

# Improving Phrase-Based Machine Translation

Alexandre Bouchard-Côté, John DeNero, Dan Gillick, James Zhang

December 20, 2005

## 1 Overview

Current state-of-the-art machine translation systems use a phrase-based scoring model for choosing among candidate translations in a target language, typically English. These models are deemed phrase-based because candidate sentence scores are in large part a product of phrase translation probabilities. These translation probabilities must be learned in some unsupervised manner from a pair of sentence-aligned corpora.

With the end goal of improving upon the published results of such systems, our project proceeded in two stages. First, we attempted to duplicate the performance results of existing end-to-end translation systems by piecing together available components and engineering the remainder guided by published techniques. Second, we identified two significant shortcomings of published systems and attempted to remedy them via machine learning techniques. In particular, we chose to learn phrase translation probabilities directly rather than deriving them heuristically. We also augmented the scoring model to relax a troublesome independence assumption across phrases.

## 2 Background

### 2.1 Phrase-based machine translation

A series of experiments from 1999 to 2002 showed that systems using multi-word phrases as basic units in machine translation outperformed those that strictly modeled word-level effects. Koehn et al. [4] provided a unifying framework for phrase-based translation and investigated several approaches.

For historical reasons, the task of translation is typically cast as translating some foreign sentence  $\mathbf{f}$  into an English sentence  $\mathbf{e}$ . Like the original word-level statistical translation models proposed and tested by Brown et al. [1], phrase-based systems are based on the noisy channel model for decoding messages. Using Bayes rule, the model is factored into a language model  $p(\mathbf{e})$  and translation model  $p(\mathbf{f}|\mathbf{e})$ , which are calculated independently.

$$\operatorname{argmax}_{\mathbf{e}} p(\mathbf{e}|\mathbf{f}) = \operatorname{argmax}_{\mathbf{e}} p(\mathbf{f}|\mathbf{e})p(\mathbf{e}) \quad (1)$$

The language model is typically a smoothed n-gram model identical to those found in speech recognition or other noisy channel problems, perhaps augmented by an additional factor to counteract n-gram models' bias toward shorter sentences.

The translation model for phrase-based systems is meant to capture how well the foreign sentence explains the English one. Through the set of independence assumptions below, the model

decomposes and simplifies into a product of phrase-level translation probabilities. Throughout the paper, we will adopt the following notational conventions when describing phrase-based models:  $\mathbf{e}$  is an English sentence,  $e_i$  is an English word,  $\bar{e}_i$  is an English phrase, and  $\bar{e}_1^I$  is a sequence of  $I$  phrases that comprise a sentence. Note that  $\bar{e}_1^I$  also represents a particular segmentation of a sentence  $\mathbf{e}$  into phrases. The following independence assumptions characterize a general family of translation models.

1. The English sentence  $\mathbf{e}$  is segmented into a sequence of  $I$  multi-word phrases  $\bar{e}_1^I$  according to a uniform probability distribution  $p(\bar{e}_1^I|\mathbf{e})$ .
2. The probability of each French phrase  $\bar{f}_i$  given a corresponding English phrase  $\bar{e}_i \in \{\bar{e}\}$  is independent.
3. The positional deviation of each foreign phrase from the position of the corresponding English phrase is either independent or a first-order Markov process dependent on the previous position.

Then, the translation model decomposes into

$$p(\mathbf{f}|\mathbf{e}) = \sum_{\bar{f}_1^I \rightarrow \mathbf{f}, \bar{e}_1^I \rightarrow \mathbf{e}} p(\bar{f}_1^I|\bar{e}_1^I) = \sum_{\bar{f}_1^I \rightarrow \mathbf{f}, \bar{e}_1^I \rightarrow \mathbf{e}} \prod_{i=1}^I \phi(\bar{f}_i|\bar{e}_i) d(a_i) \quad (2)$$

where  $\phi(\bar{f}_i|\bar{e}_i)$  is parameterized by a table of phrase translation probabilities for each phrase pair and  $d(a_i)$  is a distortion parameter that can take multiple simple forms, such as  $d(a_i) = \alpha^{|a_i - i| \frac{|\bar{f}_1^I|}{|\bar{e}_1^I|}}$ .

Several proposals for generating a table of phrase translation probabilities  $\phi(\bar{f}_i|\bar{e}_i)$  appear in the literature, the most successful of which were compared by Koehn et al. [4] using a common decoder, called Pharaoh [3]. They found that the best performing method was a particular heuristic approach to creating a table of these probabilities, counting all phrases consistent with a word-level alignment for each training sentence pair. We reproduced this method in phase 1 of our project.

## 2.2 Word alignment models

The top-performing heuristic for generating phrase translation probabilities leverages a word-level alignment between each pair of matching sentences  $\mathbf{e}$  and  $\mathbf{f}$  in the training corpus. Thus, before delving into the specifics of generating phrase translation probabilities, we shall review the unsupervised models for aligning words. This exposition will also lay the groundwork for our approach to learning phrase-level alignments in phase 2 of our project.

The word alignment models of Brown et al. [1] (IBM models) also assumed a noisy channel decomposition for machine translation. As the name suggests, the word-level translation model decomposes into word translation probabilities and distortion parameters. These models are generative models of sentence pairs, describing the process of generating a foreign sentences from an English one. Several models are proposed that increase in both complexity and training difficulty. For instance, to capture the phenomenon that certain English translate into multiple foreign words, the more sophisticated IBM models 3, 4 and 5 explicitly model the fertility of each English word: a distribution over how many foreign words align to it. The opposite phenomenon – a foreign word translating into multiple English words – is consistently handled rather naively as a null English word generating foreign words.

While a full exposition of all five IBM models would be peripheral to this paper, we shall inspect one in particular, IBM model 4, which is used as a component to our system. The generative process of a foreign sentence given an English sentence consists of three steps. First, Each English word  $e_i$  selects a fertility set  $k_i$ , a set of positions in the foreign sentence to generate, according to the product of a fertility and a distortion model. Then, each word  $e_i$ , now endowed with fertility  $k_i$ , generates a foreign word in each foreign position designated by  $k_i$ . Note that in addition to the English words actually present, a null English word is assumed that generates foreign words which do not align to anything in the English sentence. The fertility of this null word  $e_0$  is denoted  $k_0$ . This generative process corresponds to the following probability model.

$$p(\mathbf{f}, \mathbf{a}|\mathbf{e}) = \prod_{i=0}^I p(k_i|e_i, k_{i-1}) \prod_{i=0}^I \prod_{j \in k_i} p(f_j|e_i) \quad (3)$$

This model is learned as an unsupervised problem through an approximate version of the EM algorithm.

IBM model 4 is important to our project in two contexts. First, it is the model used to learn word-level alignments for our training set (via a publicly available software package GIZA++). Second, we structure our learning process for phrase translation probabilities according to a generative model of similar structure.

### 2.3 Related work

Marcu and Wong [5] were the first to attempt to train a statistical model with phrase translation probabilities at its core. Previous research into phrase-based systems had proceeded by heuristically generating phrase translation probability tables from an underlying word alignment model, while Marcu and Wong trained a generative model with multi-word phrases as its basic unit of translation using approximate EM. In phase 2 of our project, we adopt a similar approach. Their model and training procedure both differ significantly from ours, but it is instructive to review their work and results to set the stage for our effort.

Unlike the IBM word-level models, the generative process assumed for their phrase-level model is non-directional. That is, it is a process for jointly generating both an English and foreign sentence without regard to which is a translation of which. In this way, it diverges from the noisy-channel structure of later phrase-based work. The joint generative process of two sentences proceeds as follows:

1. Generate a bag of concepts  $C$ .
2. For each concept  $c_i \in C$ , generate a pair of phrases  $(\bar{e}_i, \bar{f}_i)$  according to a distribution  $\phi(\bar{e}_i, \bar{f}_i)$ , where both  $\bar{e}_i$  and  $\bar{f}_i$  contain at least one word.
3. Order the phrases in each sentence so as to create two linear sequences according to some distortion model  $d(\bar{e}_i, \bar{f}_i)$ .

