Department of Computer Science
Series of Publications C
Report C-2009-22

Phrase table pruning

for Statistical Machine Translation

Esther Galbrun

University of Helsinki
Finland

Phrase table pruning for Statistical Machine Translation

Esther Galbrun

Department of Computer Science
P.O. Box 68, FIN-00014 University of Helsinki, Finland
esther.galbrun@cs.helsinki.fi

Technical report, Series of Publications C, Report C-2009-22
Helsinki, July 2009, 38 pages

Abstract

Phrase-Based Statistical Machine Translation systems model the translation process using pairs of corresponding sequences of words extracted from parallel corpora. These biphrases are stored in phrase tables that typically contain several millions such entries, making it difficult to assess their quality without going to the end of the translation process. Our work is based on the examplifying study of phrase tables generated from the *Europarl* data, from French to English. We give some statistical information about the biphrases contained in the phrase table, evaluate the coverage of previously unseen sentences and analyse the effects of pruning on the translation.

Computing Reviews (1998) Categories and Subject Descriptors:
I.2.6 Learning
I.2.7 Natural Language Processing

General Terms:
Machine translation

Additional Key Words and Phrases:
Phrase-Based Statistical Machine Translation, Phrase table

Contents

Chapter 1

Introduction

The aim of *Machine Translation* is to automatically translate sentences from a given language into another. *Phrase-Based Statistical Machine Translation* systems, one approach to solve this problem, use pairs of corresponding sequences of words in the source and target languages to build a probabilistic model of the translation process.

Extracting pairs of corresponding phrases together with their word to word links, the biphrases, from sentence aligned bilingual corpora using statistical and heuristic models to compute the word alignments and storing them in a phrase table is the first step to set such a translation system up.

When no intermediate evaluation is available, the full training procedure of the translation system has to be completed before any evaluation can be conducted. The complete loop often requires several days to be carried out, making incremental improvements impractical. Besides the phrase table, many factors are involved in training the system and generating the translations. Thus, in the absence of intermediate evaluation, determining which part of the system is at fault when the translation quality is unsatisfactory can also be challenging.

Therefore, after acquiring a better undertanding of what phrase tables actually contain by computing statistics about the basic characteristics of the biphrases (phrase length, number of occurrences, etc.), we would like to find a way to evaluate their intrinsic quality. Since they typically contain several millions of entries, a manual evaluation by browsing through the biphrases is simply unfeasible. For this reason, we try to estimate the ability of the phrase table to cover previously unseen sentences, without making any assumption on the system that uses them. We want to determine whether the biphrases needed to construct the translations are present in the phrase table, regardless of how the system can combine them.

Finally, we investigate how manipulating the phrase table by filtering out some of the biphrases impacts the translation. We consider the effects of different pruning methods on the translation quality as well as on the size of the model and the translation speed.

The work reported here is based on the examplifying study of translation from French to English, using distinct subsets from the *Europarl* corpus to train and evaluate the systems.

After giving a overview of the field of *Machine Translation in general* and of *Phrase-Based Statistical Machine Translation* more specifically in Chapter 2, we focus on the phrase tables, explaining how they are generated and briefly reviewing related work in Chapter 3. Chapter 4 describes our experimental settings and the results obtained. Chapter 5 concludes this work.

Chapter 2

Preliminaries

2.1 Machine Translation

After defining the main goals of *Machine Translation (MT)*, we briefly present the history of this field. Next, we outline the main approaches that have been used in solving this problem.

2.1.1 Goals

Machine Translation aims at translating sentences from a source language X to a target language Y. The ultimate goal for *MT* would be to obtain perfect translations, i.e. translations that could not be discriminated from human translations. Yet, this still seems to be too high a target.

There are two main purposes for machine translation outputs, *assimilation* and *dissemination.* When used for *assimilation* purposes, the translation should help the reader in understanding texts originally available in a language he does not read. On the other hand, when used for *dissemination* purposes, the output is typically post-processed by a human translator in order to obtain high quality translation to be published.

In the case of *assimilation* the main objective is to retain as much as possible from the original meaning of the text while in the case of *dissemination* it should output sentences that require minimal post-editing before being acceptable translations for the original sentences. Of course, these two objectives are closely related. A perfect translation would reflect the original content and would not need any edition. But while redundant translations of some difficult words might help to understand the meaning of the translation it would only slow down the post-editing.

MT systems can also be used as part of larger systems. For example, they can be used in cross-lingual information retrieval or for automatic speech processing. In those cases, again, as the end-use changes, the way of characterizing a good candidate translation also varies.

2.1.2 Brief history

A detailed history of *MT* can be found in [20]. Only the main steps are reported here.

Long before computers became available did intellectuals envision the use of machines to translate from one language into another. Shortly after the Second World War, Warren Weaver [36] suggested that some of the innovations made during the war in the field of cryptography could be applied to *MT*. He compared translating a text from chinese into english to deciphering some encrypted text, the cipher being chinese language. In the 50's and 60's, the first attemps to make the old dream of automatic translation become true, resorting mainly to *rule-based systems*, failed to fulfill the high expectations they had generated. Bar-Hillel, one of the first *MT* researchers, concluded in 1960 in his review [6] that the objective of producing automatic translations undistinguishable from human translations is unrealistic and had to be abandonned. In 1966, a report published by the Automatic Language Processing Advisory Committee (ALPAC) in the United States presented *MT* as a failure and put an end to almost all research in the field [18]. In the 80's, some operational systems were released and attracted back attention. *Systran* [33] is probably the most famous of them. From the late 80's, as more resources became available, approaches based on corpora - *exemple-based* and *statistical MT* - started to be developed. The latter in particular keeps attracting increasing attention as the amount of data available to feed to those systems continues to grow.

2.1.3 Approaches

We will now introduce the main approaches for solving the problem of *Machine Translation.*

Expert systems

The first operational *MT* systems were *Rule-Based systems.* Such systems use bilingual dictionaries and a large set of rules that are automatically applied to generate a translation. Generally the set of rules needs to be written by a linguist for each specific pair of languages. Alternatively, an artificial representation, an *interlingua*, can be used as a universal intermediate representation of the semantic content.

Data-driven systems

Example-Based and *statistical MT* systems both rely on bilingual corpora. But while the former generates translations based on analogies retrieved from the parallel texts at runtime [19], the latter requires a training step to carry out a statistical analysis of the corpora in order to extract relevant knowledge to be used while translating.

Statistical systems *Statistical Machine Translation (SMT)* [24] uses a probabilistic representation of natural languages and the translation process. To all possible pairs of source language sentence x and target language sentence y is associated a value $Pr(y|x)$. This value represents the probability that given the sentence x a translator would choose y as its translation. The best translation given a sentence x is then defined as the sentence $\hat{y}$ that maximizes $Pr(y|x)$. Using Bayes' theorem this can be rewritten as

$$Pr(y|x) = \frac{Pr(y)Pr(x|y)}{Pr(x)}.$$

For a given source sentence the denominator is constant. Therefore the sentence

$$\hat{y} = \arg\max_{y} Pr(x|y)Pr(y)$$

is the best translation for the source sentence x.

$Pr(y)$ models the probability that the sentence y is a valid sentence in the target language, while $Pr(y|x)$ models the probability that y is a good translation for x. The former model is called the *language model*, the latter is the *translation model.*

The most common *language models* are based on counts of occurrences of sequences of n successive words, the *n-grams*, in large monolingual texts.

- **Phrase-Based and Syntax-Based SMT**

  The knowledge extracted from the bilingual corpora in *SMT* systems to model the translation probabilities can take different forms. It can be syntactic rules, typically represented as operations on parse trees [14], in the case of *Syntax-Based SMT*, or pairs of corresponding sequences of words in the source and target languages, *aligned phrases*, in the case of *Phrase-Based SMT.* The extracted sequences of words in the source and target languages may have varying size. This should take care of *fertility* issues, i.e., the fact that a word in language may not be translated into exactly one word in the other language. This should also alleviate the problem of *reorderings*, i.e., the fact that the words in the target language do not necessarily appear in the same order as the source words they translate and may need to be reordered. However, it cannot entirely solve this issue since the length of the extracted phrases is limited and cannot cover sentence wide reorderings. The set of corresponding sequences of words in the source and target languages is called a phrase table.

- **Phrase-Based Machine Translation and Machine Learning**

  Phrase-Based machine translation can be cast as a machine learning problem. One way of doing so is, for a given input sentence, to predict a label that indicates which phrases appear in the translation and at what position.

2.2 Phrase-Based Statistical Machine Translation

In this section we will present in more detail *Phrase-Based Statistical Machine Translation (PB-SMT)* systems, in which phrase tables, the object of our interest in this work, are used as the base element to model the translation probability $Pr(y|x)$. In the following discussion, we shall distinguish between *sentences*, the linguistic units of meaning and *phrases*, sequences of words of varying lengths whose boundaries do not necessarily have a linguistic motivation. A pair of a source language phrase and a target language phrase is called a biphrase and is generally associated with its word to word correspondence relation, called *alignment*.

2.2.1 System architecture

Here we want to give an overview of the components that make up a *Phrase-Based SMT* system. As an example, we describe the architecture of `Sinuhe` translation system [22], one of the systems we studied in this work. In both this system and `Moses` [17], the system used as a baseline in our work, the phrase table is generated the same way, as described below. However the two translation models use it differently, so the training processes as well as the *decoding* are different.

Training material

The starting material for *Phrase-Based SMT* systems is a large bilingual corpus, typically two large texts in the source and target languages which are translations of each other and are aligned at sentence level. Alignment at sentence level means that corresponding lines of the two texts contain sentences that are translations of each other. The bilingual material is separated into a training set, from which the biphrases are extracted and their weights learnt, a tuning set, used to adjust the values of the parameters of the decoder and an evaluation set, to assess the translation quality. A separate monolingual corpus in the target language is needed to train the *language model*.

Alignment and phrase extraction

Before extracting the aligned biphrases, the training set is tokenized and lowercased. The most common way to perform the phrase extraction is to generate the word to word alignment and then extract the set of biphrases that are compatible with it, called phrase table. The `GIZA++` implementation of the IBM models [7] is generally used to perform the word alignment. The IBM models are statistical models of the translation process that are used to evaluate the probabilities of word to word alignements for all pairs of source word and target word given a pair of aligned sentences. There are five models of increasing complexity to take into account effects such as *distortion*, i.e., the fact that the translations of some words may be swapped, and *fertility*, i.e., the fact that one word is not always translated into exactly one word in the other language. The parameter values estimated for one model are used as initial values for the next estimation. To obtain a good quality word alignment a series of successive estimations is needed. This iterative process generally requires several hours to be carried out for roughly one million sentences. The models allow one word of a target sentence to be linked to only one word of the source language. To by-pass this constraint, the word alignments are computed in both directions, from source to target and target to source, and the results are combined as a final step called symmetrization. The result of this alignment process is for each pair of training sentences a set of links between the source words and the target words. An example is given below where x is a source sentence, y a target sentence, and a their word to word alignment, i.e., a set of link between the words of x and y. For example the link $3-2$ indicates that the fourth word of x, `meme` is aligned with the third word of y, `also`.

```
x: je me permettrai meme , bien qu' ils soient  [...]
y: i would also like , although they are absent [...]
a: 0-0 1-1 3-2 2-3 4-4 5-5 6-5 7-6 8-7 9-8 10-9 [...]
```

A biphrase (x', a', y'), where x' is a source phrase, y' a target phrase and a' the alignment between the words of x' and y' induced by a is considered valid if it contains links but none of them crosses the boundaries of the biphrase. All valid biphrases are stored in a phrase table, along with their count of occurrences.

Learning

From this point `Sinuhe` and `Moses` differ in the use they make from the extracted biphrases to model the translation probabilities.

In `Sinuhe` the biphrases are used to construct $\phi(x, a, y)$, a vector indicating which biphrases occurs in (x, a, y), a pair of aligned sentences x and y and their word to word alignment a. More precisely $\phi(x, a, y)_{i,j}$ indicates whether the i^{th} biphrase of the phrase table occurs at position j of the source sentence. Then $\tilde{\phi}(x, a, y)_i$ is defined as $\tilde{\phi}(x, a, y)_i = \sum_{j=J} \phi(x, a, y)_{i,j}$, so it is the count of occurrences of the i^{th} biphrase in (x, a, y) over the set of all starting positions J.

The translation model in `Sinuhe` doesn't estimate the translation probability distribution $Pr(y|x)$ directly but $Pr(\phi(x, a, y)|x)$ instead, using the features $\tilde{\phi}(x, a, y)$ to build a conditional exponential model

$$Pr(\phi(x, a, y)|x) = \frac{exp(w.\tilde{\phi}(x, a, y))}{\sum_{\tilde{\phi} \in \Phi_x} exp(w \cdot \tilde{\phi})},$$

where Φ_x represents the set of all possible candidates for the sentence x. The *maximum a posteriori (MAP)* [31] estimates of the weights for the biphrases features w_i are computed using stochastic gradient ascent, where the gradients are computed by dynamic programming. The counts of occurrences associated to the biphrases can be used at this stage to compute regularization terms for the weights. Biphrases with unaligned end words are discarded from the phrase table as they cannot be handled by the dynamic procedure, as well as all biphrases occurring only once to prevent overfitting the training data. Since the weights are learnt from the same corpus as the features have been extracted from, if no pruning was applied to the phrase table prior to the learning phase, the system could simply use all the biphrases that were extracted from an aligned pair of sentences to reconstruct it.

Decoding

Decoding is the dynamic procedure of finding the translation that maximizes the translation probability. Once the weights have been learnt, the translation can be generated by selecting the vector $\tilde{\phi}(x, a, y)_i$ that receives the highest translation model probability and reconstructing the translation induced by the target side of the biphrases active in that vector.

