual dictionary according to translation quality. Melamed (1997) presented an algorithm to identify non-compositional compounds from bilingual corpus and showed that treating each of them as one token can improve Word-Based SMT systems. Ittycheriah and Roukos (2005) introduced a discriminative word alignment model that incorporated segmentation information as a feature. Ma et al. (2007b) proposed an approach to improve word alignment by optimising the segmentation of both source and target languages. However, the reported experiments are based on a poor PB-SMT baseline and the issue of scalability is not addressed. Using word segmentation to improve word alignment and/or translation quality is also discussed in Bai et al. (2008), Chang et al. (2008) and Huang et al. (2008).

Xu et al. (2005) were the first to propose the use of word lattice decoding in PB-SMT, in order to address the problems that segmentation posed on the decoding. Dyer et al. (2008) extended this approach to hierarchical SMT systems and other language pairs. However, both methods require some monolingual segmentation in order to generate word lattices.[21] Our approach facilitates word lattice generation given that our segmentation is driven by the bilingual dictionary, making the training and decoding processes more coherent. More recently, Xu et al. (2008) proposed a Bayesian semi-supervised model for word segmentation by combining knowledge from both monolingual segmentation and bilingual word alignment. Our approach is not specifically designed for segmentation; it is a new mechanism that can automatically perform bidirectional segmentation optimisation. The boostrapping step can help the statistical aligners overcome the limitations posed by the first-order assumption. The bidirectional word packing can also overcome the shortcomings of the 1-to-n assumption inherent in IBM models by facilitating m-to-n alignment structures (cf. Fraser and Marcu (2007a)). On the other hand, our approach can be generalised to perform word segmentation without relying on any monolingual resources, such as dictionaries and the like.

3.8 Summary

In this chapter, we have introduced a simple yet effective method to pack words together in order to give a different and simplified input to automatic word aligners. We use a bootstrapping approach in which we first extract 1-to-n word alignments using an existing word aligner, and then estimate the confidence of those alignments

[21] Dyer (2009) represents an attempt to avoid using monolingual segmenters for European languages which have productive compounding.

to decide whether or not the n words have to be grouped; if so, this group is considered as a new basic unit. We can finally reapply the word aligner on the updated sentences. This approach can be used for bootstrapping word alignments based on any monolingual word segmentation; it can also be used for direct word segmentation without relying on any monolingual segmenters.

We evaluated the performance of our approach by measuring the influence of this process on the Chinese–English MT task based on the IWSLT 2007 evaluation campaign with a reasonally small amount of training data. We report a 1.84 points absolute (5.44% relative) increase in Bleu score over a standard PB-SMT system. We verified that this process actually reduces the number of 1-to-n alignments with $n \neq 1$, and that it is independent of the (Chinese) segmentation strategy. We then generalise our approach for direct Chinese word segmentation without relying on monolingual word segmenters and demonstrate that (i) our approach is not as sensitive to the domain as monolingual segmenters, and (ii) the SMT system using our word segmentation can achieve state-of-the-art performance. Moreover, our approach can be scaled up to larger data sets and achieves competitive results if the small data used is a representative sample of the larger one. Since our approach does not rely on monolingual segmenters, it is particularly useful for languages which lack manually segmented resources in the context of SMT.

However, this algorithm creates new words in each iteration and the incorrect packing of words can introduce noise into following iterations. In this sense, the segmentation in this algorithm acted as a hard constraint on word alignment. In the following three chapters, we will investigate methods to incorporate linguistically motivated segmentation information (i.e. syntactic dependencies) as soft constraints into discriminative and generative models. In these models, instead of directly packing a sequence of words into one, we avail of the syntactic dependencies in such a way that the dependency information is respected by the alignment model in both training and alignment.

Chapter 4

Syntactically Enhanced Discriminative Word Alignment

In this chapter, the role of syntactic dependencies as soft constraints is explored in discriminative word alignment. A two-stage word alignment approach is introduced to combine the merits of generative and discriminative word alignment models, where the first stage is to use generative models to produce a set of anchor alignments and in the second stage we take advantage of the syntactic dependencies induced by the available anchor alignments. We also show that our approach is flexible enough to be tuned according to different optimisation criteria.[1]

4.1 Introduction

Syntactic annotation of bilingual corpora, which can be obtained more efficiently and accurately with the advances in monolingual language processing, is a potential information source for word alignment tasks. For example, Part-of-Speech (POS) tags of source and target words can be used to tackle the data sparseness problem in discriminative word alignment (Liu et al., 2005; Blunsom and Cohn, 2006). Shallow parsing has also been used to provide relevant information for alignment (Sun et al.,

[1] The contents in this chapter has partly been published, albeit in a different form, in Ma et al. (2008a) and Ma et al. (2009a).

2000; Ren et al., 2007). Deeper syntax (e.g. phrase or dependency structures) has been shown to be useful in generative models (Wang and Zhou, 2004; Lopez and Resnik, 2005), association-based models (Ayan et al., 2004; Ozdowska, 2004) and even for syntactically motivated models such as ITG (Wu, 1997; Cherry and Lin, 2006b).

While generative models trained in an unsupervised manner can produce high-quality alignments given a reasonable amount of training data, it is difficult to incorporate richer features into such models (Moore, 2005). On the other hand, discriminative models are more flexible to incorporate arbitrary features. However, these models need a certain amount of annotated word alignment data, which is often subject to criticism since the annotation of word alignment is both difficult to obtain and a highly subjective task. Moreover, parameters optimised on manually annotated data are not necessarily optimal for MT tasks (Fraser and Marcu, 2007b; Ma et al., 2009a). Recent research has focused on combining the merits of both generative and discriminative models, most notably Fraser and Marcu (2006).

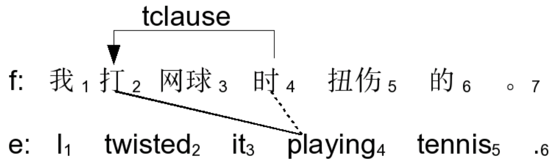

tclause
f: 我1 打2 网球3 时4 扭伤5 的6 。7
e: I1 twisted2 it3 playing4 tennis5 .6

Figure 4.1: An example of using syntactic dependencies for word alignment (1)

In this chapter, we introduce a simple yet flexible framework for word alignment (Ma et al., 2008a). To take advantage of the strength of generative models, we maintain a set of anchor alignments obtained using these models. We then incorporate syntactic features induced by the anchor alignments into a discriminative word alignment model. The syntactic features used are syntactic dependencies. This decision is motivated by the fact that if words tend to be dependent on each other, so does the alignment. If we can first obtain a set of reliable anchor links, we could take advantage of the syntactic dependencies relating unaligned words to aligned anchor words to expand the alignment. Figure 4.1 gives an illustrative example.

Note that the link (f_2, e_4) can be easily identified, but the link involving the fourth Chinese word (a function word denoting "time") (f_4, e_4) is hard. In such cases, we can make use of the dependency relationship ("tclause") between f_2 and f_4 to help the alignment process. Figure 4.2 shows another example, where the link (f_3, e_3) is easier to identify while (f_2, e_2) is difficult. Once the link (f_3, e_3) is established, the source syntactic dependency between f_2 and f_3 ("vmod"), and the target syntactic dependency between e_2 and e_3 ("vc") can be deployed to facilitate the alignment.

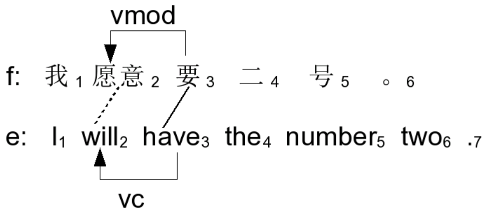

vmod
f: 我 1 愿意 2 要 3 二 4 号 5 。6
e: I1 will2 have3 the4 number5 two6 .7
vc

Figure 4.2: An example of using syntactic dependencies for word alignment (2)

We demonstrate via a series of experiments that using our word alignment approach in a PB-SMT system can significantly improve the system over a strong baseline. The experiments also show that dependency syntax is beneficial in word alignment. Given that the intrinsic quality of word alignment measured using F-score does not correlate well with PB-SMT performance measured using Bleu (Fraser and Marcu, 2007b), we conducted experiments that can directly optimise the word alignment according to Bleu score (Ma et al., 2009a). Experiments show that we can achieve higher performance using such an optimisation procedure and our word alignment approach is more flexible than state-of-the-art generative models in a PB-SMT framework.

4.2 Syntax for Word Alignment

To investigate the potential role of syntax in word alignment, an experiment is designed to investigate what types of words are harder to align, and whether the syntactic information can help the alignment of these words. We performed Chinese–

English word alignment on IWSLT 2007 data.[2] A precision-oriented word alignment was carried out using Giza++. In order to acquire a set of high-precision word alignments (anchor alignments), we run bidirectional word alignment and obtained the intersection (cf. Section 2.4.1). We then focused on the recall yielded by this high-precision alignment. The English and Chinese words were classified into a number of classes based on their corresponding POS tags.

POS	frequency	a. recall	dep	POS	frequency	a. recall	dep
,	83	0.55	0.57	PRP$	53	0.70	0.69
.	554	1.00	0.00	RB	177	0.65	0.47
CC	29	0.79	0.67	RBR	3	0.67	0.00
CD	44	0.84	0.71	**RP**	14	**0.36**	**0.67**
DT	322	**0.40**	**0.70**	**TO**	91	**0.25**	**0.72**
EX	7	0.71	0.00	UH	21	0.90	0.50
IN	192	**0.34**	**0.98**	VB	340	0.62	0.93
JJ	143	0.78	0.68	VBD	48	0.54	1.00
JJR	5	0.80	1.00	VBG	26	0.81	0.60
MD	132	0.76	1.00	VBN	29	0.76	0.43
NN	525	0.80	0.81	VBP	170	0.44	1.00
NNP	84	0.68	0.74	VBZ	135	0.27	1.00
NNPS	1	0.00	1.00	WDT	14	0.57	1.00
NNS	85	0.82	0.87	WP	41	0.51	0.75
POS	8	0.00	1.00	WRB	72	0.46	0.64
PRP	413	0.74	0.52				

Table 4.1: Syntactic dependency for aligning each type of English word using IWSLT gold-standard

The recall of anchor alignment (a. recall) is calculated against a set of manually aligned data. For words left unaligned, we check if they are involved in any dependencies with any anchor words (source or target words that are involved in the anchor alignments) and calculate the percentage of these words (dep). This quantity reflects to what extent the unaligned words can benefit from the dependency information between unaligned words and aligned anchor words. The results are shown in Table 4.1 and Table 4.2, where we can see that function words (e.g. words with POS tags DT (determiner), IN (preposition or subordinate conjunction), RP (particle), TO (to) for English and POS tags AD (adverb), BA ("ba" in a ba-construction), DEC ("de" in a relative clause), DEG (associative "de"), P (preposition excluding

[2] Here, we focus on Chinese–English word alignment. However, the methodology used may also apply to other language pairs.

"bei" and "ba") for Chinese) tend to be harder to align between Chinese and English, which is indicated by the low recall for these types of words.[3] The low recall for many of the function words is a strong indicator of the difficulty of alignment. More importantly, some function words such as adverbial (AD) and prepositions (P) in Chinese are very frequent and play an important role in the alignment process. At the same time, a large portion of the unaligned words are involved in dependencies with anchor words as shown in column "dep", implying that syntax is potentially beneficial in aligning these words. Similar phenomena can be observed for English sentences.

POS	frequency	a. recall	dep	POS	frequency	a. recall	dep
AD	195	**0.55**	**0.72**	NN	587	0.81	0.78
AS	35	0.09	0.78	NR	30	0.57	0.31
BA	14	**0.43**	**1.00**	NT	34	0.82	0.50
CC	23	0.91	0.50	OD	3	0.33	1.00
CD	165	0.60	0.48	**P**	125	**0.29**	**0.97**
CS	6	0.67	1.00	PN	495	0.79	0.48
DEC	57	**0.18**	**1.00**	PU	614	0.98	0.42
DEG	91	**0.11**	**1.00**	SB	4	0.00	1.00
DER	3	0.00	0.33	SP	134	0.07	0.95
DT	73	0.60	0.72	VA	62	0.81	0.83
JJ	39	0.69	0.50	VC	53	0.64	1.00
LC	11	0.36	1.00	VE	46	0.61	1.00
M	114	0.30	0.86	VV	776	0.70	0.95
MSP	6	0.50	1.00				

Table 4.2: Syntactic dependency for aligning each type of Chinese word using IWSLT gold-standard

4.3 Syntactically Enhanced Word Alignment Model

In this section, we describe our syntactically enhanced word alignment model, including the sub-models it can be decomposed into and how to interpolate these sub-models.

[3] (Deng and Gao, 2007) also pointed out the weakness of generative word alignment models in aligning function words. Their solution is to add constraints into generative word alignment models to guide the alignment of function words.

4.3.1 General Model

Given a source sentence $\mathbf{f} = f_1^I$ that consists of I Chinese words $\{f_1, \cdots, f_I\}$ and target sentence $\mathbf{e} = e_1^J$ which consists of J English words $\{e_1, \cdots, e_J\}$, we seek to find the optimal alignment $\hat{\mathbf{a}}$ such that in (4.1):

$$\hat{\mathbf{a}} = \underset{\mathbf{a}}{\operatorname{argmax}} P(\mathbf{a}|f_1^I, e_1^J) \tag{4.1}$$

We use a model (4.2) that directly models the links between source and target words, in a similar manner to Ittycheriah and Roukos (2005). The Chinese–English word alignment a_1^J is modelled as shown in (4.2). The difference of our model compared to Ittycheriah and Roukos (2005) is that we assume the availability of a partition over the target English word indices $\{1 \cdots J\}$ into the anchor word indices Δ and non-anchor word indices $\bar{\Delta}$. The partition can be induced from any alignment model or algorithm which can construct a set of reliable links, i.e. anchor alignments. Those target English words involved in these anchor alignments are anchor words, for which the set of indices is indicated with Δ. Here we assume that the partition between anchor words and non-anchor words is available;[4] we thereafter transform the alignment a_1^J into an anchor alignment $A_\Delta = \{i | j \in \Delta, j \rightarrow i = A_j\}$ containing the links involving anchor words, and a non-anchor alignment $A_{\bar{\Delta}}$ with the links involving non-anchor words, where $a_1^J = A_\Delta \cup A_{\bar{\Delta}}$, as shown in (4.2):

$$\begin{aligned} P(a_1^J|f_1^I, e_1^J) &\simeq p(A_\Delta, A_{\bar{\Delta}}|f_1^I, e_1^J) && (4.2)\\ &= p(A_\Delta|f_1^I, e_1^J) \times p(A_{\bar{\Delta}}|f_1^I, e_1^J, A_\Delta) && (4.3) \end{aligned}$$

where $p(A_\Delta|f_1^I, e_1^J)$ is an anchor alignment model and $p(A_{\bar{\Delta}}|f_1^I, e_1^J, A_\Delta)$ is a syntactically enhanced word alignment model, which can take both the available anchor links and relevant syntactic information into account. The anchor alignment model

[4] In principle, we can exploit all different possible partitions over the target words and obtain different anchor alignments; in our experiments, we have a single fixed set of anchor alignments obtained from intersected HMM or IBM alignment models.

is decomposed as the product of the probability of each link in the alignment A_Δ as in (4.4):

$$p(A_\Delta|f_1^I, e_1^J) = \prod_{j\in\Delta} p_\Delta(a_j|f_1^I, e_1^J) \tag{4.4}$$

The syntactically enhanced word alignment model, which aligns the remaining words with index set $\bar{\Delta}$ after anchoring, can be decomposed into an emission distribution (4.5) and a transition distribution (4.6) (cf. Section 4.3.3 below).

$$\begin{aligned} p(A_{\bar{\Delta}}|f_1^I, e_1^J, A_\Delta) &= \prod_{j\in\bar{\Delta}} p_{\bar{\Delta}}(a_j|f_1^I, e_1^J, a_1^{j-1}, A_\Delta) \qquad (4.5) \\ &\times \prod_{j=1}^{J} p_a(a_j|a_{j-1}, A_\Delta) \qquad (4.6) \end{aligned}$$

4.3.2 Anchor Word Alignment Model

Various models or algorithms can be used to identify a set of anchor alignments. The model $p_\Delta(a_j)$ then assigns a probability to the links in the set of anchor alignments.

We can use the asymmetric IBM models (Brown et al., 1993) for bidirectional word alignment and use the intersection as anchor alignments A_Δ. Subsequently, the confidence of each possible link under the anchor alignment model is modelled as in (4.7):

$$p_\Delta(a_j|f_1^I, e_1^J) = \begin{cases} \alpha & \text{if } a_j = A_j, \\ \frac{1-\alpha}{I} & \text{otherwise.} \end{cases} \tag{4.7}$$

The parameter α can be optimised on the development set. In our experiments we set $\alpha = 0.9$, implying that the obtained anchor link A_j for target word e_j is reliable and other possible links for e_j uniformly share the rest of the probability mass.

4.3.3 Syntactically Enhanced Word Alignment Model

Emission Distribution

The syntactically enhanced model is used to model the alignment of the words left unaligned after anchoring. We directly model the links between source and target words using a discriminative word alignment framework where various features can be incorporated. Given a source sentence f_1^I, target sentence e_1^J and anchor alignment A_Δ, the link a_j of each target word e_j is defined as in (4.8):

$$p_{\bar{\Delta}}(a_j|f_1^I, e_1^J, a_1^{j-1}, A_\Delta) \propto exp(\sum_{m=1}^{M} \lambda_m h_m(f_1^I, e_1^J, a_1^j, A_\Delta, T_\mathbf{f}, T_\mathbf{e})) \qquad (4.8)$$

In this definition, we assume that a set of highly reliable anchor alignments A_Δ have been obtained, and $T_\mathbf{f}$ (resp. $T_\mathbf{e}$) is used to denote the dependency structure for the source (resp. target) language. In such a framework, various machine learning techniques can be used for parameter estimation. The feature functions we used are described in Section 4.4.

Transition Distribution

Incorporating the anchor alignment, the first-order transition probability model can be defined as in (4.9):

$$p_a(a_j|a_{j-1}, A_\Delta) = \begin{cases} 1.0 & \text{if } j \in \Delta \text{ and } a_j = A_j, \\ \tilde{p}(a_j|a_{j-1}) & \text{otherwise.} \end{cases} \qquad (4.9)$$

Such a definition implies that the anchor alignment is always believed to be a correct alignment, and that maximum likelihood estimates obtained on a gold-standard word alignment corpus are used when the current word e_j is not involved in an anchor alignment. The estimation of $p_a(a_j|a_{j-1})$ is calculated following the homogeneous HMM model (Vogel et al., 1996). Under this model, we assume that the probability $p_a(a_j|a_{j-1})$ depends only on the jump width $(i-i')$, in order to make the parameters in the transition distribution independent of absolute word positions. Using a set of non-negative parameters $\{(i - i')\}$, the transition probability can be written as in (4.10):

$$\tilde{p}_a(a_j|a_{j-1}, A_\Delta) = \frac{c(i - i')}{\sum_{i''=1}^{I} c(i'' - i')} \tag{4.10}$$

where $c(i - i')$ is the count of the jump distance $|i - i'|$. Och and Ney (2003) refined this model by extending the HMM network with I empty words f_{I+1}^{2I}. The source word f_i has a corresponding empty word f_{i+I} (i.e. the position of the empty word encodes the previously visited target word). The constraints in (4.11)–(4.13) are enforced in the extended HMM network $(i \leq I, i' \leq I)$ involving the empty words:

$$\begin{aligned} p_a(i + I|i', I) &= p_0 \times \delta(i, i') & (4.11)\\ p_a(i + I|i' + I, I) &= p_0 \times \delta(i, i') & (4.12)\\ p_a(i|i' + I, I) &= p_a(i|i', I) & (4.13) \end{aligned}$$

where $\delta(i, i')$ is the Kronecker function, which is 1 if $i = i'$ and 0 otherwise. The parameter p_0 is the probability of a transition to the empty word, which can be estimated on the gold-standard word alignment corpus.

If a zero-order dependence is assumed, the emission models are the only information available to guide the word alignment.

4.3.4 Model Interpolation

The submodels in the general alignment model (4.2) are interpolated as in (4.14)–(4.16):

$$\begin{aligned} p(\mathbf{a}|f_1^I, e_1^J) &= \prod_{j\in\Delta} p_\Delta(a_j|f_1^I, e_1^J)^{1-\lambda} \times && (4.14) \\ & \prod_{j\in\bar{\Delta}} p_{\bar{\Delta}}(a_j|f_1^I, e_1^J, a_1^{j-1}, A_\Delta)^{1-\lambda} \times && (4.15) \\ & \prod_{j=1}^{J} p_a(a_j|a_{j-1}, A_\Delta)^\lambda && (4.16) \end{aligned}$$

The factor λ is used to weight the emission model and transition model probabilities so that the system can be optimised according to different objective functions.

4.4 Feature Functions for Syntactically Enhanced Model

The various features used in our syntactically enhanced model can be classified into three groups: statistics-based features, syntactic features and relative distortion features.

4.4.1 Statistics-Based Features

IBM Model 1 score

IBM Model 1 (Brown et al., 1993) is a position-independent word alignment model which is often used to bootstrap parameters for more complex models. Model 1 models the conditional distribution and uses a uniform distribution for the dependencies between a source word position j and target word position i, as in (4.17):

$$P(e_1^J, a_1^J|f_1^I) = \frac{p_l(J|I)}{(I+1)^J} \prod_{j=1}^{J} p_t(e_j|f_{a_j}) \qquad (4.17)$$

Log-Likelihood Ratio

	f_i	$\neg f_i$
e_j	a	b
$\neg e_j$	c	d

Table 4.3: Contingency table for association between word pairs

The log-likelihood ratio statistic has been found to be useful for modelling the associations between rare events (Dunning, 1993). It has also been successfully used to measure the associations between word pairs (Melamed, 2000; Moore, 2005). Given the contingency table in Table 4.3 where $\neg f_i$ denotes any source words but f_i (similarly for $\neg e_j$) and each cell in the table contains the count of co-occurrences between the pairs, the log-likelihood ratio can be defined as in (4.18):

$$G^2(f_i, e_j) = -2log\frac{B(a|a+b, p_1)B(c|c+d, p_2)}{B(a|a+b, p)B(c|c+d, p)} \tag{4.18}$$

where $B(k|n, p) = \binom{n}{k}p^k(1-p)^{n-k}$ are binomial probabilities. The probability parameters can be obtained using maximum likelihood estimates as in (4.19)–(4.20):

$$p_1 = \frac{a}{a+b} \qquad p_2 = \frac{c}{c+d} \tag{4.19}$$

$$p = \frac{a+c}{a+b+c+d} \tag{4.20}$$

POS Translation Probability

The POS tags can provide effective information for addressing the data sparseness problem in solely using the lexical features (Liu et al., 2005; Blunsom and Cohn, 2006). The POS translation probability can be easily obtained using maximum likelihood estimation from an annotated corpus, as in (4.21):

$$P(t_f|t_e) = \frac{c(t_f, t_e)}{c(t_e)} \tag{4.21}$$

where t_f is a Chinese word's POS tag and t_e is an English word's POS tag. $c(t_f, t_e)$ is the count of t_f and t_e being linked to each other in the corpus, and $c(t_e)$ is the frequency of t_e in the corpus.

4.4.2 Syntactic Features

The dependency type $r_{\mathbf{e}} \in \mathcal{R}_{\mathbf{e}} = \{\mathrm{SBJ}, \mathrm{ADJ}, \cdots\}$[5] (resp. $r_{\mathbf{f}} \in \mathcal{R}_{\mathbf{f}} = \{\mathrm{SBJ}, \mathrm{ADJ}, \cdots\}$) between two English (resp. Chinese) words e_j and $e_{j'}$ (resp. f_i and $f_{i'}$) in the dependency tree of the English sentence e_1^J (resp. Chinese sentence f_1^I) can be represented as a triple $\langle e_j, r_{\mathbf{e}}, e_{j'} \rangle$ (resp. $\langle f_i, r_{\mathbf{f}}, f_{i'} \rangle$), where e_j is the dependent and $e_{j'}$ is the head. Given f_1^I, e_1^J and their syntactic dependency trees $T_{\mathbf{f}}$, $T_{\mathbf{e}}$, if e_j is aligned to f_i, $e_{j'}$ aligned to $f_{i'}$, and there is a syntactic dependency between e_j and $e_{j'}$, according to the dependency correspondence assumption (Hwa et al., 2002), there exists a triple $\langle f_i, r_{\mathbf{f}}, f_{i'} \rangle$.

While we are not aiming to justify the feasibility of the dependency correspondence assumption by testing to what extent $r_{\mathbf{f}} = r_{\mathbf{e}}$ under the condition described above, we want to investigate whether these dependencies can help word alignment. Given the anchor alignment A_{Δ}, a candidate link (j, i) and the dependency trees, we design four classes of feature functions.

Agreement Features

The agreement features can be further classified into dependency agreement features and dependency label agreement features. Given a candidate link (j, i) and the anchor alignment A_{Δ}, the dependent-to-anchor Dependency Agreement (DA-1) feature function, which covers the case where the heads ($f_{i'}$ and $e_{j'}$) in source and

[5] The full list of English and Chinese dependency types that occurred in our data can be found in Appendix C and D respectively.

target dependency triples hold an anchor link, is defined as in (4.22):

$$h_{DA-1} = \begin{cases} 1 & \text{if } \exists \langle f_i, r_{\mathbf{f}}, f_{i'} \rangle, \langle e_j, r_{\mathbf{e}}, e_{j'} \rangle \text{ and } i' = A_{j'} \\ 0 & \text{otherwise.} \end{cases} \tag{4.22}$$

By changing the dependency direction between the words f_i and $f_{i'}$, an anchor-to-head Dependency Agreement feature (DA-2) where the dependents ($f_{i'}$ and $e_{j'}$) in the source and target dependency triples hold an anchor link, can be derived as in (4.23):

$$h_{DA-2} = \begin{cases} 1 & \text{if } \exists \langle f_{i'}, r_{\mathbf{f}}, f_i \rangle, \langle e_{j'}, r_{\mathbf{e}}, e_j \rangle \text{ and } i' = A_{j'} \\ 0 & \text{otherwise.} \end{cases} \tag{4.23}$$

We can define the dependent-to-anchor Dependency Label Agreement feature (DLA-1)[6] as in (4.24):

$$h_{DLA-1} = \begin{cases} 1 & \text{if } \exists \langle f_i, r_{\mathbf{f}}, f_{i'} \rangle, \langle e_j, r_{\mathbf{e}}, e_{j'} \rangle \text{ and } i' = A_{j'}, r_{\mathbf{f}} = r_{\mathbf{e}} \\ 0 & \text{otherwise.} \end{cases} \tag{4.24}$$

Similarly an anchor-to-head Dependency Label Agreement feature (DLA-2) can be obtained by changing the dependency direction.