To simplify the corresponding probability model, we can identify  $c_i$  with  $(\bar{e}_i, \bar{f}_i)$  instead of treating each concept as a separate hidden variable. We arrive at the following joint model for sentence pairs and their corresponding concepts, from which we can marginalize out joint sentence pair probabilities.

$$p(\mathbf{e}, \mathbf{f}, C) = \prod_{(\bar{e}_i, \bar{f}_i) \in C} \phi(\bar{e}_i, \bar{f}_i) d(\bar{e}_i, \bar{f}_i) \quad (4)$$

Exact training of this model through EM is intractable, as will be the case with our model. Marcu and Wong circumvent this issue through two means: only considering high-frequency n-grams as possible phrases<sup>1</sup>, and approximating the soft counts of aligned phrases using some clever combinatorial tricks. In doing so, they relax the restriction that phrases be composed of *contiguous* sequences of words. The effect of this choice is not well understood.

This joint model of phrase-based translation can be converted into a conditional model, extracting phrase translation probabilities  $\phi(\bar{f}_i|\bar{e}_i)$ . The phrase translation probabilities learned through this model were tested by Koehn et al. [4] against heuristic approaches and found to slightly underperform the most successful heuristic approach, which we reimplemented in phase 1 of our project.

This first attempt at learning a phrase-based translation model both improved the state-of-the-art at the time and served to show that even approximate EM could yield strong results in a phrase-based model. However, two aspects of this model seemed to leave room for improvement: learning a joint model with concepts as hidden variables seemed less appropriate than a conditional model mirroring the noisy-channel approach, and the approximation techniques did not leverage any information from previously constructed word-level alignments. Our attempt to undertake the same task – learning phrase translation probabilities using a generative statistical model – differs on both of these dimensions.

## 3 Phase 1: Reproducing a Baseline Phrase-based System

### 3.1 Model components

Creating an end-to-end translation system that reproduces the state-of-the-art baseline involved piecing together a number of different components, most of which are publicly available. The structure of this composite system is outlined in figure 1. We will first describe each component, then examine in detail the phrase-based translation model we built ourselves.

#### 3.1.1 The Europarl Corpus

Statistical machine translation relies on parallel corpora – text that has been translated into multiple languages – to learn translation parameters. Parliamentary or other official text tends to serve our purposes well because legal constraints make for particularly literal translations, thus allowing for precise word alignments, so long as the translators don't get too bored. The Europarl corpus [2] was created by crawling the website for the proceedings of the European parliament, extracting parallel chunks of text, pre-processing the text (sentence splitting, standardizing format, etc.), and finally creating a mapping of sentences from one language to another (sentence alignment). The corpus has translations in 11 different languages, each consisting of around 20 million words, or 1 million sentences. For the time being, we limit ourselves to the standard French-English pairing, though we are aware that other language pairs present their own unique challenges.

#### 3.1.2 The Language Model

As a language model, we use a standard n-gram language modeling toolkit provided by SRI [8], which includes a variety of tools for smoothing, interpolation, and evaluation (data likelihood or perplexity). For our experiments, we trained a trigram language model which backs off to bigram

---

<sup>1</sup>they later use smoothing techniques to assign translation probabilities to rare phrases

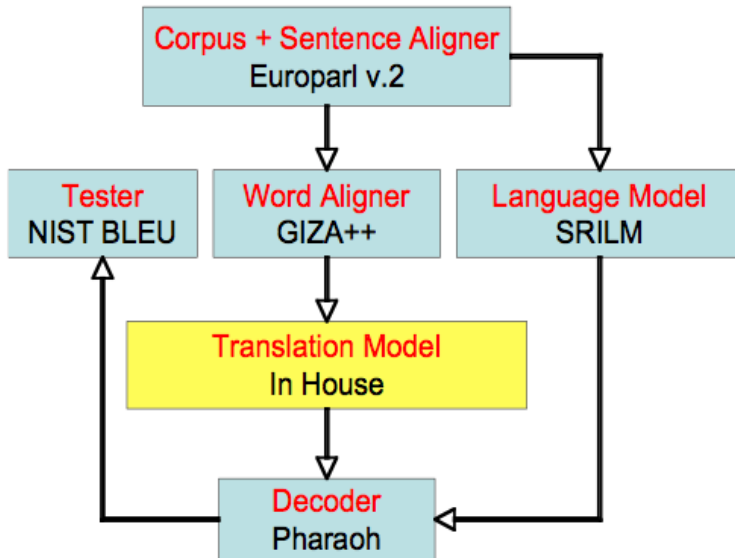


Figure 1: Structure of our end-to-end translation system.

or unigram probabilities in the absence of sufficient data. We tried a variety of backoff methods and found that Kneser-Ney backoff with absolute discounting, interpolating unigram, bigram, and trigram scores for all trigrams, gave lowest perplexity on test data. Kneser-Ney weights backoff by the number of unique contexts for the n-gram in question (so that when 'Don Francisco' backs off to 'Francisco', this gets a low score even though 'Francisco' is a common unigram). Discounting is a general smoothing technique that reserves probability mass for unseen n-grams by subtracting a little mass from each observed n-gram. This is reasonable, especially for n-grams that appeared only a few times in the training data, because a word that appeared once, say, in a million words, probably deserved a count of less than 1 in a million – it was a bit of an accident that it appeared at all. While such discounts can be learned from held-out data or via the Good-Turing method, absolute discounting takes advantage of the observation that – with the exception of the n-grams with observed counts of one – the appropriate discount is about a constant 0.75.

To replicate previous experiments, we created a language model trained on Europarl data. We also spent some time building a much larger language model trained on the Gigawords corpus – some 2.5 billion words of English newswire text. We tried interpolating this model with our Europarl language model but were unable to improve our overall translation scores. We suspect that this is because our test data, taken from Europarl as well, is very well-matched to the Europarl language model, whereas if our test set came from another source, this interpolation would help.

### 3.1.3 Giza++ Word Alignments

We generate word-level alignments for sentence-aligned data using a publicly available tool called Giza++ [6, 7]. This software package generates word-level alignments for sentence pairs governed by IBM model 4. There are a variety of parameters that need to be set or tuned, and a few iterations

suggested some good empirical values. In particular, the null-alignment parameter, which corresponds to the frequency with which the model hypothesizes a word aligning to null (disappearing in translation), proved important to translation quality.

We hope to soon replace Giza++ with state-of-the-art word alignment systems built here at Berkeley.

### 3.1.4 Translation Model

The translation model generates a table of phrase translation probabilities from the output of the word aligner. No publicly available software currently exists for this component of the system. Instead, the literature describes several heuristic approaches. In the following section, we discuss in detail how phrase translation probabilities are extracted from a word-aligned corpus.

### 3.1.5 Pharaoh: A Beam-Search Decoder

Once our phrase table has been created, we use the publicly available decoder Pharaoh to translate sentences in one language into another. Pharaoh takes a French sentence, for example, and works left to right, hypothesizing all possible English phrases that explain the current French prefix. Pharaoh uses a beam-search algorithm, pruning all English explanations whose total probability fall below a given threshold. These probabilities are calculated via a weighted combination of the translation model score, the language model score, and the distortion model score, which penalizes translations that deviate from a monotonic translation.

### 3.1.6 Evaluation

Evaluation for machine translation is a long-standing problem. How do we measure the quality of a translation? There are a number of proposed metrics, but the most popular one, of late, is the BLEU score, which simply calculates the percentage of n-grams shared between the reference translation(s) and the generated translation. BLEU score has been showed to be correlated with human judgement and is simple and cheap, though a little theoretically dissatisfying. The development of this scoring method prompted a huge increase in productivity within the machine translation community.