Alternatively, additional scores can be taken into account to select the best candidate translation:

- a language model score, given by an external language model trained separately,
- a word level lexical translation probability,
- the length of the candidate translation, and

- a distortion score, to penalize for reorderings in the translation.

The contribution of the different scores in the decoding is tuned using *Minimum Error Rate Training (MERT)* to optimize the *BLEU* score (cf. 2.2.3) on the tuning corpus.

2.2.2 Data sources

As we mentioned, bilingual corpora are the starting material for *SMT* systems.

Since the Canadian Government is officially bilingual, the proceedings of the Canadian Parlament have to be maintained both in French and English. Likewise, the European Parliament also maintains proceedings in the official languages of its member states. The proceedings of these two political institutions, the *Canadian Hansard* [27] and the *Europarl* [23, 12] respectively, have traditionally been the most important resources for *Statistical Machine Translation* between European languages. A major inconvenience of these two sources arises from the fact that they are parliament proceedings. They have a very specific focus and contain many atypical formulations that are not useful to translate texts from other domains.

A new French-English corpus generated by automatically crawling bilingual websites has recently been released for the translation task of the fourth *European Chapter of the Association for Computational Linguistics (EACL) Workshop on Machine Translation (WMT09).* This *Giga French-English* [12] corpus contains over 20 millions sentences in both languages, to be compared to *Europarl* corpora of typically slightly more than 1 million sentences.

Finding training material is a crucial point in developing a *SMT* system for a new language pair. For some language pairs, in particular those that involve rare languages, finding aligned bilingual texts can be really challenging. Therefore, some alternative approaches have been developed to take advantage of texts that are similar but not exact translations of each other [29] or even from two monolingual datasets [16], or to use a third language as an intermediate [37].

2.2.3 Translation evaluation

Evaluation of translation systems output is a hard, tedious and highly subjective task. Common criteria are *fluency* and *adequacy. Fluency* indicates whether the translation is a correct sentence in the target language and can possibly be evaluated by a person who only reads the target language. *Adequacy* measures how well the original meaning was conveyed to the translation and needs to be evaluated by a bilingual person.

Automatic evaluation tools are required not only to compare systems but also during the training process since systems are often trained to optimize a criteria on the translation quality. Various metrics have been developed to approximate human judgment. They generally require human reference translations of the test sentences to be at hand. The most widely used metric for evaluating machine translation output is the *BLEU* score [28]. This score is based on *n-grams* precision evaluation. The basic idea is to count how many *n-grams* from the candidate translation are present in the reference. This might seem a coarse criterion for evaluation and its use is subject to much critisism. But while much research effort has been directed toward inventing metrics that correlate better with human judgement [3, 8, 38], no satisfactory solution has been developed. The very existence of an automatic tool for evaluating the quality of such complex objects as instances of natural language is arguable. There are for example many different ways to translate the same idea that might be acceptable. Using several references has been shown to increase the reliability of the evaluation [34]. There is also much discussion about how to evaluate the quality of candidate translations when no reference is available. This is needed in particular to rank several candidate translations generated by one or different systems.

Automatic *Machine Translation* evaluation remains a difficult task. The use of multiple metrics has been recommended but it can be computationally heavy and the results may be difficult to interpret so that *BLEU* score alone still is widely used despite its evident flaws.

Chapter 3

Background informations on phrase tables

3.1 Motivation

Phrase tables are corner stones of *Phrase-Based Statistical Machine Translation* systems. Therefore the phrase table quality is critical in the overall quality of the translation system. The quantity of biphrases in the model is also a very important factor determining the size of the model and the speed of the learning and of the translation processes.

We focus here on three systems:

- `Moses`, a state-of-the-art open-source toolkit [17] for *Statistical Machine Translation*, is usually used as a baseline in *Phrase-Based SMT*.
- A *SMT* software that models the translation probabilities using a conditional exponential family, `Sinuhe` [22].
- An application of multiview learning to machine translation, based on the maximum margin regression algorithm [32], the `Maximum Margin Based Translator` (`MMBT`).

Both `Sinuhe` and `MMBT` where developed for the *SMART* EU project [30].

We want to consider more closely the phrase tables used by those three systems. `Moses` and `Sinuhe` both rely on the `GIZA++` implementation of the IBM models [7] to generate the word alignments but the biphrases extracted after symmetrization are scored and filtered differently. `MMBT` has its own alignment, biphrases extraction and scoring algorithm. We will analyse theses three different phrase tables, trying to get a better understanding of what they contain and looking for patterns that would enable us to discriminate between good and bad biphrases. If such characteristics were found, we could in particular reduce the search space during the decoding process, without affecting significantly the quality of the final translation.

What we call *good* biphrases are pairs of source and target phrases that are correct translation of each other, have proper boundaries and valid weights. We would also like to find a compromise for the size of the model. On one hand, an important part of large phrase tables may be constituted of biphrases which are very specific to the training corpus and rarely occur in texts to translate. On the other hand, small phrase tables that contain only frequent expressions may be unable to translate constructs slightly out of the ordinary.

3.2 Phrase table generation process

In this section, we describe how the phrase table is generated from the sentence aligned training corpus. First for `Moses` and `Sinuhe` translation systems starting from word alignments generated using `GIZA++`, then with `MMBT`.

3.2.1 Using GIZA++

Most of the current *Phrase-Based SMT* systems rely on the GIZA++ implementation of the IBM Models [7] to produce word alignments, running the algorithm in both directions, source to target and target to source. Various heuristics can then be applied to obtain a symmetrized alignment a from those two. Most of them, such as grow-diag-final-and, that we used, start from the intersection of the two word alignment and enrich it with alignment points from the union. Word sequences are then stored in the phrase table as biphrases (x', y') along with their alignment a' if they satisfy the following conditions:

1. x' and y' are consecutive word subsequences in the source sentence x and target sentences y respectively and neither of them is longer than k words.

2. a', the alignment between the words of x' and y' induced by a, contains at least one link and all links from a have either both ends in a' or none.

Moses phrase table

Moses uses directly the biphrases extracted from the GIZA++ word aligments without any further processing apart from the scoring explained in 4.2.1.

Sinuhe phrase table

Sinuhe does not use the full phrase table obtained from the GIZA++ word aligments. After associating count-based features to the biphrases as presented in 4.2.2, before proceeding to the learning of the biphrases's weights, the phrase table is pruned by filtering out biphrases that satisfiy any of the six following criteria:

1. single occurrence, the pair (source phrase, target phrase) occurs only once in the corpus,

2. rank, the biphrase is not among the N most frequent among all biphrases sharing the same source phrase (N is typically fixed to 20),

3. first source word unaligned, the first word of the source phrase is not aligned to any word on the target side,

4. last source word unaligned, similarily the last word of the source phrase is not aligned to any word on the target side,

5. first source word unaligned, and

6. last source word unaligned, similar to 3. and 4. with respect to the target phrase.

The motivation for removing biphrases that occur once is to avoid overfitting the training data by leaving out biphrases that are specific to that corpus and are unlikely to occur anywhere else. Having only biphrases whose first and last word are aligned simplifies the learning of feature weights [22]. Biphrases farther than the twentieth are likely to be assigned too low probabilities compared to the most frequent ones to be actually used in translations.

3.2.2 Using MMBT

MMBT is an application of multiview learning to machine translation. It predicts an ouput label associated to some given input features. A detailed explanation of the method can be found in [30].

Machine Translation deals with how to arrange words into sentences. Therefore, the inputs are features associated to words, both from the target language and source language. They are similarity measures, representing how closely two words are related to each other, based on how often they occur in the training data at neighboring positions. To a word w is associated a vector

$\phi_{S_s,S_t}(w)$ representing how closely w relates to the words in the source sentence S_s and in the target sentence S_t.

Consider a sentence S and a set of phrases $P_1, P_2, \ldots, P_n$ that cover it. The label to predict for a word w is a binary vector indicating which phrases w belongs to $\psi(w)_S = (w \in P_1, w \in P_2, \ldots w \in P_n)$.

Once these representations have been fixed, the learning problem is can be defined as follows:

$$\min \frac{1}{2} \parallel W \parallel^2_{Frobenius} + C \sum_{k=1}^{n_{S_s}} \xi_k, \tag{3.1}$$

$$\begin{gathered} \text{w.r.t. } W \text{ linear operator, } \xi \text{ loss,} \\ \text{s.t. } \langle \psi_{S_s}(w_k), W\phi_{S_s,S_t}(w_k) \rangle \geq 1 - \xi_k,\ w_k \in S_s\ , \\ \xi \geq 0 \text{ and } C > 0 \text{ penalty constant.} \end{gathered}$$

When W has been computed, here assuming that the source words are the training set, the label for a target word w_l is $W\phi_{S_s,S_t}(w_l)$. This vector does not contain boolean indicators but real values. The strength of the relation between a source word w_k and a target word w_l can then be computed as $R(w_k, w_l) = \langle \psi_{S_s}(w_k), W\phi_{S_s,S_t}(w_l) \rangle$.

Two such closeness matrices are computed, considering the source words as the training set and computing the labels for the target words as well as in the other direction and summed. The result is a $n_{S_s} \times n_{S_t}$ matrix, D, where n_{S_s} and n_{S_t} are the number of words in the source and target sentences respectively. $D(i,j)$ measures how closely the i^{th} word of the source sentence relates to the j^{th} word of the target sentence.

The next step is to extract biphrases from this matrix. Consider a source phrase of length k starting at position i of S_t, i.e., some adjacent lines of D with indices $[i, i+k-1]$. The aim of the phrase extraction process is to find the target phrase that gives the best match, i.e., the set of adjacent columns of D with indices $[j, j+k'-1]$ such that the similarity is maximized in the window defined by the indices and minimized outside.

The extraction is done using the following heuristic. Only values that are row or column maxima are kept. For a source phrase, all target words that are aligned to some source word are collected. Words whose score is below a certain threshold are removed and the remaining ones sorted in the order of the target sentence. The obtained biphrase is stored in the phrase table if it does not contain more than a one word gap and its target phrase length does not differ from that of the source phrase by more than one word.

3.3 Related work

3.3.1 Phrase alignment and extraction

Extracting aligned biphrases from the symmetrized word alignments generated using the IBM models, as presented in 2.2.1, is by far the most widely used technique to generate the phrase table. Alternative methods have been proposed to directly generate the phrase level alignment using statistical models [26], with a machine learning approach as in `MMBT` (described in 3.2.2) or using integer linear programming [11]. The model proposed in [25] refines the IBM models by adding agreement constraints between the alignments in direct and reverse direction, leading to improved final word alignments.

Starting from the word alignments generated by the IBM models in both directions, various symmetrization and extraction heuristics can be applied. Different criteria have been studied to discard some biphrases from the phrase table, based on usage statistics [13] or significance tests [21]. The latter reported no decrease in *BLEU* score while removing up to 90% of the biphrases. The use a Gibbs sampler initializied with the IBM word alignments to estimate biphrases frequencies is proposed in [10]. These weights were shown to allow for a better use of the phrases.

A filtering technique using triangulation with a bridge language is presented in [9]. Given an original phrase table $P_{X,Y}$ to be pruned, between languages X and Y and the two phrase tables $P_{X,Z}$ between X and the bridge language Z and $P_{Z,Y}$ between Z and Y, for a candidate biphrase

(x, y) from the original phrase table to be kept, there must be a phrase z in the third language such that $(x, z) \in P_{X,Z}$ and $(z, y) \in P_{Z,Y}$. This approach is claimed to yield increased *BLEU* score up to 2.3 points, depending on the language used as a bridge. Nevertheless, applying this technique requires to obtain the bridge phrase tables $P_{X,Z}$ and $P_{Z,Y}$, meaning that aligned corpora in the language pairs (X, Z) and (Z, Y) have to be at hand and two additional alignment processes run.

3.3.2 Alignment quality

Precision, *Recall* and *Alignement Error Rate (AER)* are generally used to evaluate the quality of word alignments. A reference alignment is required to use these metrics. Reference alignments are definied manually and contain two kinds of links, sure (S) and probable (P), with $P \subseteq S$, to which the links (A) of the studied alignment are to be compared. The metrics are then defined as follows:

$$Precision = \frac{\mid A \cap P \mid}{\mid A \mid},$$

$$Recall = \frac{\mid A \cap S \mid}{\mid S \mid},$$

$$AER = \frac{\mid A \cap S \mid + \mid A \cap S \mid}{\mid A \mid + \mid S \mid}.$$

The impact of word alignment quality on the final translation has been studied [35, 5, 15], showing that improved *AER* does not necessarily leads to better translations, in terms of *BLEU* score in particular. The results presented in [35] suggest that the usage of the biphrases by the decoder should be taken into account when tuning the alignments.

A thorough analysis of the search space of *phrase-based* systems can be found in [4]. In that work, *phrase-based* systems are compared to *hierachical phrase-based* systems, studying the reachablity of a set of translations and analysing symptomatic errors.

Chapter 4

Phrase table experiments

4.1 Material

The work we report here was made using a phrase table obtained with `GIZA++` to generate the alignment on one side and the `MMBT` alignment algorithm on the other. The phrase tables have been obtained using the *Europarl* training data, from French to English, made available for the translation task of the fourth European Chapter of the Association for Computational Linguistics (EACL) Workshop on Machine Translation (WMT09) [23, 12]. `europarl-v4.fr-en` was tokenized, lowercased and long sentences (over 40 words) filtered out before being used as the training data. We used two sets of 2000 tokenized and lowercased sentences, `dev2006` and `test2007`, as tuning and evaluation data respectively. Statistics of the three datasets are reported in Table 4.1.

Typically, the phrase tables we consider come as ordinary text files containing one line per biphrase, each line having four fields:

source phrase, a short sequence of words as it appears in the source corpus,

target phrase, a short sequence of words as it appears in the target corpus,

alignment, the mapping between the words of the source phrase and the target phrase, and

features, counts, translation probability, etc.