These agreement features can be used to capture the complex dependencies we need to consult in the process of word alignment. Even for languages that are as different as Chinese and English, many dependencies between words are preserved across different languages (Hwa et al., 2002). The dependency label agreement is an even stronger indication of the structural similarities.

[6] Note that we used the same dependency parser, Maltparser (Nivre et al., 2007), for source and target language parsing.

Source Word Dependency Features

Given a candidate link (j, i) and the anchor alignment A_Δ, source language dependent-to-anchor dependency (SRC-1) features are used to capture the dependency label between a source word f_i and a source anchor word $f_{i'}(i' \in \Delta)$. For example, a feature function relating to dependency type "PRD" can be defined as in (4.25):

$$h_{SRC-1-PRD} = \begin{cases} 1 & \text{if } \exists\langle f_i, r_{\mathbf{f}}, f_{i'}\rangle \text{ and } r_{\mathbf{f}} = \text{`PRD'} \\ 0 & \text{otherwise.} \end{cases} \tag{4.25}$$

By changing the direction we can obtain the source language anchor-to-head dependency (SRC-2) feature function $h_{SRC-2-PRD}$.

This feature, which reflects the dependency between anchor words and non-anchor words, can be seen as a "tag" of the non-anchor words, helping to overcome the data sparseness problems as the POS tag features do.

Target Word Dependency Features

Target word dependency features can be defined in a similar manner as source word dependency features. For example, we have $h_{TGT-1-PRD}$ and $h_{TGT-2-PRD}$ to encode the dependent-to-anchor dependency (TGT-1) and anchor-to-head dependency (TGT-2) features respectively if the dependency label is "PRD" in the target language.

Source Anchor Feature

Given a candidate link (j, i), the source anchor feature defines whether the source word f_i is an anchor word, as in (4.26):

$$h_{SRC-ANC} = \begin{cases} 1 & \text{if } i \in A_\Delta \\ 0 & \text{otherwise.} \end{cases} \tag{4.26}$$

This feature indicates whether the current non-anchor target word is aligned to an anchor word in the source language. If so, it implies a 1-to-n alignment between the source anchor word and target words.

4.4.3 Relative Distortion Feature

We can design features encoding the relative distortion implied by this link by computing the relative position change with respect to the established anchor links (Ker and Chang, 1997). The relative position change D of a candidate link $l = (j, i)$ is formally defined as follows:

$$D(l) = min(|d_L|, |d_R|) \tag{4.27}$$

$$d_L = (j - j_L) - (i - i_L) \tag{4.28}$$

$$d_R = (j - j_R) - (i - i_R) \tag{4.29}$$

where (j_L, i_L) is the leftmost anchor link of l, and (j_R, i_R) is the rightmost anchor link of l.[7] The less the relative position changes, the more likely the candidate link is. With a set of anchor alignments, we can obtain the distribution of the relative position changes from an annotated corpus using maximum likelihood estimation. In our experiments, we classify the relative position changes implied by a candidate link into four groups: $D = 0$, $D = 1$ or 2, $D = 3$ or 4 and $D > 4$. Subsequently, the probability of each of the groups $p_d(D = 0)$, $p_d(D = 1, 2)$, $p_d(D = 3, 4)$ and $p_d(D > 4)$, which can be estimated from the gold-standard word alignment, are the values of the relative distortion feature.

4.5 Regression Using Support Vector Machines

The parameter estimation of the syntactically enhanced model (4.5), particularly the emission distribution (4.8) regarding the link of each non-anchor word, is one

[7] The location of the leftmost and rightmost anchor links is determined based on the target word positions, i.e. the leftmost and rightmost target anchor words of e_j.

of the most crucial components in our model. Given a target non-anchor word e_j, it should be aligned (linked) to the source word f_{a_j}. This process can be cast as a binary classification problem using a binary classification function $\psi((e_j, f_i))$, by assigning each pair (e_j, f_i) a class label -1 or 1, with 1 to indicate that e_j and f_i should be aligned and -1 otherwise.

In order to have estimate how likely the alignment of e_j and f_i is, we cast the process as a regression problem so that a score can be assigned to each candidate link. Several machine learning techniques can be applied for this purpose; here we use Support Vector Machines (SVM) (Burges, 1998; Vapnik, 1998) because of its repeatedly demonstrated high performance.

The SVM model is determined by combining a number of key training examples into a functional form which can act as a classifier. These key training examples are usually selected such that the resulting classifier can maximally separate the whole set of training examples. Given m training examples $\langle x_i, y_i \rangle)$ where x_i is the training instance and y_i is the class label associated with x_i, the SVM regression model is trained to assign a score to an input instance z using the formula in (4.30):

$$f(z) = \sum_{i=1}^{m} \alpha_i y_i \psi(x_i) \cdot \psi(z) + b \tag{4.30}$$

where ψ is the transformation function which transforms the input space into the feature space, and α_i is an variable associated with the training example $\langle x_i, y_i \rangle$ to be optimised in SVM training. The regression model is trained by minimising the empirical risk as in (4.31):

$$R_{emp} = \frac{1}{2n} \sum_{i=1}^{m} |y_i - f(x_i)| \tag{4.31}$$

where $\frac{|y_i - f(x_i)|}{2}$ is the loss function.

SVMs normally resort to more sophisticated kernel functions $K(x, z) = \psi(x) \cdot \psi(z)$ to implicitly transform the input space into a feature space of higher dimensions, while the computation is still done in the input space. In our experiments, we used the simplest form of kernels, i.e. linear kernels where the input space is not transformed.

4.6 Experimental Setup

This section describes the data and baseline system for word alignment and PB-SMT experiments. The detailed partition of the IWSLT gold-standard into training, development and test sets is also displayed.

4.6.1 Data and Baseline Systems

We presented in Section 2.5.2 the details of the data we used in this thesis. For this chapter, the gold-standard data we used is Data Set 1, i.e. manually annotated IWSLT devset3; the MT experiments were carried out using Data Set 3, i.e. the IWSLT 2007 Chinese–English data set. The IWSLT 2007 test set was used for evaluation. Detailed corpus statistics are shown in Table 2.3 and 2.4 (Page 36).

Details about POS tagging and dependency parsing are described in Section 2.5.3. We evaluate the word alignment according to both intrinsic and extrinsic measures. For the intrinsic evaluation, we focus on the macro-evaluation (cf. Section 2.5.1) of the symmetrised alignment using GDF heuristics (Koehn et al., 2003) given that the evaluation of each type of alignment is not our primary focus.

Word Alignment

We used the manually annotated word alignments of IWSLT devset3, which contains 502 sentence pairs. All the links are used as sure links. The first 300 sentence pairs were used for training the SVM regression model, the following 50 sentence pairs as a development set and the last 152 sentence pairs test set of intrinsic alignment quality. The various statistics for the gold-standard corpus are listed in Table 4.4.

		Chinese	English
Train	Sentences	300	
	Running words	2,231	2,704
	Vocabulary size	636	709
	Links	2773	
Dev.	Sentences	50	
	Running words	445	451
	Vocabulary size	205	212
	Links	555	
Eval.	Sentences	152	
	Running words	1,107	1,149
	Vocabulary size	394	413
	Links	1400	

Table 4.4: Split of Chinese–English word alignment gold-standard for syntactically enhanced word alignment

4.6.2 Alignment Training and Decoding

In our experiments, we treated anchor alignment and syntactically enhanced alignment as separate processes in a pipeline. The anchor alignments are kept fixed so that the parameters in the syntactically enhanced model can be optimised.[8] The SVM toolkit, namely SVM_light[9] was used to optimise the parameters in (4.8). Our model is constrained in such a way that each source word can only be aligned to one target word.

In SVM training, we transform each possible link involving the words left unaligned after anchoring into training instance. Positive examples (aligned words pairs in the gold-standard) are assigned the target value 1 and negative examples (unaligned pairs) -1. Using this training data, we can build a regression model to estimate the reliability of a link given a pair of words. The normalised functional margin obtained by applying the regression model serves as the emission probability in our word alignment model.

For the first-order transition model, we estimate the transition probability on our gold-standard word alignment training set. In decoding, the best alignment path

[8] Note that our anchor alignment does not achieve 100% precision (cf. Table 4.8). Since we performed precision-oriented alignment for the anchor alignment model, the errors in anchor alignment will not bring much noise into the syntactically enhanced model.

[9] http://svmlight.joachims.org/

is searched for using a Viterbi-style decoding algorithm. The interpolation factor λ can be optimised on the development set. When a zero-order transition model (a uniform transition distribution) is used, we constrain the emission probability by a threshold t, which is set as the minimal reliability score for each link. Again, t can be optimised according to the development set.

The decoding is performed separately in two directions (Chinese–English and English–Chinese), and we then obtain the refined alignments obtained using GDF heuristics as the final word alignment.

4.7 Experiments

Since we conduct both intrinsic and extrinsic evaluations on our word alignment model, we present experimental results on word alignment as well as on PB-SMT experiments.

4.7.1 Word Alignment

We perform word alignment bidirectionally using our approach and obtain the refined alignments using GDF heuristics. Our results are compared with two baseline word alignment systems based on generative word alignment models. The results are shown in Table 4.5. It can be seen that the syntactically enhanced model based on IBM Model 1, HMM or IBM Model 4 anchors achieved higher F-scores than the baseline generative word alignment models (Model 1, HMM and Model 4). It can also be seen that zero-order syntactic models are better in precision and first-order models are superior is recall. The best result achieved 2.24% relative increase in F-score compared to the baseline when we use IBM Model 4 intersection as the set of anchor alignments.

Model	Precision	Recall	F-score
Model 1	65.98	70.64	68.23
+Zero-order syntax	80.71	69.93	**74.93**
+First-order syntax	72.84	73.36	73.10
HMM	73.80	73.86	73.83
+Zero-order syntax	83.65	70.14	76.30
+First-order syntax	77.17	76.07	**76.62**
Model 4	75.87	78.14	76.99
+Zero-order syntax	84.59	74.50	**79.23**
+First-order syntax	80.21	77.57	78.87

Table 4.5: Macro-evaluation of syntactically enhanced word alignment (%)

Model	Precision	Recall	F-score
Model 1			
no syntax	75.90	67.50	71.46
with syntax	80.71	69.93	74.93
HMM			
no syntax	80.75	69.27	74.54
with syntax	83.65	70.14	76.30
Model 4			
no syntax	83.97	70.36	76.56
with syntax	84.59	74.50	**79.23**

Table 4.6: The effect of syntactic dependencies for the zero-order syntactically enhanced word alignment (%)

The Influence of Syntactic Dependencies on Word Alignment

The influence of incorporating syntactic dependencies into the word alignment process is shown in Tables 4.6 and 4.7. Syntax plays a positive role in all different anchor alignment configurations. The influence grows proportionally to the strength of the anchor alignment model. With the Model 4 intersection used as the set of anchor alignments, adding syntactic dependency features yields a 3.57% relative increase in F-score for the zero-order syntactically enhanced model and 1.97% relative increase for the first-order syntactically enhanced model.

By comparing Tables 4.6 and 4.7, we can see that syntax is less useful when a more powerful transition model is deployed, which is not surprising because the transition model itself encodes the dependency information over the states of the alignment.

Model	Precision	Recall	F-score
Model 1			
no syntax	73.99	71.71	72.83
with syntax	72.84	73.36	73.10
HMM			
no syntax	77.45	74.07	75.72
with syntax	77.17	76.07	76.62
Model 4			
no syntax	81.27	73.79	77.35
with syntax	80.21	77.57	**78.87**

Table 4.7: The effect of syntactic dependencies for the first-order syntactically enhanced word alignment (%)

The Influence of Anchor Alignment Quality

As we can see in Table 4.8, our approach to acquire anchor alignments achieved quite high precision (96.08% for Model 4), and the recall varies depending on the models used (only 39.00% for Model 1). Unsurprisingly, IBM Model 4 achieved the highest precision and recall, while Model 1 receiving the lowest.

Anchor model	Precision	Recall	F-score
Model 1	89.51	39.00	54.33
HMM	94.57	44.79	60.79
Model 4	**96.08**	**50.17**	**66.39**

Table 4.8: Macro-evaluation of anchor alignments (%)

To investigate the influence of the anchor alignment model on the alignment of non-anchor words, we first obtained the intersection of the words left unaligned after anchoring using each of the anchor alignment models. The alignment of these words is evaluated against the gold-standard alignments involving these words. The influence of the anchor alignment on the performance of the syntactically enhanced model can be seen in Tables 4.9 and 4.10. The performance of the syntactically enhanced model is closely related to that of the anchor alignment method.

As can be seen from Tables 4.8 and 4.9, IBM Model 4 anchoring achieves the best precision, so does the syntactically enhanced alignment; IBM Model 4 achieves the best recall, so does the syntactically enhanced alignment. Finally, the best alignment performances are obtained with IBM Model 4 anchoring. By comparing

Anchor model	Precision	Recall	F-score
Model 1	37.69	44.13	40.66
HMM	38.19	41.99	40.00
Model 4	**41.78**	**50.18**	**45.59**

Table 4.9: Impact of anchor alignment on zero-order syntactically enhanced word alignment (%)

Table 4.9 and 4.10, we can see that first-order models achieve higher performance than zero-order model in aligning these words with an advantage of notably higher recall.

Anchor model	Precision	Recall	F-score
Model 1	36.62	53.56	43.50
HMM	39.18	56.05	46.12
Model 4	**41.27**	**58.01**	**48.22**

Table 4.10: Impact of anchor alignment on first-order syntactically enhanced word alignment (%)

Weights of Different Feature Classes

The weights for the most discriminative features in each feature class in Chinese–English word alignment (using HMM intersection as anchor alignment) are shown in Table 4.11. All statistics-based features appear to be informative (positive weights). The dependency agreement (DA) and dependency label agreement (DLA) features are useful too. Two target dependency features are informative: PRD denoting "predicative" dependency, and AMOD denoting "adjective/adverb modifier" dependency.

4.7.2 Machine Translation

Table 4.12 shows the influence of our word alignment approach on MT quality.[10] From Table 4.12, our zero-order syntactically enhanced model based on Model 4 anchors achieved 1.84 absolute Bleu score (5.38% relative) improvement compared

[10] Note that the only difference between our MT system and the baseline PB-SMT system is the word alignment component.

	weight
Model 1 Score	0.1416
POS	0.0540
Log-likelihood Ratio	0.0856
relative distortion	0.0606
DA-1	0.0227
DLA-2	0.0927
TGT-1-PRD	0.0961
TGT-2-AMOD	0.0621

Table 4.11: Weights of some informative features in zero-order syntactically enhanced word alignment

to its baseline counterpart on the test set, which is statistically significant ($p < 0.002$). However, the first-order model suffers from overfitting problems, with a significant improvement on the development set and no improvement on the test set.

	dev	test
Model4	24.13	33.85
+Syntax-zero-order	25.41	**35.67**
+Syntax-first-order	25.47	33.70

Table 4.12: Performance of Phrase-Based SMT using syntactically enhanced word alignment optimising Bleu (Bleu)

Different Optimisation Criteria

The parameter t (threshold) for zero-order models (cf. Section 4.6.2) can be optimised with either F-score (OFscore) obtained on a gold-standard word alignment corpus, or Bleu score (OBleu) on a development set of an MT system as the objective function. Similarly for first-order models, parameters λ and p_0 (cf. Section 4.3.3 and 4.6.2) can be optimised according to these two criteria. Given that we have a very limited number of parameters to optimise (just two, i.e. $t_{c \to e}$ for Chinese–English and $t_{e \to c}$ for English–Chinese in the zero-order model, and three parameters, i.e. $\lambda_{c \to e}$, $\lambda_{e \to c}$ and p_0 in the first-order model), we used a simple greedy search algorithm by searching a predefined set of possible parameter settings. For example, we tried different value combinations from a set $\{-1.7, -1.6, \cdots, 0.0\}$ for

$t_{c \to e}$ and for $t_{c \to e}$. The search graph of these two parameters is shown in Figure 4.3, showing that tuning these two parameters can lead to significantly improved F-scores.

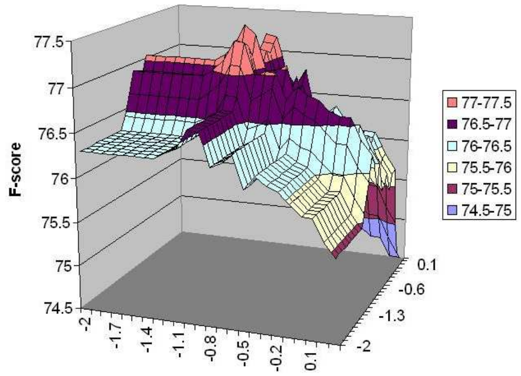


Figure 4.3: Search graph obtained when optimising F-score (%)

Table 4.13 shows the results obtained using different optimisation criteria using IBM Model 4 intersected alignments as anchors. For the zero-order model, the best parameter setting is $t_{c \to e} = -1.0$ and $t_{e \to c} = -0.6$ according to F-score; however, according to Bleu, the best parameters are $t_{c \to e} = -0.8$ and $t_{e \to c} = -0.9$. From Table 4.13, we can see that the Bleu score obtained when word alignment is optimised according to F-score is slightly inferior (not statistically significant) to that when optimised according to Bleu. The search graph of optimisation according to Bleu is shown in Figure 4.4. The different optimisation criteria do not have much impact on the F-score.

		Bleu		F-score	
		dev	test	dev	test
Zero-order	OFscore	24.74	35.21	77.49	79.23
	OBleu	25.41	**35.67**	76.98	**79.25**
First-order	OFscore	23.75	**34.32**	76.41	**78.87**
	OBleu	25.60	33.70	70.75	72.33

Table 4.13: Optimising syntactically enhanced word alignment according to Bleu and F-score (%)

For the first-order model, the best parameter setting is $\lambda_{c \to e} = 0.2$, $\lambda_{e \to c} = 0.2$ and $p_0 = 0.6$ according to F-score. However, according to Bleu, it is $\lambda_{c \to e} = 0.9$, $\lambda_{e \to c} = 0.3$ and $p_0 = 0.8$. From Table 4.13, we can observe that parameters optimised

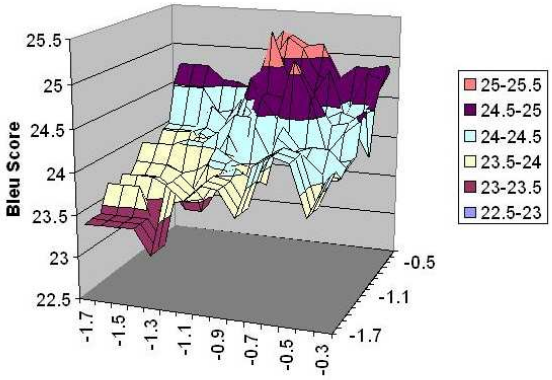


Figure 4.4: Search graph obtained when optimising Bleu

according to Bleu suffer from overfitting, yielding a low Bleu score on the test set and 6.54 absolute points lower F-score compared to the system optimised according to F-score. The word alignment optimised according to F-score not only yields a higher F-score, but also achieves better performance on the test set when used in a PB-SMT system. This reveals the necessity of more informative objective functions in parameter optimisation.

Phrase Extraction

To further investigate the impact of our word alignment on PB-SMT, we compared the extracted phrase table using our word alignment against the baseline phrase table. Figure 4.5 shows the size of the phrase tables when the system uses different word alignments. It can be observed that using the zero-order syntactically enhanced word alignment extracted 10.40% fewer phrase pairs (more word alignment links) when optimising Bleu compared to optimising F-score. The first-order word alignment which suffered from overfitting extracted 36.05% more phrase pairs (fewer word alignment links) when optimised according to Bleu compared to optimising F-score. All syntactically enhanced word alignments lead to larger phrase tables.[11] This indicates that some prior information on the number of links within the alignment is crucial in tuning and again a more informative objective function, e.g. a combination of Bleu and F-score, can hopefully lead to a better performance.

[11] Please note that the size of phrase table does not necessarily reflect the quality of the translation model; recent research (Koehn et al., 2009) uses entropy to measure the quality of the phrase table.

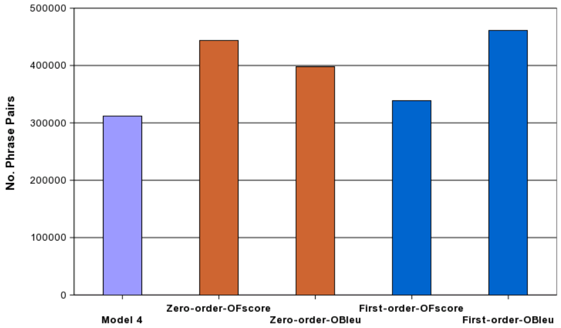


Figure 4.5: A comparison of the number of phrase pairs obtained using different models and different optimisation criteria

Scaling Up

To test the scalability of our approach, we added in a further 130K sentence pairs from the HIT corpus provided for the IWSLT 2008 evaluation campaign. The parameters obtained from the IWSLT 2007 corpus were re-used in these experiments. Table 4.14 shows the results. For the zero-order syntactically enhanced model optimised according to Bleu, we observed an increase of 1.69 absolute (6.05% relative) Bleu points over the baseline on the development set; on the test set, however, no improvement was achieved. For the first-order model, given that the parameters we obtained on the IWSLT 2007 data set by optimising Bleu suffered from overfitting, the consequence can also be seen on the experiments using this larger data set. From these results, the limitation of the optimisation process can be seen and a more informative objective function is needed to achieve better performance.

		dev	test
Baseline-Model4		27.05	**35.65**
Syntax-zero-order	OFscore	26.93	35.35
	OBleu	**28.74**	35.47
Syntax-first-order	OFscore	27.05	35.16
	OBleu	28.17	34.95

Table 4.14: Scaling up syntactically enhanced word alignment (Bleu)

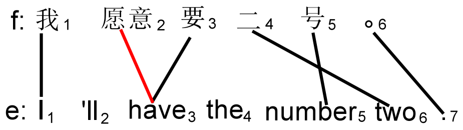

f: 我1 愿意2 要3 二4 号5 。6
e: I1 'll2 have3 the4 number5 two6 .7

Figure 4.6: An example of IBM Model 4 word alignment

4.7.3 Manual Evaluation

Some manual evaluation of the word alignment and MT output was undertaken. Figure 4.6 shows an example of IBM Model 4 word alignment where the links (f_2, e_3) (indicated with the red line) and (f_0, e_4) (the English word "the" is aligned to NULL) are incorrect. Both of these two words are function words which are normally believed to be hard to align.

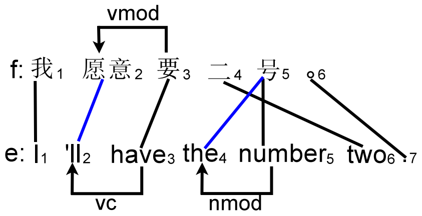

vmod
f: 我1 愿意2 要3 二4 号5 。6
e: I1 'll2 have3 the4 number5 two6 .7
vc
nmod

Figure 4.7: An example of word alignment with syntactic dependencies

Using our two-stage syntactically enhanced word alignment model, the alignment results are shown in Figure 4.7. Two links (f_2, e_2) and (f_5, e_4) (indicated with blue lines) are correctly recalled. We also show the dependency $\langle f_2, vmod, f_3\rangle$ in the Chinese sentence, and dependencies $\langle e_2, vc, e_3\rangle$ and $\langle e_4, nmod, e_5\rangle$ in the English sentence which contributed to the alignment process.

Figure 4.8 exhibits three translation examples using PB-SMT systems with IBM Model 4 word alignment (Baseline) and zero-order syntactically enhanced word alignment (Syntax) respectively. The focus phrases or words in the Chinese sentence are highlighted in blue and the corresponding translation using the Baseline system and Syntax system are also highlighted in blue. Examples (a) and (b) demonstrate that the PB-SMT system constructed using our syntactically enhanced word alignment has the advantage of selecting better phrase translations. Example (c) shows that the PB-SMT system using the syntactically enhanced word alignment

(a) f: 请 给 我 两 张 二 楼 座位 的 票 。
reference: two seats in the upper deck , please .
Baseline: can i have two balcony seats , please .
Syntax: please give me two seats on the second floor .

(b) f: 您 是 在 这儿 用餐 还是 带走 ？
reference: is that for here or take out ?
Baseline: are you here meal or take-out ?
Syntax: are you eat here or take it out ?

(c) f: 有 近路 吗 ？
reference: is there a shortcut ?
Baseline: do you have 近路 ?
Syntax: is there a shorter way ?

Figure 4.8: Translation examples using IBM Model 4 and zero-order syntactically enhanced word alignments

model has better coverage, i.e. some unknown words according to the Baseline can be correctly translated by the Syntax system.

4.8 Related Work

Our syntactically enhanced model is a discriminative word alignment model where syntactic features may be incorporated. Some previous research also tried to make use of syntax in word alignment. Wang and Zhou (2004) investigated the benefit of monolingual parsing for alignment. They learned a generalised word association measure (crosslingual word similarities) based on monolingual dependency structures and improved alignment performance over IBM Model 2 and certain heuristic-based models. Cherry and Lin (2006b) used dependency structures as soft constraints to improve word alignment in an ITG (Wu, 1997) framework. Compared to these models, our approach directly takes advantage of dependency relations as they are transformed into feature functions incorporated into a discriminative word alignment framework.

Fraser and Marcu (2007b) proposed a semi-supervised model that can take advantage of both generative and discriminative models. However, in their model word alignment is still a standalone component in a PB-SMT system and cannot be tuned for PB-SMT performance. Lambert et al. (2007) attempted to tune a discriminative word alignment model directly with MT in mind. Our work investigates the tuning of word alignment that takes advantage of both generative and discriminative word alignment models. Ma et al. (2008a) is a preliminary presentation of our word alignment framework; however, their word alignment was only tuned according to AER and the improvement for the PB-SMT system was not statistically significant. Our work shows that by tuning word alignment according to PB-SMT performance, we can achieve significantly better results.

4.9 Summary

In this chapter, we proposed a model that can facilitate the incorporation of syntax into word alignment, and measured the combination of a set of syntactic features. Experimental results showed that syntax can be useful in word alignment, and is especially effective in improving the recall. We also observed that in our word alignment framework, the two sub-models are closely related and the quality of the anchor alignment model plays an important role in system performance.

Our model can be tuned according to different end-tasks. Experimental results show that this model is superior to generative word alignment models in terms of both intrinsic and extrinsic quality. We observed a 2.99% relative increase in F-score compared to the baseline system. Using our word alignment in a PB-SMT system yields a 5.38% relative increase in Bleu score.

