REAL-WORLD USE OF PIVOT LANGUAGES TO TRANSLATE LOW-RESOURCE LANGUAGES

by

Rohit Dholakia

B.Tech., SASTRA University, 2010

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Science

in the
School of Computing Science
Faculty of Applied Sciences

© Rohit Dholakia 2014
SIMON FRASER UNIVERSITY
Spring 2014

APPROVAL

Name: Rohit Dholakia

Degree: Master of Science

Title of Thesis: Real-world use of pivot languages to translate low-resource
                                   languages

Examining Committee: Dr. Ramesh Krishnamurthi
                                   Chair

                                   _____

                                   Dr. Anoop Sarkar, Senior Supervisor
                                   Associate Professor, Computing Science,
                                   Simon Fraser University

                                   _____

                                   Prof. Fred Popowich, Supervisor
                                   Professor, Computing Science,
                                   Simon Fraser University

                                   _____

                                   Dr. Greg Mori, Examiner
                                   Associate Professor, Computing Science,
                                   Simon Fraser University

Date Approved: 29 January, 2014

Partial Copyright Licence

SFU

Abstract

Triangulation refers to the use of a pivot language when translating from a source language to a target language. Previous research in triangulation has only focused on large corpora in the same domain. This thesis conducts the first in-depth study on the use of triangulation for four real-world low-resource languages with realistic data settings, Mawukakan, Maninkakan, Haitian Kreyol and Malagasy, where fluent translations using statistical machine translation are difficult to obtain due to limited amounts of training data in the source-target language pair. We compare and contrast several design choices one needs to consider when using triangulation. We observe that triangulation via French improves translations significantly for Mawukakan and Maninkakan, two languages spoken in West Africa. We also improve translations for real-world short messages sent in the aftermath of the Haiti earthquake in 2010 and news articles in Malagasy.

As part of the dissertation, we build the first effective translation system for the first two of these languages and outperform the state-of-the-art for Haitian Kreyol. We improve translation quality by injecting more data via pivot languages and show that in realistic data settings carefully considering triangulation design options is important. Furthermore, in all four languages since the low-resource language pair and pivot language pair data typically come from very different domains, we propose a novel iterative method to fine-tune the weighted mixture of direct and pivot based phrase pairs to significantly improve translation quality.

To Mom

Acknowledgements

This dissertation would not be possible without my advisor, Dr. Anoop Sarkar. He let me choose a topic that got me excited, helped me through the lows and high and his door was always open for guidance and advice. It has been a joy to work with him. It was made doubly more fun because I could always talk to him about good food.

Feedback from my committee members, Prof. Fred Popowich and Dr Greg Mori, made the dissertation significantly better. Thanks!

Without my Mom, I would have never dreamt of coming to Vancouver from a small town in rural India. She always told me to dream and work equally hard towards them. This dissertation and everything before and after is dedicated to her.

To my family for all the support and encouragement.

To Sudha for standing by me all these years through thick and (quite a bit of) thin.

Thanks to my friends at SFU for humoring my sense of humor and being happy victims of my amateurish cooking skills. To my roommate Sushant for always cheering me up and Diljot for always being happy to explore new hole-in-the-wall restaurants in Vancouver. And to all my lab mates for being awesome.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Why study Low-Resource languages?

Statistical Machine Translation (SMT) has enabled translation between several languages such as French, Spanish, Finnish. Translation systems are now available on the web. Google Translate now supports translation between 81 languages. The success of Google Translate in covering many different languages and producing translations of high quality for at least some language pairs is largely due to the fact that statistical machine translation uses machine learning methods over large amounts of previously translated material (which can be obtained online) in order to build fluent, accurate and fast translation systems.

However, more than 90% of the world languages do not have a publicly available SMT system. Most of the world's languages (over 7000 languages are currently spoken around the world) have not been studied in the context of SMT research. In Table 1.1, we show that the major languages typically the topic of SMT research and development have many more speakers than the languages we study in this dissertation.

Language	#speakers
French	120M
Spanish	466M
Mandarin Chinese	1026M
Haitian Kreyol	12M
Malagasy	18M
Mawukakan	2M
Maninkakan	2M

Table 1.1: Number of speakers for major and low-resource languages

Studying languages with insufficient resources leads to interesting and unique linguistic challenges. Providing a solution for these challenges take us a little closer towards the goal of a universal translator. While there are many languages spoken around the world, each language does not sit in isolation. Languages are often connected with other languages, either in a synchronic or diachronic way. For instance, Malagasy has influence from French and Arabic. While there are some loan words from French, the numbers are written right-to-left like Arabic, while also having some vocabulary overlap with Bantu. Diacritics are used but only in certain circumstances. Haitian Kreyol is a French-based Creole but does not share any vocabulary with Parisian French. The influence is from 18th century French when Haiti was ruled by France. Haitian Kreyol became an official language of Haiti only in 1961. Kreyol does not have a very complex morphology, a fact that is typical of creoles. Furthermore, in informal contexts such as short text messages, the spelling is often not used in any standard way. In our training data, there are a few English words that can be spelt in Kreyol six different ways.

1.2 Phrase-based SMT Pipeline

We use a fairly standard SMT toolkit in this thesis. To the interested reader, we refer to the comprehensive and readable SMT survey by Adam Lopez [Lopez, 2007]. In this section, we discuss the framework for phrase-based SMT [Koehn et al., 2003] which has been used for all the experiments in the dissertation, and discuss how a low-resource language pair can create a stumbling block in each stage.

SMT uses data-driven models to translate sentences in a source language to a given target language. Given a parallel corpus between s and t, a phrase-based SMT system has a generic pipeline that looks as described in Algorithm 1.

Algorithm 1: Building a phrase-based system

Input: Parallel corpus between s and t: sentence-by-sentence translations of language s into language t

Output: A translation model "tm"

    Alignments: Learn bi-directional alignments from the parallel corpus
    Extraction: Extract phrase pairs from the alignments and compute probability-based feature values for each translation pair. This is called the translation model.
    Tuning: Learn the weights for the features by maximizing BLEU score on a development set using discriminative Minimum Error Rate Training (MERT)
    Decoding: Using a language model and translation model, translate test sentences

---

The reason this is called phrase-based SMT is because the base unit of translation are
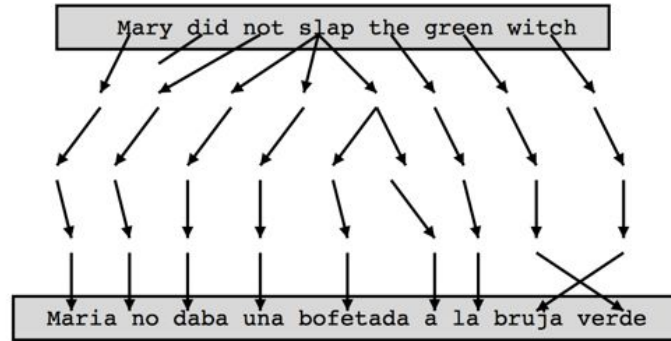
Figure 1.1: Model 3: Fertility [Knight and Koehn, 2003]

phrases. The phrases do not have to be linguistically motivated. Phrases in this context means a continuous group of words. In this dissertation, we only use phrase-based SMT for all the experiments but there are other types of SMT models proposed in the literature. [Chiang, 2007, Galley et al., 2004]

Each step of Algorithm 1 outlined above raises questions when faced with a low-resource language pair. Low-resource languages are those with insufficient resources to use for Machine Translation into and/or from the language. To provide perspective, French has a corpus with 10^9 parallel sentences with English. On the other hand, the language with the highest amount of data in this dissertation is Haitian Kreyol, with 121K sentences. Out of those 121K, only 16% are from the target domain, the sentences are noisy and with punctuation and spelling mistakes. What do we mean by noisy here? Noisy corpus here refers to parallel data with misalignments (sentence 1 in Haitian Kreyol aligned to sentence 3 in English), spelling mistakes (cafe -> caf*) and one sentence split into multiple without proper delimiters (often the case in Malagasy). Note that no preprocessing was done on any of the languages to resolve the noisy nature.

Let us consider each step and discuss the problems that come up. Given parallel data, the goal of the alignment models [Brown et al., 1993, Vogel et al., 1996] is to learn which word in source language s translates to target language t and assign a likelihood to the pair of words. The advanced alignment models use initial alignments from IBM Model 1 (Model 1, henceforth). To take into account the fact that one source word can align to multiple target words, we use IBM Model 3 which uses the concept of "fertility" to model source words that align to multiple target words (shown in Figures 1.1 and 1.2). Model 1 is typically initialized uniformly and uses Expectation Maximization [Dempster et al., 1977]
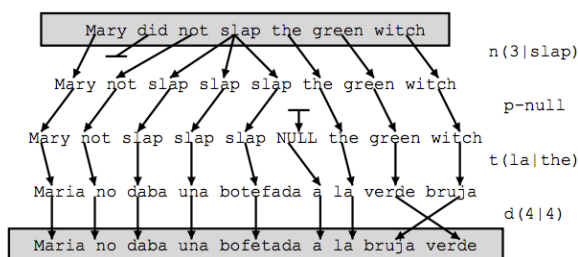
Figure 1.2: Model 3: Fertility [Knight and Koehn, 2003]

direction	tgt	src & alignment
forward	enpe tout kote	NULL ( ) a ( ) bit (1) everywhere (2 3)
backward	a bit everywhere	NULL ( ) enpe (1 2) tout ( ) kote (3)

Table 1.2: Example of a forward and backward alignment

to learn the parameter values. Facing a corpus of a small size, the alignment models will end up making inferences that are not always true. They will place higher likelihood on pairs seen fewer number of times due to lack of data. At the end of the alignment process, we will have two alignment files. The forward alignment file will say which words in the target language align to which words in the source language. The backward alignment file will say vice-versa. An entry from the backward and forward alignment files would look as shown in Table 1.2. The *forward* line says that the target word "a" is aligned to nothing on the source side, while bit is aligned to *enpe* and everywhere is aligned to both *tout* and *kote*. The *backward* line says the same thing but the target side is Haitian Kreyol.