The features assigned to each biphrase depend on the scoring procedure. In the next section, we will present in more detail how the phrase tables are generated from the parallel corpora, what features are associated with them for each of the three different systems studied here, and how we could use them to discriminate the biphrases.

All three translation systems were used as of March 2009. The source code of `Moses` translation system can be downloaded from [1]. A step by step guide from installation to translation and a detailed manual can be found on the website of the WMT09 [12] where it is used as a baseline. The source code of `Sinuhe` translation system can be downloaded from [2] and contains installation instructions. The source code of `MMBT` is not available online but can be obtained from its author.

4.2 Phrase table features

After extracting the biphrases from the aligned corpus, numerical features are associated to them which can be used for a preliminary pruning or in the training phase and decoding. In this section we will present in more detail the different features generated by the three scoring procedures. Two are applied to the `GIZA++` alignments, in `Moses` and `Sinuhe` respectively and one in the `MMBT` system.

	europarl-v4.fr-en		dev2006		test2007	
	French	English	French	English	French	English
sentences	1050377		2000		2000	
words	23812195	21617161	64331	58762	64339	59156

Table 4.1: Statistics of the data

4.2.1 Moses features

`Moses` phrase table contains five features per biphrase ($mos_i, i \in [1, 5]$):

1. the phrase translation probability $\varphi(f|e)$,
2. the lexical weighting $lex(f|e)$,
3. the phrase inverse translation probability $\varphi(e|f)$,
4. the inverse lexical weighting $lex(e|f)$,
5. the phrase penalty, currently always $e = 2.718$.

The first four features are probabilites, so they take values between zero and one. The fifth one, being constant is not used in our experiments. These features are directly used by the decoder.

4.2.2 Sinuhe features

`Sinuhe` phrase table also contains five features per biphrase ($sin_i, i \in [1, 5]$):

1. the number of occurrences of the triple (source phrase, target phrase, alignment) in the training data,
2. the number of occurrences of the source phrase,
3. the number of occurrences of the target phrase,
4. the rank of the pair (source phrase, target phrase) among all such pairs sharing the same source phrase,
5. the rank of the pair (source phrase, target phrase) among all such pairs sharing the same target phrase.

These features are not used directly by the decoder but for filtering the phrase table prior to learning and determing the regularization during the weights estimation.

Note that we have the following correspondence between `Moses`'s features and `Sinuhe`'s, from the definition of the translation probabilities:

$$mos_1 = \frac{sin_1}{sin_2} \quad \text{and} \quad mos_3 = \frac{sin_1}{sin_3}. \tag{4.1}$$

`Sinuhe`'s features, as they are not normalized, are not as easy to handle as `Moses`'s probabilities but allow for finer distinctions. Let us take some biphrases and consider the scores they are given in `Moses` and `Sinuhe` phrase table respectively to illustrate this point. The two lines below are extracted from `Moses` phrase table, they have very similar scores.

```
absence de mme ||| absence of mrs ||| (0) (1) (2) ||| (0) (1) (2)
     ||| 1 0.239303 1 0.0623942 2.718
abstention exprime ||| abstention expresses ||| (0) (1) ||| (0) (1)
     ||| 1 0.293781 1 0.0659091 2.718
```

Here are the two same biphrases, extracted from `Sinuhe`'s phrase table.

```
absence de mme ||| absence of mrs ||| 0-0 1-1 2-2
     ||| 5 5 5 1 1
abstention exprime ||| abstention expresses ||| 0-0 1-1
     ||| 1 1 1 1 1
```

Both biphrases are scored $mos_1 = mos_3 = 1$, because the source and target phrases always co-occur in the training data, but the first biphrase occurs five times, the second only once. As a consequence while the two will be handle in the same way in `Moses`, with `Sinuhe` the first will be kept but the second pruned out.

Another example follows, again the first group of biphrases are extracted from `Moses`'s phrase table and second one from `Sinuhe`. Here all biphrases have scores $mos_1 = 0.25$ and $mos_3 = 0.5$ since the pair (source phrase, target phrase) occurs with every other occurrence of the source phrase but only every fourth occurrence of the target phrase. Nevertheless, the first biphrase occurs five times in the training data while the last one occurs only once.

```
abolit ||| abolishes ||| ...
     ||| 0.25 0.263158 0.5 0.25 2.718
abandonnera ||| will abandon ||| ...
     ||| 0.25 0.00621805 0.5 0.078478 2.718
adopte par le conseil et ||| adopted by the council and ||| ...
     ||| 0.25 0.00637363 0.5 0.0748383 2.718
affirmation que l' ||| insistence that the ||| ...
     ||| 0.25 0.000199703 0.5 0.000491218 2.718
```

```
abolit ||| abolishes ||| ...
     ||| 5 10 20 1 2
abandonnera ||| will abandon ||| ...
     ||| 3 6 12 1 1
adopte par le conseil et ||| adopted by the council and ||| ...
     ||| 2 4 8 1 1
affirmation que l' ||| insistence that the ||| ...
     ||| 1 2 4 1 1
```

The lexical probabilities mos_2 and mos_3 are the product of the lexical translation probabilities $lex(w_f|w_e)$ and $lex(w_e|w_f)$ respectively over aligned word-pairs (w_f, w_e) in the biphrase. Thus they are linked to the number of occurrences of the individual words of the biphrase in the training data and to the length of the biphrase, but they do not give a direct indication of the number of occurrences of the biphrase as a whole.

4.2.3 `MMBT` features

`MMBT` phrase table only contains two features per biphrase ($mmbt_i, i \in [1, 2]$):

1. the number of occurrences of the triple (source phrase, target phrase, alignment) in the training data,
2. the sum of the margins computed for each occurrence.

Figure 4.1 is a plot of the features of the `MMBT` phrase table, second feature as a function of the first feature, for the biphrases verifying $mmbt_1 < 5000$ (includes over 99% of the points).

The points seem to lies along two different lines. Using biphrases such that $1000 < mmbt_1 < 5000$ to obtain a more reliable estimation from biphrases occurring multiple times, the linear regressions obtained for the two clusters have coefficients close to $a_1 = 2.886$ and $a_2 = 2.305$ respectively. Yet, the lower cluster is more subject to noise. The two affine functions $mmbt_2 = a_1 mmbt_1$ and $mmbt_2 = a_2 mmbt_1$ are displayed on Figure 4.1 in red and green respectively.

As the second feature accumulate the margins when the biphrase occurs multiple times, the two features are clearly linearly dependent. However the existence of two biphrases clusters based on the value of the quotient $\frac{mmbt_2}{mmbt_1}$ should be considered to check whether they have different properties.

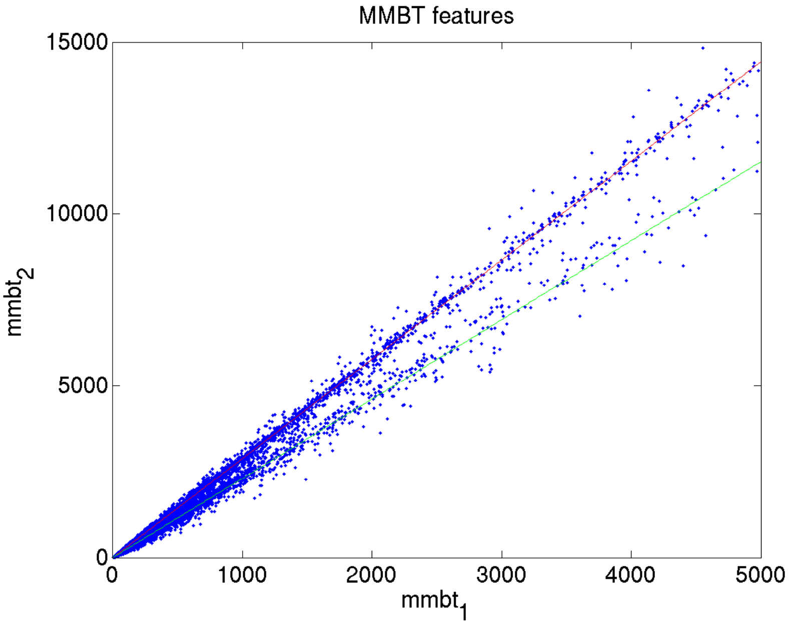


Figure 4.1: MMBT features, mmbt2 vs. mmbt1

phrase table	# biphrases	size
Moses	46058264	5.9 GB
MMBT	1936768	138.5 MB

Table 4.2: Sizes of the phrase tables

4.3 Further phrase table characteristics

In this section we present some characteristics of the phrase tables, trying to identify some that could be used along with their features to discriminate between good and bad biphrases.

4.3.1 Size of the phrase tables

Trained on the same data, the phrase tables obtained using GIZA++ or MMBT have very different sizes.

The phrase tables obtained using GIZA++ are about 23 times larger than those obtained with MMBT with respect to the number of biphrases but over 40 times larger with respect to the size of the file on disk as one can see from Table 4.2.

The distribution of biphrases with respect to their source length and target length as a ratio of the whole set of biphrases is shown on Figure 4.2. The GIZA++ algorithm allowed to retrieve phrases up to length 7 on both sides while MMBT was limited to 6. But regardless of that limit, MMBT concentrates much more on shorter biphrases and allows for much less distortion between the lengths on the source and target sides. This observation and the fact that the MMBT phrase table contains only two features per biphrase while Moses' contains five features probably explains the difference in disk size.

4.3.2 GIZA++ characteristics

Since Moses and Sinuhe both rely on GIZA++ alignment and the same symmetrization heuristic, their characteristics are the same, only the scoring methods vary. Therefore, the results presented in this section are valid for both of them.

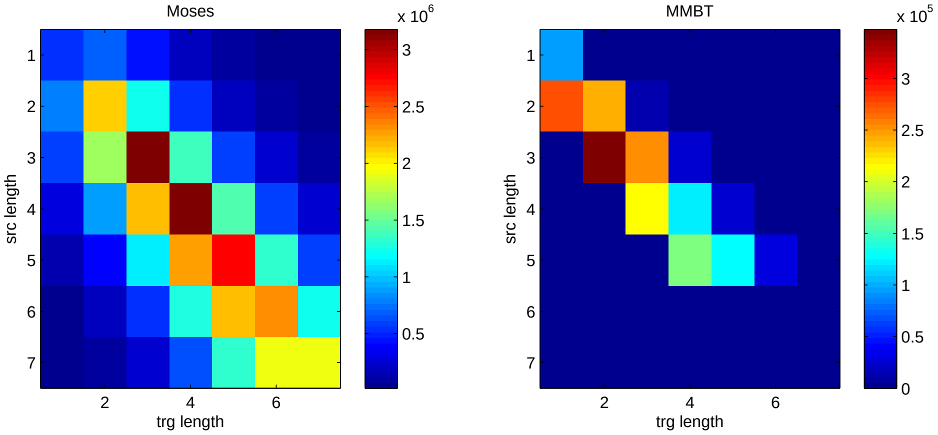


Figure 4.2: Distribution of the biphrases depending on the source length and target length

Pruning criteria

We look at the repartition of the biphrases to the groups made up by the six criteria used by `Sinuhe` to prune the phrase table. Using the phrase table generated by `GIZA++` from the whole *Europarl* training data and scored by `Sinuhe`, we extracted every fiftieth biphrase (the whole phrase table is too large to be handled in *Matlab*).

Figure 4.3 shows the number of biphrases contained by the following different groups:

full all extracted biphrases,

single biphrases occurring only once,

multiple biphrases occurring multiple times,

src non aligned biphrases whose first source word, last source word or both are not aligned,

trg non aligned biphrases whose first target word, last target word or both are not aligned,

rank over 20 biphrases that are ranked lower than twenty with regard to the count of occurrences for the same source phrase, and

kept biphrases that are retained in the phrase table after pruning.

Of course, there are overlaps between some of these groups. For example, a biphrase may occur once and have its first source word unaligned. Groups of biphrases that are left out by the pruning process are represented in red/pink while blue/cyan are used for biphrases that are retained in the phrase table.

Over 93% of the biphrases occur only once, about 20% have deficient alignment on their source side and about 16% on their target side. After pruning, only about 4.6% of the biphrases are retained.

The proportion of well-aligned biphrases is significantly higher among multiple-occurring biphrases than among single-occurring ones (76% in the former group, 67% in the latter).

Figure 4.4 and Figure 4.5 show for each of the different groups the distribution of biphrases according to their source or target phrase length and length distortion (i.e., length of the source phrase minus length of the target phrase), respectively.

Phrases lengths

The distributions of biphrases according to their source length or target length in the full phrase table are alike, since the extraction process is symmetrical and does not depend on the direction.