In the next chapter, we investigate the role of syntactic dependencies for generative word alignment models. A set of syntactically constrained HMM word-to-phrase alignment models will be presented and the efficient parameter estimation procedures for these models will be described.

Chapter 5

Syntactically Constrained HMM Word-to-Phrase Alignment Models

In Chapter 4, we showed that syntactic information can be used to enhance a discriminative word alignment model. Current work is dedicated to an investigation of the role of syntactic dependencies in a generative word alignment model, i.e. HMM word-to-phrase alignment (Deng and Byrne, 2005, 2008). We choose this model as our starting point for two reasons. Firstly, this model can produce high-quality word alignments while maintaining an efficient parameter estimation procedure. Secondly, the implementation of this model in the open-source toolkit MTTK (Deng and Byrne, 2006) offers a good baseline for comparison and reproducibility of our experimental results.[1]

5.1 HMM Word-to-Phrase Alignment Model

The HMM word-to-phrase alignment model performs simultaneous segmentation and alignment while maintaining the efficiency of the models. It models the process

[1] Part of the work in this chapter was conducted while the author visited Cambridge University Engineering Department under the supervision of Dr. Bill Byrne.

of how each of the source words generates a target phrase in sequence, as opposed to word-to-word alignment models (Brown et al., 1993; Vogel et al., 1996) which model the process of how each source word generates a target word. By modelling this process, it extends the HMM word-to-word alignments with stronger modelling power, i.e. the fertility phenomena or the 1-to-n alignments can be explicitly covered. It additionally insists that the target phrases generated by source words be consecutive so that efficient parameter estimation procedures can be deployed. Therefore, this model sets a good example of addressing the tradeoffs between modelling power and modelling complexity. This model can also be seen as a more generalised case of the HMM word-to-word model (Vogel et al., 1996; Och and Ney, 2003) because this model can be reduced to an HMM word-to-word model by restricting the generated target phrase length to one.

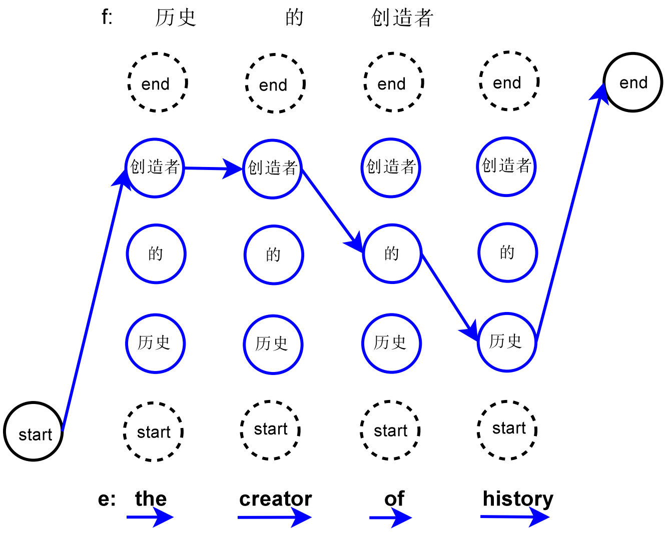


Figure 5.1: An example of an HMM word-to-word alignment trellis

Figure 5.1 is an example of an HMM word-to-word alignment trellis, where the target words are the generated observation sequence and the source words are the hidden states (cells with circles). Under this model, the target sequence is generated word-by-word; therefore, only staying in the same state can allow a 1-to-n alignment, e.g. “the” and “creator” should be both aligned to the third Chinese word.

The HMM word-to-phrase alignment models offer another route to 1-to-n alignment as shown in Figure 5.2, where the red cells and arrows indicate the generation

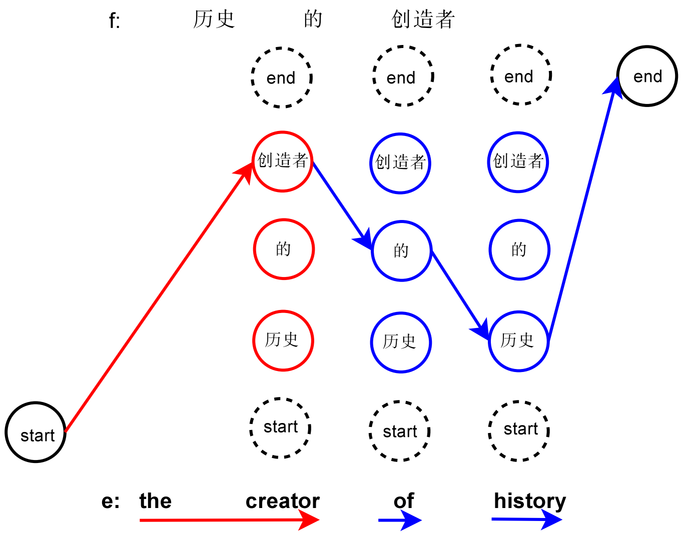


Figure 5.2: An example of HMM word-to-phrase alignment trellis

of a target phrase "the creator" from one source word. Note that introducing phrase generation does not prohibit the generation shown in Figure 5.1. Given a source sentence f_1^I, a target sentence e_1^J and the number of target words in each target phrase ϕ, we denote each cell in the trellis with a triple $\langle j, i, \phi \rangle$, implying that a source word f_i generates a target sequence $e_{j-\phi+1}, e_{j-\phi}, \cdots, e_j$ which consists of ϕ words and ends with e_j. As shown in Figure 5.3, two paths (start$\rightarrow$ $\langle 2, 3, 2 \rangle \rightarrow \langle 3, 2, 1 \rangle \rightarrow \langle 4, 1, 1 \rangle$ $\rightarrow$end and start$\rightarrow$ $\langle 1, 3, 1 \rangle \rightarrow \langle 2, 3, 1 \rangle \rightarrow \langle 3, 2, 1 \rangle \rightarrow \langle 4, 1, 1 \rangle$ $\rightarrow$end) can lead to the same correct alignment results; however, HMM word-to-phrase alignment has a stronger descriptive power that shows the advantages of tackling more complicated alignment structures. Figure 5.3 depicts the full layout of the HMM word-to-phrase alignment trellis, where the generated target phrase length ϕ can range from one to four and each possible value of ϕ gives rise to a layer in the trellis, e.g. the layer with $\phi = 2$ is denoted with red cells (the layers with $\phi = 3$ and 4 are not depicted due to space limits).

Formally, a phrase count variable K is introduced to indicate that the target sentence $\mathbf{e}$ is segmented into a sequence of consecutive phrases: $\mathbf{e} = v_1^K$, where v_k represents the k^{th} phrase in the target sentence. The assumption that each phrase v_k generated as a translation of one single source word is consecutive is made to allow efficient parameter estimation. Similarly to word-to-word alignment models, a

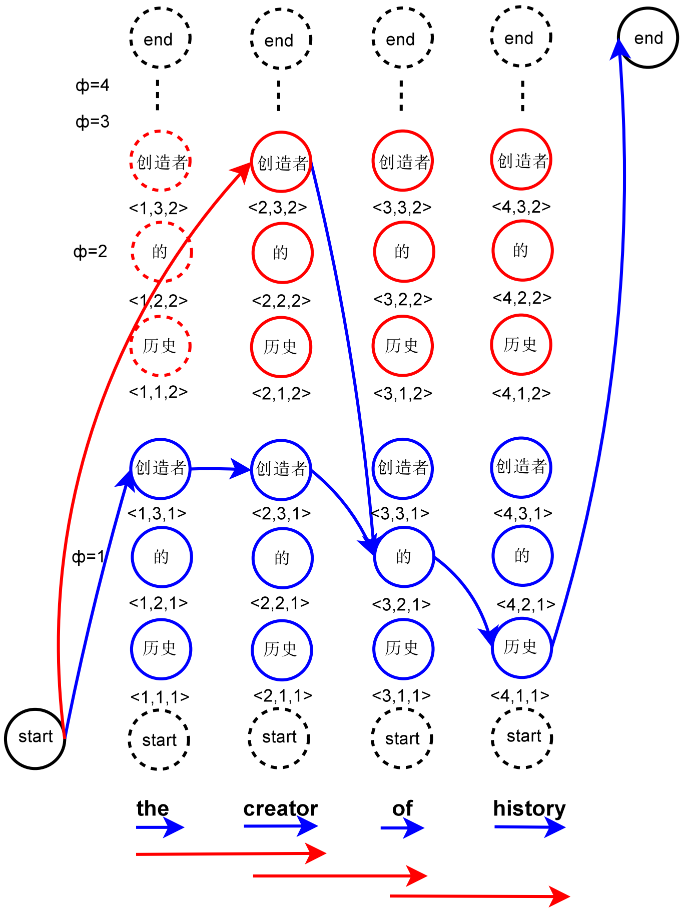


Figure 5.3: Relation between HMM word-to-word and word-to-phrase alignment models

variable a_1^K is introduced to indicate the correspondence between the target phrase index and a source word index: $k \rightarrow i = a_k$ indicating a mapping from target phrase v_k to source word f_{a_k}. A random process ϕ_k is used to specify the number of words in each target phrase subject to the constraints $J = \sum_{k=1}^{K}$, implying that the total number of words in the phrases agrees with the target sentence length J.

The insertion of target phrases that do not correspond to any source words is also modelled. This is done by allowing a target phrase to be aligned to a non-existent source word f_0 (NULL). Formally, to indicate whether each target phrase is aligned to NULL or not, a set of Kronecker functions $\varepsilon_1^K = \{\varepsilon_1, \cdots, \varepsilon_K\}$ is introduced (Deng and Byrne, 2008): if $\varepsilon_k = 0$, then NULL $\rightarrow v_k$ indicating that target phrase v_k is aligned to NULL; if $\varepsilon_k = 1$, then $f_{a_k} \rightarrow v_k$ indicating that target phrase v_k is aligned to source word f_{a_k}. The intuition behind introducing this Kronecker function ε_1^K is to model the tendency that a target phrase v_k should be aligned to NULL through the probability $p_\varepsilon(\varepsilon_k = 0) = p_0$ ($p_\varepsilon(\varepsilon_k = 1) = 1 - p_0$), where in practice p_0 is normally set as a constant independent of the target phrase v_k, e.g. $p_0 = 0.4$ for Chinese–English and $p_0 = 0.2$ for English–Chinese in MTTK, implying that there are more English words aligned to NULL than Chinese words. In HMM models, p_0 is used to indicate the probability of a transition to NULL as described in Section 4.3.3 and integrated into the HMM transition model as in (4.11)–(4.13).

To summarise, an alignment $\mathbf{a}$ in an HMM word-to-phrase alignment model consists of the elements in (5.1):

$$\mathbf{a} = (K, \phi_1^K, a_1^K, \varepsilon_1^K) \tag{5.1}$$

The modelling objective is to define a conditional distribution $P(\mathbf{e}, \mathbf{a}|\mathbf{f})$ over these alignments. Following Deng and Byrne (2008), $P(\mathbf{e}, \mathbf{a}|\mathbf{f})$ can be decomposed into a phrase count distribution (5.2) modelling the segmentation of a target sentence into phrases, an transition distribution (5.3) modelling the dependencies between the current link and the previous links, and a word-to-phrase translation distribution (5.4) to model the degree to which a word and a phrase are translational to each other.

$$\begin{aligned} P(\mathbf{e}, \mathbf{a}|\mathbf{f}) = P(v_1^K, K, a_1^K, \varepsilon_1^K, \phi_1^K|\mathbf{f}) &= P(K|J, \mathbf{f}) && (5.2) \\ &\times P(a_1^K, \varepsilon_1^K, \phi_1^K|K, J, \mathbf{f}) && (5.3) \\ &\times P(v_1^K|a_1^K, \varepsilon_1^K, \phi_1^K, K, J, \mathbf{f}) && (5.4) \end{aligned}$$

The **phrase count distribution** (5.2) is modelled using a single parameter distribution shown in (5.5). The scalar $\eta \geq 1$ is used to control the number of target phrases, in that larger values of η favour many short segmentations over the target sentence. For example, in MTTK the default value for η is 8.

$$P(K|J,\mathbf{f}) = P(K|J,I) \propto \eta^K (\eta \geq 1) \tag{5.5}$$

The **transition distribution** (5.3) is modelled as a first-order Markov process as shown in (5.6):

$$\begin{aligned} P(a_1^K, \varepsilon_1^K, \phi_1^K | K, J, \mathbf{f}) &= \prod_{k=1}^{K} P(a_k, \varepsilon_k, \phi_k | a_{k-1}, \phi_{k-1}, \varepsilon_{k-1}, K, J, \mathbf{f}) \\ &= \prod_{k=1}^{K} p_a(a_k | a_{k-1}, \varepsilon_k; I) \times p_\varepsilon(\varepsilon_k) \times p_n(\phi_k; \varepsilon_k \cdot f_{a_k}) \end{aligned} \tag{5.6}$$

where $\varepsilon_k \cdot f_{a_k}$ is shorthand for (5.7):

$$\varepsilon_k \cdot f_{a_k} = \begin{cases} f_{a_k} & \varepsilon_k = 1 \\ \text{NULL} & \varepsilon_k = 0 \end{cases} \tag{5.7}$$

$p_a(a_k|a_{k-1}, \varepsilon_k; I)$ is a first-order transition model for source length I with the link of previous phrase a_{k-1} as conditioning.[2] The conditioning also includes ε_k in order to model the transition into or out of NULL states (cf. Section 4.3.3). $p_n(\phi_k; \varepsilon_k \cdot f_{a_k})$ is the target phrase length model which can be viewed as a form of source word fertility. It specifies the probability that a source word f generates a target phrase of ϕ words. A distribution $p_n(\phi_k; \varepsilon_k \cdot f_{a_k})$ over the values $\phi = \{1, \cdots, N\}$ is maintained as a table for each source word. Finally, $p_\varepsilon(\varepsilon_k)$ is a simple distribution to model NULL alignments, where $p_\varepsilon(\varepsilon_k = 0) = p_0$ and $p_\varepsilon(\varepsilon_k = 1) = 1 - p_0$.

The **word-to-phrase translation distribution** (5.4) is formalised as in (5.8):

$$P(v_1^K | a_1^K, \varepsilon_1^K, \phi_1^K, K, J, \mathbf{f}) = \prod_{k=1}^{K} p_v(v_k | \varepsilon_k \cdot f_{a_k}, \phi_k) \tag{5.8}$$

Note here that we assume that the translation of each target phrase is conditionally

[2] Following the traditional notations in probability theory, we use a semicolumn to separate the variables from the parameters in a distribution.

independent of other target phrases given the individual source words. If we further assume that each word in a target phrase is translated independently of other words in the same phrase given the source word, we can derive an unigram translation model as shown in (5.9):

$$p_v(v_k|f_{a_k}, \varepsilon_k, \phi_k) = \prod_{j=1}^{\phi_k} p_{t_1}(v_k[j]|\varepsilon_k \cdot f_{a_k}) \tag{5.9}$$

If we assume that each word in a target phrase is translated with a dependence on the previously translated word in the same phrase given the source word, we derive the bigram translation model as shown in (5.10):

$$p_v(v_k|f_{a_k}, \varepsilon_k, \phi_k) = p_{t_1}(v_k[1]|\varepsilon_k \cdot f_{a_k}) \times \prod_{j=2}^{\phi_k} p_{t_2}(v_k[j]|v_k[j-1], \varepsilon_k \cdot f_{a_k}) \tag{5.10}$$

where $v_k[1]$ is the first word in phrase v_k and $v_k[j]$ is the j^{th} word in v_k. The intuition behind (5.10) is that the first word in v_k is firstly translated by $f_{a_k}$and the translation of the remaining words $v_k[j]$ in v_k from f_{a_k} are dependent on the translation of the previous word $v_k[j-1]$ from f_{a_k}. The use of a bigram translation model can address the coherence of the words within the phrase v_k so that the quality of phrase segmentation can be improved.

5.2 Syntactically Constrained HMM Word-to-Phrase Alignment Models

More details of the HMM word-to-phrase alignment model described in Section 5.1 can be found in Deng and Byrne (2008). In this section, we illustrate our syntactic extensions to the HMM word-to-phrase alignment model by elaborating on the components of our syntactically constrained HMM word-to-phrase alignment model and describing the parameter estimation procedures using the Baum-Welch (Forward-Backward) algorithm (Baum, 1972). We adopt the notations of Deng and

Byrne (2008) and focus on a syntactic extension to the original HMM word-to-phrase alignment model.

5.2.1 Syntactic Dependencies in HMM Word-to-Phrase Alignment

As can be seen in (5.5), the HMM word-to-phrase model has a very simple segmentation model over the target sentence. If there exist any syntactic dependencies within the target phrase, we can constrain the segmentation by availing of these syntactic dependencies.

As a preliminary test of our hypothesis, we performed dependency parsing on the GALE gold-standard word alignment corpus using Maltparser (Nivre et al., 2007). From Table 5.1, we can see that 82.54% of the consecutive English words have syntactic dependencies and 77.46% non-consecutive English words have syntactic dependencies in 1-to-2 Chinese–English word alignment.

	with dependency	no dependency	dep. ratio [%]
Consecutive	21108	4464	82.54
Non-consecutive	7178	2089	77.46

Table 5.1: Syntactic dependencies between English words in 1-to-2 Chinese–English word alignment

For English–Chinese word alignment, we can observe from Table 5.2 that 75.62% of the consecutive Chinese words and 71.15% of the non-consecutive Chinese words have syntactic dependencies.

	with dependency	no dependency	dep. ratio [%]
Consecutive	7969	2569	75.62
Non-consecutive	1184	480	71.15

Table 5.2: Syntactic dependencies between Chinese words in 1-to-2 English–Chinese word alignment

Given that a large proportion of the two consecutive words have syntactic dependencies for both English–Chinese and Chinese–English word alignment, a further investigation was conducted to uncover which dependency types were informative in a 1-to-2 alignment, i.e. given a dependency type between two target words, to what extent we believe these two target words should be aligned to one single source word. A straightforward measure is to calculate the ratio for each target language dependency type r implied by the 1-to-2 word alignment in the gold-standard word alignment $\mathcal{G}$, as shown in (5.11):

$$ratio = \frac{c(r;\mathcal{G})}{\sum_{r \in \mathcal{R}} c(r;\mathcal{G})} \tag{5.11}$$

where $c(r;\mathcal{G})$ is the count of dependency type r connecting two target words which are involved in 1-to-2 word alignment according to the gold-standard $\mathcal{G}$, and the denominator is the total count of all possible dependency types $\mathcal{R}$ involved in 1-to-2 word alignment.

Given that our goal is to identify which dependency type is more informative for a 1-to-2 alignment, this measure is not very useful because the most frequent dependency types such as NMOD and PMOD in English tend to receive higher scores according to (5.11). Alternatively, we replace the direct count in $c(r;\mathcal{G})$ with a normalised count $nc(r;\mathcal{G})$, as shown in (5.12):

$$nc(r;\mathcal{G}) = \frac{c(r;\mathcal{G})}{c(r)} \tag{5.12}$$

where $c(r;\mathcal{G})$ is calculated with respect to the gold-standard 1-to-2 word alignment, while $c(r)$ is the total count of dependency labels r in the target language, calculated without consulting the word alignment. Based on (5.12), we can obtain the normalised ratio for each dependency type as in (5.13):

$$nratio = \frac{nc(r;\mathcal{G})}{\sum_{r \in \mathcal{R}} nc(r;\mathcal{G})} \tag{5.13}$$

Table 5.3 shows the ratio and normalised ratio for each English dependency type. For consecutive English words, the ratio of dependency type NMOD is the highest; however, the normalised ratio is low (6.24%), for which COORD is the highest

	consecutive		non-consecutive		total	
Label	**ratio**	**nratio**	**ratio**	**nratio**	**ratio**	**nratio**
ADV	3.42	2.41	0.75	0.57	2.74	1.85
AMOD	1.78	5.01	0.00	0.00	1.33	8.61
CC	0.06	0.09	0.01	0.03	0.05	0.09
COORD	0.12	**19.95**	2.76	2.42	0.79	1.05
DEP	1.13	3.94	0.00	0.00	0.84	6.79
IOBJ	0.00	0.24	0.00	0.00	0.00	0.41
LGS	0.03	0.38	0.00	0.00	0.02	0.66
NMOD	**54.51**	6.24	**61.20**	16.49	**56.21**	9.02
OBJ	1.24	3.28	0.64	0.53	1.09	1.13
P	0.12	0.20	0.10	0.05	0.11	0.07
PMOD	17.64	13.01	24.09	12.44	19.27	10.36
PRD	0.47	2.64	0.59	2.29	0.50	2.04
PRT	1.52	15.73	0.07	19.14	1.15	**26.00**
SBJ	1.22	0.80	0.14	0.26	0.95	0.93
VC	7.79	9.16	7.76	**40.54**	7.78	15.73
VMOD	8.94	16.92	1.89	5.24	7.16	15.33

Table 5.3: English dependency types in 1-to-2 Chinese–English word alignment

indicating that NMOD is less informative to a 1-to-2 alignment than COORD despite NMOD being more frequent. For non-consecutive English words, again NMOD has the highest ratio; however, the label that has the highest normalised ratio is VC (40.54%) implying that a non-consecutive English verb group is more likely to be aligned to a single Chinese word. In general, English dependency label PRT is the most informative for Chinese–English 1-to-2 alignment.

	consecutive		non-consecutive		total	
Label	**ratio**	**nratio**	**ratio**	**nratio**	**ratio**	**nratio**
AMOD	5.48	6.13	0.34	0.64	4.82	6.15
DEP	22.27	17.41	6.59	5.14	20.24	14.88
NMOD	17.64	3.67	0.59	0.11	15.44	2.91
OBJ	2.21	4.30	1.60	0.96	2.13	1.98
P	1.33	1.44	0.34	0.08	1.20	0.48
PMOD	1.00	2.88	**78.21**	**78.00**	10.99	16.46
PRD	0.01	0.88	0.08	0.37	0.02	0.19
SBAR	19.32	**42.85**	3.63	11.25	17.29	**41.07**
SUB	1.88	1.93	0.08	0.05	1.65	1.26
VC	3.24	8.44	0.34	2.12	2.86	9.36
VMOD	**25.60**	10.06	8.19	1.29	**23.35**	5.25

Table 5.4: Chinese dependency types in 1-to-2 English–Chinese word alignment

Table 5.3 shows the ratio and normalised ratio for each Chinese dependency type with respect to English–Chinese 1-to-2 word alignment. For consecutive Chinese words, the ratio of dependency type VMOD is the highest; however, SBAR has the highest normalised ratio indicating that two consecutive Chinese words with one being a complementiser tend to be aligned to one English word. For non-consecutive Chinese words, again PMOD has both the highest ratio and normalised ratio implying that two non-consecutive Chinese words with one being preposition modifier is more likely to be aligned to one English word. In general, Chinese dependency label SBAR is the most informative for English–Chinese 1-to-2 alignment.

We hypothesise that making use of the dependency information can potentially constrain the behaviour of word alignment models to give improved performance. We now introduce the models that incorporate syntactic constraints and carry out a series of experiments in the next chapter to test our hypothesis.

5.2.2 Component Variables and Distributions

Syntactic dependency can be used to indicate the coherence of a phrase. We showed in Table 5.1 (resp. Table 5.2) the pervasive existence of syntactic dependencies between the words in English (resp. Chinese) phrases if they are aligned to one single Chinese (resp. English) word, which motivates the addition of syntactic constraints into the HMM word-to-phrase alignment model. To accomplish this, we constrain the word-to-phrase alignment model with a syntactic coherence model. Given a target phrase v_k consisting of ϕ_k words, we use the dependency label r_k between words $v_k[1]$ and $v_k[\phi_k]$ to indicate the level of coherence. The dependency labels are a closed set obtained from dependency parsers, e.g. using Maltparser, we have 20 dependency labels for English and 12 for Chinese in our data (cf. Appendix C and D). Therefore, we have an additional variable r_1^K associated with the sequence of phrases v_1^K to indicate the syntactic coherence of each phrase, as shown in (5.14):

$$\begin{aligned} P(r_1^K, v_1^K, K, a_1^K, \varepsilon_1^K, \phi_1^K | \mathbf{f}) &= P(K|J, \mathbf{f}) \\ &\times P(a_1^K, \phi_1^K, \varepsilon_1^K | K, J, \mathbf{f}) \\ &\times P(v_1^K | a_1^K, \varepsilon_1^K, \phi_1^K, K, J, \mathbf{f}) \\ &\times P(r_1^K | a_1^K, \varepsilon_1^K, \phi_1^K, v_1^K, K, J, \mathbf{f}) \end{aligned} \tag{5.14}$$

The **syntactic coherence distribution** (5.14) is simplified as in (5.15):

$$P(r_1^K | a_1^K, \varepsilon_1^K, \phi_1^K, v_1^K, K, J, \mathbf{f}) = \prod_{k=1}^{K} p_r(r_k; \varepsilon \cdot f_{a_k}, \phi_k) \tag{5.15}$$

The syntactic coherence model measures whether the target phrases are syntactically well-formed. Note that the coherence of each target phrase is conditionally independent of the coherence of other target phrases given the source words f_{a_k} and the number of words in the current phrase ϕ_k. A distribution over the values $r_k \in \mathcal{R} = \{\mathrm{SBJ}, \mathrm{ADJ}, \cdots\}$ ($\mathcal{R}$ is the set of dependency types for a specific language) is maintained as a table for each source word associated with all the possible lengths $\phi \in \{1, \cdots, N\}$) of the target phrase it can generate, e.g. we set $N = 4$ for the Chinese–English alignment and $N = 2$ for English–Chinese alignment in our experiments.

Given a target phrase v_k containing ϕ_k words, it is possible that there are no dependencies between the first word $v_k[1]$ and the last word $v_k[\phi_k]$. To account for this fact, we introduce a Kronecker function φ as in (5.16):

$$\varphi(v_k[1], \phi_k) = \begin{cases} 1 & \text{if } v_k[1] \text{ and } v_k[\phi_k] \text{ have syntactic dependencies} \\ 0 & \text{otherwise} \end{cases} \tag{5.16}$$

We can thereafter introduce a distribution $p_\varphi(\varphi)$, where $p_\varphi(\varphi = 0) = \zeta$ $(0 \leq \zeta \leq 1)$ and $p_\varphi(\varphi = 0) = 1 - \zeta$ with ζ indicating how likely it is that the first and final words in a target phrase do not have any syntactic dependencies. We can set ζ to a small number to favour target phrases satisfying the syntactic constraints and to a larger number otherwise. The introduction of this variable enables us to tune the model towards our different end goals. We can now refine (5.15) as (5.17):

$$P(r_1^K|a_1^K, \varepsilon_1^K, \phi_1^K, v_1^K, K, J, \mathbf{f}) = \prod_{k=1}^{K} p_r(r_k|\varphi; \varepsilon \cdot f_{a_k}, \phi_k) \times p_\varphi(\varphi) \tag{5.17}$$

where $p_r(r_k|\varphi; \varepsilon \cdot f_{a_k}, \phi_k) = 1$ if $\varphi = 0$ (the first and last words in the target phrase do not have syntactic dependencies), and $p_r(r_k|\varphi; \varepsilon \cdot f_{a_k}, \phi_k) = p_r(r_k; \varepsilon \cdot f_{a_k}, \phi_k)$ if $\varphi = 1$ (the first and last words in the target phrase have syntactic dependencies).