The phrase extraction step (Line 2) looks at alignments learnt from Line 1 in both directions and determines which phrases can translate from one language to another using the intersection of the alignments. There are several ways of computing the intersection in the literature, but we consider the approach outlined in [Koehn et al., 2003]. After the intersection, points are added if they are in the union of the bi-directional alignments and connects a previously unaligned word. This heuristic is known as *grow-diag-final-and*. This heuristic produces phrase pairs with more accurate alignments. At the end of this step, we have a phrase table which has rules shown in Table 1.3:

The Table 1.3 says that the source phrase **! la situacion de haiti ,** translate to the target phrase **concerned about the situation in haiti ,** with the feature values shown on the right.

src	tgt	features
! la situacion de haiti	concerned about the situation in haiti	0.5 8.16237e-09 1 0.000483004 2.718

Table 1.3: Example of a phrase pair in the Haitian Kreyol to English table

Feature	Explanation
$p_w(f \mid e)$	probability of seeing phrase "f" given phrase "e"
$p_{lex}(f \mid e)$	lexical probability of seeing phrase "f" given phrase "e"
$p_w(e \mid f)$	probability of seeing phrase "e" given phrase "f"
$p_{lex}(e \mid f)$	lexical probability of seeing phrase "e" given phrase "f"
phrase penalty	a constant value penalizing distortion (2.718)

Table 1.4: Features of the phrase pairs, where "f" is Foreign/source & "e" is target/English

Log-linear models can define a relationship between K features of data with a function of our interest, which in this case is $p(e \mid f)$. Equation (1.1) shows the equation for a log-linear model. The denominator is a normalizer that makes the quantity a probability. The equation says that to find the best translation e for a given source sentence f, we will multiply the weight of the features with the values and then normalize that over the complete n-best list (represented by Y()). n-best list is the top-n translations of a given source sentence.

$$P(e \mid f) = \frac{\exp \sum_{k=1}^{K} \lambda_k h_k(e, f)}{\sum_{e':Y(e')} \exp \sum_{k=1}^{K} \lambda_k h_k(e', f)} \qquad (1.1)$$

When using log-linear models to find the best output translation for a given sentence, we use :

$$p(e \mid f) = \prod_i h_i(e, f)^{\lambda_i} \qquad (1.2)$$

where f is the input sentence, e is the output translation, h_i are the feature functions and λ_i are the weights. Typically, the log values are used by the decoder, resulting in the equation (1.3)

$$\log p(e \mid f) = \sum_i \log(h_i(e, f))\lambda_i \tag{1.3}$$

Typically, the log-linear model shown in equation (1.1) has 13 parameters, λ_1 to λ_{13}. The components typically used are :

- phrase translation model (4 features)

- phrase penalty (2.718)

- language model (1 feature)

- distance-based reordering (1 feature)

- lexicalized reordering model (6 features)

The language model score is the score of the given target translation given by the language model. The phrase penalty is fixed at 2.718 (value of Euler's number e). Lexicalized reordering models are learnt using the alignments obtained above. Five features are mentioned in Table 1.4. The two p_w are the phrasal features, features that determine the likelihood of the source phrase translating to target and vice-versa. The phrasal translation likelihood is computed by using relative frequencies, as shown in equation (1.4).

$$p_w(f \mid e) = \frac{c(f, e)}{\sum_{f'} c(f', e)} \tag{1.4}$$

The counts referred to in equation (1.4) are obtained from the alignments. Note that the alignment models that were learnt on a small-sized corpus will cause some propagation of errors in the phrasal probabilities.

The lexical features [Koehn et al., 2003] are computed as shown in equation (1.5) :

$$p_{lex}(f \mid e, a) = \prod_{i=1}^{n} \frac{1}{\{j | (i, j) \in a\}} \sum_{\forall (i, j) \in a} w(f_i \mid e_j) \tag{1.5}$$

The intuition behind having a pair of lexical features is to reward phrases that contain high probability alignments while penalizing phrases with poor alignments, which are likely to be spurious and lead to worse translations. As shown in equation (1.5), the lexical

probability is the product of the lexical alignment probabilities of the constituent words in the phrase table.

Having learnt translation pairs with their respective features, we now want to know which features are better indicators of good translations and vice-versa. For weight learning, we use Minimum Error Rate Training. Before discussing MERT, its important to know about BLEU [Papineni et al., 2002], **Bi**lingual **E**valuation **U**nderstudy. BLEU is the error metric used most often when comparing output translations with reference translations. BLEU compares an output translation with a reference translation according to equation (1.6)

$$BLEU_{score} = BP \cdot \sum_{i=1}^{n} w_i p_i \qquad (1.6)$$

where w_i is the weight to the n-gram while p_i is the modified n-gram precision and BP is the *Brevity Penalty*. Brevity Penalty is used to penalize phrases that are much shorter compared to the reference translations. It's a way of guarding against relatively short translations with common words. Modified n-gram precision is a corpus-based count of the n-gram, which is modified to not count the co-occurences which are repeated in the same sentence. For instance, for an output translation,

```
the the the the the the the
```

with a reference translation

```
the cat on a mat
```

The co-occurence of "the" is only counted once and not 7 times.

The modified precision explained above is defined as in equation (1.7)

$$p_n = \frac{\sum\limits_{c \in \{Candidates\}} \sum\limits_{n-gram \in c} Count_{clip}(n-gram)}{\sum\limits_{\acute{c} \in \{Candidates\}} \sum\limits_{n-gram\acute{} \in \acute{c}} Count(n-gram\acute{})} \qquad (1.7)$$

where Candidates refers to the target set of sentences.

Minimum Error Rate Training, abbreviated as MERT from here onwards, chooses weights for features that minimize BLEU score loss given a "tuning" set. A "tuning" set is a set of parallel sentences between source and target that is in the same domain as the test and of the same type. For instance, when trying to improve translations for Haitian Kreyol

short messages, we have tuning and test in the same domain, SMS, although our training is 85:15 mix of out of domain data versus in-domain SMS data. MERT takes translation pairs generated from a mixture of domains corpus and tunes the weights such that the translations are more like that target domain. In Haitian Kreyol, as our training rules have been extracted from a smaller corpora that has not been manually sentence-aligned, MERT is learning weights for features that have values which are not always true. This is why we re-tune our weights for the interpolated model after obtaining a translated table with scaled values from the much larger French-English table.

Now, we come back to the problem of learning weights for our log-linear model in equation (1.1). The goal of MERT is to find the best model, with the best model producing the smallest error with respect to a given error function. Hence, assuming we have an error function that quantifies how erroneous is an output translation when compared to the reference translation, MERT can provide us the best model [Lopez, 2007].

Formally, as discussed in [Lopez, 2007], if we have an error function $E(\hat{e}, e)$ defining the amount of error in a translation $\hat{e}$ w.r.t reference translation e, the objective function is :

$$\lambda_1^K = \operatorname*{argmin}_{\lambda_1^K} \sum_{(e,f) \in C} E(\operatorname*{argmax}_{\hat{e}} P_{\lambda_1^{\hat{K}}}((\hat{e} \mid f), e) \tag{1.8}$$

The key to MERT is doing a line search along one feature while keeping the others constant. Lets assume a corpus of one line. In the first iteration of MERT, a n-best list will be generated. These are the top-n translations by the decoder by using the default weights for all the features. For a given sentence, the n-best list might look as shown in Table 1.5. After this n-best list is generated, the task is to find the best translation for the given source sentence. Remember that each sentence has several features and MERT has to learn weights for each. The overall likelihood for the sentence is defined by equation (1.9). The best is defined by the sentence which minimizes the error (equation (1.10)).

$$p(x) = \exp \sum_{i=1}^{n} \lambda_i h_i(x) \tag{1.9}$$

$$x_{best}(\lambda_1, ..., \lambda_n) = argmax_x \exp \sum_{i=1}^{n} \lambda_i h_i(x) \tag{1.10}$$

At this point, MERT decides to do a line search. We can learn the best weight for one feature, say at index c, by keeping all the other features constant. This is shown in

Source	n-best list
ki kote y ap bay manje ?	how can i find help for my province of aquin . how can i find help for my in aquin . how can i find help for my part of aquin . how can i find help for my province of aquin ? how can i find help for in my province of aquin . how can i find aid for in my province of aquin . how can i find help for my part aquin . how can i find help for my in aquin ? how can i find help for my country aquin . how can i find help for my in aquin .

Table 1.5: Example of a n-best list, where n $\leq$ 100

equation (1.11)

$$u(x) = \sum_{i \neq c} \lambda_i h_i(x) \tag{1.11}$$

Now, the equation (1.10) will look like equation (1.12).

$$x_{best}(\lambda_c) = argmax_x \lambda_c h_c(x) + u(x) \tag{1.12}$$

Now, each translation in the n-best list is the line of an equation and the points at which the best λ for this line will change is at the points where the line intersects. The best λ can be found in the same way for all the lines in the tuning set and the one that minimizes BLEU loss over the whole corpus becomes the weight for this feature.

The process outlined above explains both the good and bad about MERT. The good being that MERT works effectively around the fact that we have a scenario of minimizing BLEU loss, which is not smoothed and which is a corpus level error metric, inside an *argmax* as in equation (1.8). The bad is that MERT does not scale to many features. At each iteration, weights have to be learnt for all the features. After the iteration, the n-best list is regenerated to have the maximum number of entries, done to cover the hypothesis space as much as possible. To avoid local minima, in practice, MERT is started from not one but a few random points.

Having obtained the weights and a language model on the target side, decoding refers to the process of finding the best translation for the source sentence as shown in equation (1.13).

$$
\begin{aligned}
e_{best} &= \underset{e}{\text{argmax}}\, p(e \mid f) \\
&= \underset{e}{\text{argmax}}\, p(f \mid e)p_{LM}(e) \\
&= \underset{e}{\text{argmax}} \prod_{i=1}^{I} p(\hat{f}_i \mid \hat{e}_i) d(start_i - end_{i-1} - 1)p_{LM}(\hat{e})
\end{aligned}
$$

To find the e_{best}, we want to go over all possible translations of the foreign sentence **f**. Using Bayes Rule, we flip the search to now have a translation model $p(f \mid e)$ and a language model $p_{LM}(e)$. The foreign sentence **f** is segmented into a sequence of phrases f_1 to f_I. **d** is defined as the distortion penalty. Some languages show long distance reorderings (e.g In Japanese, the verb comes at the end of the sentence) but most languages do not. In those cases, allowing long distance reordering leads to poorer translations. Hence, in the decoding model, we add a parameter for the distortion, which is defined as the distance between the next word we are choosing and the current word.

Phrase-based SMT has been used with great success before in the literature. But, as described above, the approach is quite data-driven and it is not clear how to achieve fluent translations with only a little parallel data.

1.3   Examples using triangulation

The easiest way to get better translations is to have more data between the source and target languages. As the amount of data increases, the models will learn the correct alignments which leads to more meaningful translation rules with accurate feature values and thus, more fluent translations. The second easiest way is to improve tokenization for the source and/or target language. Better tokenization goes a long way in pre-processing the text correctly. But, what do we do when these two options are not available?

In the Table 1.6, we mention a sentence in Mawukakan from the heldout data. The direct translation is the output translation we obtain by only using the 3K training sentences we have. The interpolated output shown is the output after interpolating a triangulated model with the direct translation model. The reference translation is the best translation for that sentence. The word *yàngáláà* and *lákwé* are out-of-vocabulary words for the direct system. Words that have no translations in the phrase table are out-of-vocabulary words. By using an English translation via the Europarl corpus, we translate all the source words in the interpolated table. In the second example, all the source words are known. But, we get a

Language	Category	Example translation
Mawukakan	Before After Reference	*her father yángàlóò has lákwè everything* the disease her father is not in a position to everything the illness has rendered her father invalid
Mawukakan	Before After Reference	*the entrance of the child behind her back and let us go home* the child behind her back and let us go home take the baby in your back and let 's go home
Haitian Kreyol	Before After Reference	*do we still have earth-shock for haiti ?* are there always earthquake in haiti ? are there any more earthquakes in haiti ?