The scores that we report in this paper are BLEU scores for the somewhat standard 4-gram case. While multiple reference translations (created by a number of different human translators) significantly improves BLEU score, we use a single reference translation (this is all that is available with Europarl). Baseline scores for translating French to English are in the range of 0.20 to 0.35 depending on parameter settings and training conditions. We believe that there is much room for improvement.

## 3.2 The Phrase-Based Translation Model

We implemented the heuristic algorithm described in [3, 4] to produce a phrase translation table used as input to the decoder (which finally outputs translations). Building our phrase table involves three steps:

1. Creating high-precision, bi-directional word-alignments using Giza++ output (IBM model 4)
2. Extracting aligned phrase pairs from these alignments

3. Estimating phrase-translation probabilities from the counts of phrase-alignments over the training set

### 3.2.1 Bi-directional Word Alignment

Giza creates an alignment in one direction, in our case, either from an English sentence to a French sentence, or the other way around. Word alignment is trained as a one-to-many mapping like this as a consequence of the generative fertility model. We can improve our alignment error rate (AER) by intersecting alignments in each direction, thus creating sparse high-precision alignments, and heuristically growing near the diagonal to fill in some of the gaps.

The exact heuristic used to grow alignments is ambiguous in the literature. We deferred to [ref. Koehn pharaoh manual] for the details. We begin with the intersected alignments and add additional word alignments according to two constraints:

- Only add new alignment points that are in the union of the two word alignments.
- Require that a new alignment point connects at least one previously unaligned word (so you can't add an alignment if both words in question are already aligned). This process is diagrammed in figure 2.

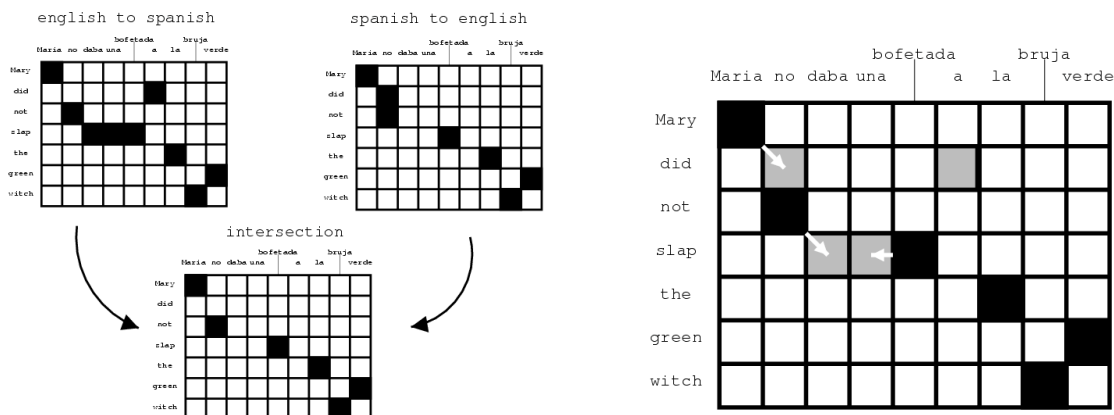


Figure 2: Intersecting and growing word alignments in two directions.

### 3.2.2 Extracting Phrases

Next, given a bi-directional alignment, we extract phrase pairs: one-to-one mappings of sequential words on one side to sequential words on the other side. Phrases can be conveniently represented by drawing rectangles on the word alignment grid.

The primary constraint for phrase pairs forces them to be self-contained: the aligned words must not be aligned to anything else outside the boundaries of the phrase. Beyond this intuitive definition, one can imagine a variety of subtleties. We consider two variations which we call the “enhanced phrase extractor” and the “null phrase extractor”.

The enhanced phrase extractor requires that there be alignments in all edges of the rectangle that represents the phrase pair. That is, the words that begin and end each phrase must align to some word in the phrase of the other language.

The null phrase extractor relaxes this constraint: the phrase pair rectangle need not have alignments on its edges. That is, phrases can begin and/or end with null alignments. This change dramatically increases the number of extracted phrases, especially give sparser alignments. The motivation here is extend the set of possible phrase pairs so that fewer valuable phrases are overlooked.

There are a few things to keep in mind about the phrases that are extracted:

1. Phrases often overlap. In particular this means that while we have many multi-word phrases, every aligned word pair becomes a phrase pair.
2. We use a maximum phrase length. Past research has shown that it is difficult to improve translations by using phrases longer than three words. While following this recommendation helps keep our phrase table to a reasonable size, we would like to experiment eventually with incorporating longer phrases as we see cases where they would almost certainly be valuable.

### 3.2.3 Phrase Translation Probabilities

Given the set of collected phrase pairs, we estimate the maximum likelihood phrase translation probability distribution by the relative frequency:

$$\phi(\bar{f}|\bar{e}) = \frac{\text{count}(\bar{f}, \bar{e})}{\sum_{\bar{f}} \text{count}(\bar{f}, \bar{e})} \quad (5)$$

As in the literature, no smoothing is performed, though it seems reasonable that incorporating some sort of discounting could improve performance. We have not tried this yet.

Sometimes, the probability associated with a phrase pair computed in this way suffers from training data sparsity. To improve the quality of these scores, we factor in a measure of how well the individual words in the phrase pair translate to each other. We implemented this technique, called Lexical Weighting, according to Koehn et al. [4]. First, we estimate lexical translation probabilities with the same relative frequency method described above for phrases:

$$w(f|e) = \frac{\text{count}(f, e)}{\sum_f \text{count}(f, e)} \quad (6)$$

A special English NULL token is added to each English sentence and aligned to all unaligned foreign words. Then, given a phrase pair  $(\bar{f}, \bar{e})$  and a word alignment  $a$  between the foreign word positions  $i = 1, \dots, n$  and the English word positions  $j = 0, 1, \dots, m$ , we compute the lexical weight  $p_w$  by

$$p_w(\bar{f}|\bar{e}, a) = \prod_{i=1}^n \frac{1}{|\{j | (i, j) \in a\}|} \sum_{\forall (i, j) \in a} w(f_i|e_j) \quad (7)$$

If there are multiple alignments  $a$  for a phrase pair, we take the one with the greatest lexical weight.

We combine the lexical weight  $p_w$  with our simple phrase pair probabilities to get new phrase translation scores:

$$p(\bar{f}|\bar{e}) = \phi(\bar{f}|\bar{e})p_w(\bar{f}|\bar{e})^\lambda \quad (8)$$

The parameter  $\lambda$  sets the strength of the lexical weight. We use 0.25 as suggested by Koehn et al. [4] and see BLEU score improvements as large as 0.01.

QUERY		Il est donc nécessaire d ' adopter des réglementations adéquates pour régir ces transports.
GUESS		It is necessary to take the appropriate rules to regulate the transport.
ANSWER		this makes it necessary to have proper rules governing transport of this kind.
QUERY		Mais nous constatons également que certaines pratiques sont désuètes.
GUESS		We also see that certain practices are entrenched.
ANSWER		But we also recognise that some of the practices are outdated.
QUERY		Nous nous trouvons à un tournant décisif dans l ' union.
GUESS		We are at a turning point in the european union.
ANSWER		We are at a watershed in the history of the european union.
QUERY		Monsieur le président , cette situation ne peut continuer et il faut y remédier.
GUESS		Mr president , this situation cannot continue and must be done.
ANSWER		Mr president , that situation cannot continue and needs remedying.

Figure 3: Randomly selected sentences translated by our system.

### 3.3 Results

Our efforts thus far have resulted in a system that achieves performance very close to the published standard. We expect that with further minor adjustments we will reach the BLEU score of 0.32. See figures 3 and 4 for details.

### 3.4 Future work

#### 3.4.1 Language Model Experiments

The history of language modeling is a frustrating one: one interesting idea after another has shown to decrease perplexity, and yet, almost nothing has helped the standard practical test – speech recognition. Speech recognition, another example of the noisy channel model, combines a language model and an acoustic model to generate text. Usually, the acoustic model is weighted 10 to 20 times more heavily than the language model, and tends to take care of most language ambiguities. Most grammatically implausible strings of words, for example, are ruled out implicitly by the acoustic model, leaving less responsibility for the language model. In machine translation, however, the translation model and the language model are typically given nearly equal weight, and yet, very few attempts at improving the language model beyond the standard set by speech recognition have been made.

