# Do we need phrases? Challenging the conventional wisdom in Statistical Machine Translation

**Chris Quirk** and **Arul Menezes**
Microsoft Research
One Microsoft Way
Redmond, WA 98052 USA
{chrisq,arulm}@microsoft.com

## Abstract

We begin by exploring theoretical and practical issues with phrasal SMT, several of which are addressed by syntax-based SMT. Next, to address problems not handled by syntax, we propose the concept of a Minimal Translation Unit (MTU) and develop MTU sequence models. Finally we incorporate these models into a syntax-based SMT system and demonstrate that it improves on the state of the art translation quality within a theoretically more desirable framework.

## 1. Introduction

The last several years have seen phrasal statistical machine translation (SMT) systems outperform word-based approaches by a wide margin (Koehn 2003). Unfortunately the use of phrases in SMT is beset by a number of difficult theoretical and practical problems, which we attempt to characterize below. Recent research into syntax-based SMT (Quirk and Menezes 2005; Chiang 2005) has produced promising results in addressing some of the problems; research motivated by other statistical models has helped to address others (Banchs et al. 2005). We refine and unify two threads of research in an attempt to address all of these problems simultaneously. Such an approach proves both theoretically more desirable and empirically superior.

In brief, Phrasal SMT systems employ phrase pairs automatically extracted from parallel corpora. To translate, a source sentence is first partitioned into a sequence of phrases $I = \boldsymbol{s}_1 \ldots \boldsymbol{s}_I$. Each source phrase $\boldsymbol{s}_i$ is then translated into a target phrase $\boldsymbol{t}_i$. Finally the target phrases are permuted, and the translation is read off in order. Beam search is used to approximate the optimal translation. We refer the reader to Keohn et al. (2003) for a detailed description. Unless otherwise noted, the following discussion is generally applicable to Alignment Template systems (Och and Ney 2004) as well.

### 1.1. Advantages of phrasal SMT

**Non-compositionality**
Phrases capture the translations of idiomatic and other non-compositional fixed phrases as a unit, side-stepping the need to awkwardly reconstruct them word by word. While many words can be translated into a single target word, common everyday phrases such as the English *password* translating as the French *mot de passe* cannot be easily subdivided. Allowing such translations to be first class entities simplifies translation implementation and improves translation quality.

**Local re-ordering**
Phrases provide memorized re-ordering decisions. As previously noted, translation can be conceptually divided into two steps: first, finding a set of phrase pairs that simultaneously covers the source side and provides a bag of translated target phrases; and second, picking an order for those target phrases. Since phrase pairs consist of memorized substrings of the training data, they are very likely to produce correct local re-orderings.

**Contextual information**
Many phrasal translations may be easily subdivided into word-for-word translation, for instance the English phrase *the cabbage* may be translated word-for-word as *le chou*. However we note that *la* is also a perfectly reasonable word-for-word translation of *the*, yet *la chou* is not a grammatical French string. Even when a phrase appears compositional, the incorporation of contextual information often improves translation



*Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL*, pages 9–16,
New York, June 2006. 

quality. Phrases are a straightforward means of capturing local context.

### 1.2. Theoretical problems with phrasal SMT

**Exact substring match; no discontiguity**
Large fixed phrase pairs are effective when an exact match can be found, but are useless otherwise. The alignment template approach (where phrases are modeled in terms of word classes instead of specific words) provides a solution at the expense of truly fixed phrases. Neither phrasal SMT nor alignment templates allow discontiguous translation pairs.

**Global re-ordering**
Phrases do capture local reordering, but provide no global re-ordering strategy, and the number of possible orderings to be considered is not lessened significantly. Given a sentence of *n* words, if the average target phrase length is 4 words (which is unusually high), then the re-ordering space is reduced from *n*! to only (*n*/4)!: still impractical for exact search in most sentences. Systems must therefore impose some limits on phrasal reordering, often hard limits based on distance as in Koehn et al. (2003) or some linguistically motivated constraint, such as ITG (Zens and Ney, 2004). Since these phrases are not bound by or even related to syntactic constituents, linguistic generalizations (such as SVO becoming SOV, or prepositions becoming postpositions) are not easily incorporated into the movement models.

**Probability estimation**
To estimate the translation probability of a phrase pair, several approaches are used, often concurrently as features in a log-linear model. Conditional probabilities can be estimated by maximum likelihood estimation. Yet the phrases most likely to contribute important translational and ordering information—the longest ones—are the ones most subject to sparse data issues.

Alternately, conditional phrasal models can be constructed from word translation probabilities; this approach is often called lexical weighting (Vogel et al. 2003). This avoids sparse data issues, but tends to prefer literal translations where the word-for-word probabilities are high Furthermore most approaches model phrases as bags of words, and fail to distinguish between local re-ordering possibilities.