Table 1.6: Examples of improvements in translations. These examples show how the pivot language can provide new useful candidate translations missing from the direct system.
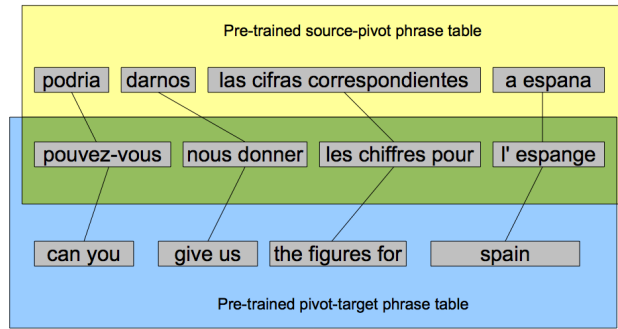
Figure 1.3: Example of triangulation from [Clifton, 2012]

weird "the entrance" phrase in the beginning. What went wrong? When we look deeper, we see that "the entrance" is one of the top translations for the word là, out of 23 translations in the direct table. After triangulation and interpolating with the direct system, it has 1615 translations and "the entrance" is nowhere to be seen. This is because some of the translations of là have common pivot phrases which end up making the translation model give lower value to the existing ones. Another example of triangulation is shown in Figure 1.3 from [Clifton, 2012].

The approach of triangulation [Cohn and Lapata, 2007, Utiyama and Isahara, 2007, Wu and Wang, 2007] aims to add translations for new source phrases while also improving translations for existing source phrases. Both the aims are contingent on the common pivot phrases between the source pivot and pivot target tables. New source phrase translations (like shown in Table 1.6) can be added if one has the source phrase in a source pivot corpora that leads to a new target phrase in the pivot target corpus. In a low resource scenario, its

important to achieve both aims with triangulation. Owing to less training data, the direct system has several out-of-vocabulary (OOV) words. We aim to reduce the number by using triangulation. At the same time, it is reasonable to assume that the source phrases we do have translations for are not always right, as seen in the second example on Table 1.6. We put our trust in triangulation to improve existing translations.

1.4 Contributions of this dissertation

We conduct the first in-depth study of the design choices in triangulation, the first using four real low-resource languages with realistic data settings. As part of the dissertation, we also build the first translation systems for two of the four languages. Our Haitian Kreyol system outperforms the best system from the Sixth Workshop on Machine Translation, 2011. Our Haitian Kreyol system gets 34.00% accuracy while a leading online translation system gets 16.72% on the same heldout set. As part of our study, we compare and contrast the models for computing phrase scores, lexical scores and also propose a novel iterative method for doing linear interpolation.

1.5 Experimental Setup

Moses [Koehn et al., 2007] was used for all the experiments. Moses is a leading publicly-available, open source SMT system with a rich documentation and active contributors.

To build our baseline systems, we followed the standard set of steps: generated bi-directional alignments using GIZA++ [Och and Ney, 2003], followed by phrase extraction using the *–grow-diag-final-and* heuristic. The heuristic intersects the alignments in both directions and takes the longest alignment that is common. The decoder parameters were optimized using Minimum Error Rate Training [Och, 2003] by minimizing BLEU [Papineni et al., 2002] loss on a development set. All scores reported are case-insensitive BLEU. All language models were generated using SRILM [Stolcke, 2002].

KenLM [Heafield, 2011] was used for language model scoring when decoding. SRILM is a language modeling toolkit for generating language models covering several smoothing and interpolation models. KenLM enables fast lookups in large language models by using efficient data structures.

1.6 Remarks

In this chapter, we described a generic phrase-based SMT pipeline and discussed the challenges that come up when facing a low-resource language pair. We also mentioned examples

of how triangulation improves translations. In the next chapter, we will discuss triangulation in more detail and also describe the various design choices involved in effective usage of triangulation.

Chapter 2

Triangulation

2.1 What is Triangulation?

To explain triangulation, let's use an example from the previous chapter. We saw in the previous chapter that the sentence *yàngáláà wɛ́ɛ́ à à lákwé kóó bɛ́ mà .* had two OOVs resolved after interpolating with a triangulated table. In other words, the direct system did not have any knowledge about two phrases, *yàngáláà* and *lákwé*. We observe that in the source pivot table, we have 1 rule for the word *yàngáláà*.

yàngáláà ||| la maladie ||| 0.285714 0.0926127 1 0.16 2.718

The French phrase "la maladie" has 215 translations in the Europarl table. We use all 215 options to compute feature values for new (*yàngáláà*, tgt) translation pairs and select the top-n and add it to the table. These steps are explained in a formal manner below.

Consider a source language, s, a target language, t, and a pivot language i. You have a little parallel data between s and t and believe that triangulation will increase the quality of translations between s and t. What steps one would follow to get the desired result?

The algorithm for triangulation is described in Algorithm 2. Having obtained new source target pairs by using the common pivot phrases in Line 1, we proceed to compute the feature values for the new phrase pairs (Line 4). To minimize the noise, we only select the top-n translations for any given source phrase (Line 6). Line 1 reiterates the importance of having a source pivot corpus of reasonable size. The triangulated translation model is contingent upon common pivot phrases and without a reasonably-sized source pivot corpus, we cannot fully utilize the large pivot target corpus (2M Europarl sentences in this dissertation). Having generated a triangulated translation model, one can combine it with the existing baseline

---

Algorithm 2: Vanilla Triangulation

Input: phrase table between s and i, p_{s-i},
phrase table between p and t, p_{i-t},
n for selecting top-n phrase pairs
Output: p_{trian}, initially empty
1: **for all** (src, pivot) in top-n p_{s-i} **do**
2:    **if** pivot phrase in p_{i-t} **then**
3:       **for** all (pivot, tgt) pairs in p_{i-t} **do**
4:          compute feature values for (src, tgt)
5:       **end for**
6:       select top-n src-tgt pair, add to p_{trian}
7:    **end if**
8: **end for**

---

model in several ways [Bertoldi et al., 2008, Nakov and Ng, 2012, Cohn and Lapata, 2007].

Having obtained a new source target pair, how best to compute the feature values? For source phrase src, target phrase tgt and pivot phrase pvt, we can compute the feature values like in [Utiyama and Isahara, 2007] using the following equations :

$$p_{lex}(tgt \mid src) = \sum_{pvt} p_{lex}(tgt \mid pvt)p_{lex}(pvt \mid src) \tag{2.1}$$

$$p_{lex}(src \mid tgt) = \sum_{pvt} p_{lex}(src \mid pvt)p_{lex}(pvt \mid tgt) \tag{2.2}$$

$$p_{w}(tgt \mid src) = \sum_{pvt} p_{w}(tgt \mid pvt)p_{w}(pvt \mid src) \tag{2.3}$$

$$p_{w}(src \mid tgt) = \sum_{pvt} p_{w}(src \mid pvt)p_{w}(pvt \mid tgt) \tag{2.4}$$

For all the feature values, we multiply the corresponding values for source pivot and pivot target entries and marginalize over the pivot phrase. Note that we are making an independence assumption shown in equation below.

Source phrase	translations
à lá báárá ɲúmá kóśń	for the good job he has accomplished
à lá báárá ɲúmá kóśń	her good work the good work the good work carried out the good work done the good work done by the good work he has done the good work that the sound work

Table 2.1: 1 translation before and 8 after triangulation for a source phrase in Maninkakan

$$
\begin{aligned}
p(tgt \mid src) &= \sum_{pvt} p(tgt, pvt \mid src) \\
&= \sum_{pvt} p(tgt \mid pvt, src)\, p(pvt \mid src) \\
&\approx \sum_{pvt} p(tgt \mid pvt)\, p(pvt \mid src)
\end{aligned}
$$

We are assuming that the pivot phrase fully represents the information and thus, neglect the tgt phrase in the equation (2.5). We call this approach the *Product* approach.

Multiplying all the features on the source pivot and pivot target is an obvious first way to obtain feature values for the triangulated table. Most previous papers follow the same route for initial scores. As we will see later, using the joint probability and changing the lexical scores leads to small but consistent improvements.

In Table 2.1, we see that the source phrase *à lá báárá ɲúmá kóśń*as only 1 translation before using triangulation. Note that the training corpus has the word "accomplished" only once. After using the target phrases in Europarl, we get 8 translations for the same phrase. We also see that "work" has changed to "job" and the possible target phrases are shorter.

2.2 Models

2.2.1 Top-*n* filtering

The size of the triangulated phrase table is controlled by the number of translations n considered for a given source phrase. Consider a source phrase p_s that translates to p_p in the pivot language. The phrase p_p has 1293 translations in the pivot target table. Considering all

Setting	Utiyama:07	Cohn:07
Phrase Scores	Product approach	Joint Model
Lexical Scores	Product approach	IBM Model 1
Interpolation	n/a	Uniform
Corpus	Europarl	Europarl

Table 2.2: Comparison of [Utiyama and Isahara, 2007] and [Cohn and Lapata, 2007]

the 1293 translations will result in 1293 translations for the phrase p_s via one pivot phrase. It is reasonable to expect the phrase p_s to have multiple pivot translations, all having a higher number of translations in pivot target. Considering all translations is not recommended for several reasons. Firstly, this will lead to a very large phrase table. Table 2.3 shows the number of rules we can end up with if we consider all possible paths to a target phrase. To put it in perspective, the direct table for Mawukakan and Maninkakan have 51K and 60K phrase pairs respectively. Secondly, along with valid translations, triangulation also adds some noise to the translations by considering several translations of the common pivot phrase. Considering all translations would add even more noise to our triangulated phrase table. Having said that, when one has only 5000 parallel sentences for the direct system, how large a value of n is enough? We found that using n = 100 was the same as using n = 1000 in terms of the BLEU scores. We also observed that using a value of more than 100 added noise for Haitian Kreyol and Malagasy. Thus, it is important to pay attention to the fan-out limit in triangulation.

Language	Direct Table	Triangulated Table
Maninkakan	51K	106.7M
Mawukakan	60K	151.6M

Table 2.3: Number of rules if all possible paths are considered

2.2.2   Connectivity features

The phrase pairs in the triangulated phrase table are contingent upon the common pivot phrases. As a result, we can have phrase pairs that map "!" to a target phrase "and making the soup thick !" in Haitian Kreyol to English triangulated phrase table. Computing feature values using equations from section 2.2.1, it is reasonable to assume that spurious phrase pairs like above can get a high enough feature value to end up as a translation during decoding. To reward phrase pairs that have more alignment links between and to penalize

pairs that don't, we add two connectivity features to the phrase table, as used in [Kholy and Habash, 2013].

For a source phrase p_s, target phrase p_t, and with the number of alignment links between them N, the strength will be calculated as follows :

$$source_{strength} = \frac{N}{S}$$

$$target_{strength} = \frac{N}{T}$$

where S is the length of the source phrase p_s and T is the length of the target phrase p_t. To compute the connectivity strength feature, the alignments in the source pivot phrase pair are intersected with the pivot target phrase pair. If the resulting alignment has a higher strength, it implies that a majority of the source words do have an alignment with the target.