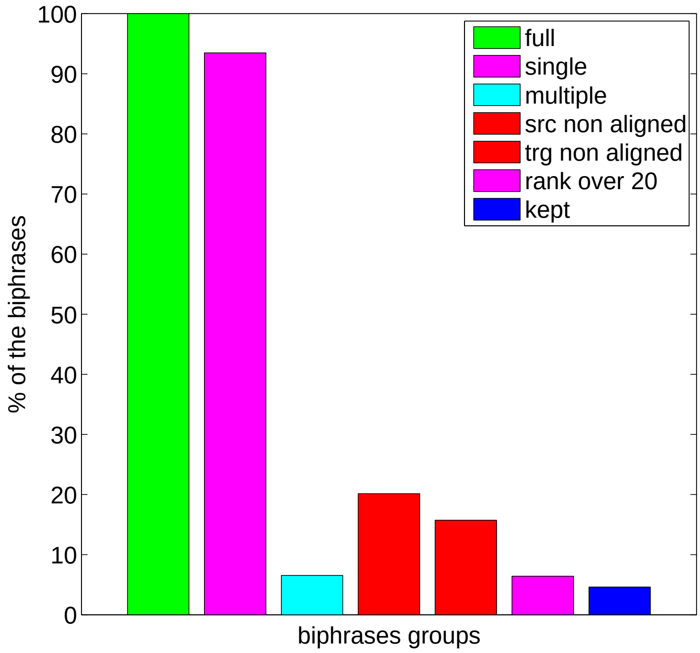


Figure 4.3: Percentage of `Sinuhe` biphrases in each group

The lengths are very similar to the distribution of n-grams depending on their length n that can be observed in *N-gram language models.*

The number of different possible n-grams formed using words taken from a dictionary containing K words grows exponentially with respect to n since there are K times as many possible combinations of n words as there are combinations of $n-1$ words. Nevertheless, k_n, the number of different n-grams actually observed in a corpus does not follow this exponential growth because of syntactic and semantic limitations on the possible combinations in natural languages. This property is the basis of *N-gram language models.* For $n > 4$, k_n even decreases. Indeed, beyond four words, there are fewer and fewer distinct phrases that occur in the data because of the data sparsity and sentence boundaries.

The distribution of source length for the biphrases with deficient source alignment is approximatetly the same as for the whole phrase table apart that there are no biphrases of length one for which that single word is not aligned. This would yield an empty alignment and this is not allowed during the phrase extraction. The same pattern is repeated on the target side.

Length distortion

The distribution of biphrases depending on the difference between the length of their source phrase and the length of their target phrase is centered on zero with variance around 1.99. This means, as we would expect, that most of the biphrases have equal length on source and target sides. The variance is quite high, some of the biphrases that were retrieved even have for example a source phrase of length one while the target phrase contains the maximum allowed seven words, yielding in this case a negative distortion of 6 words.

Many asymmetric biphrases are obtained by gluing an unaligned word on one end of the phrases, creating an extension with a deficient alignement. Unaligned words preceeding or following well-aligned symmetric biphrases are attached to them, producing asymmetric extensions. For that reason, the distribution of biphrases with unaligned end word on source side and on target side are very similar to the distribution of the complete phrase table with a one word shift toward

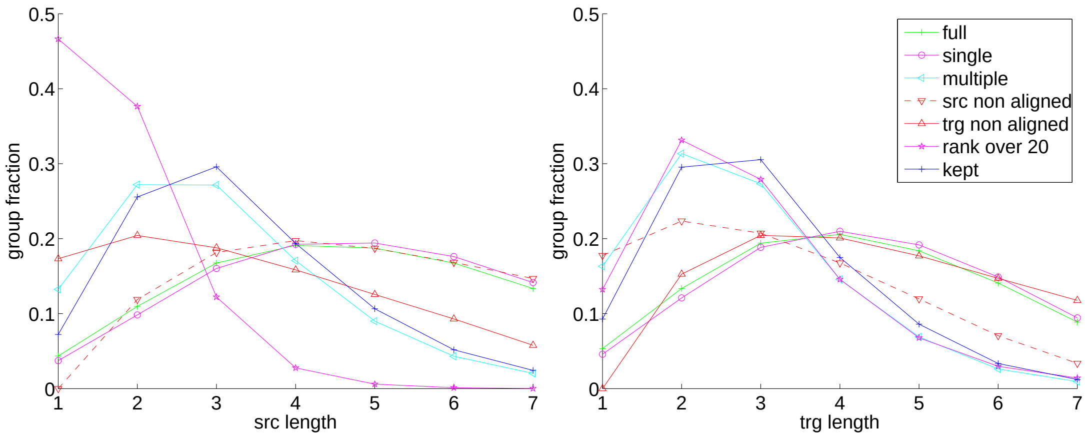


Figure 4.4: Distribution of `Sinuhe` biphrases depending on their length

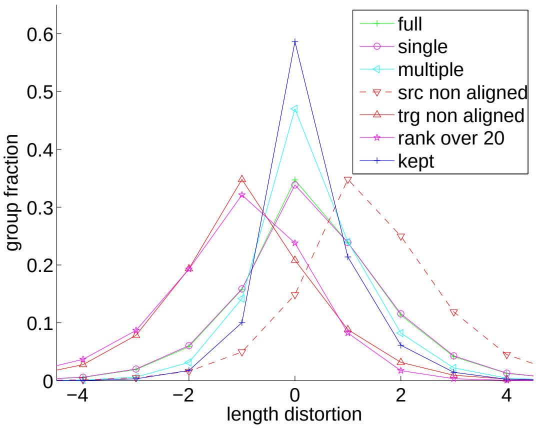


Figure 4.5: Distribution of `Sinuhe` biphrases depending on their length distortion

source length or target length respectively. Removing biphrases with deficient alignment (33% of the biphrases) reduces the variance in distortion to about 1.29.

Keeping only multiple-occurring biphrases favors shorter phrases, reducing the potential for distortion. As a consequence this further lowers the variance in distortion to 0.78. An asymmetry toward longer biphrases on the source side remains. It is probably due to the fact that French is more wordy than English and French texts are generally longer than their English translation by a few words.

Lox rank biphrases

The last pruning criteria discards biphrases that are ranked lower than twenty among the candidates for a given source phrase. Source phrases that have more than twenty candidates must be very frequent, so as to be extracted along with more than twenty different target phrases. These different target phrases can be generated by different alignments in various sentences or as extensions, by gluing unaligned words on one or both sides of the target phrases. For example, the french phrase *affaire* occurs 3243 times in the training data. It has been extracted with 514 different target phrases. Among them are 169 single words, 34 well-aligned 2-grams and 2 well-aligned 3-grams, the remaining 309 candidates are extensions with deficient alignment.

# candidates	# source phrases
1	1227906
2	118914
3	51493
4	26725
5	41931

Table 4.3: Number of candidate biphrases per source phrase for MMBT

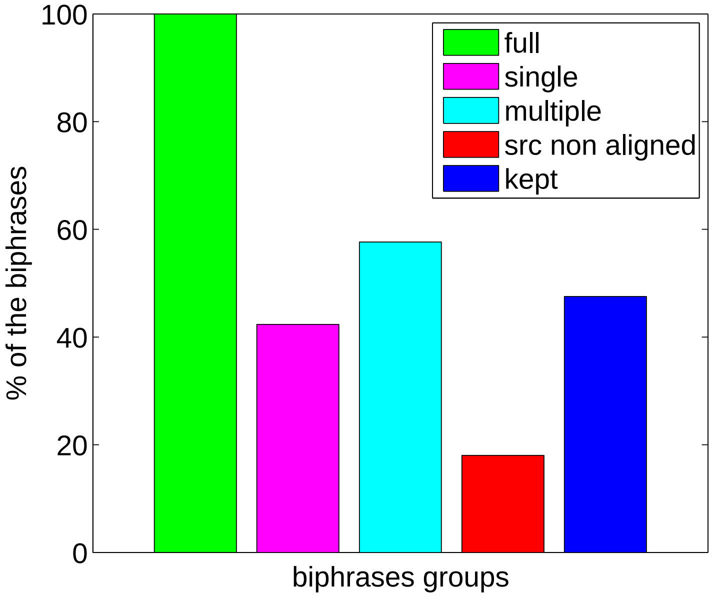


Figure 4.6: Percentage of MMBT biphrases in each group

4.3.3 MMBT characteristics

Pruning criteria

To compare the characteristics of the two different phrase tables, we studied the repartition of MMBT biphrases to the goups made up by Sinuhe pruning criteria. Note that as a consequence of the biphrase extraction process there are no biphrases with deficient alignment on the target side, since a target word has to be linked to some word of the candidate source phrase to be retrieved in the candidate target phrase. Therefore, deficient alignments can be found only on the source side.

In the phrase table generated using MMBT there were at most five different biphrases for one given source phrase. 86% of the source phrases have only one candidate biphrase, as shown in Table 4.3. The pruning criterion based on rank is therefore not applicable here.

Thus, only five groups are taken into consideration: *full*, *single*, *multiple*, *src non aligned* and *kept*. As for Sinuhe's biphrases, Figure 4.7 and Figure 4.8 show for each of the different groups the distribution of the MMBT's biphrases according to their source or target phrase length and length distortion (i.e., length of the source phrase minus length of the target phrase) respectively.

Length distortion

As can be seen from Figure 4.2, the alignment obtained with MMBT is strongly biased toward asymmetric biphrases, compared to the one obtained with GIZA++. It retrieves a much larger proportion of biphrases whose source phrase is longer than the corresponding target phrase by one

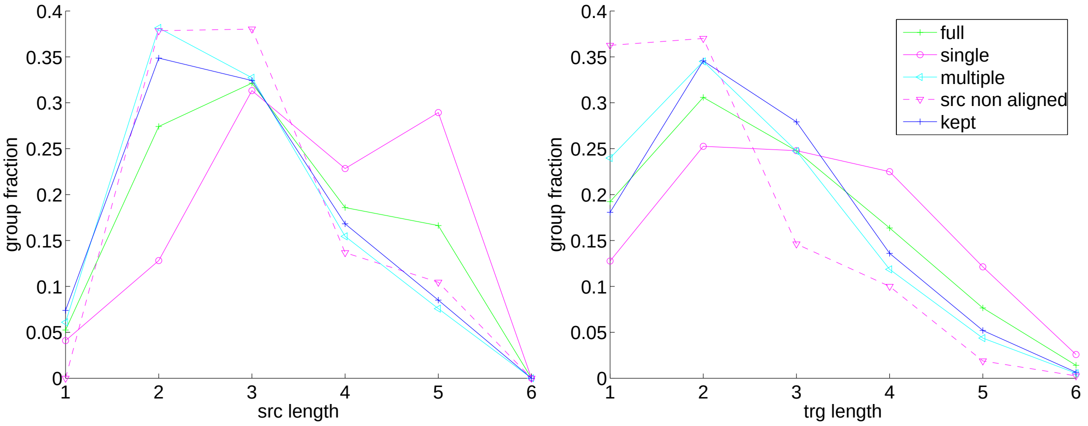


Figure 4.7: Distribution of MMBT biphrases depending on their length

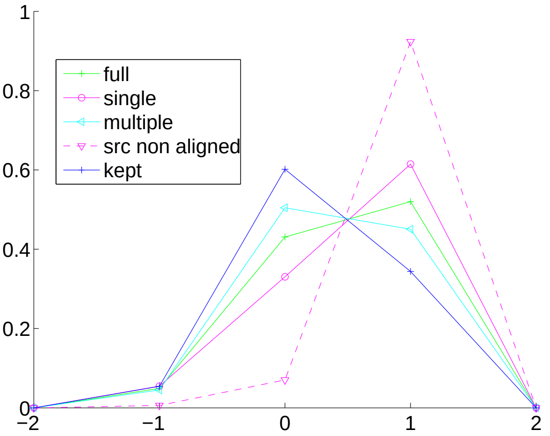


Figure 4.8: Distribution of MMBT biphrases depending on their length distortion

word.

In fact, the bias toward positive length distortion of the alignment algorithm may even be stronger, producing a large number of biphrases whose source phrase is longer than the corresponding target phrase by two or more words. Since only one word distortion is allowed during the extraction such biphrases are filtered out and we cannot assert whether this phenomena really occurs or not.

We also noticed that a gaussian kernel, instead of the default polynomial kernel configuration, produced very symmetrical alignments. However this was only tested for a small corpus and we cannot be certain that this result would generalize to a larger corpus as we were not able to run it on the larger corpus for computational reasons.

Almost all biphrases with unaligned words on the source side are asymmetric, problably produced by extending some aligned biphrase by gluing an unaligned word to its end or beginning.

Applying pruning rules to the phrase table allows to obtain characteristics that are more in line with what we would expect, in terms of symmetry in particular. This is possible only at the expense of a reduction of the phrase table size. Modifying the extraction algorithm and maybe the alignment algorithm might allow to correct this bias and help to further enhance the quality of the phrase table while keeping it at a reasonable size.

By default, a biphrase must occur at least once if the source phrase contains only one word, four times if it contains between two and four words and twice if it contains five or six words. This criteria is responsible for the irregularities in the distribution of source lengths, which is also closely linked to the distribution of target lengths.

4.4 Phrase table coverage

Our aim with this experiment is to compare how well a test set is covered by the biphrases of the different phrase tables, not making any assumptions on subsequent components such as the decoder or the language model, the capability of the system to handle reorderings, etc. The procedure we followed is explained in more details in the next section before presenting the results we obtained.

4.4.1 Experiment description

Metrics

For this test, we propose to retrieve the biphrases whose source phrase match n-grams in the source test sentence using some feature of the biphrase or a combination of features to threshold the phrase table. We analyse how the bag of words obtained from the biphrases' words on target side covers the target test sentence using metrics similar to the common precision and recall:

$$P = \frac{inter}{sumT} \quad \text{and} \quad R = \frac{inter}{sumR} \quad , \tag{4.2}$$

where

inter is the number of words common to the target sentence and the aggregate bag,

sumT is the number of words contained in the aggregate bag (test), and

sumR is the number of words contained in the target sentence seen as a bag of words (reference).

R quantifies how well the test sentences were covered and can be assimilated to a measure of recall, while P quantifies how large a bag of word has been retrieved from the phrase table and can be assimilated to a precision measure. These measures can also be computed so that only the presence of the words, not their number of occurrences, is taken into account.

To evaluate the coverage at corpus level one can either

1. use *micro-averaging*, denoted with subscript *mic*, i.e., calculate the total *inter*, *sumT* and *sumR* for all sentences then compute P and R, or
2. use *macro-averaging*, denoted with subscript *mac*, i.e., compute P and R for each sentence and then average over all sentences.