The definition of syntactic coherence proposed above does not cover the case where the conditioning variable $\phi_k = 1$. Since we cannot derive any syntactic dependencies when the target phrase contains only one word, we use a Kronecker function (5.18) to indicate the coherence:

$$r_k = \begin{cases} 1 & \text{if } v_k \text{ is coherent} \\ 0 & \text{otherwise} \end{cases} \tag{5.18}$$

Given this definition, to favour 1-to-1 alignments, we can express that v_k is completely coherent by insisting that the constraint in (5.19) be met, and we name this model the syntactically constrained HMM word-to-phrase alignment model 1 (SSH1).[3] Under this model, each target phrase (or word) v_k is considered as fully coherent.

$$p_r(r_k = 1; \varepsilon \cdot f_{a_k}, \phi_k = 1) = 1.0 \tag{5.19}$$

Similarly, to favour 1-to-n alignments by penalising the 1-to-1 cases,[4] we assume

[3] SSH is an abbreviation from Syntactically constrained Segmental HMM (Ostendorf et al., 1996) following the fact that HMM word-to-phrase alignment model is a Segmental HMM model (SH).

[4] Regarding the tradeoffs between 1-to-1 and 1-to-n links, we introduced our word packing algorithm in Chapter 3 to reduce the number of 1-to-n links, i.e. to increase the number of 1-

(5.20) and name this model the syntactic word-to-phrase model 2 (SSH2).

$$p_r(r_k = 1; \varepsilon \cdot f_{a_k}, \phi_k = 1) \propto p_n(\phi_k = 1; \varepsilon \cdot f_{a_k}) \tag{5.20}$$

where the coherence $p_r(r_k = 1; f_{a_k}, \phi_k = 1)$ of the target phrase (word) v_k is defined to be proportional to the probability of target phrase length $\phi_k = 1$ given the source word f_{a_k}. The intuition behind this model is that the syntactic coherence is strong iff the probability of the source f_{a_k} fertility $\phi_k = 1$ is high.

5.2.3 Parameter Estimation

The Forward-Backward Algorithm, a version of the EM algorithm, is specifically designed for unsupervised parameter estimation of HMM models. The parameters in our HMM word-to-phrase alignment model include the above-described component distributions, i.e. transition distribution, target phrase length distribution, unigram/bigram translation distribution. The Forward algorithm can be deployed to determine the likelihood of an observation sequence, which can be used for collecting posterior statistics for each component parameter. This algorithm starts with a uniform distribution for each component distribution in the HMM and iteratively updates the parameters in each distribution. In the context of word alignment, the observation sequence is the target words and this sequence is generated by some hidden states, i.e. the source words. The Forward procedure can be used to compute the likelihood of the generating the target words from the source words given current parameters. The parameters are updated in such a way that the likelihood is maximised.

to-1 links, by iteratively packing n words into one. In the first iteration of word packing, we can identify a considerable number of 1-to-n links and this number can be reduced in later iterations because there are more 1-to-1 correspondences after a few iterations of packing. In our syntactically constrained HMM word-to-phrase alignment model, we derive the word alignment in one pass; therefore, encouraging the identification of 1-to-n links is necessary given the pervasive existence of 1-to-n links in Chinese–English word alignment. Naturally if we use this syntactically constrained HMM word-to-phrase alignment model instead of IBM Model 4 in our word packing algorithm, we can encourage the identification of 1-to-n links in the first iteration and 1-to-1 links in later iterations.

The Forward algorithm is a kind of dynamic programming algorithm, i.e. an algorithm that uses a table to store intermediate values as it builds up the probability of the observation sequence. The Forward algorithm computes the likelihood by summing over the probabilities of all possible hidden state paths that could generate the observation sequence, but it does so efficiently by implicitly folding each of these paths into a single Forward trellis (Jurafsky and Martin, 2008). In the context of word alignment, each of the possible hidden state paths corresponds to an alignment structure.

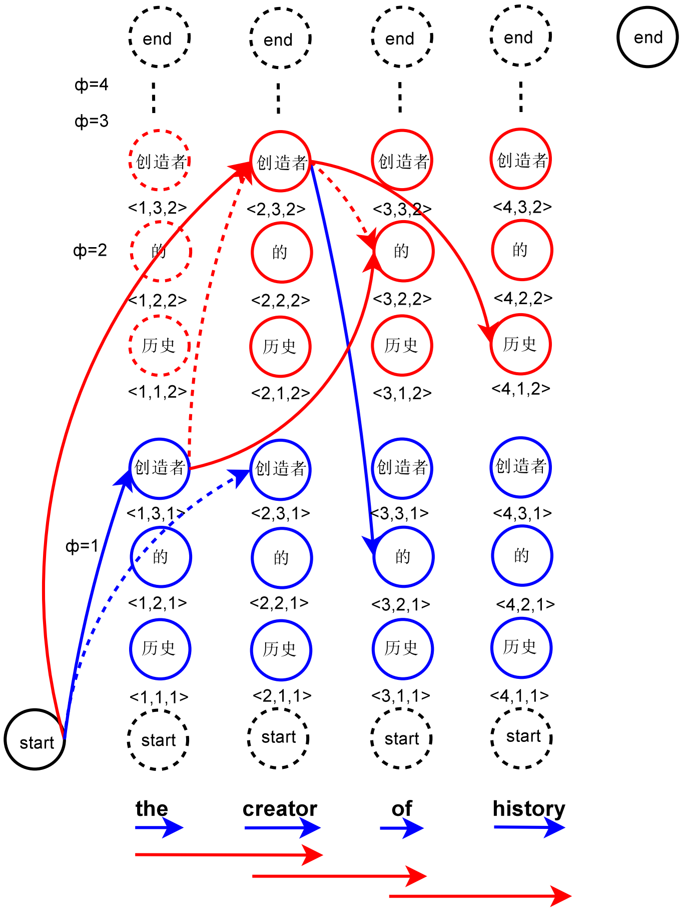


Figure 5.4: An example of valid paths in the HMM word-to-phrase alignment model Forward trellis

For the HMM word-to-word alignment model, the observation sequence and the sequence of the hidden states have the same length, i.e. for each target word, there is a corresponding source word aligned to it. For a sentence pair (f_1^I, e_1^J), the state space $\{(i, \varepsilon) : 1 \leq i \leq I, \varepsilon = 0 \text{ or } 1\}$ is created for HMM word-to-word alignment model and the Forward-Backward algorithm is carried out over a trellis of $2 \times I \times J$ cells.[5] For the HMM word-to-phrase alignment model, however, this does not hold because the target sentence is generated phrase-by-phrase rather than word-by-word. Given this, the state transition paths have to be properly constrained such that each target word can be generated only once. In Figure 5.4, the red and blue solid lines indicate valid transition while the dotted lines are invalid ones. For example, the cell $\langle 2, 3, 2\rangle$ generates the English phrase "the creator" and the cell $\langle 1, 3, 1\rangle$ generates the English word "the". The transition from $\langle 1, 3, 1\rangle$ to $\langle 2, 3, 2\rangle$, which causes a double generation of word "the", is invalid. Instead, the transitions from start to $\langle 1, 3, 1\rangle$ or to $\langle 2, 3, 2\rangle$ are both valid. For these models, a larger state space $\{(i, \phi, \varepsilon) : 1 \leq i \leq I, 1 \leq \phi \leq N, \varepsilon = 0 \text{ or } 1\}$ taking the various target phrase segmentations into account is created and the Forward-Backward algorithm is carried out over a trellis of $2 \times N \times I \times J$ cells. One of the advantages of the syntactic coherence model is that it does not change the state space of the baseline HMM word-to-phrase alignment model. The Forward statistic $\alpha_j(i, \phi, \varepsilon)$ is notated as in (5.21):

$$\alpha_j(i, \phi, \varepsilon) = \begin{cases} P(e_1^j, e_{j-\phi+1}^j \leftarrow f_i | f_1^I) & \varepsilon = 1 \\ P(e_1^j, e_{j-\phi+1}^j \leftarrow \text{NULL} | f_1^I) & \varepsilon = 0 \end{cases} \tag{5.21}$$

where $\alpha_j(i, \phi, \varepsilon)$ is the probability of being in state i with current target phrase length ϕ after seeing the first j observations and associated cell in the trellis is notated as $\langle j, i, \phi\rangle$. The value of $\alpha_j(i, \phi, \varepsilon)$ can be calculated recursively over the trellis as in (5.22):

$$\alpha_j(i, \phi, \varepsilon) = \{\sum_{i', \phi', \varepsilon'} \alpha_{j-\phi}(i', \phi', \varepsilon') p_a(i | i', \varepsilon; I)\} p_n(\phi; \varepsilon \cdot f_i)$$

$$\eta \cdot p_{t_1}(e_{j-\phi+1} | \varepsilon \cdot f_i) \prod_{j'=j-\phi+2}^{j} p_{t_2}(e_{j'} | e_{j'-1}, \varepsilon \cdot f_i) \cdot p_r(r_k; \varepsilon \cdot f_i, \phi) \tag{5.22}$$

[5] The reason why we have $2 \times I \times J$ cells rather than $I \times J$ cells is that each hidden state (source word) has an additional associated NULL state (cf. Section 4.3.3).

which sums up the probabilities of every path that could lead to the cell $\langle j, i, \phi \rangle$. Note that the syntactic coherence term $p_r(r_k; \varepsilon \cdot f_i, \phi)$ can efficiently be added into the Forward procedure. Similarly, the Backward probability $\beta_j(i, \phi, \varepsilon)$ is notated as in (5.23):

$$\beta_j(i, \phi, \varepsilon) = \begin{cases} P(e_{j+1}^J | e_{j-\phi+1}^j \leftarrow f_i, f_1^I) & \varepsilon = 1 \\ P(e_{j+1}^J | e_{j-\phi+1}^j \leftarrow \text{NULL}, f_1^I) & \varepsilon = 0 \end{cases} \tag{5.23}$$

where $\beta_j(i, \phi, \varepsilon)$ is the probability of being in state i with current target phrase length ϕ seeing the observations from $j + 1$ to the end. The value of $\beta_j(i, \phi, \varepsilon)$ can be calculated over the trellis as in (5.24) which sums up the probabilities of every path that starts from the cell $\langle j, i, \phi \rangle$, and leads to end state. Note also the syntactic coherence term $p_r(r_k; \varepsilon' \cdot f_{i'}, \phi')$ can also be integrated into Backward procedure efficiently.

$$\begin{aligned} \beta_j(i, \phi, \varepsilon) = \sum_{i', \phi', \varepsilon'} \beta_{j+\phi'}(i', \phi', \varepsilon') p_a(i' | i, h'; I) p_n(\phi'; \varepsilon' \cdot f_{i'}) \\ \eta \cdot p_{t_1}(e_{j+1} | \varepsilon' \cdot f_{i'}) \prod_{j'=j+2}^{j+\phi'} p_{t_2}(e_{j'} | e_{j'-1}, \varepsilon' \cdot f_{i'}) \cdot p_r(r_k; \varepsilon' \cdot f_{i'}, \phi') \end{aligned} \tag{5.24}$$

After the Forward recursion, the conditional probability of a sentence $\mathbf{e}$ given $\mathbf{f}$ can be calculated as in (5.25):

$$P(\mathbf{e}|\mathbf{f}) = \sum_{i', \varepsilon', \phi'} P(e_1^J, e_{J-\phi'+1}^J \leftarrow \varepsilon' \cdot f_{i'} | \mathbf{f}) = \sum_{i', \varepsilon', \phi'} \alpha_J(i', \phi', \varepsilon') \tag{5.25}$$

where $\sum_{i', \varepsilon', \phi'} \alpha_J(i', \phi', \varepsilon')$ sums up all the possible paths of the states throughout the observation sequence, i.e. all the possible alignment structures.

The probability that a target phrase $e_{j-\phi+1}^j$ is generated by any source word can be calculated as in (5.26):

$$\begin{aligned} P(\mathbf{e}, e_{j-\phi+1}^{j} \leftarrow \varepsilon \cdot f_i | \mathbf{f}) &= P(e_1^j, e_{j-\phi+1}^{j} \leftarrow \varepsilon \cdot f_i | f_1^I) \\ &\times P(e_{j+1}^J | e_{j-\phi+1}^{j} \leftarrow \varepsilon \cdot f_i, f_1^I) \\ &= \alpha_j(i, \phi, \varepsilon) \beta_j(i, \phi, \varepsilon) \end{aligned} \tag{5.26}$$

which corresponds to the product of Forward probability to the cell $\langle j, i, \phi \rangle$ and Backward probability from the cell $\langle j, i, \phi \rangle$.

With (5.25) and (5.26), we can compute the posterior probability of any target phrase generated by any source word. The posterior probability $\gamma_j(i, \phi, \varepsilon)$ that a target phrase $t_{j-\phi+1}^{j}$ is aligned to a source word $\varepsilon \cdot f_i$ can be calculated as in (5.28) by applying a simple Bayesian transformation:

$$\begin{aligned} \gamma_j(i, \phi, \varepsilon) &= P(e_{j-\phi+1}^{j} \leftarrow \varepsilon \cdot f_i | \mathbf{f}, \mathbf{e}) \\ &= \frac{P(\mathbf{e}, e_{j-\phi+1}^{j} \leftarrow \varepsilon \cdot f_i | \mathbf{f})}{P(\mathbf{e}|\mathbf{f})} \qquad (5.27) \\ &= \frac{\alpha_j(i, \phi, \varepsilon) \beta_j(i, \phi, \varepsilon)}{\sum_{i', \varepsilon', \phi'} \alpha_J(i', \phi', \varepsilon')} \qquad (5.28) \end{aligned}$$

Therefore, the posterior probability that a particular target phrase is aligned to a particular source word can be computed by the product of the Forward probability and Backward probability divided by the likelihood.

To reestimate the Markov transition matrix, we require the posterior probability of observing two consecutive target phrases. The probability that a phrase $e_{j-\phi'+1}^{j}$ and its successor $e_{j+1}^{j+\phi}$ are respectively generated by $\varepsilon' \cdot f_{i'}$ and $\varepsilon \cdot f_i$ can be calculated as in (5.29):

$$\begin{gathered} P(\mathbf{e}, e_{j-\phi+1}^{j} \leftarrow \varepsilon \cdot f_i, e_{j+1}^{j+\phi} \leftarrow \varepsilon \cdot f_i | \mathbf{f}) = \alpha_j(i', \phi', \varepsilon') \eta \\ p_a(i|i', \varepsilon; I) p_n(\phi; \varepsilon \cdot f_i) p_v(e_{j+1}^{j+\phi} | \varepsilon \cdot f_i, \phi) p_r(r_k; \varepsilon \cdot f_i, \phi) \beta_{j+\phi}(i, \phi, \varepsilon) \end{gathered} \tag{5.29}$$

Note the addition of the syntactic coherence term $p_r(r_k; \varepsilon \cdot f_i, \phi)$. The posterior probability $\xi_j(i', \phi', \varepsilon', i, \phi, \varepsilon)$ can be calculated as the ratio of (5.29) to (5.25) as shown in (5.30):

$$\xi_j(i', \phi', \varepsilon', i, \phi, \varepsilon) = \frac{P(\mathbf{e}, e_{j-\phi+1}^{j} \leftarrow \varepsilon \cdot f_i, e_{j+1}^{j+\phi} \leftarrow \varepsilon \cdot f_i | \mathbf{f})}{\sum_{i', \varepsilon', \phi'} \alpha_J(i', \phi', \varepsilon')} \tag{5.30}$$

5.2.4 EM Parameter Updates

As mentioned earlier, we use the Forward-Backward algorithm to estimate the parameters of the transition model, the target phrase length model, the translation model and the syntactic coherence model. The Expectation step accumulates fractional counts for each parameter during the Forward-Backward passes, and the Maximisation step normalises the counts in order to generate updated parameters. For the translation model, we only describe the EM procedures of unigram translation model. The estimation of the bigram translation model requires more considerations regarding smoothing; a detailed bigram translation model estimation mothod can be found in Deng and Byrne (2008).

Given a parallel text $\mathbf{T}$ used for training, in the E-step, the posterior counts of a source word f being translated into a target word e are accumulated over all the training sentences as in (5.31):

$$c(f, e) = \sum_{(\mathbf{f}, \mathbf{e}) \in \mathbf{T}} \sum_{i, j, \phi, f_i = f} \gamma_j(i, \phi, \varepsilon = 1) \tau_j(e, \phi) \tag{5.31}$$

where $\tau_j(e, \phi) = \sum_{j'=j-\phi+1}^{j} \delta(e, e_{j'})$, and $\delta(e, e_{j'})$ is Kronecker function with value 1 if e and $e_{j'}$ are the same words and 0 otherwise. The M-step normalises the posterior counts, as in (5.32):

$$p_{t_1}(e|f) = \frac{c(f, e)}{\sum_{e'} c(f, e')} \tag{5.32}$$

Similarly, the E-step for calculating the transition probability is shown in (5.33):

$$c(i, i'; I) = \sum_{(\mathbf{f},\mathbf{e})\in\mathbf{T},|\mathbf{f}|=I} \sum_{j,\phi',h',\phi} \xi_j(i', \phi', \varepsilon', i, \phi, \varepsilon = 1) \tag{5.33}$$

where $|\mathbf{f}|$ is the number of words in $\mathbf{f}$. The M-step is shown in (5.34):

$$p_a(i|i'; I) = \frac{c(i, i'; I)}{\sum_{i''} c(i'', i; I)} \tag{5.34}$$

The E-step for the target phrase length model is shown in (5.35):

$$c(\phi'; f) = \sum_{(\mathbf{f},\mathbf{e})\in\mathbf{T}} \sum_{i,j,\phi,f_i=f} \gamma_j(i, \phi, \varepsilon = 1)\delta(\phi, \phi') \tag{5.35}$$

where $\delta(\phi, \phi')$ is a Kronecker function with value 1 if $\phi' = \phi$ and 0 otherwise. The M-step is shown in (5.36):

$$p_n(\phi'; f) = \frac{c(\phi'; f)}{\sum_{\phi} c(\phi; f)} \tag{5.36}$$

Finally the E-step for the syntactic coherence model proceeds as in (5.37):

$$c(r'; f, \phi') = \sum_{(\mathbf{f},\mathbf{e})\in\mathbf{T}} \sum_{i,j,\phi,f_i=f} \gamma_j(i, \phi, \varepsilon = 1)\delta(\phi, \phi')\delta(\varphi_j(e, \phi), r') \tag{5.37}$$

where $\varphi_j(e, \phi)$ is the syntactic dependency label between $e_{j-\phi+1}$ and e_j. The M-step performs normalisation, as shown in (5.38):

$$p_r(r'; f, \phi') = \frac{c(r'; f, \phi')}{\sum_r c(r; f, \phi')} \tag{5.38}$$

5.2.5 Perplexity

Perplexity is a measurement in information theory which can be used to measure how well the model fits the training data. This measure is a useful indicator of the behaviour of the EM algorithm which aims to find a model that best fits the data. Given a probability distribution P, it is defined as 2 raised to the power of distribution entropy $H(P)$ as in (5.39):

$$PP = 2^{H(p)} = 2^{-\sum_x P(x)\log_2 P(x)} \tag{5.39}$$

However, the true probability distribution P is often unknown and consequently (5.39) cannot be directly computed. Instead, we use another distribution p (normally a model of P) based on a set of training examples which were drawn from P. We can evaluate p by asking how well it can predict a held-out test sample $z_1, z_2, ..., z_M$ which is also drawn from P. The perplexity of model p is defined as in (5.40):

$$PP = 2^{-\sum_{i=1}^{M}\frac{1}{M}\log_2 p(z_i)} \tag{5.40}$$

The exponent in (5.40) can be considered as the cross entropy between the empirical distribution of the test sample $\tilde{p}(z)$ (i.e. $\tilde{p}(z) = \frac{m}{M}$ if z occurred m times out of the test sample size M) and distribution p as shown in (5.41):

$$H(\tilde{p}, p) = -\sum_z \tilde{p}(z)\log_2 p(z) \tag{5.41}$$

In word alignment models, the held-out sample may or may not be used. If no held-out data is used, the perplexity is directly computed on the training data $\mathbf{T}$ as in (5.42):

$$PP = 2^{-\sum_{(\mathbf{f},\mathbf{e})\in\mathbf{T}}\frac{1}{|\mathbf{T}|}\log_2 p(\mathbf{e}|\mathbf{f})} \tag{5.42}$$

where $|\mathbf{T}|$ denotes the number of sentence pairs in the training data $\mathbf{T}$.

5.3 Tuning Generative Word Alignment Models

Tuning generative word alignment models is rarely conducted in PB-SMT. This is mainly because generative word alignment models are generally more mathematically grounded, normally with or without very few free parameters,[6] which is generally considered as a major advantage of generative models over association-based models. Generative word alignment models are grounded in statistical estimation theory and the parameters of the models are adjusted such that the likelihood of the models on the training data are maximised (Och and Ney, 2003). However, the alignment models with maximum likelihood on the training data do not necessarily imply the highest intrinsic word alignment quality nor high performance of SMT systems (Ganchev et al., 2008). Therefore, designing generative models with a few tunable parameters is essential in order to guide the word alignment models towards our final goal.

The HMM word-to-phrase alignment model detailed in Deng and Byrne (2008) already has a parameter η in the phrase count distribution as described in Section 5.1. In Section 5.2.2, we introduced the constant ζ as an adjustable variable in the syntactically constrained HMM word-to-phrase alignment model. In the next chapter we will show with experiments how fine-tuning of this variable can influence the performance of the word alignment model.

5.4 Related Work

Incorporating syntax into generative word alignment models has drawn plenty of attention in recent years. Wang and Zhou (2004) investigated the benefit of monolingual parsing for alignment. They learned a generalised word association measure (crosslingual word similarities) based on monolingual dependency structures and improved alignment performance over IBM Model 2 and certain association-based models.

Toutanova et al. (2002) and Deng and Gao (2007) introduced some extensions to the original HMM models to better handle the irregularities in word alignment

[6] For example, in HMM word alignment models, p_0 as the probability of transition into the empty word NULL normally requires some empirical tuning or estimation on gold-standard word alignment in order to achieve better performance.

process using POS tags; Toutanova et al. (2002) also proposed the addition of the "staying" probability to approximately model the "fertility" phenomena. Lopez and Resnik (2005) proposed a syntax-rich transition model using syntactic dependencies to replace the simple transition model in HMM in a resource-scarce scenario, and DeNero and Klein (2007) constrained the HMM training with target language constituent structure in the scenario of translation rule extraction.

Brunning et al. (2009) introduced a context-dependent model for HMM word alignment. The syntactic information used in this approach was the POS tags of a fixed window of words.

5.5 Summary

In this chapter, we proposed a model that constrains the HMM word-to-phrase alignment model with syntactic dependency information. We first illustrated the difference between HMM word-to-word and word-to-phrase alignment models and showed how syntactic dependencies can be used to constrain the HMM word-to-phrase alignment model. This is followed by a detailed description of the component variables and parameter distributions in the model. The EM algorithm for unsupervised parameter estimation is also described. We also introduced the concept of perplexity as an indicator of how well the model fits the data and the possibility of tuning our syntactically constrained model according to different objective functions.

In the next chapter, we will present the various experiments we conducted using this model and detail the advantages and disadvantages of using this model in a PB-SMT system.

Chapter 6

Experiments on Syntactically Constrained HMM Word-to-Phrase Alignment

In this chapter, we present the experimental results using our syntactically constrained HMM word-to-phrase alignment models and conduct a comparison of HMM word-to-word, standard HMM word-to-phrase and variants of our syntactically constrained HMM word-to-phrase alignment models. We show the advantages and disadvantages of using syntactic dependencies to constrain an HMM word-to-phrase model and some good improvements over other state-of-the-art models using a properly tuned syntactically constrained model.[1]

6.1 Experimental Setup

Most experiments were conducted on both IWSLT and GALE data sets, i.e. Data Set 1, 2, 3, 5 and 6 as described in Section 2.5.2 (Page 36). The baseline system for word alignment and PB-SMT also follows Section 2.5. As a difference from Chapter 3 and 4, we conducted extensive evaluations on the word alignment including the

[1] Part of the work in this chapter was carried out while the author visited Cambridge University Engineering Department.

intrinsic micro-evaluation based on different alignment types.

We first report the word alignment results obtained using two variants of syntactically constrained word alignment models described in Chapter 5, i.e. SSH1 and SSH2 and the tunable parameter ζ, which indicates how likely it is that the first and final words in a target phrase do not have any syntactic dependencies, is set to 0.05. ζ is initially set to such a small value to encourage the first and final words in a target phrase to have syntactic dependencies. Further experiments conducted include English dependency clustering which aims to test the impact of the number of English dependency labels on word alignment quality, testing the effect of parsing quality on word alignment quality, the influence of the number of EM training iterations, and the effectiveness of fine-tuning the model parameter ζ. Finally, we also briefly present the results of a manual evaluation.

6.2 Experimental Results

In this section, we present the experimental results using our syntactically constrained HMM word-to-phrase alignment models. Both intrinsic and extrinsic evaluations are conducted, and an extensive analysis is provided.

6.2.1 Intrinsic Evaluation

The intrinsic quality is measured against the manually annotated IWSLT and GALE gold-standard corpus, i.e. Data Set 1 and 2. We report both the macro-evaluation and micro-evaluation results.

Macro-Evaluation

Table 6.1 shows the alignment quality in terms of precision, recall and F-score for different alignment models. For the GALE data, the results are mixed. The baseline HMM word-to-phrase alignment model (SH) achieved the highest F-score for Chinese–English alignment. For English–Chinese, the syntactic HMM word-to-

		Chinese-to-English			English-to-Chinese		
		Precision	Recall	F-score	Precision	Recall	F-score
GALE	H	53.06	37.52	43.96	52.82	30.14	38.38
	SH	53.78	38.02	**44.55**	55.23	31.51	40.13
	SSH1	52.04	36.80	43.11	56.38	32.17	40.96
	SSH2	53.72	37.99	44.51	56.61	32.30	**41.13**
IWSLT	H	67.14	54.93	**60.42**	74.26	59.62	**66.14**
	SH	66.11	54.08	59.49	73.74	58.88	65.32
	SSH1	65.56	53.64	59.00	67.28	54.02	59.92
	SSH2	63.65	52.07	57.28	69.52	55.82	61.92

Table 6.1: Macro-evaluation of various HMM word alignment models (%)

phrase model 2 (SSH2) is superior to the other models; compared to SH, there is a 1 point absolute improvement in F-score, as a result of the improvements in both precision (1.38 points) and recall (0.79 points). The HMM word-to-word alignment model (H) performs reasonably well for Chinese–English alignment, i.e. the best model (SH) has only 0.59 points improvement in F-score over H. For the English–Chinese direction, H is inferior to other sophisticated models, with a 2.75 points gap compared to the best model (SSH2).