2.2.3   IBM Model 1 Alignment

In section 2.2.1, we computed the lexical scores by multiplying the component scores and marginalizing over the pivot phrase. The component lexical scores are a measure of the word-to-word alignment [Koehn et al., 2003] and by multiplying them, the final lexical scores are implying some strength-of-tie for each pair in the source target translation. But, as was discussed in 2.2.1, using triangulation adds some noise to the translation model by proposing spurious translations.

An alternative way to compute the lexical score is to use a Model 1 [Brown et al., 1993] score between the phrase pairs in the triangulated table. Treating the triangulated phrase table as a parallel corpus, we learn the model 1 alignment scores in both directions using 5 iterations of the EM algorithm [Dempster et al., 1977]. Given a Foreign sentence $f = f_1 \ldots f_m$, English sentence $e = e_1 \ldots e_l$, the model 1 score between the sentences is calculated as follows:

$$p(f, a \mid e) = \frac{\epsilon}{(l+1)^m} \prod_{j=1}^{m} t(f_j | e_{a(j)}) \tag{2.5}$$

The strength features in section 2.2.2 assign a phrase-level score to a given translation pair. The score does not reflect the actual many-to-one alignments between the phrases. A Model 1 score learns the likelihood of the alignment of the individual words, while also

considering the fact that a triangulated table will have less number of source phrases translating into good and some noisy translations. Noisy translations will automatically get a lower Model 1 score, something less likely to happen when using the simpler approach of multiplying the lexical scores. This effect of noisy translations ending up as a viable translation during decoding is also because of the limited source pivot training corpora available. Several translations have been only seen once and the phrase lengths are not very long either (85% of Mawukakan and Maninkakan phrase table has entries less than or equal to 3 words). A modified IBM Model 1 score is also used in [Cohn and Lapata, 2007] in the absence of word alignments. They report a BLEU score improvement of 2 points over the standard feature set when using the Model 1 score, but we observe a different pattern altogether across all the four resource-poor languages which is explained in more detail in Section 3.4.

2.2.4   Joint and Conditional Distributions

Another way of calculating the triangulated phrase scores $p_{tr}(e \mid f)$ and $p_{tr}(f \mid e)$ would be to take the joint probability $p_{tr}(s, t)$ and decompose it to get the conditional distributions. But, we do not have the counts in the triangulated phrase table. The pairs that end up in the triangulated table are contingent on the common pivot phrases that connected the source target pair, thus, counting the pairs after triangulation will not be a true reflection of the joint probability. For a triangulated table between *src* and *tgt*, using source pivot table *sp* and pivot target table *pt*, we can compute the joint probability of a phrase pair (s, t) as follows:

$$
\begin{aligned}
p_{tr}(s, t) &= \sum_i p_{sp}(s, i) p_{pt}(i, t) \\
&= \sum_i p_{sp}(s \mid i) p_{sp}(i) p_{pt}(i \mid t) p_{pt}(t)
\end{aligned}
$$

This is a more accurate description of the joint probability of the (s, t) phrase pair in the triangulated table because we are using source pivot and pivot target counts, both of which have been extracted from the alignments.

We compute $p_{sp}(s, i)$ using equation (2.6)

$$
p_{sp}(s, i) = p_{sp}(s \mid i) p_{sp}(i) \tag{2.6}
$$

As we have the counts for the direct system, computing the joint and the conditional

distributions is relatively straight-forward. When interpolating the triangulated and direct translation models (2.7), the three new features are added to the log-linear pipeline. Owing to the smaller size of the source pivot corpora, we observed it was better to add the three new features to the log-linear pipeline and let MERT decide which features lead to a better BLEU score. This is in contrast to [Cohn and Lapata, 2007] where they combine the joint probability of the phrase pair from direct and triangulated uniformly, and use the resulting conditionals as part of the log-linear pipeline.

2.3 Translation Model Combination

Combining translation models, trained on corpora from different domains, is an inherently difficult task. We want to make our translations better on the domain of the test set, while also correcting errors in our baseline translation model. In case of low-resource languages, the baseline translation model has been trained on completely out-of-domain corpora or some in-domain and a lot of out-of-domain corpora. This results in translation pairs that are missing altogether or translation pairs with so low probability that decoding misses them altogether. The aim of Interpolation is to add translation pairs that are missing and give more weight to translations that are more valid in the given domain.

Consider the translations from Haitian Kreyol to English. We have a baseline model trained on a little in-domain parallel data (less than 17K sentences). We aim to make our translations better on the same domain using a lot of out-of-domain data, which in our case is parliamentary proceedings. It's important that we do not make the baseline model translations end up at the bottom of the stack because they are in-domain. At the same time, we do not want to miss out on the valid translations introduced by the larger, clean parliamentary proceedings based translation model.

2.3.1 Example

Consider a phrase pair, (jan nou, that you). Each phrase pair has a set of scores associated with it in the phrase table. They are the forward and backward lexical probabilities, and the forward and backward phrase probabilities.

From the direct phrase table, we have the following scores for the phrase pair mentioned above. The last score, 2.718, is a constant which is the phrase penalty.

```
jan nou  ||| that you ||| 0.000786782 2.11603e-05 0.125 0.00906772 2.718
```

The triangulated table also happens to have the same phrase pair with different scores.

These scores have been obtained by using the equations shown above.

```
jan nou ||| that you ||| 0.00318015 7.75194e-05 0.0715829 0.00214831 2.718
```

We know that our direct system has been trained on in-domain data, hence, it should get more weight intuitively. A heuristic approach to this problem would choose a pair of values and see which one does best. For instance, if you choose 0.85 for the direct table and 0.15 for the triangulated table, the end result for the phrase pair would look like the following :

```
jan nou ||| that you ||| 0.0011 2.961416503e-05 0.116 0.0080 2.718
```

There are several flaws with the approach outlined above. Firstly, an intuitive idea about the importance of the in-domain or out-of-domain phrase table is not enough. The direct Haitian Kreyol to English phrase table has been trained on only 120K parallel sentences and cannot always be right. Hence, starting with 0.9 for the direct table and 0.1 for the triangulated table is an extreme step. So is 0.5 and 0.5 because we want translations with more influence from the cleaner, larger Europarl data. Moreover, as we will discuss in the other chapters, we report results on several combinations of triangulation, based on changes in phrase scores, lexical scores and adding connectivity features. With every improvement, the importance of the triangulated table might increase or decrease. The heuristic approach will not be able to take that into account.

We use CONDOR [Thain et al., 2005] to perform an efficient grid search over the pairs of co-efficients based on the BLEU score of the interpolated system on the heldout set. Our interpolation method would have the steps outlined in Algorithm 3

For instance, consider the word "tranblemanntè". It gets translated to *shaking* by our best baseline system. After interpolating our top-20 triangulated translation model, it gets translated to earthquake. Note that the word earthquake is present in the baseline translation model but does not end up as a translation for the source word "tranblemanntè".

$$p_{interp}(s \mid t) = \lambda_d p_d(s \mid t) + (1 - \lambda_d) p_t(s \mid t) \qquad (2.7)$$

Given a source target phrase pair (s, t), we use uniform linear interpolation as shown in equation (2.7)to scale all the feature values.

Algorithm 3 explains the process of using CONDOR [Thain et al., 2005] to find the best interpolation co-efficient for a given direct and triangulated model. Note that the

---

Algorithm 3: Grid Search for Interpolation

Input: triangulated phrase table p_t,
direct phrase table p_d,
λ_d, $\lambda_t = 1 - \lambda_d$, $\text{prev}_{bleu} = 0$,
$\text{minimum} = e^{-2}$
Output: best_{λ_d}

  1: **while** $\delta_{bleu} >$ minimum **do**
  2:     interpolate p_d, p_t to give p_{interp}
  3:     Run MERT using p_{interp} as translation model
  4:     find $\text{bleu}_{heldout}$
  5:     $\delta_{bleu} = \text{bleu}_{heldout}$ - prev_{bleu}
  6:     $\text{prev}_{bleu} = \text{bleu}_{heldout}$
  7:     Based on δ_{bleu}, find $\text{new}\lambda_d$
  8: **end while**

---

approach can be easily extended to multiple triangulated models. Line 2 interpolates the two translation models using equation (2.7). We re-tune the log-linear weights using MERT for the interpolated feature values and use the tuned model to find BLEU score on the heldout set. Based on the difference between the BLEU score obtained and the previous BLEU (line 7), CONDOR searches for the new co-efficient in the corresponding direction. The search will culminate when consecutive BLEU scores show a marginal difference (Line 1). For instance, we start with a value of 0.85 for the direct system from Mawukakan to English we obtain a BLEU score of 9.10. If we use uniform weights for both the tables, we get BLEU scores on heldout as shown in Table 3.7. In three of four cases, we would not have out-performed our baseline. We can try 0.50, 0.60, 0.70 and 0.80 [Nakov and Ng, 2012] and run MERT for each choice. Although 0.70 would have given us our best BLEU for this pair, we observed that different languages led to different interpolation weights (Table 2.4), and this was different for different design choices for each language pair (Haitian Kreyol and Malagasy have disjoint systems). Our method automates grid search for the mixture weight and combines it with minimum error rate training of the log linear models for both direct and triangulated systems.

2.4 Summary

In this chapter, we discuss the approach of triangulation and how it helps in introducing new phrase pairs. We also discuss various models that affect the performance of the triangulated system, top-n filtering, adding connectivity features, using IBM Model 1 alignments and

Language	Best λ_d
Mawukakan	0.84
Maninkakan	0.75
Haitian Kreyol	0.95
Malagasy	0.82

Table 2.4: Different languages have different interpolation co-efficients that lead to the best system. Although we always start with 0.85, we iterate systematically over different values to obtain the best co-efficient.

computing the joint distribution for phrase pairs. Finally, we discuss the importance of using insights from Domain Adaptation to learn the relevant weights for the direct and translation models with the goal of maximizing BLEU score on a given heldout set.

Chapter 3

Triangulation for Very Low-Resource Languages

3.1 Four Very Low-Resource Languages

Faced with a low-resource language pair, several questions arise when combining a direct translation model with a triangulation model:

- In [Utiyama and Isahara, 2007] all possible triangulated phrases are used, even very unlikely ones. We show that it is better to eliminate unlikely triangulated phrases.

- In [Utiyama and Isahara, 2007, Cohn and Lapata, 2007, Wang et al., 2012, Wu and Wang, 2007] many different feature functions are provided for the log-linear model over triangulated phrase pairs. We conduct extensive experiments to show which features should be used for real world low-resource languages based on the data settings for each language pair.