Compared with *micro-average*, *macro-average* puts more emphasize on the shortest sentences. Nevertheless, the difference between the two measures is only noticeable for the first thresholds and vanishes as the size of the bags of words grows. Later in this report we use *micro-average*, unless otherwise stated, as it proved to be more stable for the first thresholds.

Outline

To carry out our experiment we follow the outline described below.

- Run the algorithm to obtain a scored phrase table (`GIZA++` and `Moses` or `Sinuhe` scoring and pruning / `MMBT`).
- Associate to each biphrase a unique score using some function of the features found in the original phrase table. For that scoring function choose a set of K thresholds, defining bins in which to categorize the biphrases depending on their score, such that they give some nice partition of the biphrases. With `Moses`, the biphrases are scored using $mos_1.mos_3$, the product of the first feature (direct translation probability) and the third feature (reverse translation probability) of the phrase table (cf.4.2.1). With `Sinuhe` we used the equivalent

score formula $\frac{sin_1}{sin_2.sin_3}$, to obtain comparable results (cf. 4.2.2). For `MMBT`, the score is $\frac{mmbt_2}{mmbt_1}$, the quotient of the second feature by the first feature (count) of the phrase table, i.e., the average margin (cf. 4.2.3).

- Choose a test dataset, an aligned corpus of J sentences.
- For each of the sentences of the source corpus look for all possible n-grams that appear in the phrase table on the source side, with n varying from 1 to a chosen length N, generally the maximum length of the source phrases in the studied phrase table. Split the corresponding target phrase into words. For each score bin, construct a bag containing all target words obtained from phrases whose score fall in that particular bin. There is no order between the words in the bag, but each of the words is associated to its count of occurrences.
- Progressively aggregate the bags from bins with increasing or decreasing thresholds and evaluate how the target sentence is covered at each step using P and R.

More formal definition

Our framework is defined by the following parameters:

- Different algorithms to generate phrase tables are compared: L phrase tables $(1 \dots l)$,
- A set of thresholds $(t_1, t_2, \dots t_k, \dots)$ defines bins, such that the bin k contains biphrases whose scores s are such that $t_k - 1 < s <= t_k$: K bins $(1 \dots k)$,
- An aligned test corpus: J test sentences $(1 \dots j)$,
- All words that appear on the target side, either in the phrase table or test dataset makes up a dictionary containing I words indexed from 1 to i: I words $(1 \dots i)$.

This allows use to define:

- $W_t(i, j, k, l)$, non-negative integer, as the number of times the word i is predicted for the sentence j, using the bin k of the phrase table l.
- $W_r(i, j)$, non-negative integer, as the number of times the word i appears in the target sentence j.
- $Z_t(i, j, k, l)$, boolean, indicating whether the word i is predicted for the sentence j, using the bin k of the phrase table l.
- $Z_r(i, j)$, boolean, indicating whether the word i appears in the target sentence j.

$$inter(j,k,l) = \sum_I \min(W_t(i,j,k,l), W_r(i,j)), \tag{4.3}$$

$$sumT(j,k,l) = \sum_I W_t(i,j,k,l), \tag{4.4}$$

$$sumR(j) = \sum_I W_r(i,j), \tag{4.5}$$

$$P_{mic}(k,l) = \frac{\sum_J inter(j,k,l)}{\sum_J sumT(j,k,l)}, \tag{4.6}$$

$$P_{mac}(k,l) = \frac{1}{J} \sum_J \frac{inter(j,k,l)}{sumT(j,k,l)}, \tag{4.7}$$

$$R_{mic}(k,l) = \frac{\sum_J inter(j,k,l)}{\sum_J sumR(j,k,l)}. \tag{4.8}$$

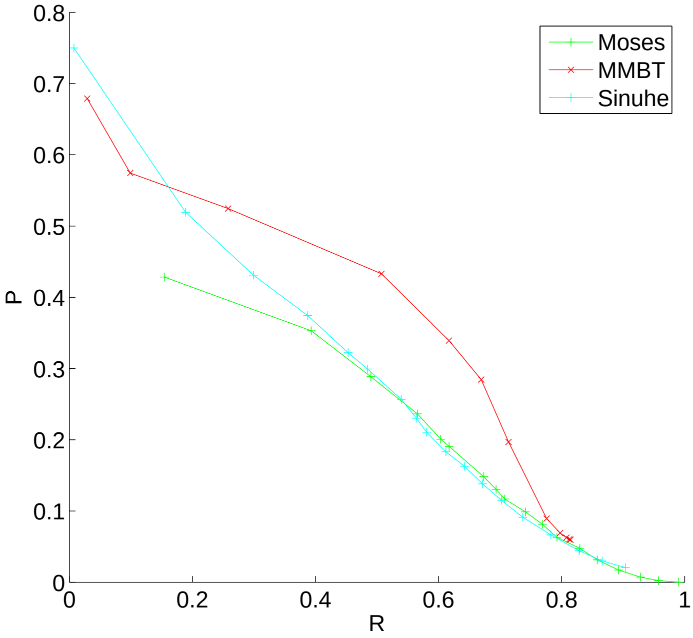


Figure 4.9: `MMBT`, `Moses` and `Sinuhe` phrase tables coverage comparison over 100 test sentences, including biphrases with decreasing scores, P vs. R

$$R_{mac}(k,l) = \frac{1}{J}\sum_{J}\frac{inter(j,k,l)}{sumR(j,k,l)}, \tag{4.9}$$

One can similarily define $interU$, $sumTU$, $sumRU$, PU and RU, by leaving out the number of occurrences and using the boolean indicators (0/1) instead of counts:

$$interU(j,k,l) = \sum_{I}(Z_t(i,j,k,l) \text{ AND } Z_r(i,j)), \tag{4.10}$$

$$sumTU(j,k,l) = \sum_{I} Z_t(i,j,k,l), \tag{4.11}$$

and so on.

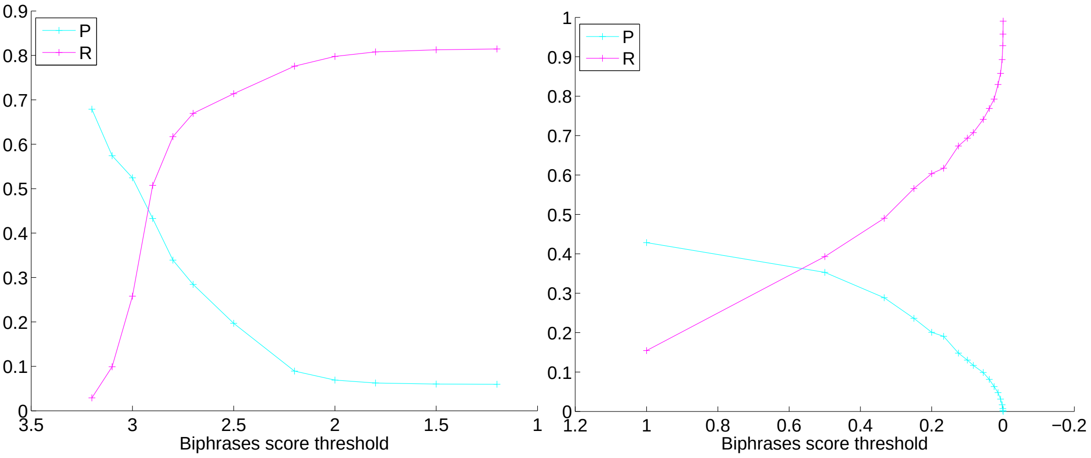
4.4.2 Results

`MMBT` and `Moses`

Figure 4.9 is a plot of R as a function of P for `MMBT`, `Moses` and `Sinuhe`.

We note that for the same value of R `Moses` generally has lower P than `MMBT`. This means that to cover the same amount of words in the test sentences, more words have been retrieved from `Moses`'s phrase table, leading to a larger raw search space to construct the translation from. In this respect, we can say that `Moses`'s phrase table contains more alternatives than `MMBT`'s. Among these alternatives, some are probably clearly wrong candidates but others can be correct translations that would have been used in other contexts. The prediction of the words is in that sense less deterministic with `Moses`.

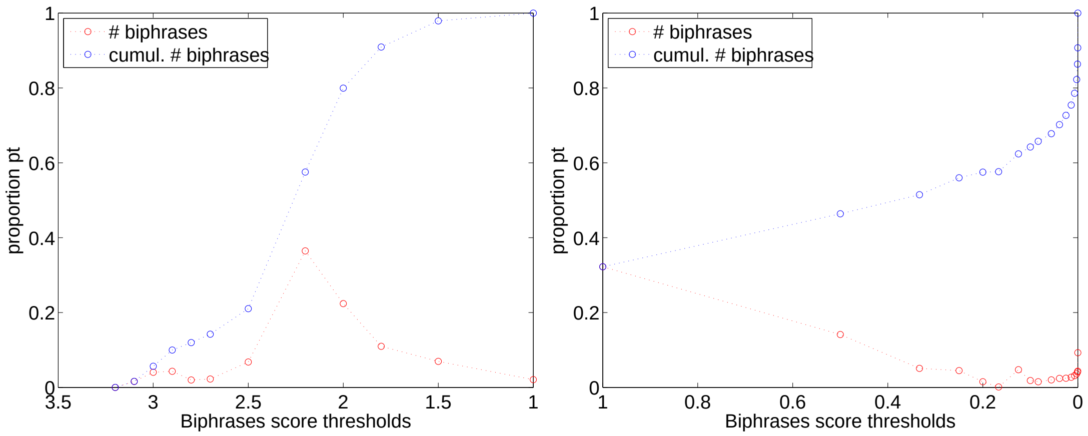
When the full phrase table is taken into account (right end of the curves) `Moses` reaches a much better coverage, close to full coverage ($R = 0.99$) while with `MMBT` one fifth of the sentences remains uncovered ($R = 0.81$). The value of P when the full phrase table is taken into account is 6×10^{-2} for `MMBT` and 2.8×10^{-5} for `Moses`.



(a) `MMBT` (score=mmbt2/mmbt1)

(b) `Moses` (score=mos1.mos3)

Figure 4.10: `MMBT` and `Moses` phrase tables coverage comparison over 100 test sentences, including biphrases with decreasing scores, P and R vs. thresholds



(a) `MMBT` (score=mmbt2/mmbt1)

(b) `Moses` (score=mos1.mos3)

Figure 4.11: `MMBT` and `Moses` biphrases distribution depending on their scores

Coverage and biphrases distribution

The evolution of R and P when more and more biphrases are included, aggregating words from phrases with decreasing scores, for `MMBT` and `Moses`, is shown on Figure 4.10a and Figure 4.10b respectively.

We can see that the increase of R is directly linked to the decrease of P. Increasing the coverage (as measured by R) is obtained by lowering the threshold to take in more biphrases, therefore building a larger bag of words, loosing precision (as measured by P). The gain in coverage always remains proportional to the loss in precision.

The slope of the curve is mainly caused by the size of the portions of the phrase table that are aggregated when lowering the threshold on the score. We tried to use a score and thresholds set in order to cut the phrase table in equally sized portions to cut out this effect, but some large portions of the phrase table may have the same score and thus cannot be discriminated. The best example is the set of `Moses`'s biphrases scoring 1 which represent about a third of that phrase table.

With `Moses` this effect is particularly strong, the cumulative distribution of the biphrases de-

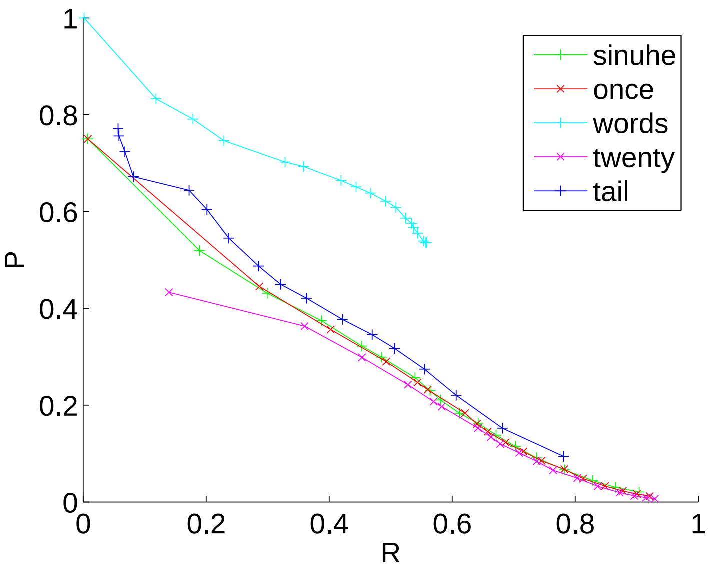


Figure 4.12: `Sinuhe` phrase table coverage comparison over 100 test sentences, including biphrases with decreasing scores, pruning using various methods, P vs. R

pending on their score shown on Figure 4.11a is very similar to the R curve of Figure 4.10b. Different scoring methods and threshold sets have been tried but we could not identify any subset of biphrases for which R and P were atypically related.

With `MMBT`, the R curve has a sharp increase for scores between 3 and 2.5, while the cumulative distribution of the biphrases (Figure 4.11a) does not show an equally sharp growth in the total number of biphrases between thoses scores but a rather small bump corresponding to the cluster with slope $a_1 = 2.886$ mentioned in 4.2.3. Among a rather small number of biphrases from that cluster, many have been retrieved and allowed to cover a significant part of the test sentences. This might indicate that this cluster contains biphrases of better quality.

Phrase table pruning

Next we look at how pruning the phrase table according to `Sinuhe` criteria affects the coverage. We used the score formula $\frac{sin_1}{sin_2.sin_3}$. It corresponds directly to $mos_1.mos_3$, the formula used in the previous section, allowing us to compare the results obtained with `Moses`'s phrase table and those obtained with `Sinuhe`'s.