When examining the errors from our translation system, we see that many of the strikingly poor translations seem to be caused by poor grammar. While sometimes it is clear that a more complex parse-tree structure is needed to clean up these issues, the problems often look like they should be corrected by a trigram language model. Since the phrase table is filled with mappings from common words to common words, the translation model often hypothesizes locally-ungrammatical trigrams. Since our language model backs off to lower-order n-grams, it will give unreasonably good scores to trigrams like “to the the” or “like to this”. For the purposes of translation, we need a language model that is capable of rejecting improper English as opposed to a model that just accepts well-formed English.

We plan to try reranking n-best lists of hypotheses with some different language models to see if this intuition is correct.

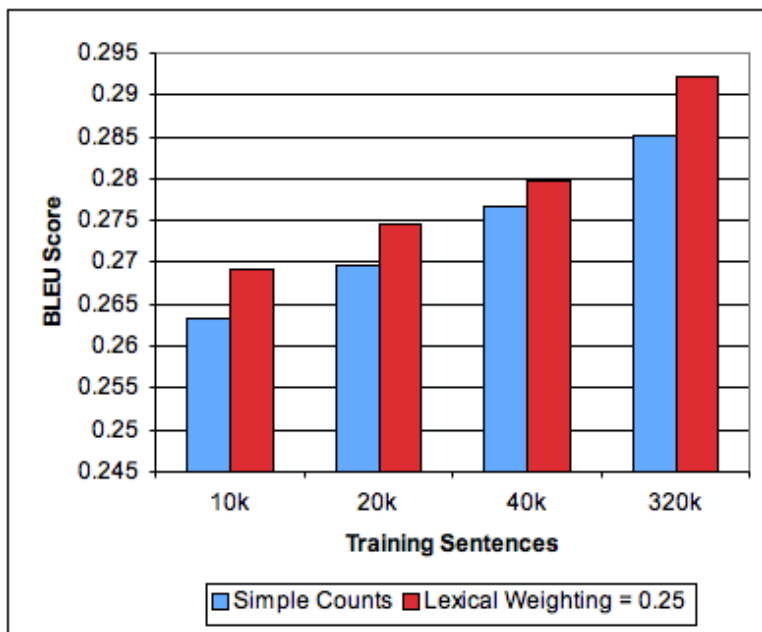


Figure 4: Results of Phase 1

## 4 Phase 2: Applications of Machine Learning

### 4.1 Learning phrase translation parameters

Much of the success of recent phrase-based machine translation systems can be attributed to clever decoding. The beam search decoder, Pharaoh, aggressively and cleverly prunes the search space of translation hypotheses given a set of model parameters. Learning parameters for the statistical model assumed by this decoder, however, has proven challenging indeed. At the heart of this overall model lies the phrase translation parameters,  $\phi(\bar{f}_i|\bar{e}_i)$ . However, the best translation results come from a fitting these parameters to the training data rather indirectly via word-level translations and a heuristic counting process. While in practice this approach outperforms the learned phrase translations from Marcu and Wong [5], this method appears to hold potential for improvement. One of our primary goals in this project is to substitute learned parameters for these heuristics.

The challenge with this project is that learning phrase translation parameters via the expectation-maximization algorithm is clearly intractable. Not only must we sum over exponential numbers of sentence segmentations and phrase alignments, but we also must store the distribution of  $\phi(\bar{t}_j|\bar{s}_i)$  for a daunting number of phrase pairs. In 100,000 sentences of the Europarl corpus, there are over 250 million phrase pairs that co-occur in aligned sentences given a maximum phrase length of only three words.

In the brief exposition of Marcu and Wong [5]’s work, we identified two areas of potential improvement over previous work. First, the generative model previously learned was a joint model without directionality, whereas the decoding process requires a conditional model. Second, the

approximate training approach left many alternatives unexamined. Previous work concentrated only on common n-grams, whereas we considered the space of all n-grams in the corpus, pruning only those that were disallowed by word-level alignments. We also found that the combinatorial methods used to approximate EM in previous work could instead be replaced by aggressively pruning the space of possible segmentations of a sentence into phrases. We implement this pruning approach by again using word-level alignments as a guide.

#### 4.1.1 A new generative model

While the Pharaoh decoder and corresponding general model for phrase-based translation do not assume a generative process, they certainly allow for a generative interpretation. The noisy channel model of machine translation views foreign sentences as generated from English sentences in some way. Furthermore, the phrase-based perspective assumes that this generation is done on a phrasal level, with each English phrase independently generating a foreign one. We wanted to learn from a model consistent with these assumptions. The generative process we chose to model produces a pair of sentences, English and foreign, where the latter is a translation of the first:

1. Generate an English sentence  $\mathbf{e}$ .
2. Segment  $\mathbf{e}$  into a sequence of  $I$  multi-word phrases that span the sentence,  $\bar{e}_1^I$ .
3. For each phrase  $\bar{e}_i \in \bar{e}_1^I$ , choose a corresponding position in the foreign sentence and establish the alignment  $a_j = i$ , then generate exactly one phrase in the foreign language  $\bar{f}_j$ .

The corresponding conditional statistical model for this generative process is:

$$P(\mathbf{f}, \bar{e}_1^I, \bar{f}_1^I, a | \mathbf{e}) = P(\bar{e}_1^I | \mathbf{e}) \prod_{\bar{f}_j \in \bar{f}_1^I} \phi(\bar{f}_j | \bar{e}_i) d(a_j = i | \mathbf{e}) \quad (9)$$

The parameters for each component of this model are estimated differently:

- The segmentation model  $p(\bar{e}_1^I | \mathbf{e})$  is assumed to be uniform over all possible segmentations for a sentence.<sup>2</sup>
- The phrase translation model  $\phi(\bar{f}_j | \bar{e}_i)$  is parameterized as a table of phrase translation probabilities, the parameters we want to learn.
- The distortion model  $d(a_j = i | \mathbf{e})$  is calculated via a parameterized discounting equation akin to the one used in IBM model 4.<sup>3</sup>

#### 4.1.2 Training

Given this generative model, we have a straightforward but thoroughly intractable application of EM: learning a sprawling set of phrase translation parameters by summing over exponentially many possible segmentations for each of several hundred thousand training sentences. Through a series

<sup>2</sup>Note that this segmentation model is deficient given a maximum phrase length. Many segmentations are disallowed because they violate the phrase length restriction.

<sup>3</sup>In the current implementation, we use a simplified distortion model based on absolute position of the sentences rather than their position relative to the previous phrase. This simplification (the difference between IBM models 3 and 4) is commonly made and will eventually be relaxed as we continue our research.

of assumptions, we can constrain this problem significantly, even to a worst-case polynomial time problem.

Our goal is to fit  $\phi(\bar{f}_j|\bar{e}_i)$  to best explain the training data.<sup>4</sup> To reestimate each of these, we extract counts from the data given the current model parameters.

$$\phi_{new}(\bar{f}_j|\bar{e}_i) = \frac{c(\bar{e}_i, \bar{f}_j)}{c(\bar{e}_i)} = \sum_{(\mathbf{e}, \mathbf{f})} \frac{\sum_{\bar{e}_1^I: \bar{e}_i \in \bar{e}_1^I} \sum_{\bar{f}_1^I: \bar{f}_j \in \bar{f}_1^I} \sum_{a: a_j=i} P(\mathbf{f}, \bar{e}_1^I, \bar{f}_1^I, a|\mathbf{e})}{\sum_{\bar{e}_1^I} \sum_{\bar{f}_1^I} \sum_a P(\mathbf{f}, \bar{e}_1^I, \bar{f}_1^I, a|\mathbf{e})} \quad (10)$$

Here we can explicitly see the sources of intractability for this problem. The number of phrase pairs is enormous, while summing over possible segmentations and alignments is exponential in time and space. Thus, pruning will be required to select which phrase pairs are of interest and which segmentations and alignments contain the majority of the probability mass. We use the information generated from word-level alignments to guide this pruning process.