- In [Cohn and Lapata, 2007] a mixture of the direct system and the triangulated system is shown to work better. However, they used uniform weights. In [Wang et al., 2012] a few different weights were selected heuristically while in [Wu and Wang, 2007], 0.9 is assumed for the baseline. We provide an algorithm that combines grid search for learning the mixture weights and minimum error rate training of the direct and triangulated log-linear models.
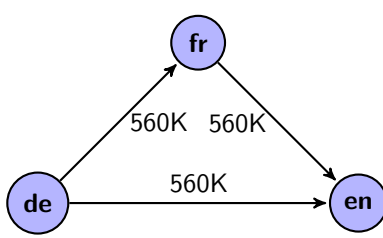
(a) Setting for [Utiyama and Isahara, 2007]
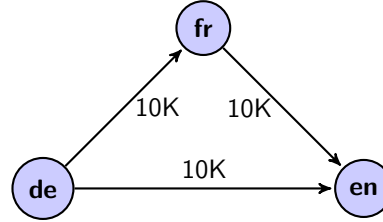
(b) Low-Resource Setting for [Cohn and Lapata, 2007]
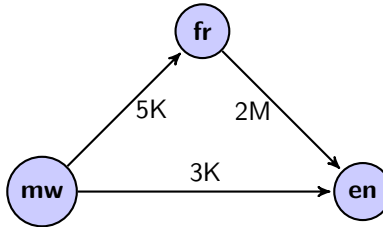
(a) Setting for Mawukakan
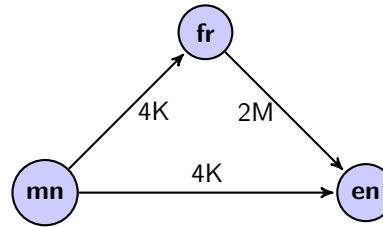
(b) Setting for Maninkakan

Figure 3.2: Comparison of our low-resource scenario with triangulation for Europarl. In our setting, the source pivot corpus is quite constrained, thus, limiting the fan-out for triangulation

To answer some of the above questions, we study the effectiveness of pivot-based triangulation for languages with insufficient resources, Mawukakan, Maninkakan, Malagasy and Haitian Kreyol. Figure 3.2 compares our data settings with previous research into triangulation. Mawukakan and Maninkakan are two languages from the Mandekan family, spoken by almost 3.5 million people in West Africa. The Mandekan languages are a part of the Niger-Congo language family. Maninkakan and Mawukakan have little writing tradition, are written using multiple alphabets [1] and have very little resources for Machine Translation. Malagasy is the national language of Madagascar, spoken by 18 million people worldwide. Haitian Kreyol is the national language of the Republic of Haiti and data used is from the Sixth Workshop on Machine translation, 2011 [Callison-Burch et al., 2011]. It comprises short messages sent to the number 4636 after the devastating earthquake in January, 2010. Although nine systems participated in the workshop on Haitian Kreyol, the approach of triangulation was not used. To our best knowledge, this is the first in-depth study of triangulation in a real-world low-resource setting and also the first for the four languages mentioned above. Mawukakan, Maninkakan and Malagasy do not have publicly available SMT systems.

---

[1] data we have used has Latin script, obtained via LDC

In the aftermath of the earthquake in Haiti in January, 2010, Mission 4636 set up a service where anyone in Haiti could send a message for free to a phone number 4636[2]. A group of volunteers translated the messages into English and helped the relief organizations provide swift help to the affected masses. Microsoft Research released a translation system to the public, for Haitian Creole, 5 days after the devastating earthquake [Lewis and Munro, 2011]. The fast turnaround time[3] and the usefulness of Machine Translation in the time of crisis inspired the featured task in the 6th Workshop on Statistical Machine Translation. Although Haitian Kreyol is a French-based Creole, the approach of inducing a Haitian Kreyol to English phrase table by pivoting via French was not used.

Malagasy is an Austronesian language and the national language of Madagascar, spoken by 18 million around the world. Although it shares several words with Ma'anyan, it has influences from Arabic, French, Swahili and Bantu. Characters can have diacritics but not always. Numbers are written right-to-left like Arabic, while some words are in common with French. It follows the Latin alphabet but with 21 characters. Finally, the dataset we have is real-world news articles translated by volunteers across the world[4] and aligned using a sentence aligner, thus, introducing some inconsistencies.

Mawukakan[5] and Maninkakan[6] are two of the four languages of the Mandekan family. They have no writing tradition, are spoken by a few million people around the world and are unique in several ways. Several characters have diacritics but they can have different stress depending on the nearby words. The lengths of sentences are relatively longer when compared to English. By using triangulation and significantly improving the output translations, we hope to preserve the existing data and encourage more monolingual and parallel data production.

3.2 Datasets

All the source sentences in Mawukakan have both French and English translations. Not all sentences in Maninkakan have both translations. The numbers for each of the datasets are mentioned in Table 3.2. The training data for Haitian Kreyol is the same as released in the Workshop. Malagasy training data also has not been changed in any manner. Both Haitian

---

[2]*http://www.mission4636.org*

[3]*To know the exact timeline, refer to http://languagelog.ldc.upenn.edu/nll/?p=2068*

[4]http://www.ark.cs.cmu.edu/global-voices/

[5]http://catalog.ldc.upenn.edu/LDC2005L01

[6]http://catalog.ldc.upenn.edu/LDC2013L01

language	src/tgt
mawu	*à à f á nè kò búlàmá mùsò kwáò à yá wɛ́ɛ́ó lé é à mátá à*
	people say that she performs magic
manin	*àlù bárá álámándí bèn à kàn , kà à màs̀ ̀r sɛ́bɛ́ tɛ́ à lá mòrìfa lá*
	they fined him because his gun is not legally registered
ht	*j' aimerais avoir quelques informations svp , concernant ce numero 4636 en quoi je peux l' utiliser*
	i would like to have information regarding the number 4636. how do i use it
mlg	*takelaa facebook ho an ' i laura sy euna efa manana mpikambana maherin ' ny dumy arivo sahady*
	a facebook page for laura and euna already has more than five thousand members

Table 3.1: An example for each language: mawu = Mawukakan, manin= Maninkakan, ht = Haitian Kreyol, mlg = Malagasy

Kreyol and Malagasy have no parallel data with other languages except English. To use triangulation, we needed parallel data with one more language to use as a pivot.

To enable us to reach French phrases, we have used the Bible as our source pivot text for Haitian Kreyol and Malagasy. The Bible gives us 30K sentences of text that is relatively clean. To align the Bible in source languages and French, we used hunalign [Varga et al., 2005], a sentence aligner. No manual alignment was done. As a result of using Bible, our source pivot, pivot target and source target models are all trained on disjoint and unrelated domains for Haitian Kreyol and Malagasy. For Haitian Kreyol, we aim to improve translations for short messages using the Bible to reach French phrases present in parliamentary proceedings. For Malagasy, we aim to improve poorly aligned news articles using the Bible to reach the same French phrases. As shown in our results, we improve the translations for both over the target system.

3.2.1   Pre-processing

The English and French side of Mawukakan and Maninkakan parallel data sometimes have forward slashes separating equivalent English and French translations. For both, the feminine form was chosen. For instance, a sentence

```
he/she/it goes to school
```

was replaced by the English sentence

```
she goes to school
```

Text between square brackets was removed. As development, heldout and test sets are not separately released, the last 2000 sentences was used for development, heldout and test together, for both Mawukakan and Maninkakan. The top 1000 was kept aside for development, while 500 each was kept aside for heldout and test. The last 2000 sentences make up 40% of the total data for Mawukakan and 33% of the total data for Maninkakan. We kept aside a large percentage for development and testing to get a better idea about the difference between the various models.

Both Haitian Kreyol and Malagasy are tokenized using the French tokenizer that is part of the Moses toolkit while Mawukakan and Maninkakan are tokenized using the English tokenizer.

3.2.2 Development and evaluation data

For Haitian Kreyol, the same development, heldout and test data has been used as the Workshop on Machine Translation. For Malagasy, the development data has been used as-is. As there is no separate heldout set, we have used the top 500 sentences of the test data as heldout, keeping aside the rest as unseen test data.

We used 40% and 33% of total data for Mawukakan and Maninkakan respectively for development, heldout and test data. A larger proportion was kept aside to make sure evaluation can be done over a range of sentences. The distribution of the evaluation data is shown in Table 3.5. The development, heldout and test sets for Haitian Kreyol have *raw* and *clean* versions. The raw versions are the short messages sent as-is, while the clean versions are the same messages with some words corrected or removed, e.g caf* in raw is cafe in clean version.

In Table 3.4, we observe that even with a constrained source pivot in Europarl, where we only use the top 50K sentences as source pivot, the triangulated table still finds 1.3M rules common, which is much larger than we observe for the four languages shown. Having perfectly sentence aligned data which is also multi-parallel goes a long way in expanding the triangulated phrase table.

setting	src tgt	src pivot	pivot tgt	domains
[Utiyama and Isahara, 2007]	560K	560K	560K	multi-parallel
[Cohn and Lapata, 2007]	700K	700K	700K	multi-parallel
[Cohn and Lapata, 2007]	10K	10K	10K	multi-parallel
Mawukakan	3K	5K	2M	different
Maninkakan	4K	4K	2M	different
Haitian-Creole	120K	30K	2M	different
Malagasy	88K	30K	2M	different

Table 3.2: Comparison of the low-resource scenario with Europarl

Language	#baseline-rules	#triangulated-rules
Mawukakan	51066	1.3M
Maninkakan	60097	0.8M
Haitian Kreyol	4.8M	13.6M
Malagasy	5.5M	7.1M

Table 3.3: Number of phrase pairs before and after triangulation

System	#rules
mawu-fr-en	1.3M
manin-fr-en	0.8M
ht-fr-en	13.6M
mlg-fr-en	7.1M
de-fr-en	28.7M
fr-es-en	13.7M
es-fr-en	12.95M

Table 3.4: Comparison of triangulated phrase table sizes for Europarl(50K src pivot and 2M pivot tgt) and four languages we study

Language	dev	heldout	test
Mawukakan	1000	500	500
Maninkakan	1000	500	500
Haitian Kreyol	900	900	1274
Malagasy	1133	500	633

Table 3.5: Training, development, heldout and test sets for all 4 languages

System	d(cl)	d(r)	t(cl)	t(r)
just-ood	27.56	20.77	26.72	20.14
just-sms	32.85	29.15	32.09	27.56
full	33.52	29.76	*33.1*	28.19
full-bigLM	33.6	29.83	33.07	28.91

Table 3.6: Different baselines for Haitian Kreyol

3.3   Baselines

Broadly, the training data for Haitian Kreyol can be divided into 3 parts, *SMS*, *Out-of-domain* and *Wikipedia* with 16k, 88k and 17k parallel sentences respectively. We observe, as shown in Table 3.6 that only using the OOD data does not take us very far. Just using the 16.6K in-domain short messages leads to a better BLEU score than not using it. Using all of the data leads to the best baseline. The *bigLM* refers to an interpolated language model comprising the English side of Haitian Kreyol workshop data and the English side of Europarl. For all the other experiments, baseline-bigLM is the baseline and the same language model has been used throughout, for Haitian Kreyol.