Figure 4.12 is a plot of R as a function of P for `Sinuhe`'s phrase table. The curve denoted *sinuhe* was obtained after applying the pruning used in `Sinuhe` described in 3.2.1, it was the same as the corresponding curve in Figure 4.9. The curve denoted *once* was obtained when applying the pruning used in `Sinuhe` but relaxing the pruning criterion on single co-occurrence of the pair (source phrase, target phrase) to a single occurrence of both source phrase and target phrase separately. Using only biphrases of length one on both sides restricting to the highest ranking candidate for each source phrase, i.e., using only word to word translation with the first candidate, we obtained the curve denoted as *words*. The curve denoted *twenty* was generated by removing only candidates ranked lower than twenty and the one denoted *tail* with multiple occurring biphrases, cutting the tail of candidates in a somewhat more elaborate way that we will detail later.

The values obtained for P and R for the different pruning methods when including the whole pruned phrase table (this corresponds to the rightmost point of each curve) are reported in Table 4.4.

The fact that using word to word matching with the best candidate (*words*) only allows to cover about half of the test data is an argument for using phrases with multiple candidates. Relaxing

pruning	P	R
sinuhe	0.0207	0.9038
once	0.0120	0.9210
words	0.5358	0.5583
twenty	0.0065	0.9294
tail	0.0941	0.7813

Table 4.4: P and R for different pruning methods

the pruning criterion of one occurrence of the biphrase to one occurrence of both the source phrase and the target phrase (*once*) does not lead to a significant improvement over the original pruning when only multiple occurring biphrases ar kept (*sinuhe*). Only a little increase in coverage R at the expense of a smaller P, i.e., of a larger search space. Including all biphrases but still cutting the tail of candidates at the twentieth (*twenty*) has about the same effect, the coverage still increases by a few hundredth parts and the search space becomes still larger. This means that adding once occurring biphrases only brings few new useful terms but makes the search space larger, about twice as many distinct words to select the translation from in *twenty* as in *sinuhe*. Therefore, the decoding process might be slown down significantly only for little gain in translation quality. The last pruning technique, including only multiple occurring biphrases and cutting the tail of candidates depending on its shape allows to reach equivalent coverages with smaller search space than the three methods with fixed number of candidates. On the other hand its maximum coverage is only 0.78, letting a rather large fraction of data uncovered, and might be harmful to the translation quality.

Low rank biphrases

When using the complete phrase table, the value for R is very close to 1 for `Moses`, i.e., almost perfect recall is reached. In fact, this result appears to be misleading. We run the same coverage test, this time using only biphrases ranked further than the twentieth for a given source phrase. That phrase table contains only 11708 very frequent source phrases. Each of them is associated with a number of possible candidates. For example, the french word *de* is associated to 42569 different target phrases, *pour* to 11885 and *pour les* to 1713. In fact, almost all relatively common words from the target vocabulary have been extracted in some low ranking candidate. Since thoses source phrases are very frequent, several of them typically occur in each of the sentences to be translated. Then, almost all relatively common words are included in our target word bag and those that do occur in the target sentence can be found there even though they might have been generated only by chance from a distant word of the source sentence. This is how only rare words are left untranslated, yielding an almost perfect coverage, which is in fact only virtual, since as we explained it is generated merely by chance and cannot be handled by the decoder.

The number of candidates taken into account in the decoding process with `Moses` is generally limited to twenty, but the ranking of the candidates is not established on the same criteria. The values obtained for *twenty* ($P = 0.0065$, $R = 0.9294$), when removing only the low rank biphrases, therefore give a more accurate estimate of the operative setting also for `Moses`, even if it does not exactly correspond to the restriction applied on the search beam during the decoding process.

4.5 Tail cutting, translation quality and speed

Some phrases have a very large number of candidate translations, as we pointed out in the preceding section. Some among these candidates are erroneous translations, others are more or less literal translations. In this section we concentrate on the question of how many candidates to keep for each phrase.

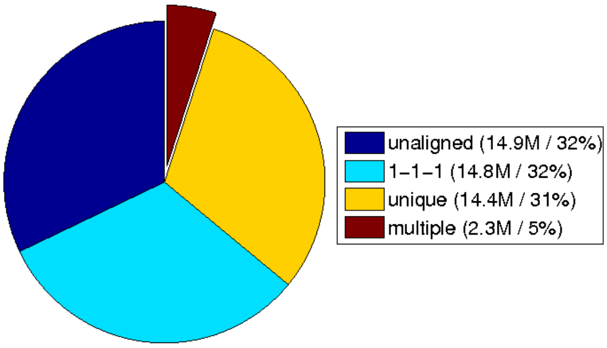


Figure 4.13: Partition of `Sinuhe` biphrases depending on their occurrence counts

4.5.1 Sinuhe pruning

Out of the 46.4 million biphrases of the original phrase table, about a third (14.9 MB) contains an unaligned end word (Figure 4.13). About another third is made up by biphrases that occur once and whose source phrase and target phrase occur once. These biphrases are denoted by their characteristic *1-1-1* occurrence count. Almost all of the last third of the phrase table is composed of once occurring biphrases for which either the source phrase, the target phrase or both occur more than once. This group of biphrases will later be referred to as *unique* biphrases. The remaining *multiple* occurring biphrases represent only about 5% of the original phrase table. The original prunig method for `Sinuhe` keeps only the last group of biphrases.

An additional criterion for filtering the biphrases is one based on their rank. The biphrases that share the same source phrase are ranked in decreasing order of occurrence count and only those ranked over k are kept, where k is a parameter than can be modified in the configuration and is typically set to twenty. Note that since some biphrases can occur the same number of time there can be several biphrases having the same rank. In this case, if for example three biphrases have equal rank four, there won't be any biphrase with rank five or six and the next biphrase will have rank seven. For that reason, phrases with more than twenty multiple occurring biphrases may not have exactly twenty candidates after pruning.

Furthermore, the ranking is established before any pruning. The once occurring biphrases do not have an influence on the ranking since they would always be ranked last. On the other hand, since the ranking is done before filtering out unaligned biphrases, some of them may be ranked among the twenty most frequent candidates, leaving holes in the ranking when they are removed.

As an example, the tails of candidates of two source phrases are given as the sequence of the counts of occurrence of the candidate biphrases in decreasing order of occurrence. The subscripts indicates the rank and unaligned biphrases are displayed between parenthesis.

devions: 78_1 60_2 50_3 42_4 29_5 23_6 12_7 11_8 9_9 9_9 7_{11} 6_{12} 6_{12} 5_{14} 3_{15} 3_{15} 3_{15} 3_{15} 2_{19} 2_{19} 2_{19} 2_{19} 2_{19} $(2)_{19}$ $(2)_{19}$ $(2)_{19}\ldots$

de manière à: $(108)_1$ 85_2 84_3 $(82)_4$ $(54)_5$ $(51)_6$ 49_7 40_8 36_9 $(35)_{10}$ 27_{11} $(26)_{12}$ 25_{13} $(23)_{14}$ $(19)_{15}$ 18_{16} 16_{17} $(16)_{17}$ $(15)_{19}$ $(15)_{19}$ 14_{21} $(13)_{22}$ $(13)_{22}$ 12_{24} 12_{24} 12_{24} $(11)_{27}$ $(11)_{27}$ $(11)_{27}$ 9_{30} 9_{30} 8_{32} $(8)_{32}$ $(7)_{34}$ $(7)_{34}$ $(7)_{34}$ $(6)_{37}$ $(6)_{37}$ $(6)_{37}$ $(6)_{37}$ 5_{41} $(5)_{41}$ $(5)_{41}$ $(5)_{41}$ $(5)_{41}\ldots$

The first case shows how more than twenty candidates can be kept for the same source phrase. After pruning, the source phrases *devions* will have 23 candidates. The second case is an extreme example of how frequent unaligned biphrases can lead to a short tail of candidates after pruning. The source phrase *de manière à* will only have 9 candidates after pruning.

A similar fixed rank tail cutting also happens in `Moses` since the number of candidates for a given input phrase is limited during the decoding. This parameter is defined in the configuration file and is typically set to twenty.

qui a permis (119, 0.04)		pallier (112, 0.08)		diminuera (49, 0.08)	
5	which enabled	9	alleviate	4	will reduce
3	which led	8	compensate for	4	will be reduced
3	which has enabled	7	remedy	3	will diminish
3	that allowed	5	overcome	3	will decrease
2	which has allowed	3	offset	3	will
2	which caused	3	deal	2	will lessen
2	which allowed	3	alleviating	2	will fall
2	that made it possible	2	plug	2	will drop
2	that enabled	2	mitigate	2	reduce
2	that brought	2	make up for	2	declining

nous avons toujours (259, 0.63)		dans les nouveaux (353, 0.76)		mesdames (6458, 0.96)	
13	we still have	268	in the new	6170	ladies
9	we always	10	in new	125	honourable
6	we have consistently	4	to the new	7	i
4	we always have	3	of the new	5	– ladies
3	we have repeatedly	3	into the new	4	dear
3	we have constantly	3	in the newly	3	members
2	we have continually	2	with the new	3	by
2	we have actually always	2	within the new	2	rapporteur
2	we continue	2	from the new	2	parliament
-	-	2	among the new	2	onorevoli

Table 4.5: Example of tails of candidates. The source phrases are followed by their count of occurrences and the ratio of these occurrences taken by the first candidate. The candidate translations are listed in decreasing order of co-occurrences with the source.

4.5.2 Tail of candidates

A fixed rank tail cutting is intuitively a simple but suboptimal solution to the problem of deciding which candidates to keep for a phrase with multiple candidates. Indeed, some phrases clearly have one good translation and the rest is merely noise while other phrases might have several equally acceptable translations, as it is the case with polysemic terms. Consider the six source phrases presented in Table 4.5. Those six source phrases have ten multiple occurring candidate biphrases, ordered by decreasing count of co-occurrences with the source. Only the aligned biphrases are reported here, this is why one of the phrases only have nine candidates.

For some source phrases, *qui a permis* or *pallier* for example, all ten candidates can be considered as correct translations while the first candidate is the only acceptable translation for *mesdames.* Some cases are more ambiguous, the candidates cannot be considered as incorrect but they are not exact translations either. For example *to the new* is acceptable for *dans les nouveaux* in some situations but it better translates to *aux nouveaux.*

The numbers in parenthesis after the source phrases are the count of occurrences of the source phrase in the training data, O_s, and the proportion of those occurrences taken by the first candidate biphrase, r_f, respectively. Let o_f be the count of occurrences of the first candidate biphrases of a source phrase. Then r_f is defined as $\frac{o_f}{O_s}$. This ratio gives a good indication of how much the translation of the source phrase is spread between different candidate biphrases. The larger r_f is, the more the occurrences are concentrated on one translation, the fewer candidates need to be taken into account.

From this idea we derive a criterion to cut the tail of candidates. From the count of occurrences of the source phrase and the number of candidates we compute o_e, the number of occurrences of each candidate under the hypothesis that all candidates are equally good translations for the source phrase, in which case they would appear equally often. $o_e = \frac{O_s}{N_s}$, where N_s is the number of candidates biphrases of the source phrase. Then we compute $o_t = (1 - r_f)o_e + r_f o_f$, the mean between the number of occurrences of the first candidate and the number of occurrences under

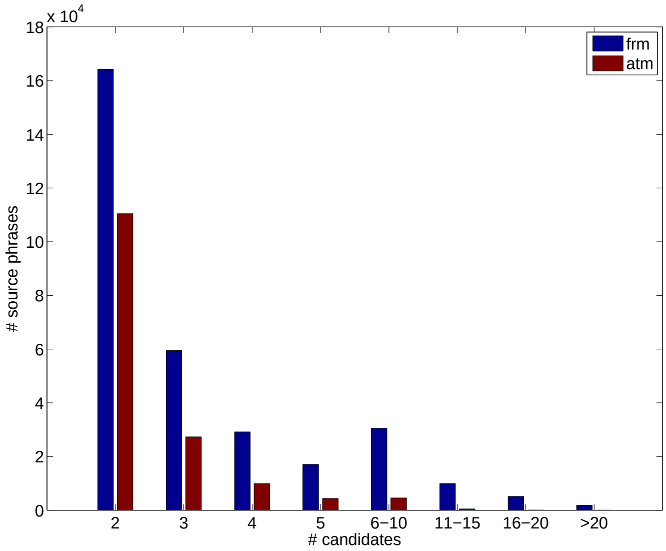


Figure 4.14: Number of candidates per source phrase for fixed rank (frm) and adaptive thresholding (atm) tail cutting methods

equiprobability assumption, weighted by the ratio of occurrences taken by the first candidate. We use k_t as an adaptive threshold, only biphrases with a count of occurrences greater than o_t are kept. The greater the r_f the closer o_t will be from o_f, the count of occurrences of the first candidate, the fewer candidates will remain in the phrase table. Note that when r_f is large, the threshold o_t will be high too and at the same time, since most of the mass is concentrated on the first candidate, the following candidates will have low count of occurrences, much lower than o_t and will not be retained. In the special case where the first candidate is an unaligned biphrase which collects most of the occurrences, no aligned candidate will occur more than o_t times and the source phrase will not appear in the phrase table. These entries are typically phrases ending with a character that is often left unaligned, there are a few thousands of such source phrases. Leaving them out is rather safe as it is difficult to assess the quality of the aligned candidates in such cases.

Using the *fixed rank* method, 932587 source phrases, i.e., slightly less than 75% of the source phrases, are associated to only one candidate. Using the *adaptive thresholding* method reduces the number of candidates of 160152 additional source phrases to one (87% of the source phrases then have a single candidate). The histogram in Figure 4.14 plots the number of source phrases depending on the number of candidate biphrases they are associated to, considering only multiple occurring biphrases. The original *fixed rank* method is denoted as *frm* while the *adaptative thresholding* method we just presented is denoted as *atm*. The number of source phrases having many candidates is smaller using the *adaptive thresholding* method, only a few hundred source phrases have more than ten candidates while with the former, there were almost 17000 such source phrases.