First, we asserted that  $P(\bar{f}_j|\bar{e}_i) = 0$  for all phrase pairs not generated by a heuristic approach of extracting phrases from word-level alignments. We found the phrase extraction heuristic used in phase 1 (enhanced phrase extraction) to be too constricting, so we developed a more aggressive extraction heuristic we call null extraction. In this approach, we extract all phrases such that no word in either sentence aligns to a word outside of the phrase pair. Thus, an empty sentence alignment will allow all possible pairs of contiguous strings as phrase pairs in two aligned sentences, whereas the original heuristic would allow none. While this restriction ruled out very few apparently desirable phrases, it successfully reduced the total legal phrase pairs from approximately 250 million to 17 million for 100,000 training sentences.

Imposing a similar restriction at the sentence level has an even more important pruning effect. We considered only segmentations and cross-segmentation alignments that consisted of phrase pairs allowable by our null phrase extraction heuristic. That is, we allowed  $P(\mathbf{f}, \bar{e}_1^I, \bar{f}_1^I, a|\mathbf{e}) > 0$  only for tuples  $(\bar{e}_1^I, \bar{f}_1^I, a)$  such that  $a$  provided a one-to-one mapping from  $\bar{e}_1^I$  to  $\bar{f}_1^I$  where all phrase pairs  $(\bar{e}_{a_j}, \bar{f}_j)$  could be extracted from the word alignment for that particular sentence pair.

While in the worst case this restriction does not impose a tractable upper bound on the exponential running time of computing the required expectation, in practice it allowed us to compute the contribution of most sentences in under 10 seconds. Sentences with sparse word alignments still required several hours to process, however. All sentence pairs that either could not be explained under this restriction or required too long to process were dropped from the training set.

While in practice these restrictions allowed us to successfully train on 30k sentences in under 12 hours, we were left with an exponential time/space algorithm that would not function with sparse word alignments. A final restriction imposed upon the model did in fact yield a polynomial-time algorithm for computing the desired expectation. We assumed that all of the probability mass for  $P(\mathbf{f}, \bar{f}_1^I, \bar{e}_1^I, a|\mathbf{e})$  lay in segmentations with monotonic alignments. That is, for all  $j$ ,  $a_j = j$ . The phrases in the source and target must have the same order. Viewed alternatively, when moving along the source sentence generating target phrases, each newly generated target phrase must be appended to the end of the target sentence. We also considered extensions in which sentences were partially monotonic, consisting of two groups of phrases that were each internally monotonic. Tests relating to these assumptions showed that they did not match well with the content of the training set. Thus, for our preliminary results presented below, we used only the worst-case exponential-time algorithm.

---

<sup>4</sup>We have thus far limited our model to learning the parameters for the translation model,  $\phi(\bar{f}_j|\bar{e}_i)$ , leaving the distortion parameters fixed and the segmentation model uniform.

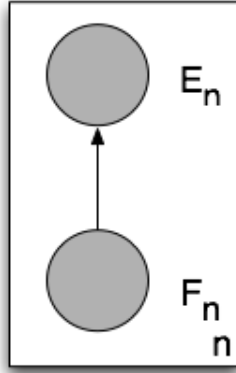


Figure 5: The baseline: a simplistic graphical model.

## 4.2 Using Contextual Information to Score Phrase Translation

Even with an improved translation model, the current phrase-based approach is hindered by its independence assumptions. In particular, phrases are translated independently of each other. Only the language model captures multi-phrase interactions.

In this section we approach a different problem, orthogonal to what we have discussed so far. We will consider an extension to the standard phrase-based approach that deviates from the noisy channel model. Instead of considering probabilities of translating french phrase given an english phrase, we consider *scores* of the phrase pairs, which do not need to be conditional probabilities of the foreign phrase given the English one. In particular, we will consider the “opposite” conditional probabilities, an English phrase given a foreign one. This is actually a very common practice in machine translation.

Let us again concentrate on the evaluation of the scores for the different phrase translations pairs,  $\phi(\bar{f}, \bar{e})$  and relax some independence assumptions. Note that in the Pharaoh system, these were computed using a very simplistic graphical model (refer to figure 5).

There is one such graphical model for each french phrase type  $f^*$ . The english translation is modeled by a random variable  $E_n$  with a range over the english phrase types and a multinomial distribution given the french phrase. Since  $F_n$  (the french phrase token that were seen to be aligned with  $E_n$ <sup>5</sup>) is always observed and  $E_n$  is observed during training, the MLE of the parameters  $\theta(\bar{e}|\bar{f})$  can be computed and used as a phrase translation score during decoding:

$$\hat{\theta}(\bar{e}|\bar{f}) = \frac{\sum_n \mathbb{1}[F_n = \bar{f}] \mathbb{1}[E_n = \bar{e}]}{\sum_n \mathbb{1}[F_n = \bar{f}]}$$

In order to understand the limitations of this model, consider the example in figure 6: the french

<sup>5</sup>Note that for a fixed  $f^*$ , the random variables  $F_n$  will always have the same value. This detour will make more sense in the next section when we will consider french phrase token in their context, in which case the values of the homologue of  $F_n$  will vary.

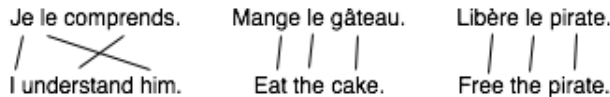


Figure 6: The french word “le” can be either a determiner, in which case it is translated by “the”, or a pronoun, in which case it is translated by “him”.

QUERY	I would like to explain our <i>thinking</i> here.
GUESS	Je tiens à défendre les <i>réfère</i> .
ANSWER	Je voudrais expliquer notre <i>pensée</i> .
QUERY	Le parlement doit <i>les</i> évaluer et devrait les rendre juridiquement contraignantes.
GUESS	Parliament has <i>the</i> evaluate and should be the legally binding.
ANSWER	They must be assessed by parliament and should be made legally binding.

Figure 7: Some sentences (QUERY, ANSWER) that the baseline system (GUESS) got wrong and that motivated this new approach for scoring phrases. “thinking” can indeed translate to “réfère”, but not in this context, in which “pensée” is a much better translation. Similarity, “les” can translate either to “the” or a pronoun, and the context must be used in order to disambiguate them.

word “le” can be either a determiner, in which case it is translated by “the”, or a pronoun, in which case it is translated by “him”. The MLE estimate for the parameters of the graphical model 5 is then  $\hat{\theta}(\text{“the”}|\text{“le”}) = \frac{2}{3}$  and  $\hat{\theta}(\text{“him”}|\text{“le”}) = \frac{1}{3}$ .

So this means that if the system is asked to translate the sentence “Je le déteste” (“I hate him”), it will assign higher score to the translation “him” for “le”, which is not what is required in this case. We may hope that this would be corrected by the n-gram language model, but it is easy to come up with examples that cannot be fixed this way (see figure 7).

As we will see, the proposed solution actually has the potential to solve many other types of errors, including:

- inflection errors where the source language has less inflection than the destination language,
- special case translations where the sentence is a question (e.g. wh-words, “est-ce que” in french),
- penalty for troublesome phrases (e.g. those that include punctuation).

### 4.3 Proposed Solution

In order to fix these errors, we will let the score of the phrase pairs depends on the *context* in which the french phrase occurred. The *context* includes the surrounding french sentence (it could even be extended to the entire french text), the french phrase type and its position in the sentence. For each french phrase type  $f^*$ , we maintain a graphical model with the structure depicted in figure 8. Note that the part in the plate corresponds to the graphical model of a discriminative classifier, thus, we will use the term “discriminative phrase scoring” for this part of the project.