The baseline BLEU score for all the four languages are reported on Table 3.7. Note that the baseline for Haitian Kreyol outperforms the best system from the Workshop.

3.4   Results

Despite using a disjoint and out-of-domain Bible as source pivot and Europarl as pivot target, both Haitian-Creole and Malagasy lead to a better BLEU score on using an interpolated model comprising the direct and triangulated model. As indicated in the example in section 2.3, words that were mistranslated earlier get the right translations after pivoting via a large and clean fr-en phrase table.

For both Mawukakan and Maninkakan, the BLEU scores show a more significant increase of 2.2 and 1.5 BLEU points respectively for the top-n interpolation model. As the training and source pivot corpora for both comprises commonly spoken sentences that are not very long, the English side of Europarl effectively augments the limited target side of the training corpus, thus, leading to better translations after interpolation.

Intuitively, the two connectivity features should penalize the spurious and less aligned phrases, thus, reducing the noise and rewarding the just translations. But, the effect is not observed in the BLEU scores. Except in the case of Haitian Kreyol where it improves by a

small margin, adding the two connectivity features reduces the BLEU score. This could be owing to the fact that the source pivot data is tiny and the intersection of the alignments with the clean Europarl alignments is leading to feature values that do not effectively discriminate between the good and bad. For instance, in Mawukakan and Maninkakan, 60% and 66% phrase pairs have a source connectivity strength of more than 0.5 while 67% and 69% have more than 0.5 in the backward direction. With feature values that show a more skewed distribution, the connectivity features are not helping distinguish good from bad.

Setting	Mawu	Manin	Haitian-Creole	Malagasy
Baseline	7.08	9.41	33.6	18.8
Uniform	9.35	10.50	33.44	18.51
top-n	9.29	10.91	33.84	19.17
top-n + Strength	9.17	10.80	33.92	19.03
Model 1	9.02	10.69	**34.00**	**19.20**
Joint	**9.62**	**11.06**	33.85	19.10

Table 3.7: Results for all languages: Uniform is interpolated model with uniform weights
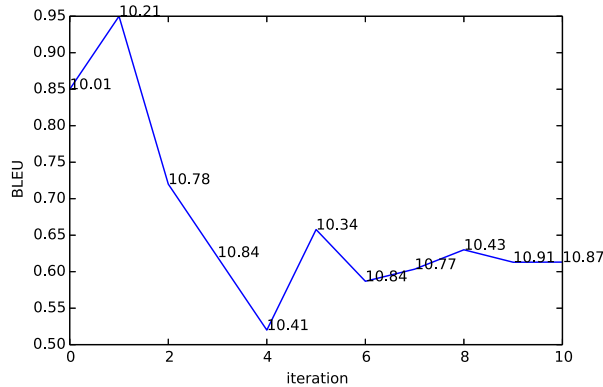
Figure 3.3: Grid search over interpolation co-effs leading to a best BLEU of 10.91 using $\lambda_d = 0.612962$

Although Model 1 helps in the case of Malagasy and Haitian Kreyol, they do not help in the other two languages.

Adding the joint and decomposed conditional distributions as features does well for Mawukakan and Maninkakan, leading to the best system for both. For Haitian Kreyol, we replicated [Cohn and Lapata, 2007] setting of using uniform weights for the joint probability and decomposing to find the conditional distributions, but adding the values to the log-linear

pipeline outperforms uniform interpolation of joint probabilities.

The advantages of using Grid Search over interpolation co-efficients is denoted by the BLEU scores on the row **Uniform**. Uniform is the interpolated model that uses uniform weights for both the direct and the top-n triangulated phrase tables. We observe that our best system outperforms the uniform system in all the four languages, although the improvements are slightly lower in the cases of Mawukakan and Maninkakan. Previously, the weights for the interpolation were not learnt systematically. They were assumed heuristically and the process was not iterative.

Language	baseline v/s best	uniform v/s best
Mawukakan	0.00	0.4
Maninkakan	0.00	0.4
Haitian Kreyol	0.07	0.02
Malagasy	0.08	0.01

Table 3.8: **baseline v/s best** indicates the p-value when the baseline system is compared to our best system;**uniform v/s best** indicates the p-value when an interpolated model with uniform weights is compared to our best system

3.4.1   Significance Testing

The final translations for a sentence are an output of a hypothesis. A hypothesis that, given the previous words translated with a score, assigns the highest likelihood to the output we see. But, this hypothesis is a result of an optimization algorithm (MERT) that uses a non-smoothed error count (BLEU) on a corpus level. To how the statistical significance of our results, we use **multeval** [Clark et al., 2011] to perform bootstrap resampling on our BLEU score hypotheses. We compare our baseline score in Table 3.7 to our best system for all four languages. In Table 3.8, we observe that our best systems for Mawukakan and Maninkakan are better than the baseline, while the other two systems have a p-value of more than 0.05, thus, indicating some instability in the best system. We find that our best systems for Mawukakan and Maninkakan are not significantly better than using uniform weights. This is not very surprising because our target domain is not different like short messages or news articles, they are simple sentences spoken in the real world about everyday happenings (i go to school, she married him). The source pivot and direct source target system is also extremely small. These factors enable the uniform model to have a comparable performance to our best system. For Haitian Kreyol and Malagasy, where our target domain is different

from Europarl and also our source pivot system, we find that our best system is significantly better than a uniform interpolated model.

3.5   Summary

In this chapter, we report our observations on using triangulation for four real-world low-resource languages with data settings that are significantly different and more constrained when compared to previous research into triangulation. We find that using triangulation helps but it is more helpful to make careful design choices, all of which provide small but consistent gains. We observe that a tiny source pivot corpus can still help in improving translations by using alternative target phrases from the English side of Europarl.

Chapter 4

Related Work

4.1 Triangulation

Consider a source language s, pivot language p and target language t. When using the *cascading* approach, we build two systems, between s and p and between p and t. Given a test set in s, it is first translated to p and those output translations are then translated into the target language t, making decoding twice as expensive as well. We do not report our results on using cascading for various reasons. Firstly, translating the output of a source pivot system trained and tuned on little data will lead to propogation of errors. Secondly, we will need three development sets, one for each system. Finding standard development sets for low-resource languages is unlikely. Finally, it has been shown before that cascading does not give the most fluent translations. [Utiyama and Isahara, 2007] compared pivot-based triangulation with cascading using all of multi-parallel Europarl, observing that pivot-based methods outperformed cascading.

The second approach is the pivot-based approach where a triangulated phrase table is generated between the source and target, by using the common pivot phrases between the source pivot and pivot target tables [Utiyama and Isahara, 2007, Cohn and Lapata, 2007, Wu and Wang, 2007]. [Utiyama and Isahara, 2007] observed that the triangulated table was able to achieve comparable BLEU scores to the direct system for French, German and Spanish. This could be owing to the fact that the data comprised multi-parallel 560K sentences. [Cohn and Lapata, 2007] observe that multiple pivot languages lead to more fluent translations compared to one pivot language. Multiple pivot language lead to multiple alternative translations, thus, increasing phrase coverage and rewarding the more appropriate translations and reducing out-of-vocabulary words further. They also propose a systematic way of combining the triangulated translation model with the direct model using

linear interpolation and log-linear interpolation, although they assume a uniform weight for both the models. To "simulate" a low-resource scenario, the top 10K multi-parallel sentences are considered for source pivot, pivot target and source target systems. As we will observe later, real low-resource scenarios are significantly different from how it was simulated in [Cohn and Lapata, 2007]. [Nakov and Ng, 2012] propose a language-independent approach to improving translation for low-resource languages, but the approach assumes the presence of a resource-rich language that bears similarity to the low-resource language, the similarities helping in creating a large triangulated phrase table. In [Wang et al., 2012], the resource-rich language is adapted to be more like the resource-poor one. Notice that this also assumes both are very similar. Results are reported using both Malay-Indonesian and Bulgarian-Macedonian, the third language being English in both cases. [Gispert and Mario, 2006] translate Catalan to Spanish via English by using news corpora on both source pivot and pivot target side. [Huck and Ney, 2012] report on BLEU score improvements by using 10^9 parallel sentence between German and French.

[Paul et al., 2009] evaluate the performance of Cascading and pivot-based approach over a range of different pivot languages. But, the languages used are not low-resource, and some European languages have also been used. [Costa-jussa et al., 2011] compare Cascading with a pseudo-corpus approach. In a pseudo-corpus approach, the pivot language is used to generate a noisy source target corpus. But, the dataset for evaluation is a Chinese-Spanish corpus with 3 languages as pivot. [Kholy et al., 2013] observe that using categories for source target pairs when combining the direct and triangulated models helped in improving the BLEU score. In other words, a source target pair can be in both the direct and triangulated phrase tables, or only one of them could be in both. They enumerate the different possibilities and use them as separate decoding paths. [Wu and Wang, 2007] also approach triangulation in a similar way to [Cohn and Lapata, 2007] but use different methods to compute lexical weights. They find improvements using linear interpolation and "simulate" low-resource by using small subsets of Europarl. [Zhu et al., 2013] try to address the problem of missing translations in triangulation (as pivot phrases are not always in both tables) by using a random walk approach. The initial triangulated phrase table is extended by treating the table as a graph and using a random walk to obtain more translations. [Crego et al., 2010] focus on improving one system (German-English) in their case by using a dynamically build model from auxiliary sources. In other words, they translate the source sentence using various models and then use a framework to combine the different outputs.

A common thread that binds the previous work using the approach of Triangulation is the usage of resource-rich languages. The fundamental reason behind the effectiveness of triangulation is the reduction in the number of OOVs when using the pivot language(s). This can be observed in various forms. If the source and pivot language have a healthy vocabulary overlap, the SMT system between source-pivot is large, thus, improving translations. This factor also helps when the amount of parallel text between source-pivot is relatively low, e.g, Indonesian-English. All the Europarl languages are based on parlimentary proceedings and have minimal noise. Hence, the improvements using triangulation over the direct systems cannot be generalized for systems for low-resource languages. All the papers that use triangulation in machine translation cite either [Utiyama and Isahara, 2007] or [Cohn and Lapata, 2007], both published in 2007 (and sometimes cite both of them but use either one model or the other). However, these two papers introduce triangulation for phrase-based SMT in very different ways and their models are different from each other. "Simulating" low-resource scenarios is ineffective in various ways. Firstly, real low-resource languages are noisy, not perfectly sentence aligned, and do not have a lot of data in the target domain. Secondly, triangulation is highly dependent on how good is the source pivot bitext. If the size of source pivot bitext is comparable to the source target, and/or is in the same domain, this increases bias in triangulation by introducing several common phrases, and, this is also not seen in a real low-resource setting. [Cettolo et al., 2011] build an Arabic-Italian system by using comparable documents and cascading via English as the pivot language to improve the translations.