**Partitioning limitation**
A phrasal approach partitions the sentence into strings of words, making several questionable assumptions along the way. First, the probability of the partitioning is never considered. Long phrases tend to be rare and therefore have sharp probability distributions. This adds an inherent bias toward long phrases with questionable MLE probabilities (e.g. 1/1 or 2/2). [1]

Second, the translation probability of each phrase pair is modeled independently. Such an approach fails to model any phenomena that reach across boundaries; only the target language model and perhaps whole-sentence bag of words models cross phrase boundaries. This is especially important when translating into languages with agreement phenomena. Often a single phrase does not cover all agreeing modifiers of a headword; the uncovered modifiers are biased toward the most common variant rather than the one agreeing with its head. Ideally a system would consider overlapping phrases rather than a single partitioning, but this poses a problem for generative models: when words are generated multiple times by different phrases, they are effectively penalized.

### 1.3. Practical problem with phrases: size

In addition to the theoretical problems with phrases, there are also practical issues. While phrasal systems achieve diminishing returns due

[1] The Alignment Template approach differs slightly here. Phrasal SMT estimates the probability of a phrase pair as:

$$\phi(\bar{t} \mid \bar{s}) = \frac{count(\bar{s}, \bar{t})}{\sum_{\bar{t}'} count(\bar{s}, \bar{t}')}$$

The Alignment Template method incorporates a loose partitioning probability by instead estimating the probability as (in the special case where each word has a unique class):

$$p(\bar{t} \mid \bar{s}) = \frac{count(\bar{s}, \bar{t})}{count(\bar{s})}$$

Note that these counts could differ significantly. Picture a source phrase that almost always translates into a discontiguous phrase (e.g. English *not* becoming French *ne … pas*), except for the rare occasion where, due to an alignment error or odd training data, it translates into a contiguous phrase (e.g. French *ne parle pas*). Then the first probability formulation of *ne parle pas* given *not* would be unreasonably high. However, this is a partial fix since it again suffers from data sparsity problems, especially on longer templates where systems hope to achieve the best benefits from phrases.

to sparse data, one does see a small incremental benefit with increasing phrase lengths. Given that storing all of these phrases leads to very large phrase tables, many research systems simply limit the phrases gathered to those that could possibly influence some test set. However, this is not feasible for true production MT systems, since the data to be translated is unknown.

## 2. Previous work

### 2.1. Delayed phrase construction

To avoid the major practical problem of phrasal SMT—namely large phrase tables, most of which are not useful to any one sentence—one can instead construct phrase tables on the fly using an indexed form of the training data (Zhang and Vogel 2005; Callison-Burch et al. 2005). However, this does not relieve any of the theoretical problems with phrase-based SMT.

### 2.2. Syntax-based SMT

Two recent systems have attempted to address the contiguity limitation and global re-ordering problem using syntax-based approaches.

**Hierarchical phrases**

Recent work in the use of hierarchical phrases (Chiang 2005) improves the ability to capture linguistic generalizations, and also removes the limitation to contiguous phrases. Hierarchical phrases differ from standard phrases in one important way: in addition to lexical items, a phrase pair may contain indexed placeholders, where each index must occur exactly once on each side. Such a formulation leads to a formally syntax-based translation approach, where translation is viewed as a parallel parsing problem over a grammar with one non-terminal symbol. This approach significantly outperforms a phrasal SMT baseline in controlled experimentation.

Hierarchical phrases do address the need for non-contiguous phrases and suggest a powerful ordering story in the absence of linguistic information, although this reordering information is bound in a deeply lexicalized form. Yet they do not address the phrase probability estimation problem; nor do they provide a means of modeling phenomena across phrase boundaries. The practical problems with phrase-based translation systems are further exacerbated, since the number of translation rules with up to two non-adjacent non-terminals in a 1-1 monotone sentence pair of $n$ source and target words is $O(n^6)$, as compared to $O(n^2)$ phrases.

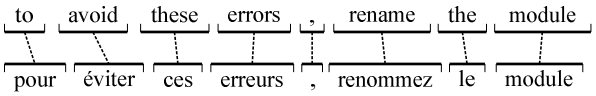
**Treelet Translation**

Another means of extending phrase-based translation is to incorporate source language syntactic information. In Quirk and Menezes (2005) we presented an approach to phrasal SMT based on a parsed dependency tree representation of the source language. We use a source dependency parser and project a target dependency tree using a word-based alignment, after which we extract tree-based phrases ('treelets') and train a tree-based ordering model. We showed that using treelets and a tree-based ordering model results in significantly better translations than a leading phrase-based system (Pharaoh, Koehn 2004), keeping all other models identical.