The main differences with the previous model are the following:

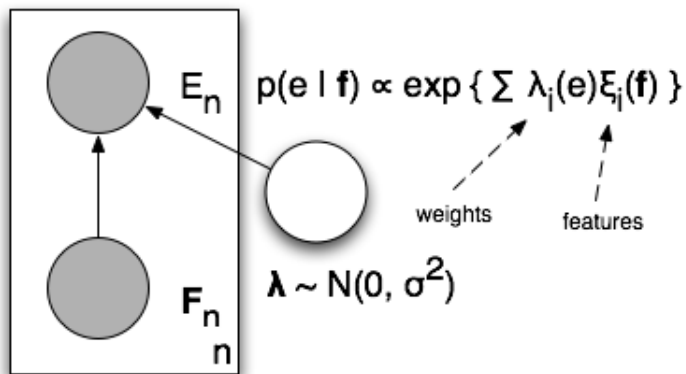


Figure 8: The new graphical model considered for scoring the phrase pairs. There is one such graphical model for each french phrase type  $f^*$ .  $E_n$  is the english phrase type,  $\mathbf{F}_n$  is the context of the french phrase token and  $\lambda$  is a prior on the parameters of the conditional distribution of  $E_n$  to avoid over-fitting.

- The random variables  $\mathbf{F}_n$  are defined on the context rather than on the french phrase only,
- the distribution of  $E_n$  is now log-linear and conditioned on features  $\{\xi_i\}$  extracted from the context,
- we introduced a gaussian prior over the parameters  $\{\lambda_i(e)\}$  to prevent over-fitting.

So we train one probabilistic discriminative classifier for each  $f^*$ . The scores for the different phrase pairs are then given by the conditional probabilities hence defined.

Since the system will have to deal with very large amounts of data, we decided to use a gradient ascent algorithm (LBFGS, for which we had a tested, scalable and efficient implementation) rather than IPF. If we forget about the prior on the parameters, the gradient of the likelihood is easily checked to be:

$$\frac{\partial l}{\partial \lambda_i(e)} = \sum_n \mathbb{1}[e_n = e] \xi(\mathbf{f}_n) - \sum_n P[E_n = e | \mathbf{F}_n = \mathbf{f}_n, \lambda] \xi_i(e_n).$$

When the gaussian prior is introduced and we compute the MAP instead of the MLE, the gradient becomes:

$$\frac{\partial l}{\partial \lambda_i(e)} = \sum_n \mathbb{1}[e_n = e] \xi(\mathbf{f}_n) - \sum_n P[E_n = e | \mathbf{F}_n = \mathbf{f}_n, \lambda] \xi_i(e_n) - \lambda_i / \sigma^2,$$

hence, the prior in this case can be described informally as a penalty imposed if the values of the parameters get too large.

In order to scale this method to the large corpora that we are using (on the order of hundreds of thousands of sentences), many clever algorithmic and engineering tricks had to be applied. This include, for instance, a sparse representation of the features vectors, “sloppy” operations on logs (to approximate  $\log(x + y)$  when we normalize), using a database to avoid heap overflow, etc.

QUERY	<i>Qu ' en est-il de la personne responsable de l ' erreur de calcul?</i>
GUESS 1	<i>That is in any of the person responsible for the mistake of calculation.</i>
GUESS 2	<i>What is in any of the person responsible for the mistake of calculation.</i>
ANSWER	<i>What happened to the person who was responsible for that miscalculation?</i>

Figure 9: Here, GUESS 1 is the original system, and GUESS 2 is the system using discriminative scoring and only two feature templates: the words in the phrase, and the indicator of a question mark in the sentence. Note that even though GUESS 2 still did not get the sentence entirely correct, the presence of the nonlocal question mark feature changed the score of “qu” → “what” from 0.0048 to 0.016 and consequently fixed this particular error in the sentence.

#### 4.4 Choosing the Right Features

Feature engineering is a crucial step for our system, as it is for many statistic nlp applications. Guided by our preliminary error analysis summarized in the previous sections, we decided to start with the following basic feature templates:

**Words in the phrase (ordered)** Probably the most important feature, these will always give the possibility to the system to achieve at least the data likelihood of the baseline. Note that if we set our prior properly, we can hope to do always at least as well as the original system.

**Part-Of-Speech (POS)** The parts-of-speech of the words in the french phrase token. It is hoped that it will help, for instance, to disambiguate homonyms that happen to have different translations.

**Questions** The presence of a question mark in the sentence.

These templates are used to create many binary features.

Clearly, we cannot assume that the training or test data are tagged with their POS. Hence, we need a subsystem that tags automatically our sentences. We decided to use a POS tagger that we built for assignment 3. Without going into the details, it is a MEMM using an extensive set of feature templates and achieving an accuracy of 96.5% (85.5% for unknown words). It was trained on the PENN treebank, a labeled training set of english sentences (this explains why some of our test go from english to french instead of the standard french to english).

#### 4.5 Preliminary Experiments

Due to the slowness of the current system (currently, one classifier must be trained for each allowed french phrase in the sentences that need to be translated), it was not possible to train on something large enough to guarantee a sensible BLEU evaluation. Instead, we decided to start with a few targeted sentences and look qualitatively at the results. Some of these sentences can be consulted in figure 9, 10.

After looking at the translations, we realized that even though our system is guaranteed to increase the likelihood of our data, it does not necessarily imply better translation results. We concluded that the main problem with our method is that we are using the same set of feature for every classification task and that for many french phrase types, there is not enough data to train the classifier properly. We think that this could be solved either using model selection techniques or by using stronger priors on the parameters for french phrase types with less training data. The second idea is discussed in more details and formalized in the next section.

QUERY	Parliament , but above all the member states , <i>demandé</i> massive improvements.
GUESS 1	Le parlement , mais surtout les états membres , <i>exigé</i> une amélioration.
GUESS 2	Parlement , mais surtout nos pays , <i>exigé</i> massive améliorations .
ANSWER	Le parlement , mais surtout les états membres , <i>ont exigé</i> massivement des améliorations .

Figure 10: Now, a more negative example. GUESS 1 is the same baseline as in figure 9, but this time GUESS 2 contains an additional feature template: POS of the words in the phrase. “Demanded” can be either the past (in which case a good translation is “ont exigé”) or past participle (in which case a good translation is “exigé”). The hope was that the discriminative scoring would fix the bad choice that was taken in this sentence. Unfortunately, it did not happened and the fluency of the overall translation slightly decreased. However, GUESS 2 was ran on a smaller training set (to speed things up) and it is hope that the future version of the discriminative scorer will be more scalable and eases the creations of better benchmarks.

## 4.6 Extensions and Future Directions

Our short-term goal is to improve the performance (both in time and space) of the system, so that we can run more extensive tests and evaluate objectively the validity of the method. Both engineering and algorithmic aspects of the system will have to be re-factored. For instance, it is clearly not a good idea to apply the probabilistic classifier to all phrases, for some of them do not occur often enough to justify the training of a probabilistic classifier. Moreover, some features are more expensive to extract than others (e.g. the POS features), and it might be possible to cast nicely this problem as a value of information/active learning problem, for which more and more theory and algorithms are available.

Next, we would like to add richer features, such as parse-tree-derived features (e.g. “SUBJECT-IS-PLURAL”, which would be motivated by the cases where we want to translate an english verb (mildly inflected) into a french verb (more inflected)). We also have an even more ambitious plan involving an automatic clustering system and the use of “PRESENCE-OF-CLUSTER-XX” features. Some words translate differently depending of the semantic context of the sentence or text in which they occur. This is especially interesting in cases such as web translation where we do not limit ourselves to testing the system on a set of sentences taken from the same corpus.