[Hal Daumè, 2007] proposed domain adaptation by straight-forward duplication of features. [Bertoldi et al., 2008] suggested using alternative decoding paths when having different translation models. In our experiments, we found that alternative decoding paths did not work so well. This could be partly be because there are not that many alternatives when having two translation models of very different sizes and from different domains. When we do have alternative paths, they may not always be useful. Making trade-offs is a constant theme especially when facing low-resource languages. We want to maximize the good influence from the out-of-domain parlimentary proceedings but we do not want to undervalue translations that we get right already. [Cohn and Lapata, 2007] propose uniform linear interpolation and log-linear interpolation. Before interpolating, they compute the joint probability of phrase pairs and use the resulting conditionals in place of the scores computed in equations (2.1) to (2.4).

When working with low-resource languages, we realized that insights from domain adaptation help us in getting to the best possible translations. [Foster et al., 2007] do instance reweighting for out-of-domain phrase pairs and use it for linear interpolation of different translation models. The work has not yet been studied in the case of triangulation and it will be an interesting experiment to find out how the reweighting would fit and in which step. Out-of-domain phrase pairs in triangulation far outnumber the in-domain phrase pairs and we have also observed that doing a grid search over interpolation co-efficients works the best.

To our knowledge, before this dissertation, there has been no in-depth study of the different choices in building an SMT system using triangulation. Cascading and pivot-based triangulation have been compared [Utiyama and Isahara, 2007, Gispert and Mario, 2006], and a hybrid method has also been proposed [Wu and Wang, 2009] but results on using just pivot-based triangulation with real low-resource languages and comparing the different choices one needs to make has not been studied before.

4.1.1 Europarl

Europarl is short for European Parliament and refers to the multi-parallel corpora generated by translating the proceedings of European parliament into several languages. Version 7 of Europarl now has 20 languages, from French to Estonian and Finnish. Release of the Europarl corpus led to a surge in research into more and more data-driven methods to enable Statistical Machine Translation. The results were easily reproducible and the data is very clean and sentence-aligned. Each language has development and test sets which are used to report and reproduce results.

What does multi-parallel imply? Consider English as the common target language. A multi-parallel corpus between 20 European languages and English comprises sentences in 20 European languages which translate to the same English sentence. In Table 4.1, an English sentence and it's corresponding translations in French, Spanish and German are shown. These are the 10th sentence in each of the files. The German, French and Spanish sentences are the same English sentence. When using triangulation for a multi-parallel corpora like Europarl, one is effectively tracing different steps to the same target. This helps in expanding the resulting phrase table due to many common phrases.

To give some perspective, we will discuss some results using triangulation for Europarl in this section. [Utiyama and Isahara, 2007] used 560K multi-parallel sentences for their

Language	sentence
en	i would like to ask what ideas have been developed in this report .
de	ich mchte fragen , welche vorstellungen man in diesem bereich entwickelt hat .
es	quiero preguntar qu ideas se han desarrollado en este contexto .
fr	je voudrais demander quelles proposition on a mis au point dans ce domaine .

Table 4.1: Multi-parallel example: en = English, de = German, fr = French, es = Spanish

System	#sentences
direct	100K
src pivot	50K
pivot target	2M

Table 4.2: Our own data setting for Europarl triangulation

source pivot, pivot target and source target systems. [Cohn and Lapata, 2007] reported results using all 700K sentences and using 10K sentences each. It is but obvious that using all the available multi-parallel sentences for triangulation leads to results that cannot be generalized to a low-resource scenario. But, consider the setting of 10K each. Considering the top 10k sentences would mean that we have a low-resource source language with which we are using a low-resource pivot language too. Even though it's likely to deal with a low-resource language pair, it's ineffective to use triangulation by also assuming that the pivot language itself is low-resource. Moreover, we are not effectively using the knowledge we have on using large-resource languages like French and Spanish. Using a manually set poor Europarl language also necessitates the usage of multiple pivot languages to resolve OOVs, something that [Cohn and Lapata, 2007] found effective when using triangulation. We believe that considering the top 10K sentences does not fully mimic a low-resource scenario.

In the experiments below, we consider a constrained data setting similar to what we faced with Haitian Kreyol and Malagasy. The distribution of data is shown in Table 4.2

We consider top 100K sentences for our direct system while considering only top 50K of those for our source pivot system. This gives us some common words to use for triangulation but not all of 100K, which would have brought us on or above par with our direct system. On the other hand, we use all available data on our pivot target system, which is approximately 2 million lines of parliamentary proceedings of Europarl version 7. We report results on 3 languages, French (fr), Spanish (es) and German (de), all translating into English. All

System	BLEU
es-en	23.32
fr-en	19.53
de-en	15.60

Table 4.3: Baselines for our setting for all three languages

src-tgt	pivot	baseline	top20	top40	top60	top80	top100
de-en	fr	*15.60*	13.32	13.33	13.35	13.42	13.03
de-en	es	*15.60*	13.37	13.17	13.49	13.34	13.36
fr-en	de	*19.53*	16.21	15.82	15.89	16.08	15.95
fr-en	es	*19.53*	17.77	18.15	17.99	18.01	18.27
es-en	fr	*23.32*	21.35	21.18	20.83	21.01	21.45
es-en	de	*23.32*	18.36	19.19	19.35	19.23	18.97

Table 4.4: BLEU scores using just the triangulated phrase table, for n = 20 to n = 100

the results are using one pivot language. The language model used is a 5-gram Kneser-Ney interpolated language model using the English side of Europarl.

4.1.2   Results

The baseline models are trained on the top 100K sentences of fr-en, es-en and de-en. The baseline results are as shown in Table 4.3

Before we report our results on interpolation, let's find how far we can reach in terms of BLEU scores when only using the triangulated table. As the triangulated table uses multi-parallel corpora, intuitively, we should perform as well as the baseline at the least. Observe the results in Table 4.4

But, we observe a consistent drop in BLEU scores compared to the baseline. Our constrained source pivot means that the number of OOVs for the triangulated table are more than the OOVs for the direct system. With more OOVs, the BLEU score reduces due to the reduction in the modified unigram precision as discussed in the Introduction chapter.

The results using the top-20 interpolation method, top20 + strength model, Model 1 substitution and the Joint model are reported in Table 4.5. We observe that vanilla interpolation brings the BLEU score close or a little more than the baseline for all the language pairs. Adding the connectivity features, Model 1 scores or the Joint feature does

src-tgt	pivot	baseline	top20	strength	M1	joint
de-en	fr	15.60	15.45	15.62	15.24	15.52
de-en	es	15.60	15.55	15.55	15.49	15.61
fr-en	de	19.53	19.85	19.85	19.92	19.76
fr-en	es	19.53	19.86	19.90	20.03	19.66
es-en	fr	23.32	23.66	23.66	23.85	23.70
es-en	de	23.32	23.68	23.55	23.84	23.65

Table 4.5: BLEU scores using different interpolated models for Europarl

not make a discernible difference to the BLEU scores.

4.2   Summary

In this chapter, we compare our models with most of the previous work done into using the approach of triangulation. We also discuss some insights from Domain Adaptation that prove to be useful when combining translation models in triangulation. For the sake of comparison, we report results using previously discussed triangulation models by applying them on a simulated low-resource setting. We find that having a constrained source pivot in multi-parallel Europarl makes it more challenging to have the BLEU score improvements that have been reported in the literature before.

Chapter 5

Conclusion & Future Work

5.1 Conclusion

We started the dissertation by discussing the questions that come up when using the approach of triangulation in a low-resource language pair. In this dissertation, we have answered those questions.

We showed that it is better to eliminate unlikely triangulated phrases by limiting our fan-out limit.

Which model out of the product approach and joint model works better for phrase scores? We found that in a multi-parallel scenario, the joint model works better while in a disjoint scenario (like Haitian Kreyol), the product approach gave better BLEU scores.

Which model out of the product approach and IBM Model 1 works better for lexical scores? We found that IBM Model 1 works better each time. Having said that, we believe that the IBM Model 1 can be improved by changing the initialization from uniform to something that takes into consideration the actual number of alignment links between the phrase pair. We plan to address this in future work.

How best to interpolate the direct and triangulated models? We found that using the approach of grid search over model weights worked out the best. Besides comparing the interpolated model with our best baseline system, we also compared how we would have done had we taken uniform weights for both our models. And we found that we are significantly better ($p<0.05$) for Haitian Kreyol and Malagasy. This implies that when faced with a disjoint scenario, it helps to do grid search over the weights significantly. When faced with a data scenario that has some/all multi-parallel data with the source pivot, using uniform weights performs comparably (although worse in terms of absolute BLEU scores). Note that significance testing using p-values was computed by using approximate randomization [Clark

et al., 2011]. To get the complete picture, we would need to do human evaluation. Human evaluation is an expensive process although Mechanical Turk has been used before for other Natural Language Processing tasks.

To summarize, we observe that the approach of triangulation helps when used with low-resource languages. For two languages, we conduct the first study in SMT literature and report significant improvements by using triangulation. We also find that paying attention to design choices leads to small but consistent improvements. Finally, insights from domain adaptation help in learning the best ways of combining models from different corpora when using triangulation in low-resource scenarios.

5.2 Future Work

5.2.1 More Sophisticated Lexical Models

We used IBM Model 1 in place of lexical scores that were summed up over all pivot phrases. In future work, we would like to explore more sophisticated lexical models. For instance, we use *–grow-diag-final-and* as our alignment heuristic, which actually is quite conservative as it considers intersection of alignments in both directions and only adds unaligned phrases if they are part of the union of the alignments. We could only use the forward alignments of source pivot and intersect with the forward of pivot target, bypassing the backward alignments, and use the counts obtained from the forward ones to initialize Model 1. This will nullify some of the aggressiveness of *–grow-diag-final-and*

As the connectivity features and the Model 1 features are complementary, we can also change the initialization of Model 1 to reflect the word alignment score and the phrase-level connectivity. Using IBM Model 3 is trickier as its not very clear what fertility is optimum for the given low-resource languages.

5.2.2 Using Hierarchical phrase-based SMT

In all the experiments, we used phrase-based SMT systems. In hierarchical phrase-based systems [Chiang, 2007], we can translate phrases with gaps. This generates a lot more phrase rules while also being able to translate phrases with longer reorderings. Hiero systems have not been used before for low-resource triangulation and it will be interesting to see if more possible rules with gaps lead to better translations.

5.2.3 Considering Sub-phrases

A restriction of pivot-based triangulation is the dependence upon exact matches of pivot phrases. A French translation "le pivot el muli" has to be the exact same phrase on the French-English table to be used for the triangulated table. An alternative would be to consider sub-phrases.

Consider a source pivot pair,

h_1 h_2 h_3 ||| a_1 a_2 a_3

Now say that a_1 a_2 a_3 does not exist in the pivot target table. The source phrase will not end up with any target phrases from triangulation. But, assume that a_2 a_3 exists in the pivot target table having 55 translations. If we could align h_1 h_2 h_3 to all those target phrases with corresponding scores computed as before, we have new rules that would not have been if we had not taken sub-phrases.

It's ambiguous to consider any source phrase. Instead of considering any sub-phrase, we might perform better by considering only consecutive words that make up sub-phrases. Also, on the source side, instead of considering the whole sub-phrase, it will be more realistic to only consider words on the source side that are actually aligned to the sub-phrase we are triangulating with.