An alternative criterion is to keep a certain proportion p of the occurrences for each source phrase. For example, if we choose $p = 0.75$ we cut the tail of candidates so that 75% of the occurrences of the source phrase remain. For *devions* (cf. 4.5.1), which occurs 392 times in the training data, we would keep eleven candidates, since the sum of their occurrences $(78 + 60 + 50 + 42 + 29 + 23 + 12 + 11 + 9 + 9 + 7 = 330)$ is larger than $0.75 * 392 = 327$. But choosing an appropriate value for p is nontrivial and our attempts did not yield any satisfactory result. It is also disputable whether to include the unaligned and once occurring biphrases or not when computing the number of occurrences and how to break ties for equally ranked biphrases.

4.5.3 Effects of the tail of candidates

Experiment

To analyse the effects of the different pruning methods on the translation we trained `Moses` and `Sinuhe` starting from a phrase table pruned using each of the following methods:

org (*original*) no pruning,

fro (*fixed rank 1-1-1*) pruning unaligned and ranked lower than twenty,

frm (*fixed rank multiple*) pruning unaligned, 1-1-1, unique and ranked lower than twenty,

atm (*adaptive thresholding multiple*) pruning unaligned, 1-1-1, unique and less than o_t occurrences,

atu (*adaptive thresholding unique*) pruning unaligned, 1-1-1, and less than o_t occurrences, and

ato (*adaptive thresholding 1-1-1*) pruning unaligned and less than o_t occurrences.

Only `Moses` was trained with the phrase table *org*, as it contains unaligned biphrases `Sinuhe` is not able to deal with. It is impossible to discriminate between aligned *1-1-1* biphrases based on their occurrence counts, rank or translation probabilities since thoses scores are equal to one for all elements of this group. The lexical weighting (mos_2) and inverse lexical weighting (mos_4) does vary for the biphrases of this group but how to use them as a filtering criterion is unclear. For example, one-to-several links are penalized a priori since the lexical weight is divided between the differents target words. For this reason, the set of aligned *1-1-1* biphrases is always handled as a whole and *ato* is simply the union of *atu* and aligned *1-1-1* biphrases.

Translation quality

The *BLEU* scores obtained by the different settings for the translation of 2000 test sentences from the *Europarl* are reported in Table 4.7. The translations have been generated with the full systems, i.e., using a language model, distortion penalty, etc. (*TM+LM*) or enabling only the translation model (*TMonly*). Table 4.8 and Table 4.9 contain figures about the training of the different models for `Moses` and `Sinuhe` respectively. The evolution of the log probabilities during the training of the models for `Sinuhe`, that indicates the convergence of the weights, is displayed in Figure 4.15.

The pruning method seems to have more impact when using only the translation model. In that case, a stricter selection of the biphrases allows to gain 4 *BLEU* points between *org* and *atm* with `Moses` and 3 *BLEU* points between *fro* and *atm* with `Sinuhe`. In general, the more biphrases the model contains the lower the *BLEU* score.

On the other hand, when the full translation system is enabled, all scores are within one *BLEU* point for `Moses` and a half *BLEU* point for `Sinuhe` and the larger phrase tables yield the higher scores. This is probably due to the fact that the larger phrase tables allow for more variety in the translations, as we argue in the next section.

However, such small differences in *BLEU* score cannot be considered as really significant in term of translation quality. When we look at the translations generated by the different systems, we only notice minor differences in word choice. The terms or expressions chosen by the systems have typically very close meanings.

Table 4.6 gives an example of translations obtained with different models. Grey lines contain translations with translation model only while translations with the fully enabled translation system are on white background. First and second lines contain the French source and English reference respectively. Following lines are translations by `Moses`, with *org* and *atm* models, then translations by `Sinuhe`, with *frm*, *fro*, *atm* and *atu* models. The sentence level and corpus level *BLEU* scores are reported, best and worse performing settings in each category (translation system / *TM+LM* or *TMonly*) in terms of corpus level *BLEU* score are identified with a + and - respectively.

The aim here is not to compare the settings on a single translation. Indeed if a *BLEU* score computed for a test corpus of significant length is hard to interpret, a comparison based on a single sentence would be even more meaningless. We will use this example only to point out typical errors and problems for the different systems. But first we note how close the different translations are from each others.

In models where once occurring biphrases have been pruned out, the french conditional *procéderait* is left untranslated while it is translated into *should undertake* by the other systems apart from $moses\_org_{TMonly}$ where it is incorrectly translated into *would not proceed*, giving the sentence a meaning in contradiction with the source. The explanation for this error can be found when looking at the entries of the Moses's phrase table containing *procéderait* that match with the source sentence, some of which are reproduced below:

```
commission procederait ||| commission would not proceed |||
    (0) (1,2,3) ||| (0) (1) (1) (1) ||| 1 0.000212282 1 0.00465698 2.718
commission procederait a une ||| commission should undertake an |||
    (0) (1,2) (2) (3) ||| (0) (1) (1,2) (3) ||| 1 1.43621e-05 1 0.000792311 2.718
procederait ||| ask |||
    (0) ||| (0) ||| 0.000153492 9.41e-05 0.333333 0.142857 2.718
procederait ||| would not proceed |||
    (0,1,2) ||| (0) (0) (0) ||| 0.5 0.000220533 0.333333 0.0058309 2.718
procederait ||| would |||
    (0) ||| (0) ||| 5.23231e-05 2.71e-05 0.333333 0.285714 2.718
procederait a ||| should undertake |||
    (0,1) (1) ||| (0) (0,1) ||| 0.05 4.90124e-05 0.5 0.0102129 2.718
procederait a ||| would |||
    (0) () ||| (0) ||| 5.23231e-05 8.05358e-07 0.5 0.285714 2.718
```

All those biphrases only occur once in the training data, so they are pruned away as unique biphrases and *atm* as well as *frm* do not contain any translation for *procéderait*. The three entries for *procéderait* can be considered incorrect since they either don't convey the full original meaning or worse, mean the opposite. Therefore, trying to translate a chunk of the source sentence cut between *procéderait* and *à*, as in $moses\_org_{TMonly}$ inevitably leads to an error. Only considering *procéderait* together with *à* can yield a correct translation. This shows how critical the sentence segmentation can be in Moses, where the source sentence is first cut into pieces that are translated independently. Moses tries to find a segmentation of the source sentence and biphrases for each of these chunks so that its translation score is maximized. The language model score should enforce the fluency of the combination of the translated parts, possibly involving reorderings. Since Sinuhe can handle overlappings, there is no such segmentation issue. It tries to maximize the translation probability by finding a combination of compatible biphrases with possible overlappings. This is a major difference in the decoding process.

It is also interesting to notice how the word *retombées* is translated differently in its two occurrences, into *repercussions*, *impact*, *consequences*, *effects*, *fallout*, *fall-out* or *spin-off*. The first four are rather common acceptable translations, though each of them as a closer french equivalent, while *spin-off* is less adequate in this context. *Fallout* is the most literal translation of *retombées* as both have the same literal meaning. But since it also occurs with the alternative spelling *fall-out*, the occurrences are divided between the two forms so that each of them is less likely to be chosen.

Translation variety

We could witness a difference in variety when using Sinuhe to generate not only one candidate translation for a given source sentence but a list of candidates. We tried to generate the list of 500 best candidates for the 2000 sentences of our test data using *fro*, *frm*, *atu* and *atm*. The list can contain less than the required number of candidates if the system is unable to generate enough different translations using all possible combinations of the biphrases at hand. With *fro* and *frm*, that include a larger number of candidates, only eight lists contained less than 500 candidates, for very short and formal statments of the parliement. The average lengtht of these shorter lists is of 243 candidates for *fro* and 180 for *frm*. For *atu* and *atm*, 88 and 105 lists were shorter, with average length of 146 and 115 candidates respectively. When looking at the lists of candidates, the variety was clearly reduced, the different candidates being generated from two or three alternative translations for a small set of words or expressions in the sentence.

When the phrase table went through a more severe pruning, there are fewer biphrases left for the translation system to build the best combination. The pruning makes the search for the best candidate translation more deterministic. The biphrases left are more exact literal translations so that when only the translation model is enabled, the outcome is better. On the other hand, the larger phrase tables, since they are less constrained and even though they contain some bad quality biphrases, allow for more variety, from which the translation system is able to choose the best translation based on more criteria (translation score, language model score, reordering penalties, etc.) With *atu* and *atm*, the fully enabled translation system has less options to choose from, moreover these options are very similar to each other, so the improvement when using additional scores is small. This effect is even more significant when the reference translation is not a literal translation of the source sentence. With a broader set of biphrases the systems seem to have a better ability to imitate the unliteral translations.

The use of overlappings in `Sinuhe` adds constraints on the candidate translations, already enforcing the fluency of the output. This might explain why `Sinuhe` obtains higher *BLEU* scores than `Moses` without language model but lower with it, as it benefits less from the additionnal scores having less variety among the candidates.

However, these differences should be put in perspective, since as we mentioned at the beginning, the translations contain only minor differences and the *BLEU* scores for the different setting are generally very close to each other.

Model size

If the impact of the tail cutting method on the translation outcome is subject to discussion, its impact on objective factors such as the size of the model, the time needed to train the model or translation speed are unequivocal.

Leaving out unaligned biphrases, *1-1-1* biphrases and finally *unique* biphrases, allows to decrease the size of the model by about one third at each step, from 6.0 GB to 4.1 GB, 1.7 GB to finally 250 MB when keeping all multiple biphrases. The method employed to cut the tail of candidates comparatively allows for rather limited size reduction, only few percents of the model. The model size is an important issue when the translation system is to be installed on portable devices for example. Before translating a text with `Moses`, the model can be filtered to keep only biphrases whose source phrase occurs in the text, dramatically reducing the size of the model that has to be loaded into memory. In most of the cases, only a few percents of the original model needs to be loaded. Nevertheless, the full model must be stored as long as new data may be submitted for translation.

When the decision not to use the unaligned biphrases or the once occurring biphrases is taken, they should be removed from the phrase table as early as possible, to minimize the quantity of data that needs to be manipulated. It is for example possible to avoid extracting unaligned biphrases from the word aligned corpora. The absence of unaligned biphrases would presumably not have a big impact when ranking the candidates. The same way, once occurring biphrases could be filtered out before the ranking process. Since they are always on the last position, this would not affect the ranking.

Translation and training speed

When translating webpages "on the fly", which is a very common usage scenario, or if it were integrated into an automatic speech translation system, the translation speed is a factor of uppermost importance for automatic translation systems. The *adaptive thresholding* method makes the decoding process for both systems much faster. Indeed, the main advantage of this method lies in the reduction of the number of alternatives to be examined, significantly shrinking the search

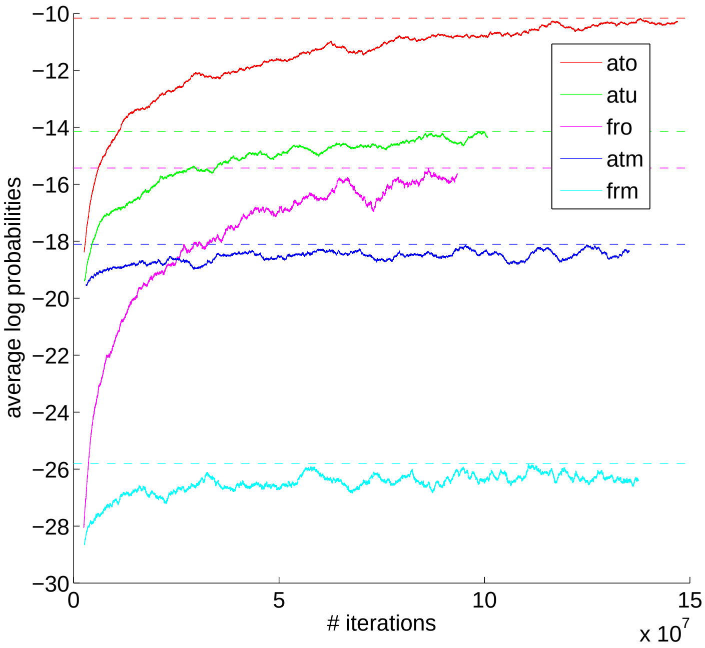


Figure 4.15: Training of the biphrases weights for `Sinuhe`: log probabilities for differents models vs. nb. of iterations

space. In a way, the systems only have good candidates at hand and simply need to assemble them. From 7.9 ms per sentence (*TM only*) and 63.1 ms/sent ($TM + LM$) with the usual *frm* for `Sinuhe`, the translation time drops to 2.7 ms/sent (*TM only*) and 26.0 ms/sent ($TM + LM$) when using *atm*. *fro* is the slowest, requiring 19.0 ms/sent (*TM only*) and 145.0 ms/sent ($TM + LM$). This is still much faster than `Moses`, even when using *atm*, which is up to ten times faster than with the full phrase table *org* (510 ms/sent with $atm\_TM + LM$ instead of 5660 ms/sent with $org\_TM + LM$).

Since the decoding process is significantly faster, the time spent in translating the tuning data during the *MERT* training is also reduced. As a consequence, the full tuning of `Moses` that required 41 hours with the original phrase table (*org*) could be completed within 7 hours or less (*atm*, *atu* and *ato*).