Results on IWSLT data show that the HMM word-to-word alignment model is better than the other more complicated models in general, indicating that the simplicity of the model is preferred for the IWSLT data setting. We also observe that as the complexity of the model increases, i.e. from H to SH to SSH1 (or SSH2), the F-score decreases. One explanation for this is that the small amount of training data does not allow a precise estimation of such complicated word-to-phrase alignment models.

Micro-Evaluation

	GALE			IWSLT		
	Correct	Redundant	Incorrect	Correct	Redundant	Incorrect
H	54.89	10.58	34.53	56.07	12.46	31.47
SH	57.71	10.54	31.75	59.29	8.15	32.56
SSH1	59.08	10.61	**30.31**	**64.88**	6.49	**28.63**
SSH2	**60.01**	**9.21**	30.78	60.90	**6.07**	33.03

Table 6.2: Micro-evaluation of Chinese–English 1-to-1 alignments (%)

In order to have a better understanding of the nature of the derived alignments, a micro-evaluation that measures the quality of the different alignment types is conducted. To measure the quality of the 1-to-1 alignment, we classify the links into different quality levels including correct, redundant and incorrect links as described in Section 2.5.1. The evaluation results consist of the ratio of links of different quality levels. Table 6.2 shows the quality of the 1-to-1 alignment. On GALE data, we observed a consistent improvement in the quality of 1-to-1 alignment by using more sophisticated models such as SH, SSH1 and SSH2. In terms of correct links, the best model SSH2 achieved a 5.12 points absolute improvement over H and 2.30 points over SH. In terms of redundant links, the best model SSH2 achieved a 1.37 points reduction over H and a 1.33 points reduction over SH. SSH1 has a 4.22 points reduction in incorrect links over H and a 1.44 points reduction over SH.

On IWSLT data, despite the fact that the more sophisticated models including SH, SSH1 and SSH2 do not improve the overall performance as described in the macro-evaluation, the Chinese–English 1-to-1 alignments can be improved using these models. The improvement is particular salient for SSH1 with a 8.81 points improvement over H and 5.59 points over SH in terms of correct links. In terms of redundant links, the best model is SSH2, which achieved a 6.39 points reduction over H and a 2.08 points reduction over SH. Finally SSH1 also leads to a 2.84 points absolute reduction in incorrect links over H and 3.93 points over SH.

These nice improvements in 1-to-1 alignments demonstrate the strength of the more sophisticated models, in that they are more precise in capturing the irregularities in the structure of word alignment. This point will be demonstrated using the examples in the manual evaluation (cf. Section 6.7).

To measure the quality of the 1-to-2 alignments, we classify the links into correct, incomplete-missing, incomplete-redundant, redundant and incorrect links as described in Section 2.5.1. Depending on whether the two target words are consecutive or not, we have separate evaluation scores for these two cases. Table 6.3 shows the quality of the Chinese–English 1-to-2 alignments. Surprisingly, the word-to-phrase alignment models are not as good as the word-to-word models in predicting

		GALE			IWSLT		
		cons.	n.c.	total	cons.	n.c.	total
H	Correct	23.95	4.29	**18.75**	26.30	6.19	**21.24**
	Incomplete-missing	34.74	59.84	41.39	42.56	67.01	48.70
	Incomplete-redundant	9.85	10.64	10.06	11.07	14.43	11.92
	Redundant	9.05	26.18	7.34	9.69	1.03	7.51
	Incorrect	22.41	22.62	22.46	10.38	11.34	10.62
SH	Correct	21.10	1.43	15.89	20.07	7.22	16.84
	Incomplete-missing	39.41	62.00	45.40	52.25	67.01	55.96
	Incomplete-redundant	10.63	12.56	11.14	15.22	16.49	15.54
	Redundant	7.31	4.11	6.46	5.54	0	4.15
	Incorrect	21.55	19.90	21.11	6.92	9.28	7.51
SSH1	Correct	8.86	2.93	7.29	3.11	8.25	4.40
	Incomplete-missing	48.46	62.60	52.21	66.78	76.29	69.17
	Incomplete-redundant	18.08	13.68	16.92	22.15	8.25	18.65
	Redundant	3.33	1.30	2.79	0.35	0	0.26
	Incorrect	21.27	19.48	20.79	7.61	7.22	**7.51**
SSH2	Correct	16.90	2.07	12.97	7.61	7.22	7.51
	Incomplete-missing	44.73	63.83	49.79	67.13	72.16	68.39
	Incomplete-redundant	12.06	12.87	12.28	15.92	11.34	14.77
	Redundant	5.16	1.90	4.30	2.08	0	1.55
	Incorrect	21.14	19.31	**20.66**	7.27	9.28	7.77

Table 6.3: Micro-evaluation of Chinese–English 1-to-2 alignment (%)

completely correct links (corresponding to the group of correct links).[2] In total, there is a 2.86 points gap between SH and H, and a 5.80 points gap between SSH2 and H on GALE data. This gap is reflected in both consecutive (cons.) and non-consecutive English words (n.c.). Unsurprisingly, the syntactically constrained alignment model which favours 1-to-1 alignments (SSH1) dramatically underperforms that in favour of 1-to-n alignments (SSH2) with a 5.68 points gap in terms of correct links. A similar trend is observed in this regard on IWSLT data, with the gap between the syntactically constrained model (both SSH1 and SSH2) and H even larger in terms of correct links.

At the same time, we observe that the word-to-phrase models, i.e. SH, SSH1 and SSH2, are effective in reducing the number of *completely* incorrect alignments (corresponding to the group of incorrect links).[3] On GALE data, our best model SSH2 achieved a total of 1.8 points reduction of incorrect links compared to H

[2] By *completely*, we mean that both of the predicted links in a 1-to-2 alignment are correct and the three words (one source word and two target words) are exclusively involved in the two predicted links, which is a very harsh constraint.

[3] By *completely* incorrect, we mean none of the predicted links involving the three words (again, one source word and two target words) are correct.

and a 0.45 points reduction compared to SH. This reduction is achieved for both consecutive and non-consecutive English words. A similar trend is observed on IWSLT data. These nice improvements show that our model has the advantage of avoiding completely incorrect links.

This advantage is further demonstrated in the case of redundant links. On GALE data, SSH1 achieved a 4.55 points reduction in redundant links compared to H and 3.67 points compared to SH. Again this reduction is achieved on both consecutive and non-consecutive English words. This trend is also observed on IWSLT data. It is arguably true that these advantages can be overwhelmed by the large increase in both Incomplete-redundant and particularly Incomplete-missing links. We will discuss the overall number of links obtained using different models in Section 6.2.2; the syntactically constrained models tend to obtain a smaller number of links. One solution to overcome this limitation is to tune the number of links by adjusting the transition probability to the NULL state p_0.

	GALE			IWSLT		
	Correct	Redundant	Incorrect	Correct	Redundant	Incorrect
H	51.77	9.80	38.43	63.88	12.18	23.93
SH	58.74	9.77	31.55	63.84	10.05	26.11
SSH1	**58.98**	9.47	31.50	**64.36**	8.20	27.44
SSH2	58.60	9.12	32.28	61.80	8.34	29.86

Table 6.4: Micro-evaluation of English–Chinese 1-to-1 alignments (%)

Similar to Chinese–English alignment, for the English–Chinese 1-to-1 alignments shown in Table 6.4, both SSH1 and SSH2 models outperform H; The SSH1 model also outperforms SH in terms of correct links with a 0.24 points improvement on GALE data and 0.52 points on IWSLT data. For both data sets, the improvement over SH using syntactically constrained models (SSH1 and SSH2) is not as pronounced as for Chinese–English 1-to-1 alignments. However, we observed a large improvement in English–Chinese 1-to-2 alignments.

Table 6.5 shows the quality of English–Chinese 1-to-2 alignments. One of the differences from the Chinese–English direction that we observed was that the word-

		GALE			IWSLT		
		cons.	n.c.	total	cons.	n.c.	total
H	Correct	18.38	15.62	**18.00**	48.63	29.03	45.52
	Incomplete-missing	28.77	44.46	30.92	27.96	48.39	31.20
	Incomplete-redundant	8.44	7.81	8.36	5.18	4.84	5.12
	Redundant	11.15	3.32	10.08	9.12	9.68	9.21
	Incorrect	33.26	28.78	32.65	9.12	8.06	8.95
SH	Correct	9.63	20.54	11.12	48.02	41.94	**47.06**
	Incomplete-missing	44.08	43.42	43.99	32.52	40.32	33.76
	Incomplete-redundant	15.51	8.12	14.50	11.25	4.84	10.23
	Redundant	3.84	2.64	3.67	2.13	4.84	2.56
	Incorrect	26.95	28.28	26.72	6.08	8.06	**6.39**
SSH1	Correct	5.22	**20.91**	7.36	1.52	43.55	8.18
	Incomplete-missing	47.91	43.3	47.28	63.22	40.32	59.59
	Incomplete-redundant	17.71	8.06	16.39	13.98	9.68	13.30
	Redundant	2.35	2.52	2.37	0.91	0	0.77
	Incorrect	26.81	25.22	**26.59**	20.36	6.45	18.16
SSH2	Correct	10.76	19.99	12.02	7.29	**48.39**	13.81
	Incomplete-missing	44.00	44.53	44.07	60.48	37.10	56.78
	Incomplete-redundant	12.63	7.63	11.94	10.94	6.45	10.23
	Redundant	5.11	2.77	4.79	12.16	0	1.02
	Incorrect	27.50	25.09	27.17	20.06	8.06	18.16

Table 6.5: Micro-evaluation English–Chinese 1-to-2 alignment (%)

to-phrase alignment models outperform the word-to-word models in the case of non-consecutive Chinese words, with the best model SSH1 achieving a 5.29 points improvement over H and 0.37 points over SH. This reveals a general insight that different alignment models should be used for different alignment directions (source-to-target or target-to-source). Another phenomenon that reaffirms this insight is that on IWSLT data, the SH model performs the best for English–Chinese alignment, as opposed to the H model performing the best for Chinese–English alignment. On IWSLT data, we also observe that using the syntactically constrained models SSH1 and SSH2 not only leads to a reduction in correct links, but also an increase in the incorrect links. This further exhibits the negative effects of using complicated models on a small data set.

To summarise our findings in our micro-evaluation, we observed that using syntactically constrained models can improve the quality of both Chinese–English and English–Chinese 1-to-1 alignments on both large (GALE) and small (IWSLT) data sets. This improvement demonstrates one of the advantages of using syntactically constrained models, i.e. stronger modelling power in handling the radical structural differences between two languages and the irregularities in alignments. This point will be revisited in Section 6.7 (cf. Figures 6.16 and 6.17). We also observed some negative effects of deploying syntactically constrained models, very pronounced in the 1-to-2 alignments, and particularly on small training corpora like IWSLT where the data sparseness problem could be a major issue in parameter estimation of more sophisticated models.

Pairwise Alignment Agreement

	Chinese–English			English–Chinese		
	H	SH	SSH2	H	SH	SSH2
H	100.0	84.38	78.17	100.0	88.91	81.89
SH	84.38	100.0	88.33	88.91	100.0	90.04
SSH2	78.17	88.33	100.0	81.89	90.04	100.0

Table 6.6: Pairwise agreement between different alignment models on GALE gold-standard (%)

Table 6.6 shows the pairwise agreement between different alignment models on GALE data sets. From this Table, we observed a stronger agreement between H and SH (84.38% for Chinese–English and 88.91% for English–Chinese alignment), SH and SSH2 (88.33% for Chinese–English and 90.04% for English–Chinese alignment). Considerable discrepancies are observed between H and SSH2 with only 78.17% agreement for Chinese–English alignment and 81.89% agreement for English–Chinese alignment.

	Chinese–English			English–Chinese		
	H	SH	SSH2	H	SH	SSH2
H	100.0	86.33	79.13	100.0	91.17	83.13
SH	86.33	100.0	89.07	91.17	100.0	90.85
SSH2	79.13	89.07	100.0	83.13	90.85	100.0

Table 6.7: Pairwise agreement between different alignment models on IWSLT gold-standard (%)

Table 6.7 shows the pairwise agreement between different alignment models on IWSLT data. From this Table, similar trends to Table 6.6 can be seen. As a difference, various models achieved more agreement on IWSLT data than GALE data, which is not surprising because word alignment on IWSLT data set is a relatively simple task with much shorter sentences and limited vocabulary.

Perplexity

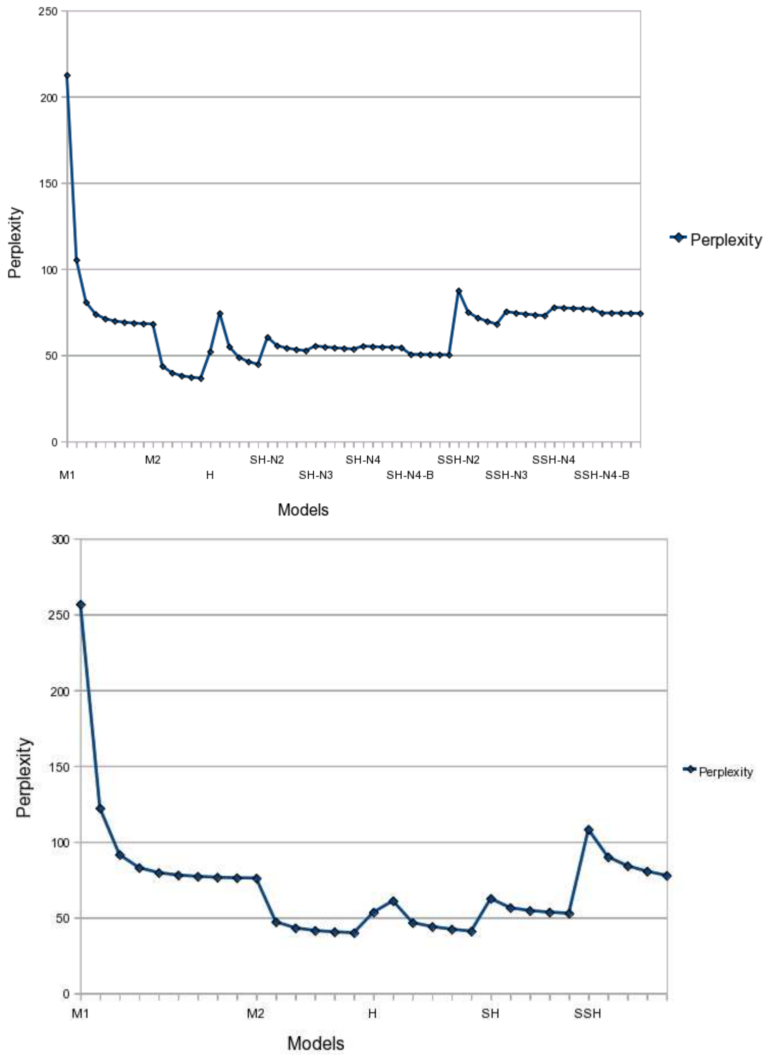


Figure 6.1: Perplexity for Chinese–English (top) and English–Chinese alignment (bottom)

Figure 6.1 shows the perplexity curves for Chinese–English and English–Chinese alignment. For both alignment directions, perplexity is considerably reduced during the 10 EM iterations of IBM model 1 (M1) training. The 5 iterations of model 2 (M2) training further reduces the model perplexity, and 5 iterations of HMM word-to-word model (H) maintains the low level of perplexity transferred from model 2. However, HMM word-to-phrase models (SH-N2, SH-N3, SH-N4 and SH-N4-B in Chinese–English and SH in English–Chinese alignment), particularly the syntactically constrained models (SSH-N2, SSH-N3, SSH-N4 and SSH-N4-B in Chinese–English and SSH in English–Chinese alignment), lead to a significant increase in model perplexity.

We also observed that the English dependency parser used a larger set of dependency labels, i.e. 20 labels observed in the training data compared to 12 labels for Chinese, which could be a source that increases the model perplexity. This phenomenon will be investigated in Section 6.3. In addition, the syntactically constrained HMM word-to-phrase model for English–Chinese word alignment does not seem to converge after 5 iterations; therefore, we conducted further experiments to see the effect of more iterations of training in Section 6.4.

6.2.2 Extrinsic Evaluation

To extrinsically evaluate the quality of our word alignment, we trained PB-SMT systems using the word alignments obtained from our syntactically constrained HMM word-to-phrase alignment models on both IWSLT and GALE training data sets, i.e. Data Set 3 and 5. The MTC data (Data Set 6), is used for development and testing purposes. The statistics of the data and the configuration of the baseline system are described in Section 2.5.2 (Page 36) and Section 2.5.3 (Page 38) respectively.

MT Results

	IWSLT06			IWSLT07		
	Bleu	Nist	Meteor	Bleu	Nist	Meteor
H	**22.01**	**6.0914**	**47.65**	**34.89**	**6.3403**	**55.09**
SH	21.09	5.7006	45.82	31.26	6.0271	52.13
SSH1	21.32	5.9564	46.53	31.92	6.1755	53.23
SSH2	20.98	5.7532	45.95	33.96	6.1723	53.16

Table 6.8: Performance of Phrase-Based SMT using syntactically constrained HMM word-to-phrase alignment ($\zeta = 0.05$) on IWSLT testsets

Table 6.8 shows the performance of the PB-SMT systems on IWSLT data when different word alignments are used. Note that the MT evaluation scores on IWSLT

2006 test set are lower than those on IWSLT 2007 test set because the translation of the IWSLT 2006 test set is more difficult due to the longer sentences (6066 versus 3166 running words for the 489 Chinese sentences) and larger vocabulary (1339 versus 862) according to Table 2.4 (Page 36). We can see the intrinsic quality indicated in Table 6.1 has been carried over to this extrinsic evaluation. The HMM word-to-word alignment model, which achieved the highest F-score, also leads to the best MT system according to all three MT evaluation metrics. It can also be seen that the SSH1 model is slightly better than SH model despite the fact that the F-score is actually lower according to Table 6.1.

	MTC2			MTC3			MTC4		
	Bleu	Nist	Meteor	Bleu	Nist	Meteor	Bleu	Nist	Meteor
H	**12.93**	**5.6027**	**41.94**	11.48	**5.3865**	**41.03**	12.44	**5.6936**	**42.39**
SH	12.72	5.4798	41.31	**11.61**	5.3132	40.61	**12.78**	5.6352	41.89
SSH1	12.57	5.5658	41.34	11.45	5.3689	40.25	12.21	5.6325	41.41
SSH2	12.70	5.5076	40.94	11.55	5.2804	39.93	12.40	5.5763	41.57

Table 6.9: Performance of Phrase-Based SMT using syntactically constrained HMM word-to-phrase alignment ($\zeta = 0.05$) on MTC testsets

The performance of the MT systems using different word alignment methods on GALE data is shown in Table 6.9. Note that the scores on MTC data sets are particularly low due to (i) the difficulty of translating news domain documents where the sentences tend to be longer and the vocabulary is larger; (ii) the relatively small number of references (4 for all the MTC test sets) used by the evaluation metrics (cf. Table 2.6 on Page 37). It can also be observed that there is inconsistency in rating different systems among the three evaluation metrics.[4] For example, on MTC3, SH achieved the best Bleu score; however, H received the best Nist and Meteor scores. The SH model which achieved the best F-score in Chinese–English alignment and SSH2 which received the best F-score in English–Chinese alignment are not necessarily the best models in achieving the highest Bleu score (e.g. on MTC2 the best model is H), which is the metric we directly optimise using MERT, not to mention other metrics we do not directly optimise. The lack of correlation between

[4] Despite the absolute score differences between different systems are quite small, this difference is still noticeable taking into account the fact that all the scores are at a quite low level.

intrinsic quality and extrinsic quality further reaffirms the necessity of optimising word alignment for MT purposes that we detailed in Chapters 3 and 4. This will be revisited in Section 6.6.

Phrase Table Analysis

To further investigate how the Chinese–English and English–Chinese word alignments are used in a PB-SMT system, we analysed the the number of links for each model after applying the symmetrisation heuristic "Grow-Diag-Final" (GDF). The resulting number of derived phrase pairs are exploited.

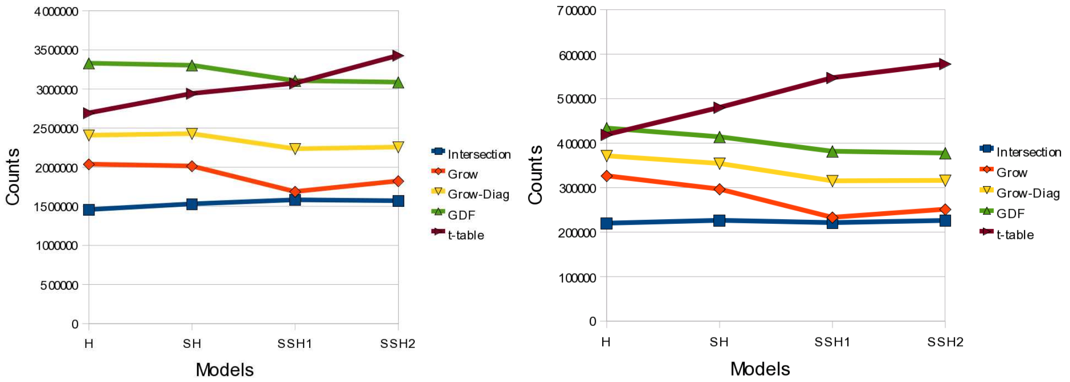


Figure 6.2: Number of links vs. number of phrase pairs on GALE (left) on IWSLT training data (right)

Figure 6.2 shows the number of derived links using "Intersection", "Grow", "Grow-Diag" and GDF heuristics for symmetrisation from different word alignment models and the number of phrases extracted from the GDF word alignment. For both GALE and IWSLT data, we obtained fewer links using our SSH2 model and consequently many more phrases are extracted. On GALE data, using the SSH2 model leads to a 7.30% decrease in the number of links compared to the H model, which resulted in a massive 27.34% increase in the number of phrase pairs. On IWSLT data, this effect is more pronounced; the SSH2 model has 12.96% fewer links than H model, leading to a 38.01% increase in the number of phrase pairs. There can be two possible reasons accounting for the smaller number of links using GDF heuristics; either (i) many links obtained using SSH1 and SSH2 model are discarded by the GDF procedure, or (ii) both Chinese–English and English–Chinese word alignment using the SSH2 model lead to fewer links (more NULL links).

A first investigation into this phenomenon is conducted by a detailed examination of the GDF procedure. Given the fact that GDF heuristics start with intersection of bidirectional alignments and expand the links within the intersection with neighbourhood links, there are two possible reasons for the smaller number of links, either (i) SSH1 and SSH2 models end up with fewer intersected links (implying that the results from the Chinese–English alignment and the English–Chinese alignment are quite divergent), or (ii) fewer neighbourhood links can be identified on top of the intersected links for SSH1 and SSH2. From Figure 6.2 which also shows the counts of intersection links, we can see an increase in intersection links using SSH1 and SSH2 compared to H or SH on GALE data, with the SSH1 model leading to a 8.60% increase in intersected links over H and 3.41% increase over SH, SSH2 model leading to a 7.78% increase over H and 2.63% over SH. On IWSLT data, SSH1 achieved a 0.59% increase over H and a 2.77% increase over H for SSH2; no increase over the SH model using SSH1 or SSH2 can be seen. However, in the process of horizontal neighbour expansion ("Grow" step), SSH1 and SSH2 alignments fail to provide more links and result in fewer "Grow-Diag" and GDF links. This also confirms the characteristics of the alignment derived from SSH1 and SSH2 models we explained in the micro-evaluation in Section 6.2.1, namely that 1-to-1 alignments are improved and consecutive 1-to-2 alignments worsened. However, the GDF neighbourhood link expansion procedure requires consecutive 1-to-n alignments; therefore, fewer links can be derived from models like SSH1 or SSH2.

As a further investigation into the reasons for the smaller number of links in our syntactically constrained models, we counted the number of NULL alignments for different alignment models as shown in Figure 6.3, where we can see a dramatic increase in NULL alignments for the SSH1 and SSH2 models for both Chinese–English (ZH–EN) and English–Chinese (EN–ZH) alignment directions. This might also explain why fewer links are derived in our SSH1 and SSH2 models.

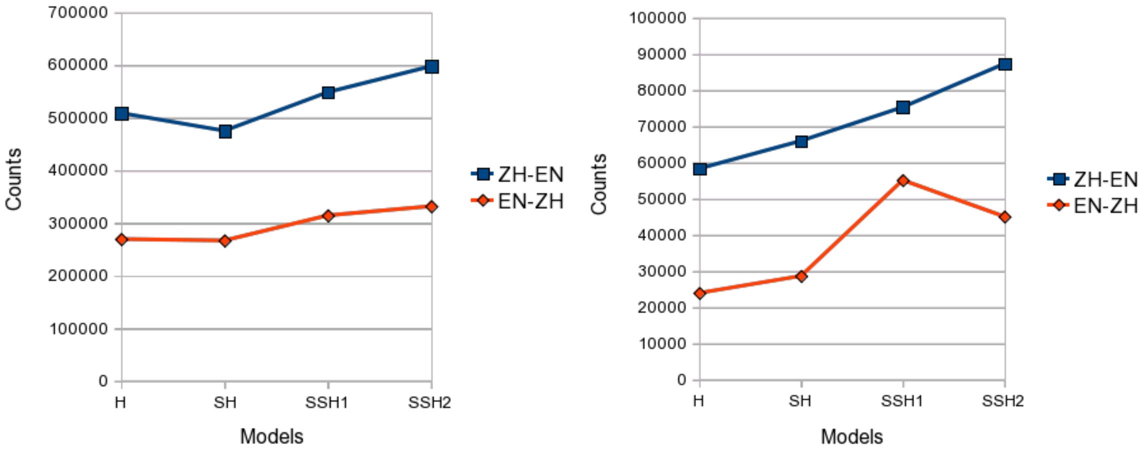


Figure 6.3: NULL links on GALE (left) and IWSLT training data (right)

To summarise, the smaller number of links from our syntactically constrained models can be attributed to (i) as a whole, fewer links (more NULL links) have been identified in Chinese–English and English–Chinese alignments, despite the fact that the alignments from the two directions reached more agreement (more intersection links); (ii) GDF heuristics fail to offer a proper expansion over the intersection based on links provided by SSH1 and SSH2 models.