Considering sub-phrases is an even more interesting challenge when using hierarchical phrase-based systems, which have synchronous context-free grammars as rules, like below:

$$IX_1X_2|||toX_1X_2 \tag{5.1}$$

X_1 and X_2 are known as "gaps", essentially they are gaps that can be filled by other phrases. But, there are constraints on how the non-terminals can be positioned. Firstly, there can only be 2 on the source side. As the pivot of source pivot is the source of pivot target, we can essentially only triangulate with rules having 2 non-terminals. When trying to consider sub-phrases, we will have to consider how to deal with non-terminals. For "to X_1 X_2", do we consider "to X_2" as a pivot? Or "to X_1"?

If we assume either of them as pivots, what about the non-terminals on the source and target side? We will also need a new feature function which models the sub-phrases themselves and their connectivity, something like the connectivity features used above but with non-terminals.

5.2.4   Faster Parameter Learning

In all the interpolation experiments, we are learning one parameter (best weight for the direct system). When using more translation models, the approach of using CONDOR is not scalable. We run an iteration of MERT to completion for each co-efficient and it will be ideal to have faster ways of achieving the best parameters. An approach like [Foster et al., 2007] of instance re-weighting for linear interpolation would be a good first attempt. The model has not been used for triangulation before and it will be interesting to see if we can use a similar approach to re-weight the phrase pairs from baseline while also giving importance to the right phrase pairs from triangulated table. There is a ratio mismatch in triangulation for low-resource languages because of the tiny size of the direct phrase tables. Hence, when doing instance re-weighting, more than 95% of the instances will actually be from the triangulated phrase table. Thus, some changes will be needed to adapt to the skewed setting of the translation models.

Bibliography

[Allen, 1998] Allen, J. (1998). Lexical variation in Haitian Creole and orthographic issues for Machine Translation (MT) and Optical Character Recognition (OCR) applications. Assocation for Machine Translation in the Americas.

[Banchs and Costa-jussa', 2011] Banchs, R. and Costa-jussa', M. (2011). A semantic feature for statistical machine translation. Fifth Workshop on Syntax, Semantics and Structure in Statistical Translation.

[Bertoldi et al., 2008] Bertoldi, N., Barbaiani, M., Federico, M., and Cattoni, R. (2008). Phrase-Based Statistical Machine Translation with Pivot Languages. International Workshop on Spoken Language Translation.

[Brown et al., 1993] Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., and Mercer, R. L. (1993). The mathematics of statistical machine translation: Parameter estimation. *Comput. Linguist.*, 19(2):263–311.

[Callison-Burch et al., 2011] Callison-Burch, C., Koehn, P., Monz, C., and Zaidan, O. F. (2011). Findings of the 2011 Workshop on Statistical Machine Translation. Sixth Workshop on Statistical Machine Translation.

[Cettolo et al., 2011] Cettolo, M., Bertoldi, N., and Federico, M. (2011). Bootstrapping arabic-italian smt through comparable texts and pivot translation. In *Proceedings of the 15th International Conference of the European Association of Machine Translation*, Leuven, Belgium. EAMT 2011 Organizing Committee.

[Chiang, 2007] Chiang, D. (2007). Hierarchical phrase-based translation. Computational Linguistics.

[Clark et al., 2011] Clark, J., Dyer, C., Lavie, A., and Smith, N. A. (2011). Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. Association of Computational Linguistics.

[Clifton, 2012] Clifton, A. (2012). Mo' languages, mo' mt problems: Statistical language machine translation beyond a single language pair.

[Cohn and Lapata, 2007] Cohn, T. and Lapata, M. (2007). Machine Translation by Triangulation: Making Effective Use of Multi-Parallel Corpora. Association of Computational Linguistics.

[Costa-jussa et al., 2011] Costa-jussa, M. R., Henriquez, C., and Banchs, R. E. (2011). Enhancing scarce-resource language translation through pivot combinations. In *Proceedings of the 5th International Joint Conference on Natural Language Processing*, Stroudsburg, PA, USA. Association of Computational Linguistics.

[Crego et al., 2010] Crego, J. M., Max, A., and Yvon, F. (2010). Local lexical adaptation in machine translation through triangulation: Smt helping smt. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 232–240, Beijing, China. Coling 2010 Organizing Committee.

[Dempster et al., 1977] Dempster, A., Laird, N., and Rubin, D. (1977). Maximum Likelihood from incomplete data via the EM algorithm. volume 39. Journal Royal Statistics Society.

[Eidelman et al., 2011] Eidelman, V., Hollingshead, K., and Resnik, P. (2011). Noisy SMS Machine Translation in Low-Density Languages. Sixth Workshop on Statistical Machine Translation.

[Foster et al., 2007] Foster, G., Goutte, C., and Kuhn, R. (2007). Discriminative instance weighting for domain adaptation in statistical machine translation. Empirical Methods in Natural Language Processing.

[Galley et al., 2004] Galley, M., Hopkins, M., Knight, K., and Marcu, D. (2004). What's in a translation rule? North American Association of Computational Linguistics.

[Gao and Vogel, 2011] Gao, Q. and Vogel, S. (2011). Corpus expansion for statistical machine translation with semantic role label substitution rules. Association of Computational Linguistics.

[Gispert and Mario, 2006] Gispert, A. D. and Mario, J. B. (2006). Statistical machine translation without parallel corpus: bridging through Spanish. pages 65–68. 5th International Conference on Language Resources and Evaluation.

[Hal Daumè, 2007] Hal Daumè, I. (2007). Frustatingly easy domain adaptation. Association of Computational Linguistics.

[Hardmeier et al., 2011] Hardmeier, C., Jrg Tiedemann, Markus Saers, M. F., and Prashant, M. (2011). The Uppsala-FBK systems at WMT 2011. Sixth Workshop on Statistical Machine Translation.

[Heafield, 2011] Heafield, K. (2011). KenLM: Faster and smaller language model queries. Sixth Workshop on Statistical Machine Translation.

[Hewavitharana et al., 2011] Hewavitharana, S., Bach, N., Gao, Q., Ambati, V., and Vogel, S. (2011). CMU Haitian Creole-English Translation System for WMT 2011. Sixth Workshop on Statistical Machine Translation.

[Huck and Ney, 2012] Huck, M. and Ney, H. (2012). Pivot Lightly-Supervised Training for Statistical Machine Translation. Association for Machine Translation in the Americas.

[Kholy and Habash, 2013] Kholy, A. E. and Habash, N. (2013). Language Independent Connectivity Strength Features for Pivot Translation . Association of Computational Linguistics.

[Kholy et al., 2013] Kholy, A. E., Habash, N., Leusch, G., Matusov, E., and Sawaf, H. (2013). Selective combination of pivot and direct statistical machine translation models. In *International Joint Conference on Natural Language Processing*, Stroudsburg, PA, USA. Association of Computational Linguistics.

[Knight and Koehn, 2003] Knight, K. and Koehn, P. (2003). What's new in statistical machine translation.

[Koehn et al., 2007] Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertold, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. Association of Computational Linguistics.

[Koehn et al., 2003] Koehn, P., Och, F. J., and Marcu, D. (2003). 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 - Volume 1*, NAACL '03, pages 48–54, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Lefebvre, 1998] Lefebvre, C. (1998). Creole Genesis and the Acquisition of Grammar: The case of Haitian-Creole. Cambridge University Press, Cambridge, England.

[Lewis and Munro, 2011] Lewis, W. and Munro, R. (2011). Crisis MT: Developing a Cookbook for MT in Crisis Situations. Sixth Workshop on Statistical Machine Translation.

[Lopez, 2007] Lopez, A. (2007). A survey of statistical machine translation. Technical report.

[Nakov and Ng, 2012] Nakov, P. and Ng, H. T. (2012). Improving Statistical Machine Translation for a Resource-Poor Language Using Related Resource-Rich Languages. Journal of Artificial Intelligence Research.

[Och, 2003] Och, F. J. (2003). Minimum Error Rate Training in Statistical Machine Translation. Association of Computational Linguistics.

[Och and Ney, 2003] Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. Association of Computational Linguistics.

[Papineni et al., 2002] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a Method for Automatic Evaluation of Machine Translation. Association for Computational Linguistics.

[Paul et al., 2009] Paul, M., Yamamoto, H., Sumita, E., and Nakamura, S. (2009). On the importance of pivot language selection for statistical machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, NAACL-Short '09, pages 221–224, Stroudsburg, PA, USA. Association for Computational Linguistics.

[R.Costa-jussa and Banchs, 2011] R.Costa-jussa, M. and Banchs, R. (2011). The BM-I2R Haitian-Creole-to-English translation system description for the WMT 2011 evaluation campaign. Sixth Workshop on Statistical Machine Translation.

[Saers et al., 2010] Saers, M., Nivre, J., and Wu, D. (2010). Word alignment with stochastic bracketing linear inversion transduction grammar. North American Chapter of the Association of Computational Linguistics.

[Stolcke, 2002] Stolcke, A. (2002). SRILM - An extensible language modeling toolkit. International Conference on Spoken Language Processing.

[Stymne, 2011] Stymne, S. (2011). Spell Checking Techniques for Replacement of Unknown Words and Data Cleaning for Haitian Creole SMS Translation. Sixth Workshop on Statistical Machine Translation.

[Thain et al., 2005] Thain, D., Tannenbaum, T., and Livny, M. (2005). Distributed computing in practice: the condor experience. *Concurrency - Practice and Experience*, 17(2-4):323–356.

[Utiyama and Isahara, 2007] Utiyama, M. and Isahara, H. (2007). A Comparison of Pivot Methods for Phrase-based Statistical Machine Translation. North American Association of Computation Linguistics.

[Varga et al., 2005] Varga, D., Németh, L., Halácsy, P., Kornai, A., Trón, V., and Nagy, V. (2005). Parallel corpora for medium density languages. Recent Advances in Natural Language Processing.

[Vogel et al., 1996] Vogel, S., Ney, H., and Tillmann, C. (1996). HMM-based Word Alignment in Statistical Translation. 16th Conference on Computational Linguistics.

[Wang et al., 2012] Wang, P., Nakov, P., and Ng, H. T. (2012). Source Language Adaptation for Resource-Poor Machine Translation. Association for Computational Linguistics.

[Wu and Wang, 2007] Wu, H. and Wang, H. (2007). Pivot language approach for phrase-based statistical machine translation. In *Proceedings of the 45th Annual Meeting of Computational Linguistics*, Stroudsburg, PA, USA. Association of Computational Linguistics.

[Wu and Wang, 2009] Wu, H. and Wang, H. (2009). Revisiting Pivot Language Approach for Machine Translation. Association of Computational Linguistics.

[Zhu et al., 2013] Zhu, X., He, Z., Wu, H., Wang, H., Zhu, C., and Zhao, T. (2013). Improving pivot-based statistical machine translation using random walk. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 524–534, Seattle, Washington, USA. Association for Computational Linguistics.