Another issue that we encountered with this method and that we mentioned in the last section is the sparsity of the training data. Indeed, even if the corpus is very large, the number of instances of a given phrase pair is generally small, and that may cause improper training in some cases (this will become more serious as we add more features—the more features, the more data we need). One way to fix this issue would be to adjust the values of  $\sigma^2$  (the parameter of our prior over the parameters). Right now, there is only one value of  $\sigma^2$  shared among all graphical models (recall that there is one graphical model like figure 8 for each french phrase type  $f^*$ ), and it seems more reasonable to have different values for  $\sigma^2$ , say  $\sigma_{f^*}^2$  depending on the abundance of examples for  $f^*$ . So we are working on a way to set those  $\sigma_{f^*}^2$  using a held-out set.

Finally, for the long-term, we are considering an application of transfer learning to train the classifiers jointly instead of all separately using a hierarchical model. We hope that all these improvements will increase both the BLEU score and the quality of the translations.

## 5 Results and Discussion

### 5.1 Monotonicity Constraints

When first formulating how to approximate the intractable e-step of our proposed training, we considered pruning assumptions that yielded a polynomial-time algorithm. Before settling on a training approach, we analyzed our training corpus to evaluate the feasibility of the various constraints that we wanted to impose on the EM training model. The two constraints we considered were the strict monotonicity of French phrases to English phrases and a relaxation of the monotonicity constraint that allowed for a single gap of monotonic phrases.

	Maximum Phrase Lengths		
	3	4	5
Monotonicity	42.6%	46.0%	49.5%
Single-Gap	46.4%	50.7%	54.6%

Table 1: Percentage of the 320k corpus that satisfies monotonicity and single-gap constraints

The result of this analysis was not encouraging. For a model with maximum phrase length of 3, over half of the corpus did not satisfy either of our constraints. This number did not significantly increase when we increased the maximum phrase size. Additionally, allowing for a single gap did not significantly increase the amount of the corpus that satisfies the model constraints. Therefore we decided to not use models that impose any monotonicity constraints for EM training. Thus, we resigned to using an exponential algorithm for the E-step.

### 5.2 Results

We performed several sets of experiments to evaluate our learned phrase translation model parameters  $\phi(\bar{\ell}_j|\bar{s}_i)$ . For all of these experiments we generated the phrase translation model parameters either by using a heuristic method, or by estimation using EM. After the phrase translation parameters have been generated, we generated English translations from a test set of 1000 French sentences. The generated English translations is then scored against the reference translations for the test sentences using the BLEU metric.

We used two heuristic methods for generating the phrase translation parameters. The first uses the enhanced phrase extractor with a maximum phrase size of 3 to produce the list of possible phrases, then we count the co-occurrences of the English phrases and the French phrases normalized by the counts of the English phrases. We tested a second heuristic method identical to the first except the possible phrases are extracted by the null phrase extractor. The only difference between the two phrase extraction heuristic is that the latter is more lenient at extracting phrases with null word alignments at both the beginning and the end of the phrase. Therefore the second heuristic extracts many more phrases than the first heuristic. The EM trained phrase translation parameters are initialized by the heuristically generated phrase translation parameter. We performed two EM sets, one for each heuristic method and both EM sets were run to 5 iterations. The training was performed on a 10k sentence subset of the Europarl corpus and repeated on a 30k sentence subset<sup>6</sup>. The results from this experiment is summarized in Table 2 and Figure 11.

<sup>6</sup>We were unable to complete all of the runs for the 30k corpus at this writing

	Training Set Size	
	10k	30k
Heuristic using Enhanced Extraction	0.268	0.278
EM (5 Iterations) using Enhanced Extraction	0.231	0.254
Heuristic using Null Extraction	0.169	N/A
EM (5 Iterations) using Null Extraction	0.226	N/A

Table 2: Comparison of BLEU scores obtained using heuristic approaches and EM

We also evaluated the BLEU score for the phrase translation model parameters after each iteration of EM. The results are summarized in Table 3 and Figure 12.

	EM Iterations					
	0	1	2	3	4	5
EM using Null Extraction on 10k corpus	0.169	0.245	0.242	0.238	0.234	0.226
EM using Enhanced Extraction on 10k corpus	0.268	0.214	0.24	0.236	0.233	0.231
EM using Enhanced Extraction on 30k corpus	0.278	0.214	0.254	0.257	0.258	0.254

Table 3: BLEU scores after successive EM iterations

### 5.3 Error analysis

All of the phrase translation model parameters generated by EM performed worse than the best phrase translation model parameters generated by heuristic methods. Hence, it is apparent that we did not use the correct model for EM. From Figure 12 we can see that the model assumed by EM is inadequate. In all three cases, the BLEU score decreased as the iterations increased. However, more notably, for EM initialized with the results of the heuristic from *EnhancedPhraseExtractor*, the first EM iteration made the BLEU score much worse. This clearly indicates that the model assumed by EM is flawed. But, with many parameters left to tweak, we cannot predict how far we are from improving upon the established baseline.

There are a few aspects of our preliminary experiments that we know to be problematic:

1. There is a mismatch between the EM training model that we used and the phrase translation model assumed by the decoder. Currently, for simplicity, we use a global distortion penalty for phrase rearrangement, but Pharaoh assumes relative distortion.
2. The parameter for the global distortion penalty was not learned. Instead we chose a parameter that we felt would work well<sup>7</sup>. However, learning the distortion parameter is desirable regardless of which distortion scheme is used.
3. EM did not use all of the data in the training set. First, we limited training to only sentences between 5 and 40 words. Second, our model explicitly assumes a 1-to-1 correspondence between French and English phrases, which did not hold for some of the sentences in the

<sup>7</sup>This parameter was selected based on previous experiences with learning word alignments using EM.

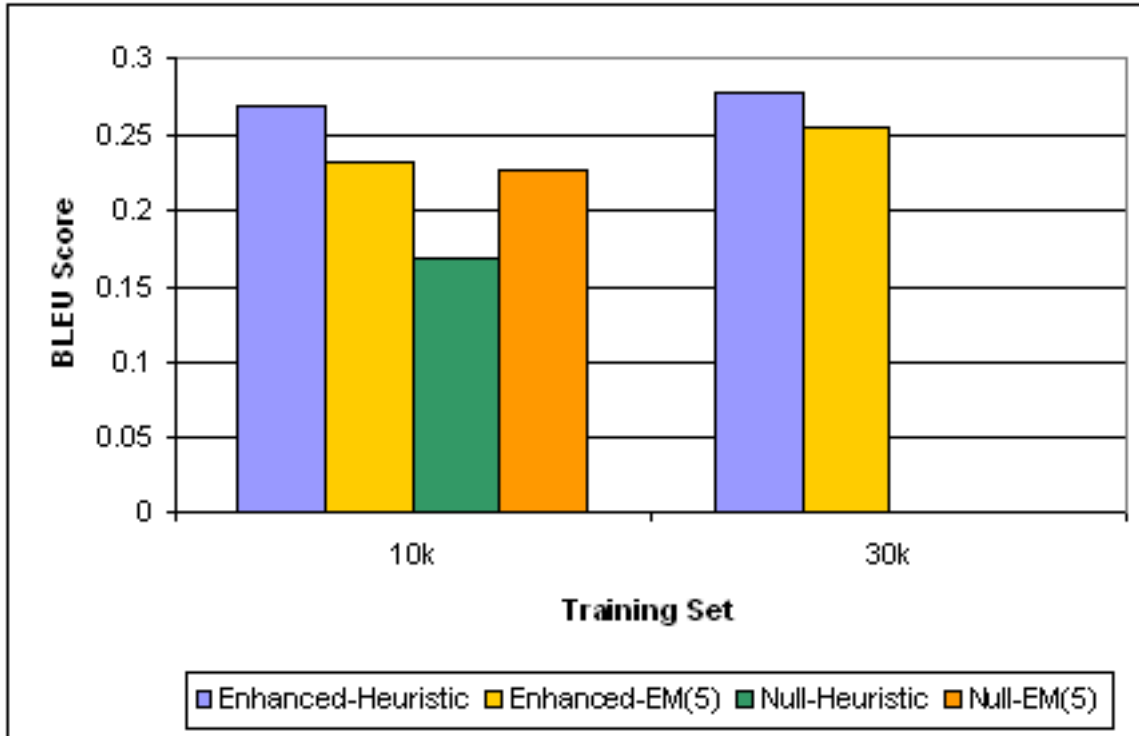


Figure 11: Comparison of BLEU scores obtained using heuristic approaches and EM

training set. Lastly, since we were using an exponential algorithm in the E-step, we placed an upper limit on the computation time for the E-step<sup>8</sup>.