There are many avenues for refining the syntactically constrained models. Instead of using GDF, we can design symmetrisation heuristics that can better fit models like SSH1 and SSH2. We can also test the word alignment results on syntax-based MT systems (e.g. SAMT (Zollmann and Venugopal, 2006)) where the word alignments are used to derive translation rules rather than phrase extraction, or using different word alignments obtained from different alignment models to construct multiple MT systems and taking the advantage of system combination (e.g. ROVER (Fiscus, 1997)). In Section 6.6, we tune the SSH1 and SSH2 models to produce more consecutive 1-to-n links such that the alignment quality can be boosted through the GDF heuristics.

Label	Meaning	Cluster
PRD	Predicative complement	**PRD**
AMOD	Modifier of adjective or adverb	**AMOD**
NMOD	Modifier of nominal	**NMOD**
P	Punctuation	**P**
PMOD	Between preposition and its child in a PP	**PMOD**
PRT	Particle	PRT
VC	Verb chain	**VC**
ROOT	Root	**ROOT**
ADV	Unclassified adverbial	**VMOD**
VMOD	General adverbial	
OBJ	Direct object or clause complement	**OBJ**
IOBJ	Indirect object	
SBJ	Subject	**SBJ**
LGS	Logical subject	
DEP	Unclassified relation	**DEP**
CC	Between conjunction and second conjunct in a coordination	
CLF	Cleft sentence	
COORD	Coordination	
EXP	Extraposed element in expletive construction	
PRN	Parenthetical	

Table 6.10: English dependency label clustering

6.3 English Dependency Label Clustering

As mentioned at the end of Section 6.2.1, there are 20 English dependency labels observed in the training data compared to 12 labels for Chinese, which could be a source that increases the model perplexity. We thereafter investigate the effect of the number of dependency labels on word alignment quality. With the Maltparser (Nivre et al., 2007), the dependency labels which occurred in our English corpus are shown in the leftmost column in Table 6.10. We cluster the 20 labels into 12 labels as in Nivre et al. (2007) which are shown in the rightmost column. The 11 labels marked in bold also occurred in Chinese texts.[5] After English dependency label grouping, we conducted Chinese–English word alignment with the same configuration as that without label grouping.

	Precision	Recall	F-score
SSH2	53.72	37.99	44.51
+Clustering	53.79	38.04	44.56

Table 6.11: Macro-valuation of Chinese–English HMM word-to-phrase alignment using English dependency label clustering (%)

6.3.1 Intrinsic Evaluation

We conducted a macro-evaluation on Chinese–English word alignment with English dependency label clustering as shown in Table 6.11. Only a modest improvement is observed implying that the number of dependency labels does not play a crucial role in determing system performance.

	GALE		
	Correct	Redundant	Incorrect
SSH2	60.01	9.21	**30.78**
+Clustering	**60.03**	**9.16**	30.81

Table 6.12: Micro-evaluation of the effect of dependency label clustering on Chinese–English 1-to-1 alignment (%)

A further micro-evaluation was also carried out. We observed very slight gains in 1-to-1 alignment, i.e. 0.02 points improvement in terms of correct links as shown in Table 6.12. Correct links in 1-to-2 alignments are also modestly improved with 0.13 points as shown in Table 6.13.

		GALE		
		cons.	n.c.	total
SSH2	Correct	16.90	2.07	12.97
	Incomplete-missing	44.73	63.83	49.79
	Incomplete-redundant	12.06	12.87	12.28
	Redundant	5.16	1.90	4.30
	Incorrect	21.14	19.31	**20.66**
SSH2 +Clustering	Correct	17.04	2.04	**13.10**
	Incomplete-missing	44.75	63.72	49.76
	Incomplete-redundant	11.87	12.73	12.09
	Redundant	5.26	2.02	4.40
	Incorrect	21.08	19.48	20.65

Table 6.13: Micro-evaluation of the effect of dependency label clustering on Chinese–English 1-to-2 alignment (%)

[5] While dependency label "PRT" is unique for our English texts, the label "SBAR" representing complementiser dependents only occurred in our Chinese texts.

Given the limited improvement in the intrinsic alignment quality, no additional MT experiments were conducted specifically for dependency label clustering. However, we conduct MT experiments in Section 6.4, where we combine the Chinese–English word alignment using English dependency label clustering with English–Chinese word alignment using more alignment iterations.

6.3.2 Perplexity Using Label Clustering

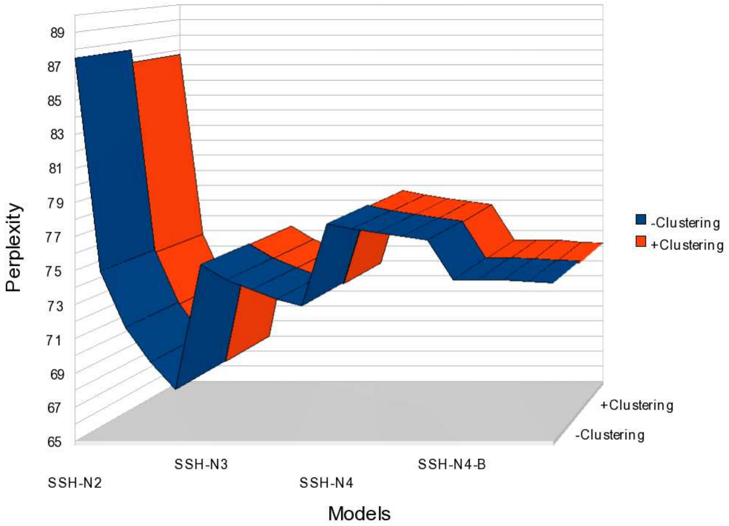


Figure 6.4: Perplexity for Chinese–English alignment using label clustering

As a confirmation of the results reported in Table 6.12 and 6.13, Figure 6.4 shows the perplexity curves during training with and without English dependency label clustering. The curves basically mirrored each other, implying that the model is not really simplified by this simple clustering strategy.

6.4 Effect of Iterations

As mentioned at the end of Section 6.2.1 where the perplexity of each EM training iteration is discussed, the HMM word-to-phrase alignment model does not converge for English–Chinese word alignment after 5 iterations of EM training of SSH2 so that an additional 5 iterations were performed. The 5 red squares in Figure 6.5 represent

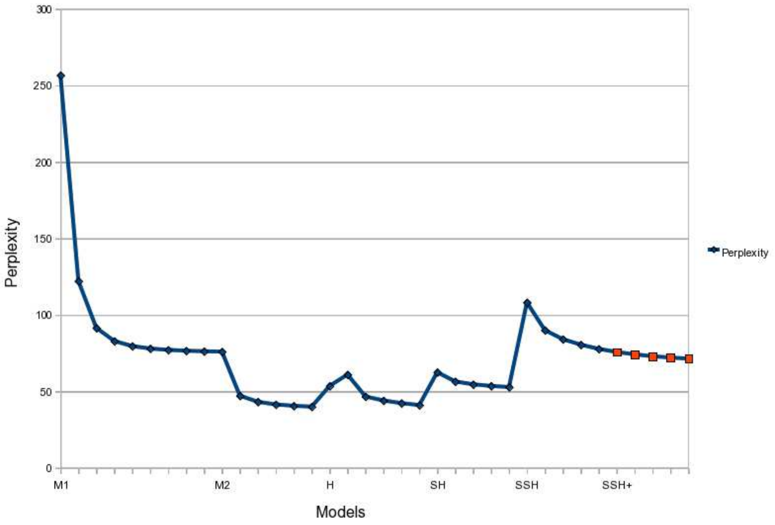


Figure 6.5: Perplexity for English–Chinese alignment with more iterations

the additional 5 iterations, after which the curve becomes flat. We conducted both intrinsic and extrinsic evaluations on the effect of additional iterations.

6.4.1 Intrinsic Evaluation

	Precision	Recall	F-score
SSH2	56.61	32.30	41.13
SSH2+	56.81	32.41	41.28

Table 6.14: Macro-evaluation of English–Chinese alignment with additional iterations (%)

Table 6.14 shows the results of the macro-evaluation on the effect of the number of iterations. With 10 iterations of EM training of SSH2 (SSH2+), we observed a modest improvement over 5 iterations (SSH2); a 0.2 points improvement in precision and 0.11 points improvement in Recall, which jointly lead to a 0.15 points improvement in F-score.

According to the micro-evaluation of the 1-to-1 alignments as shown in Table 6.15, we gain 0.41 points in the correct links and reduce the incorrect links by 0.51 points by performing more iterations. This further demonstrates the strength of our SSH2 model in the prediction of 1-to-1 alignments.

For 1-to-2 alignments as shown in Table 6.16, with more iterations, there is a

		GALE
SSH2	Correct	58.60
	Redundant	9.12
	Incorrect	32.28
SSH2+	Correct	**59.01**
	Redundant	9.21
	Incorrect	31.77

Table 6.15: Micro-evaluation of the effect of additional iterations on English–Chinese 1-to-1 alignment (%)

		GALE		
		cons.	n.c.	total
SSH2	Correct	10.76	19.99	12.02
	Incomplete-missing	44.00	44.53	44.07
	Incomplete-redundant	12.63	7.63	11.94
	Redundant	5.11	2.77	4.79
	Incorrect	27.50	25.09	27.17
SSH2+	Correct	9.56	18.14	**10.13**
	Incomplete-missing	45.66	44.65	45.52
	Incomplete-redundant	13.56	8.06	12.81
	Redundant	4.24	2.09	3.95
	Incorrect	26.97	27.06	26.98

Table 6.16: Micro-evaluation of the effect of additional iterations on English–Chinese 1-to-2 alignment (%)

1.81 points drop in correct links, particularly in the case of consecutive Chinese words; at the same time, there is a modest 0.19 points reduction in incorrect links. Overall, performing additional iterations harms 1-to-2 alignments. This also reveals the importance of performing a proper number of iterations regarding unsupervised generative word alignment models which maximise the likelihood of the training data in each iteration; related issues about IBM model 1 training are also discussed in Moore (2004), where the author commented that maximising the likelihood of the training data causes overfitting so that early stopping of IBM model 1 training is implicitly recognised by the community. In our models, given the complexity of our alignment models, an in-depth evaluation of iterations required for different data sizes is also necessary. However, this is beyond the scope of this thesis, and is left for future studies.

	MTC2			MTC3			MTC4		
	BLEU	NIST	METEOR	BLEU	NIST	METEOR	BLEU	NIST	METEOR
SSH2	12.70	5.5076	40.94	11.55	5.2804	39.93	12.40	5.5763	41.57
SSH2+ Clust.	12.32	5.4261	40.37	11.28	5.2482	39.90	12.18	5.5551	41.41

Table 6.17: Performance of Phrase-Based SMT on MTC testset using our syntactically constrained HMM word-to-phrase alignment with additional iterations and English dependency label clustering

6.4.2 Extrinsic Evaluation

We then fed both the Chinese–English word alignments derived with English dependency label clustering and English–Chinese word alignments derived with additional iterations into the PB-SMT system. This caused the MT performance to dip slightly as Table 6.17 shows. This can partly be attributed to the lower level of 1-to-2 alignments, particularly consecutive 1-to-2 alignments, in English–Chinese, as shown in Table 6.16. Again, this is closely related to the GDF heuristics which penalise the alignments with fewer consecutive alignments.

6.5 Parsing Quality

Given that our syntactically constrained HMM word-to-phrase alignment models require dependency parsing on both source and target languages, the effect of parsing quality on word alignment is worthy of investigation. We focus on English parsing quality in this section.

There are many ways to generate different quality levels of dependency parsers. One way to achieve this is through the control of the amount of data used to train the parser. However, in our case we used the pre-trained English parsing model which is regarded as a high-performance parser, we thereafter decided not to change the model itself. Another straightforward way to achieve this is to change the quality of the POS tagger. Again this can be achieved by using a varied amount of training data to train the POS tagger. Our method of obtaining a varied quality of POS tagging results is through changing the case of the words in the data. We trained the POS tagger using truecase English training data and tagged both the truecase and lowercase data used for word alignment. Here we assume the mismatch between training and testing data causes the tagged lowercase data to have a lower tagging quality compared to the truecase data and subsequently leads to a lower parsing quality.

	Precision	Recall	F-score
lowercase	53.72	37.99	44.51
truecase	53.69	37.96	44.48

Table 6.18: Macro-evaluation of syntactically constrained HMM word-to-phrase alignment with varied English dependency parsing quality (%)

Table 6.18 shows the Chinese–English word alignment results when different parses are used. This Table shows that under our experimental setting parsing quality does not have impact on alignment quality. The higher quality parsing results with truecase data as input actually leads to a minor 0.03% drop in F-score, indicating that the consistency of the parsing results is more important than the quality of labelled dependencies itself.[6]

	GALE		
	Correct	Redundant	Incorrect
lowercase	60.01	9.21	30.78
truecase	59.97	9.24	30.79

Table 6.19: Micro-evaluation of the impact of parsing quality on Chinese–English 1-to-1 alignment (%)

The micro-evaluation shown in Table 6.19 and 6.20 further demonstrates our findings above. There is a minor drop in the quality of both 1-to-1 and 1-to-2 alignments.

6.6 Fine-Tuning

As mentioned at the end of Section 6.2.2, fine-tuning of syntactically constrained models can potentially improve the alignment quality and/or the translation quality.

[6] It is also worth mentioning that the truecase data itself is noisy and lacks consistency. Furthermore, changing the case only has a limited impact on the parsing quality.

		GALE		
		cons.	n.c.	total
lowercase	Correct	16.90	2.07	12.97
	Incomplete-missing	44.73	63.83	49.79
	Incomplete-redundant	12.06	12.87	12.28
	Redundant	5.16	1.90	4.30
	Incorrect	21.14	19.31	20.66
truecase	Correct	16.87	2.02	12.85
	Incomplete-missing	44.70	63.78	49.92
	Incomplete-redundant	12.00	12.79	12.18
	Redundant	5.15	1.95	4.30
	Incorrect	21.28	19.46	20.76

Table 6.20: Micro-evaluation of the impact of parsing quality on Chinese–English 1-to-2 alignment (%)

Recall that in Section 5.2.2, we introduced a constant ζ ($0 \leq \zeta \leq 1$) to indicate how likely the first and final words in a target phrase do not have syntactic dependencies; we now show how fine-tuning of this variable can lead to different intrinsic and extrinsic word alignment quality.

6.6.1 Intrinsic Evaluation

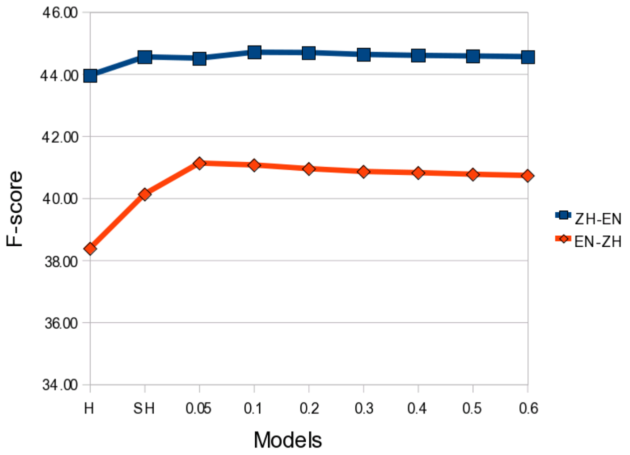


Figure 6.6: Macro-evaluation: F-score curves of different alignment models on GALE gold-standard (%)

Figure 6.6 shows the F-score curves when different models are used. The models put in comparison include the H, SH and SSH2 with ζ set to different values such as 0.05, 0.1, 0.2, 0.3, 0.4, 0.5 and 0.6. We set $\zeta = 0.05$ in the SSH2 model results reported above. For Chinese–English word alignment, we observed that the highest F-score (44.70%) is achieved when $\zeta = 0.1$; for English–Chinese, the best F-score can be obtained by setting $\zeta = 0.05$. Comparing the two curves, it can be seen that the curve for Chinese–English word alignment is more flat, implying that the use of different models does not lead to major differences in F-score. For the English–Chinese word alignment, the syntactically constrained models outperform the H and SH models, where the best syntactically constrained model leads to an improvement of 2.75 points absolute F-score over H and 1 point absolute F-score over SH.

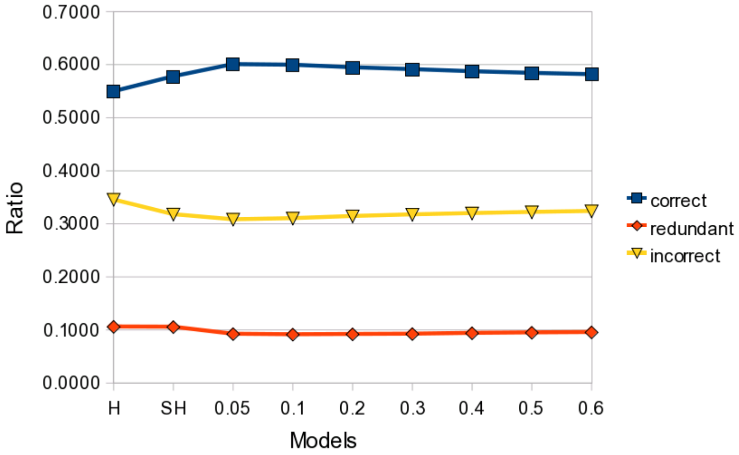


Figure 6.7: Micro-evaluation of Chinese–English 1-to-1 word alignment obtained using different alignment models

We conducted a micro-evaluation on the alignment results. Figure 6.7 shows the curves of correct, redundant and incorrect Chinese–English 1-to-1 alignment when different alignment models are used. We observed a 2.30% absolute increase in correct links, and 1.33% and 0.97% absolute reduction in redundant and incorrect links respectively using SSH2 ($\zeta = 0.05$) compared to SH. We can also see that both SH and various settings of SSH2 achieves more correct links and fewer redundant and incorrect links compared to H.

The curves of correct, redundant and incorrect English–Chinese 1-to-1 alignment are shown in Figure 6.8. All three curves turn flat after SH, indicating that no gains can be obtained using our SSH2 models. This does not conflict with our conclusions drawn above from Figure 6.6 that SSH2 models outperform H and SH in terms of F-score. We will show that there are substantial gains using SSH2 for 1-to-2 English–Chinese word alignments.

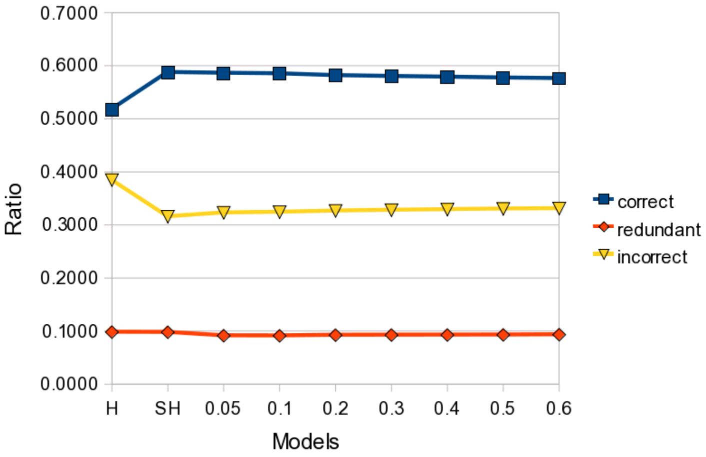


Figure 6.8: Micro-evaluation of English–Chinese 1-to-1 word alignment obtained using different alignment models

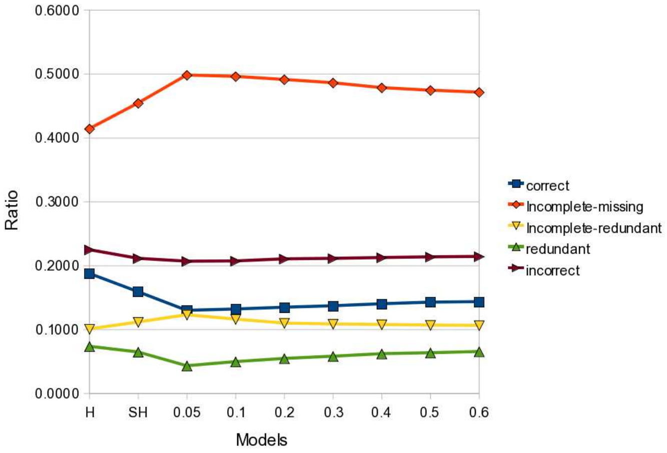


Figure 6.9: Micro-evaluation of Chinese–English 1-to-2 word alignment obtained using different alignment models

Figure 6.9 shows the curves of correct, incomplete-missing, incomplete-redundant, redundant and incorrect links for Chinese–English word alignment. We can see that using SSH2 with $\zeta = 0.05$ leads to a decrease in correct links and a sharp rise in incomplete-missing and incomplete-redundant links. Increasing the value of ζ can gradually increase the the number of correct links and reduce the number of incomplete-missing and incomplete-redundant links, demonstrating the significance of fine-tuning of variable ζ. Note also that from Figure 6.7 increasing the value of ζ can lead to a drop in correct links; therefore a careful balance between the quality of 1-to-1 and 1-to-2 alignments is crucial in order to obtain a generally high-quality word alignment. As seen from Figure 6.6, balancing these two for the highest F-score is achieved when ζ is set to 0.1.

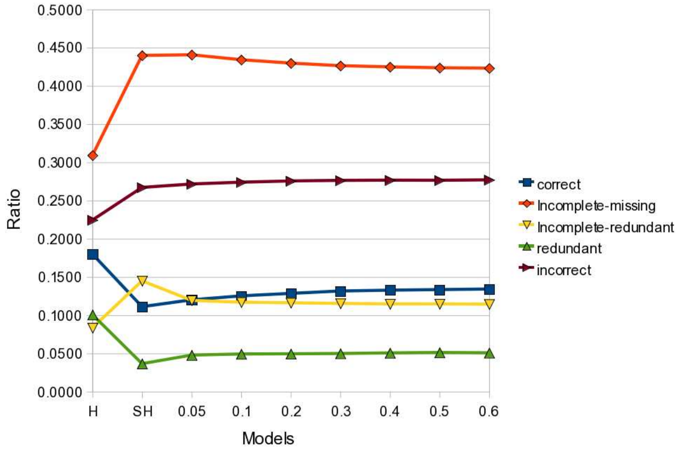


Figure 6.10: Micro-evaluation of English–Chinese 1-to-2 word alignment obtained using different alignment models

The curves of correct, incomplete-missing, incomplete-redundant, redundant and incorrect links for English–Chinese word alignment are shown in Figure 6.10. Using the SH model, we can see a sharp drop in correct links and a substantial rise in incomplete-missing, incomplete-redundant and incorrect links compared to H. Using our SSH2 model together with fine-tuning of ζ, we can see a gradual increase in correct links and a reduction in incomplete-missing and incomplete-redundant links.

6.6.2 Extrinsic Evaluation

The parameter ζ can be tuned according to MT performance. However, tuning ζ on our default development set (cf. Table 2.6) does not lead to significant differences in Bleu score. Therefore, we further make use of the three default test sets, i.e. MTC2, MTC3 and MTC4, for tuning purposes. We use the weights obtained on the default development set to translate the Chinese side of the MTC2, MTC3 and MTC4 data sets, and score the translations. The value of ζ is considered optimal if the corresponding system yields the highest combined score in translating MTC2, MTC3 and MTC4 data sets. Finally, the system is evaluated using the NIST 2006 and 2008 evaluation test sets.

PB-SMT Performance

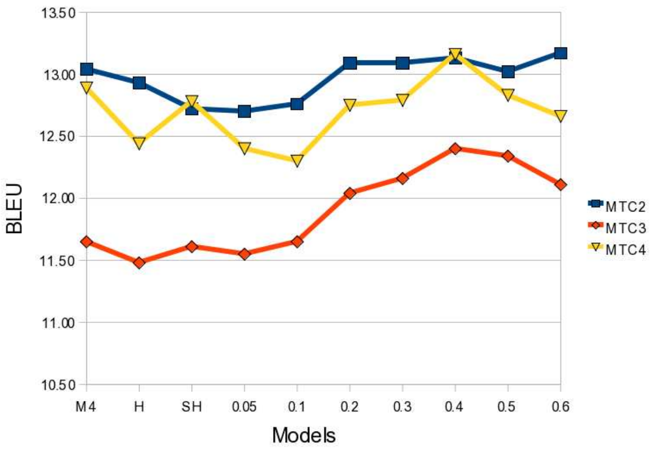


Figure 6.11: The performance of Phrase-Based SMT using different alignment models on three MTC data sets (Bleu)

Figure 6.11 shows the Bleu score curves on MTC2, MTC3 and MTC4 using different alignment models. As a contrast, we also include the results obtained using IBM model 4 alignment (M4). The curves show that tuning ζ can lead to dramatic differences in the final Bleu score. When ζ is set to 0.05, the SSH2 model results in an inferior Bleu score compared to SH on all three test sets. When we increase the value of ζ to 0.1, the resulting Bleu score starts to pick up on MTC5 and MTC3, and the Bleu score on MTC4 starts to pick up when ζ is increased to 0.2. The best Bleu score is achieved when ζ is set to 0.4 and the scores on MTC3 and MTC4 start to drop when the value of ζ further increases. This further demonstrates that the optimal ζ according to F-score ($\zeta = 0.1$ for Chinese–English and $\zeta = 0.05$ for English–Chinese alignment) does not necessarily imply that best Bleu score will be obtained. Note also that M4 achieves a similar Bleu score as the SH model on MTC3 and MTC4 and outperforms the SH model on MTC2.

Figure 6.11 shows the Nist score curves on MTC2, MTC3 and MTC4 using different alignment models. Again SSH2 model with $\zeta = 0.4$ achieved the highest Nist score. Differently from the Bleu curves, the H model is better than M4 and SH, of which SH is slightly better. Tuning ζ is also essential to achieve better performance, with a Nist score gap of 0.16, 0.22 and 0.18 on three test sets respectively between

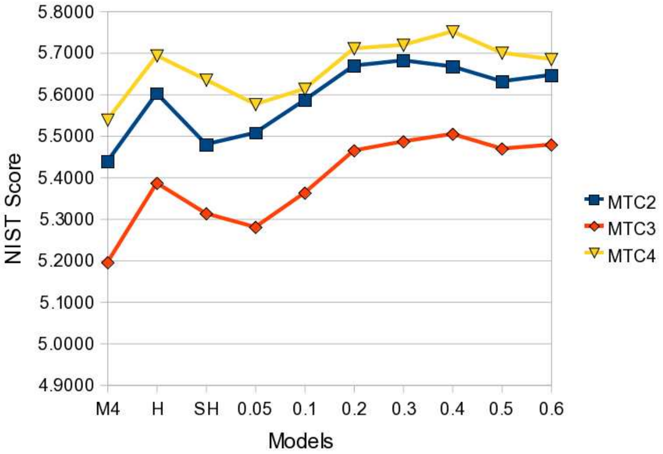


Figure 6.12: The performance of Phrase-Based SMT using different alignment models on three MTC data sets (NIST)

the optimal value ($\zeta = 0.4$) the most inferior value ($\zeta = 0.05$).