The number of iterations needed to obtain the convergence of the weights for `Sinuhe`'s translation model clearly depends on the total number of biphrases. The more biphrases, the more iterations needed to obtain the convergence. Training curves for models that contain the same groups of biphrases, *fro* and *ato* on the one hand, *frm* and *atm* on the other hand, have similar shapes, the latter two being flatter, with *atu* as an intermediate. At the same time, the fewer candidates there are for a source phrase the lower the average log probability reached at convergence. Indeed, the translation probability is divided between fewer candidates so that each of them receives higher probability. The number of iterations is not the only parameter determining the speed of the learning process, with fewer candidates the dynamic procedure to estimate weights is also faster. This procedure is very similar to the one used for decoding, so both speed improvements are comparable. Thus, also there the time required is reduced to hours instead of days.

source	ainsi , il a été très agréable d' entendre que m. hänsch , qui s' est prononcé au nom du groupe pse , espérait que la commission procéderait à une véritable évaluation des retombées de l' élargissement , ainsi que des retombées économiques au cas où l' élargissement n' aurait pas lieu .
reference	so it was very cheering to hear klaus hänsch , speaking for the pse group , hope that he commission would make a real appraisal of the effects of enlargement , including the economic impact if enlargement were not to take place .
$moses\_org\_TM + LM$ 13.76 / 33.02 +	thus , it was very good to hear that mr hänsch , who spoke on behalf of the pse group , hoped that the commission should undertake a real evaluation of the consequences of enlargement and of the economic repercussions in the event that enlargement should not take place .
$moses\_org\_TMonly$ 10.17 / 22.31 -	thus , it has been very agreeable to hear that mr hänsch , who has pronounced on behalf of the pse group , hoped that the commission would not proceed to a real evaluation of fallout in the enlargement , as well as that of economic repercussions of to the case when the enlargement would not have take place .
$moses\_atm\_TM + LM$ 13.47 / 32.23 -	thus , it was very good to hear that mr hänsch , who voted on behalf of the pse group , hoped that the commission procéderait to a genuine assessment of the impact of enlargement , and the economic effects in the event that enlargement would not have place .
$moses\_atm\_TMonly$ 10.81 / 26.77 +	thus , it has been very pleasant to hear that mr hänsch , who voted on behalf of the pse group , hoped that the commission procéderait to a genuine assessment of fall-out from enlargement , as well as spin-offs economic in case enlargement would not have take place .
$sinuhe\_frm\_TM + LM$ 12.43 / 31.00 -	thus , it has been very pleasing to hear that mr hänsch , who has spoken on behalf of the pse group , hoped that the commission procéderait to a genuine assessment of the impact of enlargement , as economic spin-offs in cases where enlargement should not take place .
$sinuhe\_frm\_TMonly$ 13.76 / 27.87	and , he was very pleasant to hear that mr hänsch , which has stated on behalf of the pse group , hoped that the commission procéderait to a genuine assessment of impact of enlargement , as repercussions economic if enlargement would not have take place .
$sinuhe\_atm\_TM + LM$ 10.73 / 31.40	thus , it has been very nice to hear that mr hänsch , who has spoken on behalf of the pse group , hoped that the commission procéderait at a proper evaluation of the impact of enlargement , as well as economic repercussions in the event that enlargement would not have place .
$sinuhe\_atm\_TMonly$ 13.14 / 28.88 +	and , he was very pleasant to hear that mr hänsch , which declared itself on behalf of the pse group , hoped that the commission procéderait to an proper evaluation consequences of enlargement , as well as consequences economic if enlargement would not have take place .
$sinuhe\_atu\_TM + LM$ 11.71 / 31.56 +	thus , it was very good to hear that mr hänsch , who has spoken on behalf of the pse group , hoped that the commission should undertake a proper evaluation of the repercussions of enlargement and the economic repercussions in the event that enlargement would never happen .
$sinuhe\_fro\_TMonly$ 14.41 / 25.88 -	so it was very good to hear that mr hänsch , which has insisted on behalf of the pse group who were hope that the commission should undertake a real evaluation of impact of enlargement and that repercussions economic if enlargement would not have take place .

Table 4.6: Source, reference and translations with different models for a sample *Europarl* sentence

setting	moses TM+LM	moses TMonly	sinuhe TM+LM	sinuhe TMonly
org	33.02	22.31	-	-
frm	32.43	26.46	31.00	27.87
fro	32.55	25.56	31.15	25.88
ato	32.40	22.61	31.45	26.82
atu	32.31	24.46	31.56	27.95
atm	32.23	26.77	31.40	28.88

Table 4.7: BLEU scores for translation of 2000 *Europarl* test sentences by different systems

setting	number of biphrases (M)	size on disk	ratio filtered (%)	MERT training time (h) / nb. iterations	translation speed TM + LM (ms/sent)	translation speed, TM only (ms/sent)
org	46.1	6.0 GB	9.2	41 /9	5660	2040
frm	2.1	240 MB	16.3	18 /7	2870	880
fro	30.5	4.1 GB	3.0	32 /10	3820	1310
ato	26.4	3.7 GB	0.6	3 /6	550	40
atu	11.7	1.5 GB	1.4	5 /7	1030	60
atm	1.4	175 MB	7.9	7 /9	510	50

Table 4.8: Comparison `Moses`

setting	number of biphrases (M)	size of model on disk	app. weights convergence (M iterations)	MERT training time (h) / nb. iterations	translation speed TM + LM (ms/sent)	translation speed, TM only (ms/sent)
frm	2.1	166 MB	20	6 /96	63.1	7.9
fro	30.5	3.4 GB	70	8 /90	145.0	19.0
ato	26.4	3.1 GB	150	2 / 124	21.0	7.8
atu	11.7	1.3 GB	80	4.5 /231	17.8	4.6
atm	1.4	144 MB	30	2 /92	26.0	2.7

Table 4.9: Comparison `Sinuhe`

Chapter 5

Conclusions

In this report we took a close look at phrase tables, the corner stones of *Phrase-Based Statistical Machine Translation* systems.

First, we reviewed how they are generated from aligned bilingual corpora, how they are used by the translation systems to model the translation probabilities and to generate translations.

Next, we gave some statistical information about the actual content of the phrase tables and the coverage of unseen sentences. Nevertheless, phrase tables are very dependent from the system that uses them. Some biphrases might be present in the phrase table but never be used in translations because they are ranked too low, for example.

Finally, we presented pruning methods that can be applied in order to reduce the size of the model or increase the speed of the training and decoding processes by filtering out large parts of the phrase table or reducing the number of candidate translations per source phrase, whithout affecting the translation quality significantly.

We based our work on the example of translation from French to English using data from the *Europarl* corpus to train and evaluate the systems. The generalization of the results obtained in this framework to other language pairs and other training corpora remains to be analysed.

Acknowledgements

This work was supported by the IST Programme of the European Community under the SMART project, IST-2005-033917 and under the PASCAL2 Network of Excellence, IST-2007-216886 and also in part by Academy of Finland grant 118653 (ALGODAN). This publication only reflects the authors' views.

Many thanks to Juho Rousu and Matti Kääriäinen for reviewing the draft and for their helpful comments and advices.

Bibliography

[1] Moses translation system. [Online]. Available: http://www.statmt.org/moses/.

[2] Sinuhe translation system. [Online]. Available: http://www.cs.helsinki.fi/u/mtkaaria/sinuhe/.

[3] A. Agarwal and A. Lavie. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation (WMT '07)*, pages 228–231, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[4] M. Auli, A. Lopez, H. Hoang, and P. Koehn. A systematic analysis of translation model search spaces. In *Proceedings of the Fourth Workshop on Statistical Machine Translation (WMT '09)*, pages 224–232, Athens, Greece, March 2009. Association for Computational Linguistics.

[5] N. F. Ayan and B. J. Dorr. 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 the 44th Annual Meeting of the Association for Computational Linguistics (ICML/ACL '06)*, pages 9–16, Sydney, NSW, Australia, July 2006. Association for Computational Linguistics.

[6] Y. Bar-Hillel. The present status of automatic translation of languages. *Advances in Computers*, 1:91–163, 1960.

[7] P. F. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

[8] C. Callison-Burch and M. Osborne. Re-evaluating the role of bleu in machine translation research. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL '06)*, pages 249–256, Trento, Italy, April 2006. Association for Computational Linguistics.

[9] Y. Chen, M. Kay, and A. Eisele. Intersecting multilingual data for faster and better statistical translations. In *Proceedings of the 2009 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT/NAACL '09)*, pages 128–136, Boulder, Colorado, June 2009. Association for Computational Linguistics.

[10] J. DeNero, A. Bouchard-Cote, and D. Klein. Sampling alignment structure under a bayesian translation model. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP '08)*, pages 314–323, Honolulu, Hawaii, October 2008. Association for Computational Linguistics.

[11] J. DeNero and D. Klein. The complexity of phrase alignment problems. In *Proceedings of the 2008 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT/NAACL '08)*, pages 25–28, Columbus, Ohio, June 2008. Association for Computational Linguistics.

[12] EACL. Fourth workshop on statistical machine translation. [Online]. Available: http://www.statmt.org/wmt09/, March 2009.

[13] M. Eck, S. Vogel, and A. Waibel. Translation model pruning via usage statistics for statistical machine translation. In *Proceedings of the 2007 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT/-NAACL '07)*, pages 21–24, Rochester, NY, USA, April 2007. Association for Computational Linguistics.

[14] M. Galley, M. Hopkins, K. Knight, and D. Marcu. What's in a translation rule ? In *Proceedings of the 2004 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT/NAACL '04)*, pages 273–280, Boston, USA, May 2004. Association for Computational Linguistics.

[15] K. Ganchev, J. de Almeida Varelas Graa, and B. Taskar. Better alignments = better translations ? In *Proceedings of the 2008 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT/NAACL '08)*, pages 986–993, Columbus, Ohio, June 2008. Association for Computational Linguistics.

[16] A. Haghighi, P. Liang, T. B. Kirkpatrick, and D. Klein. Learning bilingual lexicons from monolingual corpora. In *Proceedings of the 2008 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT/NAACL '08)*, pages 771–779, Columbus, Ohio, June 2008. Association for Computational Linguistics.

[17] H. Hoang, A. Birch, C. Callison-burch, R. Zens, R. Aachen, A. Constantin, M. Federico, N. Bertoldi, C. Dyer, B. Cowan, W. Shen, C. Moran, and O. Bojar. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45nd Annual Meeting of the Association for Computational Linguistics (ACL '07)*, pages 177–180, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[18] J. Hutchins. Alpac: the (in)famous report. *MT News International*, 14:9–12, June 1996.

[19] J. Hutchins. Towards a definition of example-based machine translation. In *Proceedings of MT Summit X, Workshop on Example-Based Machine Translation*, pages 63–70, Phuket, Thailand, September 2005.

[20] W. J. Hutchins. *Concise history of the language sciences: from the Sumerians to the cognitivists*, chapter Machine Translation: A Brief History, pages 431–445. Pergamon, 1995.

[21] J. H. Johnson and J. Martin. Improving translation quality by discarding most of the phrasetable. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL '07)*, pages 967–975, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[22] M. Kääriäinen. Sinuhe – statistical machine translation with a globally trained conditional exponential family translation model. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP '09)*, Singapore, August 2009. Association for Computational Linguistics. To appear.

[23] P. Koehn. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of MT Summit X*, pages 79–86, Phuket, Thailand, September 2005.

[24] P. Koehn, F. J. Och, and D. Marcu. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT/NAACL '03)*, pages 48–54, Edmonton, Canada, May 2003. Association for Computational Linguistics.

[25] P. Liang, B. Taskar, and D. Klein. Alignment by agreement. In *Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT/NAACL '06)*, pages 104–111, New York, USA, June 2006. Association for Computational Linguistics.

[26] D. Marcu and W. Wong. A phrase-based, joint probability model for statistical machine translation. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP '02)*, pages 133–139, Philadelphia, PA, USA, July 2002. Association for Computational Linguistics.

[27] N. L. G. of the USC Information Sciences Institute. Aligned hansards of the 36th parliament of canada. [Online]. Available: `http://www.isi.edu/natural-language/download/hansard/`, 2001.

[28] K. Papineni, S. Roukos, T. Ward, and W. jing Zhu. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL '02)*, pages 311–318, Philadelphia, PA, USA, July 2002. Association for Computational Linguistics.

[29] R. Rapp. Identifying word translations in non-parallel texts. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL '95)*, pages 320–322, Cambridge, Massachusetts, USA, June 1995. Association for Computational Linguistics.

[30] C. Saunders, G. Foster, M. Kääriäinen, S. Szedmak, Z. Wang, and M. Dymetman. D2.2: Application of markov approaches to smt. Technical report, SMART, 2008.

[31] H. W. Sorenson. *Parameter Estimation: Principles and problems*. Marcel Dekker, 1981.

[32] S. Szedmak, J. Shawe-Taylor, and E. Parado-Hernandez. Learning via linear operators: Maximum margin regression. Technical report, PASCAL, Southampton, UK, 2006.

[33] P. Toma. Systran as a multilingual machine translation system. In *Commission of the European Communities. Overcoming the language barrier.*, pages 569–581. Mnchen, Vlg. Dokumentation, 1977.

[34] J. Turian, L. Shen, and I. D. Melamed. Evaluation of machine translation and its evaluation. In *Proceedings of MT Summit IX*, pages 386–393, New Orleans, Louisiana, USA, September 2003.

[35] D. Vilar, M. Popovic, and H. Ney. Aer: Do we need to "improve" our alignments ? In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT '06)*, pages 205–212, Kyoto, Japan, November 2006.

[36] W. Weaver. Translation. *Machine translation of languages: fourteen essays*, 1:15–23, 1949.

[37] H. Wu and H. Wang. Pivot language approach for phrase-based statistical machine translation. *Machine Translation*, 21(3):165–181, 2007.

[38] Y. Zhang, S. Vogel, and A. Waibel. Interpreting bleu/nist scores: How much improvement do we need to have a better system. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC '04)*, pages 2051–2054, Lisbon, portugal, May 2004.