4. The training set was kept small to allow each EM iteration to finish in a reasonable amount of time<sup>9</sup>. Because of this, we encounter severe data sparsity issues. From the 10k training set we are estimating over 100k phrase translation parameters. Ideally we would need train on more data, and use regularization to lessen the sparsity effects.

## 5.4 Approach shortcomings

Our approach has some inherent limitations which need to be addressed before we can improve on the state-of-the-art in statistical machine translation. A few of these merit further discussion.

1. Currently we are tied to using an exponential algorithm for the E-step of EM since less than half of the sentences in our corpus do not satisfy monotonicity constraints. For the experiments we performed, the exponential algorithm functioned in practice because we do

<sup>8</sup>Most of the training set that satisfied the first two requirements took less than a second to process, but occasionally a sentence would take hours. This occurred in about 1 in 30 sentences.

<sup>9</sup>For 10k each EM iteration took 40 minutes, and for 30k each EM iteration took about 2 hours.

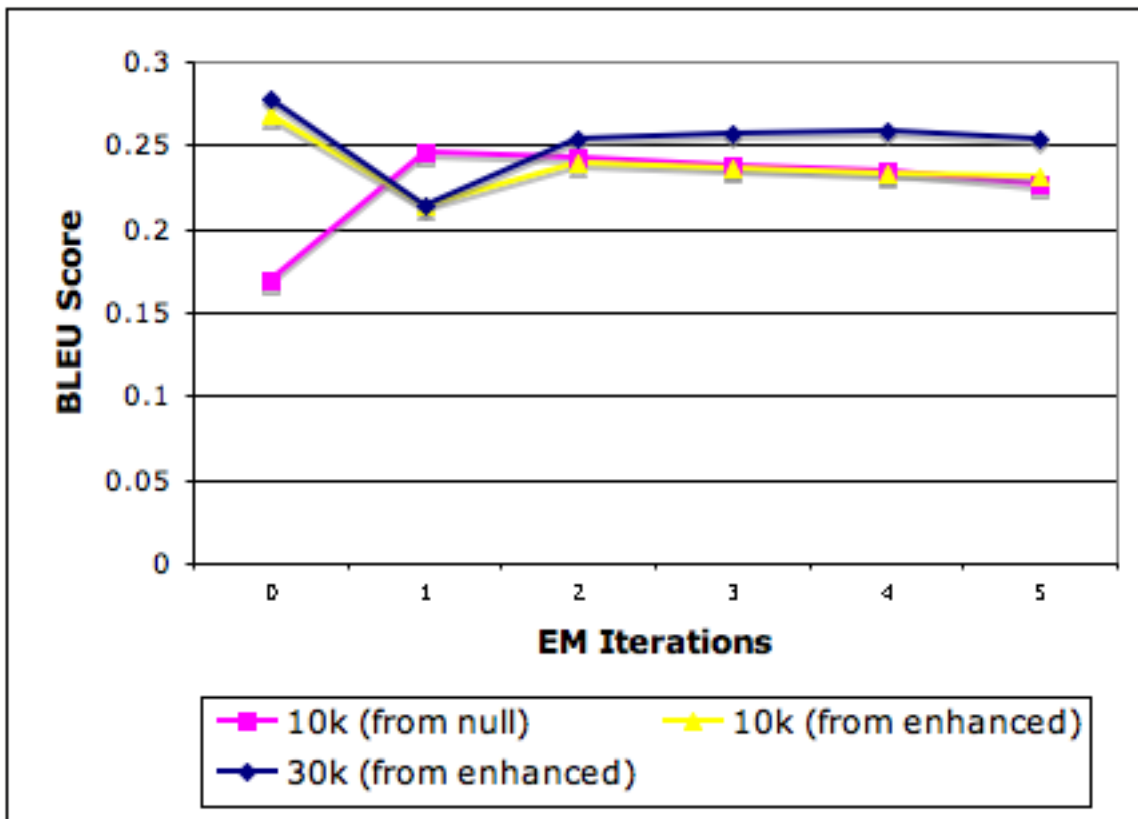


Figure 12: BLEU scores after successive EM iterations

not consider any potential phrases that violate the word alignment. However, this approach will only remain practical if word alignments are not sparse. Since we constrain the possible phrases to those that respect the word alignment in a sentence pair, we want to use the most accurate word alignments available. However, accurate alignments tend to be sparse in nature. Therefore, if we want to use more accurate word alignments, we would need to find an alternative to the exponential algorithm that we are currently using.

2. The model used to train EM assumes a 1-to-1 mapping between French phrases and English phrases. This assumption is valid for decoding since during decoding we are content with producing more literal translations. However, this assumption does not hold for all sentences in the training set. Since the training set is made by human translators, there are sentences that are translated more liberally, where some phrases are not translated at all.
3. Data sparsity is a serious problem for phrase based translation. Even using the heuristic approach, from a corpus of just over 1 million sentences we produce over 4 million phrase translation parameters. Many of these phrases occur only once and even with lexical weighting, they have high weights. This problem will be much worse when we learn the phrases using EM.

4. Translations of a phrase in a sentence currently does not depend on the context of its occurrence in the sentence. We hope this issue can be addressed by discriminative scoring of phrase translation parameters prior to decoding.
5. Implementation issues are also of important consideration since heuristic methods are much faster and uses less storage than EM. The learned phrase translation parameters need to produce much better translations than the heuristic approach to be worthwhile.

## 5.5 Next steps

There are a few things that we will implement to further investigate the feasibility of learning phrase-based translations:

**New Alignments:** Currently, we are using very rich word-level alignments for pruning in EM. Specifically, we use an output of GIZA that forces very few null word alignments. In addition, these alignments have undergone the growth process described previously, which is likely including many incorrect word alignments. We intend to substitute sparser alignments with higher precision. Doing so will require additional pruning in our EM algorithm. We intend to implement some threshold pruning of segmentations to handle these sparser alignments.

**Distortion Models:** We first intend to implement relative distortion, as used in IBM Model 4, for the model used by EM. Also, we have not yet attempted to either manually adjust or learn the distortion parameters for the model.

**Relaxing the 1-to-1 constraint:** As stated, we currently force a 1-to-1 mapping between sequences of phrases during training. We would like to include the possibility of null alignments, but doing this introduce new tractability issues that we have yet to solve. Solving this issue would allow us to use a much larger portion of the training set, including all sentence pairs with unmatchable phrases.

**Training set size:** We intend to scale the experiment to use larger training sets. The structure of our training can easily be converted to a parallel design.

## References

- [1] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2), 1993.
- [2] Philipp Koehn. *Europarl: A Multilingual Corpus for Evaluation of Machine Translation*. USC Information Sciences Institute, 2002.
- [3] Philipp Koehn. *Pharaoh: A Beam Search Decoder for Phrase-Based Statistical Machine Translation Models*. USC Information Sciences Institute, 2003.
- [4] Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. *HLT-NAACL*, 2003.

- [5] Daniel Marcu and William Wong. A phrase-based, joint probability model for statistical machine translation. *Conference on Empirical Methods in Natural Language Processing*, 2002.
- [6] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.
- [7] Franz Josef Och, Christoph Tillmann, and Hermann Ney. Improved alignment models for statistical machine translation. *ACL Workshops*, 1999.
- [8] Andreas Stolcke. Srilm – an extensible language modeling toolkit. *Proceedings of the International Conference on Statistical Language Processing*, 2002.