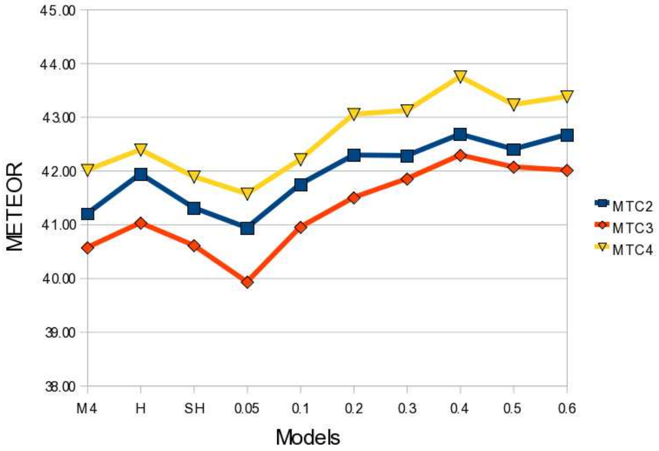


Figure 6.13: The performance of Phrase-Based SMT using different alignment models on three MTC data sets (METEOR)

The METEOR score curves are shown in Figure 6.13 which are very similar to the NIST score curves shown in Figure 6.12. Again, tuning ζ is essential to achieve better performance, with a difference of 1.74, 2.36 and 2.18 absolute METEOR points on three testsets respectively between the optimal value ($\zeta = 0.4$) the most inferior value ($\zeta = 0.05$).

Based on Figure 6.11, 6.12 and 6.13, we observe that 0.4 is the optimal value for ζ. Table 6.21 shows the translation results on NIST06 and NIST08 test sets when we seed the PB-SMT system with word alignments obtained with our fine-tuned SSH2

	NIST06			NIST08		
	Bleu	Nist	Meteor	Bleu	Nist	Meteor
SH	14.18	5.99	39.42	9.37	5.20	34.16
SSH2	14.64	6.20	40.81	10.07	5.40	35.38
(ζ=0.4)	($p < 0.07$)			($p < 0.01$)		

Table 6.21: Performance of Phrase-Based SMT using our fine-tuned syntactically constrained HMM word-to-phrase alignment model ($\zeta = 0.4$) on NIST06 and NIST08 testsets

model where ζ is set to 0.4. On two different testsets NIST06 and NIST08, the SSH2 model leads to a respective 3.24% and 7.47% relative improvement over SH in terms of Bleu score. The improvement on NIST08 test set is statistically significant using approximate randomisation (Noreen, 1989). Moreover, we can observe an improvement over SH according to all three MT evaluation measures including Nist and Meteor when SSH2 models are properly tuned.

Phrase Table Analysis

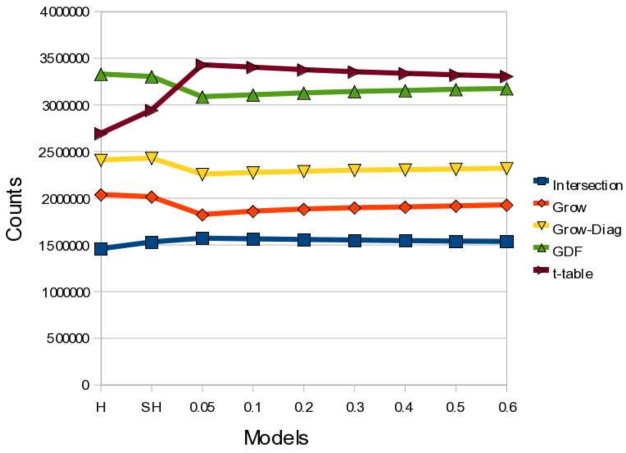


Figure 6.14: Number of links vs. number of phrase pairs

Figure 6.15 shows the number of links obtained from different alignment models using different symmetrisation heuristics and the resulting number of phrase pairs in the phrase table using GDF heuristics. With “Grow”, “Grow-Diag” and GDF heuristics, we can observe a drop in the number of links when the SSH2 model is used with $\zeta = 0.05$. Given the relation between the different heuristics as described in Section 2.4.1 (Page 24), i.e. “Grow” is performed on top of Intersection, “Grow-

Diag" performed on top of "Grow", and GDF on top of "Grow-Diag", the reason for the smaller number of links using GDF heuristics is the Grow step, where the symmetrisation algorithm tries to include the neighbouring links of the intersection links. Despite the number of links in the intersection is larger for SSH2 with $\zeta = 0.05$, the algorithm can not "grow" more links from the neighbourhood. This can also be explained by Figure 6.9, where it is shown that SSH2 models have fewer correct 1-to-2 links and far more incomplete-missing and incomplete-redundant links compared to H and SH.

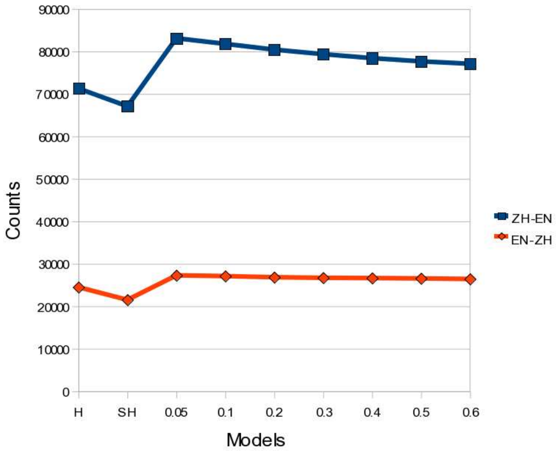


Figure 6.15: Number of NULL links

Fortunately, we can tune ζ by increasing its value so that more links can be included from the "Grow" step and subsequently more links can be identified in the GDF heuristic. The consequence of a small number of word alignment links is a larger number of phrase pairs as described in Section 2.4.1. From Figure 6.15, we can see a substantial increase in the number of phrase pairs in the t-table from SH to SSH2 ($\zeta = 0.05$). By increasing the value of ζ, more links can be identified and subsequently fewer phrase pairs are extracted. Here we are not claiming the fewer phrase pairs in the t-table, the better; the final PB-SMT performance requires a balance of the quality and coverage of the phrase table, which can be achieved by tuning the word alignment. In our case, the tunable variable is ζ in SSH2.

We also counted the number of NULL links in both Chinese–English and English–Chinese word alignment as shown in Figure 6.15. This Figure shows that setting ζ

to a small value (e.g. 0.05) results in a large number of NULL links. By increasing the value of ζ, we observed a gradual reduction in NULL links in both directions, which interestingly implies that increasing the value of ζ can indirectly encourage the model to produce fewer NULL links.

6.7 Manual Evaluation

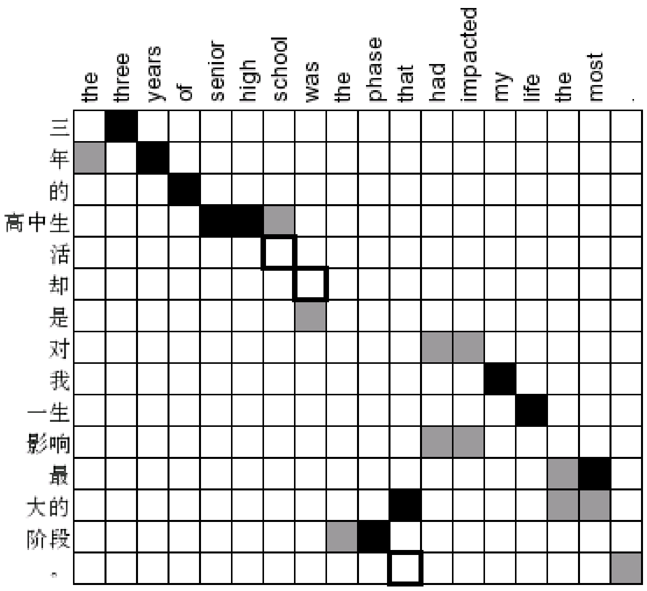


Figure 6.16: An example of Chinese–English word alignment using baseline HMM word-to-phrase alignment model

We performed a manual evaluation to compare the word alignment obtained using the baseline HMM word-to-phrase word alignment model and the syntactically constrained model ($\zeta = 0.05$) as shown in Figures 6.16 and 6.17.

For both Figures, the black squares represents the correct links identified by the aligner, the grey squares indicates the correct links not identified and the white square with black borders indicates the incorrect links proposed by the aligner. By examining the gold-standard alignment (both the grey and black squares), we can see that word alignment for this sentence pair is non-trivial. This is reflected in the long-distance reordering and the large number of 1-to-n and m-to-n alignments.

Comparing Figure 6.16 with 6.17, it can be seen that the syntactically con-

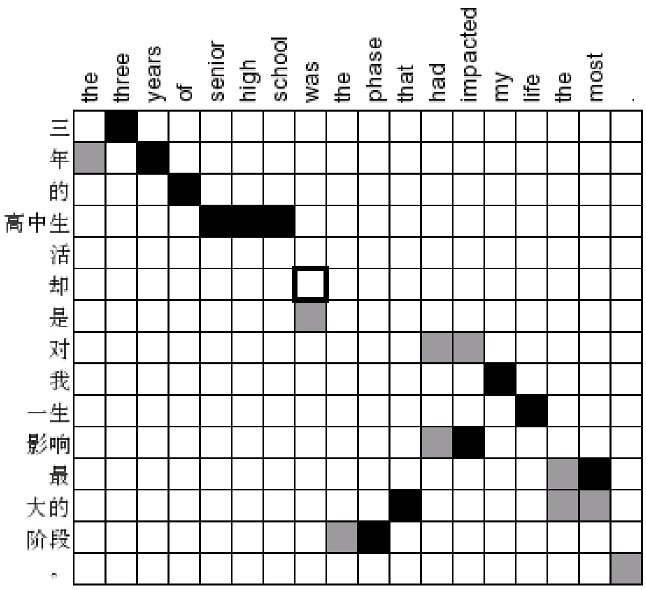


Figure 6.17: An example of Chinese–English word alignment using syntactically constrained HMM word-phrase alignment model

strained model does have the power to identify some difficult links, i.e. Figure 6.17 has more black squares. It can also be seen that the syntactically constrained models commit fewer link errors, i.e. Figure 6.17 has fewer white square with black borders. Besides the alignment, we also examined the translation outputs from the PB-SMT

(a) f: 印度 国防部 发言人 说 ： 「 即将 举行 的 演习 旨 在 进一步 提升
印度 与 美国 的 军事 合作 关系 。
reference: a spokesman from the ministry of national defense of india said : " the upcoming exercise aims at further enhancing the military cooperation between india and the united states . "
SH: " : defense , india , " said in his upcoming india and further enhance the us military partnership 。 . "
SSH: india 's department of defense : spokesperson said , " the upcoming exercise his in india and further enhance the us military cooperation relationship 。 . "

(b) f: 南非 太空 观光 客 结束 太空 之 旅 返 抵 地球
reference: south african space tourist back to earth after his space travel
SH: the south african space space trip to the visitors ' end backed earth .
SSH: the south african space tourism ' end space trip back to benefit the earth .

Figure 6.18: Translation examples using HMM word-to-phrase alignment model (fluency)

systems using different word alignment models. Figure 6.18 shows two translation examples using the SH and SSH word alignment models, where the corresponding translations of source segments are highlighted with the same colours. Examples (a) and (b) show that the system using SSH word alignment produces more fluent and adequate translations.

Figure 6.19 contains another two examples, which show that the system using SSH can produce more adequate output. For example (c), we see that the main verb of the sentence is missing using the SH system and present in the SSH system. In example (d), the information on “prevention and treatment” is included in the SSH system but not in the SH system. These examples demonstrate that the translation model obtained from SSH word alignment is of higher quality.

(c) f: 联合国 指出 非洲 南部 一千 万 人 濒临 饥饿 危机
reference: un declares 10 million in southern africa on brink of starvation
SH: the united nations that the south african 10 million people 濒临 hunger crisis .
SSH: the united nations refers to the southern africa 10 million people 濒临 hunger crisis .

(d) f: 世 卫 组织 ： 印度 中国 防治 肺结核 大有 进展
reference: who : great progress in tb prevention and treatment seen in india and china
SH: the world health organization great progress in china , india , ： for tb .
SSH: the world health organization , india , china ： great progress in the prevention and cure of tb .

Figure 6.19: Translation examples using HMM word-to-phrase alignment model (adequacy)

Note that the alignment examples in Figure 6.17 and SSH translation examples in Figure 6.18 and 6.19 are respectively obtained by setting ζ to 0.1 and 0.4. This is a further demonstration of the flexibility of our SSH model which can be tuned according to different objective functions and end tasks.

6.8 Summary

In this chapter, we reported the experimental results using the syntactically constrained HMM word-to-phrase alignment models. We showed that this model is effective in improving 1-to-1 alignments at a cost of quality reduction in 1-to-2 alignments with parameter $\zeta = 0.05$. With this setting, the resulted PB-SMT system does not outperform standard HMM word-to-phrase alignment model mainly due to the fact that the syntactically constrained models results in a smaller number of links when GDF heuristics is used. Tuning ζ can balance the quality of 1-to-1 and 1-to-2 alignments with $\zeta = 0.1$ leading to the best Chinese–English F-score, $\zeta = 0.05$ for the best English–Chinese F-score, and $\zeta = 0.4$ holding the highest Bleu score which is statistically better than the state-of-the-art word alignment models. This demonstrates the importance of flexibility of generative word alignment models. We also investigated the effect of dependency label number, parsing quality, number of EM iterations on the quality of word alignment and showed that these factors have only minor influences on word alignment, indicating the robustness of this model.

In the next chapter, we conclude this thesis and point out avenues for future research.

Chapter 7

Conclusions and Future Work

In this thesis, we covered a comprehensive set of research topics on the area of word alignment in the context of PB-SMT, and developed new algorithms and models that are demonstrated to be effective in improving word alignment quality. The research questions we tried to answer involve the following key elements:

- Segmentation and syntactic constraints for word alignment
- Word packing
- Discriminative word alignment models with syntactic constraints
- Generative word alignment models with syntactic constraints
- Tuning word alignment according to different objective functions

We started with a review of existing word alignment models and argued that existing approaches can be improved from two aspects. The first aspect is general alignment quality, and the other is the flexibility. We then introduced new algorithms and models to improve the quality of word alignment. The word packing algorithm described in Chapter 3 is a good demonstration of using segmentation constraints in word alignment. The segmentation acted as a hard constraint; however, through a few carefully controlled iterations, the negative effects of hard constraints can be partly eliminated. The new models presented in Chapter 4 and Chapter

5 improved the alignment using the syntactic dependencies as soft constraints and showed promising results in this direction. In developing our alignment algorithms and models, we also bore in mind that these alignment approaches should be as flexible as possible so that they can be optimised for the end PB-SMT tasks.

Now let us revisit the four research questions mentioned in Chapter 1.

> **(RQ1)** Can bilingually motivated word segmentation improve word alignment and PB-SMT?
>
> **(RQ2)** Can discriminative word alignment models be enhanced by syntactic dependencies?
>
> **(RQ3)** Can we extend existing generative word alignment models with syntactic constraints?
>
> **(RQ4)** Can we tune the word alignment in a bid to achieve the highest MT performance?

As an answer to (**RQ1**) and partly to (**RQ4**), Chapter 3 proposed a practical algorithm to perform bilingually motivated word segmentation optimised for PB-SMT systems. Our algorithm consists of using the output from an existing statistical word aligner (e.g. GIZA++) to obtain a set of candidate "words" to be packed. We evaluated the reliability of these candidates using simple metrics based on co-occurrence statistics. We then modified the segmentation of the respective sentences in the parallel corpus according to these candidate words; these modified sentences were then given back to the word aligner, which produces new alignments. We reported a 1.84 points absolute (5.44% relative) increase in BLEU score over a standard PB-SMT system on the IWSLT 2007 task. We also revealed that under certain conditions, monolingual segmentation can be replaced by our bilingually motivated segmentation approach. Regarding the feasibility of applying this approach on larger data, we suggested that an increase in the amount of training data is necessary if the vocabulary size increases, and the balance between the amount of training and vocabulary size is crucial to the performance of the system. An optimisation procedure that dynamically determines which words should be grouped into one word in each iteration were also proposed.

To address (**RQ2**) as well as (**RQ4**), Chapter 4 described a flexible discriminative word alignment model that can leverage syntactic dependencies for word alignment. We first obtain a set of anchor alignment using state-of-the-art alignment models. We then incorporate syntactic features induced by the anchor alignments into a discriminative word alignment model. The syntactic features we used are syntactic dependencies. This two-stage model is motivated by the fact that some words in a language require more information than others in order to be aligned. By identifying some anchor alignments in the first stage, some syntactic dependencies can also be introduced. This combined information can boost the alignment of the remaining words that are more difficult to align. Our experiments showed that using our word alignment in a PB-SMT system yields a 5.38% relative increase on IWSLT 2007 task in terms of Bleu score. In addition, the flexibility of our alignment model facilitates the tuning of the word alignment towards achieving higher MT performance. Therefore, we also introduced a simple optimisation method that can optimise the word alignment either according to the F-score of the alignment or the Bleu score when the word alignment is used in a PB-SMT system. Experimental results showed that optimising Bleu can achieve better performance; however, it is also prone to overfitting problems given that our optimisation algorithm is still somewhat primitive.

Finally, Chapter 5 and 6 aimed to tackle (**RQ3**) and (**RQ4**). In these Chapters, we proposed a mathematically grounded model that can use syntactic dependency information as soft constraints to guide the alignment. Specifically, we use a HMM word-to-phrase alignment model and insist that the first word and last word within a phrase should be dependent on each other. We added this constraint into the model in such a way that the efficient parameter estimation procedures are still preserved. Extensive experiments were conducted to evaluate the effect of the syntactic constraints and a manual evaluation was also carried out to investigate the characteristics of the resulting alignments. In general, we observed a substantial improvement in general alignment quality. We also showed that the quality of 1-to-1 and 1-to-2 alignments can be balanced via tuning the model parameter ζ. We also observed that tuning the model parameter can allow a proper control over the number of NULL alignments. Experiments on feeding this word alignment into PB-SMT systems showed a significant gain over the state-of-the-art, demonstrating the advantages of using syntactic constraints in generative word alignment models.

7.1 Contributions of This Thesis

In this thesis, we introduced three new methods for word alignment which outperform the state-of-the-art word alignment models. We measured the quality of word alignment in terms of both intrinsic and extrinsic evaluation metrics. The word packing algorithm is the first approach for "bilingually motivated word segmentation" and shows that using segmentation constraints in word alignment can improve the alignment quality. The syntactically enhanced discriminative word alignment models offer a new approach to incorporating syntactic dependencies into word alignment and demonstrate the effectiveness of syntax in the process of word alignment. These models also allows flexible fine-tuning according to different objective functions, e.g. F-score of the word alignment or the Bleu score of the resulted PB-SMT outputs. Our syntactically constrained HMM word-to-phrase alignment models improve the performance of the standard HMM word-to-phrase word alignment model in terms of both intrinsic and extrinsic measures. The tunability of these models is also novel among various generative word alignment models.

In summary, this thesis offers a series of new algorithms and models to improve the quality of word alignment and shows that properly constraining word alignment models using linguistically motivated insights benefits the word alignment. Another contribution of these models is their tunability so that these models can yield PB-SMT systems with good performance through a careful development process. This guides the word alignment research towards an application-oriented direction.

7.2 Future Work

Each of our proposed algorithms or models is faced with some limitations. In an era when large amounts of parallel data are widely available, scalability becomes an important measure to gauge the merits of a method. For the word packing algorithm, scalability is one of the problems we intend to address in the near future. A successful scaling up of our method has to meet two constraints: (i) the data has a relatively small vocabulary so that a high-quality bilingual dictionary can be obtained; and (ii) the small data set is representative enough of the larger data set. Given such limitations, in future work, we firstly intend to explore the correlation between vocabulary size and the amount of training data needed in order to achieve good results. By doing this, we can scale up this algorithm to handle larger data sets. We also plan to use more sophisticated association measures to estimate the reliability of the derived 1-to-n alignments.

The model described in Chapter 4 exploits syntactic dependencies as soft constraints in discriminative word alignment. There are several aspects we can improve over these models. First, the two sub-models in our approach are two separate processes performed in a pipeline. We plan to jointly optimise the two models in one go. Second, some of our experiments used complex IBM models, e.g. IBM Model 4, to obtain anchor alignment. We plan to boostrap the alignment using simple heuristics without relying on complex IBM models. Third, a comparison with other discriminative word alignment models is also necessary to justify the merits of our approach. Moreover, the optimisation algorithm used in tuning the word alignment model according to different objective functions is still somewhat primitive. This disadvantage is exhibited when we optimise the first-order syntactically enhanced model according to Bleu (cf. Section 4.7.2). In the future, we plan to refine the optimisation algorithm in order to produce more reliable optimised parameters. These parameters can hopefully be more useful when we scale up our experiments. Finally, we also plan to adapt our approach to larger data sets and more language pairs as long as some annotated data is available.

For the generative model described in Chapter 5 and 6, we observed a great deal of errors in POS tagging and consequently in dependency parsing. To adopt POS taggers and dependency parsers with varied quality levels is necessary in order to draw more grounded conclusions about the effect of parsing quality on word alignment. We also plan to induce dependencies from constituent structure trees using a set of head rules and compare the results with using dependency parsers (Magerman, 1995; Srivastava and Way, 2009). Moreover, the syntactic coherence model itself is very simple, in that it only covers the syntactic dependency between the first and last word in a phrase. Accordingly, we intend to extend this model to cover more sophisticated syntactic relations within the phrase. Furthermore, given that we can construct different MT systems using different word alignments, multiple system combination can be conducted to avail of the advantages of different systems. Again, we plan to test this approach with larger amounts data and other language pairs.

Bibliography

Alshawi, H., Douglas, S., and Bangalore, S. (2000). Learning dependency translation models as collections of finite-state head transducers. *Computational Linguistics*, 26(1):45–60.

Ayan, N. F., Dorr, B., and Habash, N. (2004). Multi-align: Combining linguistic and statistical techniques to improve alignments for adaptable MT. In *Proceedings of the 6th Conference of the Association for Machine Translation in the Americas (AMTA-2004)*, pages 17–26, Washington DC.

Ayan, N. F. and Dorr, B. J. (2006). Going beyond AER: An extensive analysis of word alignments and their impact on MT. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 9–16, Sydney, Australia.

Bai, M.-H., Chen, K.-J., and Chang, J. S. (2008). Improving word alignment by adjusting Chinese word segmentation. In *Proceedings of the Third International Joint Conference on Natural Language Processing*, pages 249–256, Hyderabad, India.

Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, MI.

Baum, L. E. (1972). An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. *Inequalities*, 3:1–8.

Birch, A., Callison-Burch, C., and Osborne, M. (2006). Constraining the Phrase-Based, joint probability statistical translation model. In *Proceedings of the 7th biennial conference of the Association for Machine Translation in the Americas*, pages 10–18, Cambridge, MA.

Blunsom, P. and Cohn, T. (2006). Discriminative word alignment with Conditional Random Fields. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 65–72, Sydney, Australia.

Brown, P. F., Cocke, J., Della-Pietra, S. A., Della-Pietra, V. J., Jelinek, F., Lafferty, J. D., Mercer, R. L., and Roossin, P. S. (1990). A statistical approach to Machine Translation. *Computational Linguistics*, 16(2):79–85.

Brown, P. F., Della-Pietra, S. A., Della-Pietra, V. J., and Mercer, R. L. (1993). The mathematics of Statistical Machine Translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Brunning, J., de Gispert, A., and Byrne, W. (2009). Context-dependent alignment models for Statistical Machine Translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 110–118, Boulder, CO.

Burges, C. J. C. (1998). A tutorial on Support Vector Machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167.

Carpuat, M. and Wu, D. (2007). Context-dependent phrasal translation lexicons for Statistical Machine Translation. In *Proceedings of Machine Translation Summit XI*, pages 73–80, Copenhagen, Denmark.

Chang, P.-C., Galley, M., and Manning, C. D. (2008). Optimizing Chinese word segmentation for Machine Translation performance. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 224–232, Columbus, OH.

Chen, B. and Federico, M. (2006). Improving Phrase-Based statistical translation through combination of word alignment. In *FinTAL - 5th International Conference on Natural Language Processing*, pages 356–367, Turku, Finland.

Cherry, C. and Lin, D. (2003). A probability model to improve word alignment. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 88–95, Sapporo, Japan.

Cherry, C. and Lin, D. (2006a). A comparison of syntactically motivated word alignment spaces. In *Proceedings of the 11th Conference of the European Chapter of the ACL (EACL 2006)*, pages 145–152, Trento, Italy.

Cherry, C. and Lin, D. (2006b). Soft syntactic constraints for word alignment through discriminative training. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 105–112, Sydney, Australia.

Chiang, D. (2005). A hierarchical Phrase-Based model for Statistical Machine Translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 263–270, Ann Arbor, MI.

Collins, M., Koehn, P., and Kucerova, I. (2005). Clause restructuring for Statistical Machine Translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 531–540, Ann Arbor, MI.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38.

DeNero, J. and Klein, D. (2007). Tailoring word alignments to syntactic Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 17–24, Prague, Czech Republic.

Deng, Y. and Byrne, W. (2005). HMM word and phrase alignment for Statistical Machine Translation. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 169–176, Vancouver, BC, Canada.

Deng, Y. and Byrne, W. (2006). MTTK: An alignment toolkit for Statistical Machine Translation. In *Proceedings of the Human Language Technology Conference of the NAACL*, pages 265–268, New York City, NY.

Deng, Y. and Byrne, W. (2008). HMM word and phrase alignment for Statistical Machine Translation. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(3):494–507.

Deng, Y. and Gao, Y. (2007). Guiding statistical word alignment models with prior knowledge. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 1–8, Prague, Czech Republic.

Ding, Y., Gildea, D., and Palmer, M. (2003). An algorithm for word-level alignment of parallel dependency trees. In *Machine Translation Summit IX*, pages 95–101, New Orleans, LA.

Doddington, G. (2002). Automatic evaluation of Machine Translation quality using N-gram co-occurrence statistics. In *Proceedings of the second international conference on Human Language Technology Research*, pages 138–145, San Francisco, CA.

Du, J., Ma, Y., and Way, A. (2009). Source-side context-informed hypothesis alignment for combining outputs from Machine Translation systems. In *Proceedings of the Machine Translation Summit XII*, pages 230–237, Ottawa, ON, Canada.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Dyer, C. (2009). Using a maximum entropy model to build segmentation lattices for mt. In *Proceedings of Human Language Technologies and the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 406–414, Boulder, CO.

Dyer, C., Muresan, S., and Resnik, P. (2008). Generalizing word lattice translation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1012–1020, Columbus, OH.

Eisner, J. (2003). Learning non-isomorphic tree mappings for Machine Translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL-03), Companion Volume*, pages 205–208, Sapporo, Japan.

Fiscus, J. G. (1997). A post-processing system to yield reduced Word Error Rates: Recogniser output voting error reduction (ROVER). In *Proceedings 1997 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 347–352, Santa Barbara, CA.

Fraser, A. and Marcu, D. (2006). Semi-supervised training for statistical word alignment. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 769–776, Sydney, Australia.

Fraser, A. and Marcu, D. (2007a). Getting the structure right for word alignment: LEAF. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 51–60, Prague, Czech Republic.

Fraser, A. and Marcu, D. (2007b). Measuring word alignment quality for Statistical Machine Translation. *Computational Linguistics*, 33(3):293–303.

Galley, M., Graehl, J., Knight, K., Marcu, D., DeNeefe, S., Wang, W., and Thayer, I. (2006). Scalable inference and training of context-rich syntactic translation models. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 961–968, Sydney, Australia.

Ganchev, K., Graça, J. a. V., and Taskar, B. (2008). Better alignments = better translations? In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics and Human Language Technology Conference (ACL-08:HLT)*, pages 986–993, Columbus, OH.

Germann, U. (2003). Greedy decoding for Statistical Machine Translation in almost linear time. In *Proceedings of the Joint Meeting of the Human Language Technology Conference and the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2003)*, pages 72–79, Edmonton, AB, Canada.

Gildea, D. (2003). Loosely tree-based alignment for Machine Translation. In *Proceedings of the 41st Annual Meeting of the Association of Computational Linguistics (ACL-03)*, pages 80–87, Sapporo, Japan.

Hall, J. (2006). *MaltParser - An Architecture for Inductive Labeled Dependency Parsing*. Licentiate thesis, Växjö University: School of Mathematics and Systems Engineering.

Haque, R., Naskar, S., Ma, Y., and Way, A. (2009). Using supertags as source language context in SMT. In *Proceedings of the 13th Annual Meeting of the European Association for Machine Translation (EAMT 2009)*, pages 234–241, Barcelona, Spain.

Hassan, H., Ma, Y., and Way, A. (2007a). MaTrEx: the DCU Machine Translation system for IWSLT 2007. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 21–28, Trento, Italy.

Hassan, H., Sima'an, K., and Way, A. (2007b). Supertagged Phrase-Based Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 288–295, Prague, Czech Republic.

Hassan, H., Sima'an, K., and Way, A. (2008). Syntactically lexicalized Phrase-Based SMT. *IEEE Transactions on Audio, Speech & Language Processing*, 16(7):1260–1273.

Huang, C.-C., Chen, W.-T., and Chang, J. S. (2008). Bilingual segmentation for alignment and translation. In *Proceedings of 9th International Conference Computational Linguistics and Intelligent Text Processing*, pages 445–453, Haifa, Israel.

Hwa, R., Resnik, P., Weinberg, A., and Kolak, O. (2002). Evaluating translational correspondence using annotation projection. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 392–399, Philadelphia, PA.

Ittycheriah, A. and Roukos, S. (2005). A maximum entropy word aligner for Arabic-English Machine Translation. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 89–96, Vancouver, BC.

Jelinek, F. (1977). *Statistical Methods for Speech Recognition.* MIT Press, Cambridge, MA.

Johansson, R. and Nugues, P. (2007). Extended constituent-to-dependency conversion for English. In *Proceedings of NODALIDA 2007*, pages 105–112, Tartu, Estonia.

Jurafsky, D. and Martin, J. H. (2008). *Speech and Language Processing.* Prentice Hall (2 edition), Upper Saddle River, NJ.

Ker, S. J. and Chang, J. S. (1997). A class-based approach to word alignment. *Computational Linguistics*, 23(2):313–343.

Kneser, R. and Ney, H. (1995). Improved backing-off for m-gram language modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 1, pages 181–184, Detroit, MI.

Knight, K. (1999). Decoding complexity in word-replacement translation models. *Computational Linguistics*, 25(4):607–615.

Koehn, P. (2004). Statistical significance tests for Machine Translation evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain.

Koehn, P. (2005). Europarl: A parallel corpus for Statistical Machine Translation. In *Machine Translation Summit X*, pages 79–86, Phuket, Thailand.

Koehn, P. (2009). *MOSES: A Beam-Search Decoder for Factored Phrase-Based Statistical Machine Translation Models User Manual and Code Guide.*

Koehn, P., Birch, A., and Steinberger, R. (2009). 462 machine translation systems for europe. In *Proceedings of the Machine Translation Summit XII*, pages 65–72, Ottawa, ON, Canada.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.

Koehn, P., Och, F., and Marcu, D. (2003). Statistical Phrase-Based Translation. In *Proceedings of the 2003 Human Language Technology Conference and the North*

American Chapter of the Association for Computational Linguistics, pages 48–54, Edmonton, AB, Canada.

Lambert, P., Banchs, R. E., and Crego, J. M. (2007). Discriminative alignment training without annotated data for Machine Translation. In *Proceedings of Human Language Technologies Conference and Conference of the North American Chapter of the Association for Computational Linguistics*, pages 85–88, Rochester, NY.

Lambert, P., Ma, Y., Ozdowska, S., and Way, A. (2009). Tracking relevant alignment characteristics for Machine Translation. In *Proceedings of the Machine Translation Summit XII*, pages 268–275, Ottawa, ON, Canada.

Liang, P., Taskar, B., and Klein, D. (2006). Alignment by agreement. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 104–111, New York, NY.

Lin, D., Zhao, S., Van Durme, B., and Paşca, M. (2008). Mining parenthetical translations from the web by word alignment. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics and Human Language Technology Conference (ACL-08:HLT)*, pages 994–1002, Columbus, OH.

Liu, Y., Liu, Q., and Lin, S. (2005). Log-linear models for word alignment. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 459–466, Ann Arbor, MI.

Liu, Y., Liu, Q., and Lin, S. (2006). Tree-to-string alignment template for Statistical Machine Translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 609–616, Sydney, Australia.

Lopez, A. and Resnik, P. (2005). Improved HMM alignment models for languages with scarce resources. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 83–86, Ann Arbor, MI.

Ma, Y., Lambert, P., and Way, A. (2009a). Tuning syntactically enhanced word alignment for Statistical Machine Translation. In *Proceedings of the 13th Annual Meeting of the European Association for Machine Translation (EAMT 2009)*, pages 250–257, Barcelona, Spain.

Ma, Y., Okita, T., Özlem Çetinoğlu, Du, J., and Way, A. (2009b). Low-resource Machine Translation using MaTrEx: The DCU Machine Translation system for IWSLT 2009. In *Proceedings of International Workshop on Spoken Language Translation (IWSLT09)*, Tokyo, Japan. To appear.

Ma, Y., Ozdowska, S., Sun, Y., and Way, A. (2008a). Improving word alignment using syntactic dependencies. In *Proceedings of the ACL-08: HLT Second Workshop on Syntax and Structure in Statistical Translation (SSST-2)*, pages 69–77, Columbus, OH.

Ma, Y., Stroppa, N., and Way, A. (2007a). Alignment-guided chunking. In *Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 114–121, Skövde, Sweden.

Ma, Y., Stroppa, N., and Way, A. (2007b). Bootstrapping word alignment via word packing. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 304–311, Prague, Czech Republic.

Ma, Y., Tinsley, J., Hassan, H., Du, J., and Way, A. (2008b). Exploiting alignment techniques in MaTrEx: the DCU Machine Translation system for IWSLT08. In *Proceedings of International Workshop on Spoken Language Translation (IWSLT08)*, pages 26–33, Honolulu, HI.

Ma, Y. and Way, A. (2009a). Bilingually motivated domain-adapted word segmentation for Statistical Machine Translation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2009)*, pages 549–557, Athens, Greece.

Ma, Y. and Way, A. (2009b). Bilingually motivated word segmentation for Statistical Machine Translation. *ACM Transactions on Asian Language Information Processing, Special Issue on Machine Translation of Asian Languages*, 8(2):1–24.

Macherey, W., Och, F., Thayer, I., and Uszkoreit, J. (2008). Lattice-based Minimum Error Rate Training for Statistical Machine Translation. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 725–734, Honolulu, HI.

Magerman, D. M. (1995). Statistical decision-tree models for parsing. In *In Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pages 276–283.

Marcu, D., Wang, W., Echihabi, A., and Knight, K. (2006). SPMT: Statistical Machine Translation with syntactified target language phrases. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 44–52, Sydney, Australia.

Marcu, D. and Wong, W. (2002). A Phrase-Based, joint probability model for Statistical Machine Translation. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, pages 133–139, Morristown, NJ.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Mariño, J. B., Banchs, R. E., Crego, J. M., de Gispert, A., Lambert, P., Fonollosa, J. A. R., and Costa-jussà, M. R. (2006). N-gram-based Machine Translation. *Computational Linguistics*, 32(4):527–549.

Matusov, E., Zens, R., and Ney, H. (2004). Symmetric word alignments for Statistical Machine Translation. In *Proceedings of the 20th international conference on Computational Linguistics (COLING 2004)*, pages 219–225, Geneva, Switzerland.

Melamed, I. D. (1996). Automatic construction of clean broad-coverage translation lexicons. In *Proceedings of the 2nd Conference of the Association for Machine Translation in the Americas (AMTA)*, pages 125–134, Montreal, QC, Canada.

Melamed, I. D. (1997). Automatic discovery of non-compositional compounds in parallel data. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, pages 97–108, Somerset, NJ.

Melamed, I. D. (2000). Models of translational equivalence among words. *Computational Linguistics*, 26(2):221–249.

Moore, R. C. (2004). Improving IBM word alignment model 1. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pages 518–525, Barcelona, Spain.

Moore, R. C. (2005). A discriminative framework for bilingual word alignment. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 81–88, Vancouver, BC.

Moore, R. C., Yih, W.-T., and Bode, A. (2006). Improved discriminative bilingual word alignment. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 513–520, Sydney, Australia.

Nichols, C. and Hwa, R. (2005). Word alignment and cross-lingual resource acquisition. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 69–72, Ann Arbor, MI.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S., and Marsi, E. (2007). MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Noreen, E. W. (1989). *Computer-Intensive Methods for Testing Hypotheses: An Introduction.* Wiley-Interscience, New York, NY.

Och, F. (2002). *Statistical Machine Translation: From Single Word Models to Alignment Templates*. PhD thesis, RWTH Aachen.

Och, F. (2003). Minimum Error Rate Training in Statistical Machine Translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan.

Och, F. and Ney, H. (2002). Discriminative training and maximum entropy models for Statistical Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 295–302, Philadelphia, PA.

Och, F. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Ostendorf, M., Digalakis, V. V., and Kimball, O. A. (1996). From HMMs to segment models: A unified view of stochastic modeling for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 4(5):360–378.

Ozdowska, S. (2004). Identifying correspondences between words: an approach based on a bilingual syntactic analysis of French/English parallel corpora. In *Proceedings of the COLING'04 Workshop on Multilingual Linguistic Resources*, pages 49–56, Geneva, Switzerland.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a method for automatic evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, PA.

Paul, M. (2006). Overview of the IWSLT 2006 Evaluation Campaign. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 1–15, Kyoto, Japan.

Quirk, C., Menezes, A., and Cherry, C. (2005). Dependency treelet translation: Syntactically informed phrasal SMT. In *Proceedings of the 43rd Annual Meeting* *of the Association for Computational Linguistics (ACL'05)*, pages 271–279, Ann Arbor, MI.

Ratnaparkhi, A. (1996). A maximum entropy model for part-of-speech tagging. In Brill, E. and Church, K., editors, *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 133–142, Somerset, NJ.

Ren, D., Wu, H., and Wang, H. (2007). Improving statistical word alignment with various clues. In *Machine Translation Summit XI*, pages 391–397, Copenhagen, Denmark.

Riezler, S. and Maxwell, J. T. (2005). On some pitfalls in automatic evaluation and significance testing for MT. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 57–64, Ann Arbor, MI.

Smadja, F., McKeown, K. R., and Hatzivassiloglou, V. (1996). Translating collocations for bilingual lexicons: A statistical approach. *Computational Linguistics*, 22(1):1–38.

Srivastava, A. and Way, A. (2009). Using percolated dependencies for phrase extraction in SMT. In *Proceedings of the Machine Translation Summit XII*, pages 316–323, Ottawa, ON, Canada. To appear.

Stolcke, A. (2002). SRILM – An extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken Language Processing*, pages 901–904, Denver, CO.

Stroppa, N., van den Bosch, A., and Way, A. (2007). Exploiting source similarity for SMT using context-informed features. In *Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 231–240, Skövde, Sweden.

Sun, L., Jin, Y., Du, L., and Sun, Y. (2000). Word alignment of English-Chinese bilingual corpus based on chunks. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical Methods in Natural Language Processing and very large corpora*, pages 110–116, Hong Kong.

Takezawa, T., Sumita, E., Sugaya, F., Yamamoto, H., and Yamamoto, S. (2002). Toward a broad-coverage bilingual corpus for speech translation of travel conversations in the real world. In *Proceedings of Third International Conference on Language Resources and Evaluation 2002*, pages 147–152, Las Palmas, Canary Islands, Spain.

Taskar, B., Simon, L.-J., and Dan, K. (2005). A discriminative matching approach to word alignment. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 73–80, Vancouver, BC, Canada.

Tiedemann, J. (2003). Combining clues for word alignment. In *Proceedings of the 10th Conference of European Chapter of the Association for Computational Linguistics*, pages 339–346, Budapest, Hungary.

Tinsley, J., Ma, Y., Ozdowska, S., and Way, A. (2008). MaTrEx: The DCU MT system for WMT 2008. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 171–174, Columbus, OH.

Tinsley, J., Zhechev, V., Hearne, M., and Way, A. (2007). Robust language-pair independent sub-tree alignment. In *Machine Translation Summit XI*, pages 467–474, Copenhagen, Denmark.

Toutanova, K., Ilhan, H. T., and Manning, C. (2002). Extentions to hmm-based statistical word alignment models. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 87–94, Philadelphia, PA.

Tseng, H., Chang, P., Andrew, G., Jurafsky, D., and Manning, C. (2005). A conditional random field word segmenter for sighan bakeoff 2005. In *Proceedings of Fourth SIGHAN Workshop on Chinese Language Processing*, pages 168–171, Jeju Island, Republic of Korea.

van Rijsbergen, C. (1979). *Information Retrieval.* Butterworths, London, UK.

Vapnik, V. N. (1998). *Statistical Learning Theory.* Wiley-Interscience, Hoboken, NJ.

Vilar, D., Popovic, M., and Ney, H. (2006). AER: Do we need to "improve" our alignments? In *Proceedings of the International Workshop on Spoken Language Translation*, pages 205–212, Kyoto, Japan.

Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13:1260–1269.

Vogel, S., Ney, H., and Tillmann, C. (1996). HMM-based word alignment in statistical translation. In *Proceedings of the 16th International Conference on Computational Linguistics*, pages 836–841, Copenhagen, Denmark.

Wang, W. and Zhou, M. (2004). Improving word alignment models using structured monolingual corpora. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, pages 198–205, Barcelona, Spain.

Wu, D. (1997). Stochastic Inversion Transduction Grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–403.

Wu, H., Wang, H., and Liu, Z. (2006). Boosting statistical word alignment using labeled and unlabeled data. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 913–920, Sydney, Australia.

Xu, J., Gao, J., Toutanova, K., and Ney, H. (2008). Bayesian semi-supervised Chinese word segmentation for Statistical Machine Translation. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING 2008)*, pages 1017–1024, Manchester, UK.

Xu, J., Matusov, E., Zens, R., and Ney, H. (2005). Integrated Chinese word segmentation in Statistical Machine Translation. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 141–147, Pittsburgh, PA.

Xu, J., Zens, R., and Ney, H. (2004). Do we need Chinese word segmentation for Statistical Machine Translation? In *Proceedings of the Third SIGHAN Workshop on Chinese Language Learning*, pages 122–128, Barcelona, Spain.

Xue, N., Xia, F., Chiou, F.-d., and Palmer, M. (2005). The Penn Chinese TreeBank: Phrase structure annotation of a large corpus. *Natural Language Engineering*, 11(2):207–238.

Yamada, H. and Matsumoto, Y. (2003). Statistical dependency analysis with Support Vector Machines. In *Proceedings of the 8th International Workshop on Parsing Technologies (IWPT)*, pages 195–206, Nancy, France.

Yamada, K. and Knight, K. (2001). A syntax-based statistical translation model. In *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pages 523–530, Toulouse, France.

Zhang, H., Yu, H., Xiong, D., and Liu, Q. (2003). HHMM-based Chinese lexical analyzer ICTCLAS. In *Proceedings of Second SIGHAN Workshop on Chinese Language Processing*, pages 184–187, Sapporo, Japan.

Zhang, R., Yasuda, K., and Sumita, E. (2008). Improved Statistical Machine Translation by multiple Chinese word segmentation. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 216–223, Columbus, OH.

Zollmann, A. and Venugopal, A. (2006). Syntax augmented Machine Translation via chart parsing. In *Proceedings of the Workshop on Statistical Machine Translation*, pages 138–141, New York City, NY.

Appendix A: Penn English Treebank Tag Set

Tags	Description	Tags	Description
CC	Coordinating conjunction	PP$	Possessive pronoun
CD	Cardinal number	RB	Adverb
DT	Determiner	RBR	Adverb, comparative
EX	Existential there	RBS	Adverb, superlative
FW	Foreign word	RP	Particle
IN	Preposition or subord. conj.	SYM	Symbol
JJ	Adjective	TO	to
JJR	Adjective, comparative	UH	Interjection
JJS	Adjective, superlative	VB	Verb, base form
LS	List item marker	VBD	Verb, past tense
MD	Modal	VBG	Verb, gerund or present participle
NN	Noun, singular or mass	VBN	Verb, past participle
NNS	Noun, plural	VBP	Verb, non-3rd person sing. present
NP	Proper noun, singular	VBZ	Verb, 3rd person sing. present
NPS	Proper noun, plural	WDT	Wh-determiner
PDT	Predeterminer	WP	Wh-pronoun
POS	Possessive ending	WP$	Possessive wh-pronoun
PP	Personal pronoun	WRB	Wh-adverb

Appendix B: Penn Chinese Treebank Tag Set

Tags	Descrition	Tags	description
AD	Adverb	M	Measure word
AS	Aspect marker	MSP	Other particle
BA	“ba” in a ba-construction	NN	Common noun
CC	Coordinating conjunction	NR	Proper noun
CD	Cardinal number	NT	Temporal noun
CS	Subordinating conjunction	OD	Ordinal number
DEC	“de” in a relative-clause	ON	Onomatopoeia
DEG	Associative “de”	P	Prepostion excluding “bei” and “ba”
DER	“de” in V-de construction	PN	Pronoun
DEV	“de” before VP	PU	Punctuation
DT	Determiner	SB	“bei” in short bei-construction
ETC	For words “deng”, “dengdeng”	SP	Sentence-final particle
FW	Foreign words	VA	Predicative adjective
IJ	Interjection	VC	“shi”
JJ	Other noun-modifier	VE	“you” as the main verb
LB	“bei” in long bei-construction	VV	Other verb
LC	Localiser		

Appendix C: English Dependency Types

Labels	Description
ADV	Unclassified adverbial
PRD	Predicative complement
LGS	Logical subject
SBJ	Subject
AMOD	Modifier of adjective or adverb
NMOD	Modifier of nominal
P	Punctuation
PMOD	Between preposition and its child in a PP
PRT	Particle
VC	Verb chain
ROOT	Root
VMOD	General adverbial
OBJ	Direct object or clause complement
IOBJ	Indirect object
DEP	Unclassified relation
CC*	Between conjunction and second conjunct in a coordination
COORD	Coordination
PRN	Parenthetical
EXP*	Extraposed element in expletive construction
CLF*	Cleft sentence

Appendix D: Chinese Dependency Types

Label	Description
PRD	Predicative complement
AMOD	Modifier of adjective or adverb
NMOD	Modifier of nominal
P	Punctuation
PMOD	Between preposition and its child in a PP
VC	Verb chain
ROOT	Root
VMOD	General adverbial
OBJ	Direct object or clause complement
SBJ	Subject
SBAR	Complementiser dependent
DEP	Unclassified relation

Appendix E: Head Percolation Table for Penn English Treebank

<table>
<tr><td>NP</td><td>R</td><td>POS|NN|NNP|NNPS|NNS NX JJR CD JJ JJS RB QP NP</td></tr>
<tr><td>ADJP</td><td>R</td><td>NNS QP NN $ ADVP JJ VBN VBG ADJP JJR NP JJS DT FW RBR RBS SBAR RB</td></tr>
<tr><td>ADVP</td><td>L</td><td>RB RBR RBS FW ADVP TO CD JJR JJ IN NP JJS NN</td></tr>
<tr><td>CONJP</td><td>L</td><td>CC RB IN</td></tr>
<tr><td>FRAG</td><td>L</td><td></td></tr>
<tr><td>INTJ</td><td>R</td><td></td></tr>
<tr><td>LST</td><td>L</td><td>LS :</td></tr>
<tr><td>NAC</td><td>R</td><td>NN|NNS|NNP|NNPS NP NAC EX $ CD QP PRP VBG JJ JJS JJR ADJP FW</td></tr>
<tr><td>PP</td><td>L</td><td>IN TO VBG VBN RP FW</td></tr>
<tr><td>PRN</td><td>R</td><td></td></tr>
<tr><td>PRT</td><td>L</td><td>RP</td></tr>
<tr><td>QP</td><td>R</td><td>$ IN NNS NN JJ RB DT CD NCD QP JJR JJS</td></tr>
<tr><td>RRC</td><td>L</td><td>VP NP ADVP ADJP PP</td></tr>
<tr><td>S</td><td>R</td><td>TO IN VP S SBAR ADJP UCP NP</td></tr>
<tr><td>SBAR</td><td>R</td><td>WHNP WHPP WHADVP WHADJP IN DT S SQ SINV SBAR FRAG</td></tr>
<tr><td>SBARQ</td><td>R</td><td>SQ S SINV SBARQ FRAG</td></tr>
<tr><td>SINV</td><td>R</td><td>VBZ VBD VBP VB MD VP S SINV ADJP NP</td></tr>
<tr><td>SQ</td><td>R</td><td>VBZ VBD VBP VB MD VP SQ</td></tr>
<tr><td>UCP</td><td>L</td><td></td></tr>
<tr><td>VP</td><td>L</td><td>VBD VBN MD VBZ VB VBG VBP VP ADJP NN NNS NP</td></tr>
<tr><td>WHADJP</td><td>R</td><td>CC WRB JJ ADJP</td></tr>
<tr><td>WHADVP</td><td>L</td><td>CC WRB</td></tr>
<tr><td>WHNP</td><td>R</td><td>WDT WP WP$ WHADJP WHPP WHNP</td></tr>
<tr><td>WHPP</td><td>L</td><td>IN TO FW</td></tr>
</table>

NX	R	POS|NN|NNP|NNPS|NNS NX JJR CD JJ JJS RB QP NP
X	R	

Appendix F: Head Percolation Table for Penn Chinese Treebank

ADJP	r ADJP JJ;r AD NN CS;r
ADVP	r ADVP AD;r
CLP	r CLP M;r
CP	r DEC SP;l ADVP CS;r CP IP;r
DNP	r DNP DEG;r DEC;r
DP	l DP DT;l
DVP	r DVP DEV;r
FRAG	r VV NR NN;r
INTJ	r INTJ IJ;r
IP	r IP VP;r VV;r
LCP	r LCP LC;r
LST	l LST CD OD;l
NP	r NP NN NT NR QP;r
PP	l PP P;l
PRN	r NP IP VP NT NR NN;r
QP	r QP CLP CD OD;r
UCP	r
VCD	r VCD VV VA VC VE;r
VCP	r VCP VV VA VC VE;r
VNV	r VNV VV VA VC VE;r
VP	l VP VA VC VE VV BA LB VCD VSB VRD VNV VCP;l
VPT	r VNV VV VA VC VE;r
VRD	r VRD VV VA VC VE;r
VSB	r VSB VV VA VC VE;r
WHNP	r WHNP NP NN NT NR QP;r
WHPP	l WHPP PP P;l

Appendix G: English Dependency Type Derivation Rules

1. If D is the first more than one object, r=IOBJ
2. If D is an object, r=OBJ
3. If D=PRN, r=PRN
4. If D is a punctuation category, r=P
5. If D is coordinated with M, r=COORD
6. If D=PP,ADVP or SBAR, and M=VP, r=ADV
7. If D=PRT and M=VP, r=PRT
8. If D=VP and M=VP, SQ or SINV, r=VC
9. If M=VP, S, SBAR, SBARQ, SINV, or SQ, r=VMOD
10. If M=NP,NX,NAC or WHNP, r=NMOD
11. If M=ADJP, ADVP, WHADJP or WHADVP, r=AMOD
12. If M=PP or WHPP, r=PMOD
13. Otherwise, r=DEP

Note that given a token e, D is the highest phrase that e is the head of, M is the parent of D, and r is the label on the dependency arc from e to its parent.

Appendix H: Chinese Dependency Type Derivation Rules

1. If D is punctuation category, r = P
2. If D contains the function tag SBJ, r=SBJ
3. If M=VP, H=TAG and D=NP-OBJ, r=OBJ
4. If M=VP, H=TAG and D=VP, r=VC
5. If M=VCD, VCP, VRD or VSB, R=VC
6. If M=VNV or VPT, H=TAG, r=VC
7. If M=CP and D=IP, r=SBAR
8. If M=VP, IP, SQ, SINV or CPQ, r=VMOD
9. If M=NP, NAC, NX or WHNP, r=NMOD
10. If ADJP, ADVP, QP, WHADJP or WHADVP, r=AMOD
11. M=PP or WHPP, r =PMOD
12. Otherwise, r=DEP

Note that given a token f, D is the label of the highest phrase that f is the head of, M is the parent of D, r is the label on the dependency arc from f to its parent token h, and H is the label of the highest phrase that h is the head of.