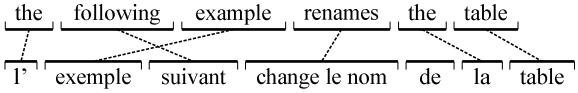
Like the hierarchical phrase approach, treelet translation succeeds in improving the global re-ordering search and allowing discontiguous phrases, but does not solve the partitioning or estimation problems. While we found our treelet system more resistant to degradation at smaller phrase sizes than the phrase-based system, it nevertheless suffered significantly at very small phrase sizes. Thus it is also subject to practical problems of size, and again these problems are exacerbated since there are potentially an exponential number of treelets.

### 2.3. Bilingual *n*-gram channel models

To address on the problems of estimation and partitioning, one recent approach transforms channel modeling into a standard sequence modeling problem (Banchs et al. 2005). Consider the following aligned sentence pair in Figure 1a. In such a well-behaved example, it is natural to consider the problem in terms of sequence models. Picture a generative process that produces a sentence pair in left to right, emitting a pair of words in lock step. Let $M = \langle m_1, \ldots, m_n \rangle$ be a sequence of word pairs $m_i = \langle s, t \rangle$. Then one can generatively model the probability of an aligned sentence pair using techniques from *n*-gram language modeling:



(a) Monotone aligned sentence pair



(b) More common non-monotone aligned sentence pair

**Figure 1**. Example aligned sentence pairs.

$$P(S,T,A) = P(M) = \sum_{i=1}^{k} P(m_i \mid m_1^{i-1}) \approx \sum_{i=1}^{k} P(m_i \mid m_{i-n}^{i-1})$$

When an alignment is one-to-one and monotone, this definition is sufficient. However alignments are seldom purely one-to-one and monotone in practice; Figure 1b displays common behavior such as one-to-many alignments, inserted words, and non-monotone translation. To address these problems, Banchs et al. (2005) suggest defining tuples such that:

(1) the tuple sequence is monotone,
(2) there are no word alignment links between two distinct tuples,
(3) each tuple has a non-NULL source side, which may require that target words aligned to NULL are joined with their following word, and
(4) no smaller tuples can be extracted without violating these constraints.

Note that *M* is now a sequence of phrase pairs instead of word pairs. With this adjusted definition, even Figure 1b can be generated using the same process using the following tuples:

$m_1$ = ‹ *the, l'* ›
$m_2$ = ‹ *following example, exemple suivant* ›
$m_3$ = ‹ *renames, change le nom* ›
$m_4$ = ‹ *the, de la* ›
$m_5$ = ‹ *table, table* ›

There are several advantages to such an approach. First, it largely avoids the partitioning problem; instead of segmenting into potentially large phrases, the sentence is segmented into much smaller tuples, most often pairs of single words. Furthermore the failure to model a partitioning probability is much more defensible when the partitions are much smaller. Secondly, n-gram language model probabilities provide a robust means of estimating phrasal translation probabilities in context that models interactions between *all* adjacent tuples, obviating the need for overlapping mappings.

These tuple channel models still must address practical issues such as model size, though much work has been done to shrink language models with minimal impact to perplexity (e.g. Stolcke 1998), which these models could immediately leverage. Furthermore, these models do not address the contiguity problem or the global reordering problem.

## 3. Translation by MTUs

In this paper, we address all four theoretical problems using a novel combination of our syntactically-informed treelet approach (Quirk and Menezes 2005) and a modified version of bilingual *n*-gram channel models (Banchs et al. 2005). As in our previous work, we first parse the sentence into a dependency tree. After this initial parse, we use a global search to find a candidate that maximizes a log-linear model, where these candidates consist of a target word sequence annotated with a dependency structure, a word alignment, and a treelet decomposition.

We begin by exploring minimal translation units and the models that concern them.

### 3.1. Minimal Translation Units

Minimal Translation Units (MTUs) are related to the tuples of Banchs et al. (2005), but differ in several important respects. First, we relieve the restriction that the MTU sequence be monotone. This prevents spurious expansion of MTUs to incorporate adjacent context only to satisfy monotonicity. In the example, note that the previous algorithm would extract the tuple ‹*following example, exemple suivant*› even though the translations are mostly independent. Their partitioning is also context dependent: if the sentence did not contain the words *following* or *suivant*, then ‹ *example, exemple* › would be a single MTU. Secondly we drop the requirement that no MTU have a NULL source side. While some insertions can be modeled in terms of adjacent words, we believe more robust models can be obtained if we consider insertions as

| | | English | French | English | Japanese |
|---|---|---|---|---|---|
| *Training* | Sentences | 300,000 | | 500,000 | |
| | Words | 4,441,465 | 5,198,932 | 7,909,198 | 9,379,240 |
| | Vocabulary | 63,343 | 59,290 | 79,029 | 95,813 |
| | Singletons | 35,328 | 29,448 | 44,111 | 52,911 |
| *Development test* | Sentences | 200 | | 200 | |
| | Words | 3,045 | 3,456 | 3,436 | 4,095 |
| *Test* | Sentences | 2,000 | | 2,000 | |
| | Words | 30,010 | 34,725 | 35,556 | 3,855 |
| | OOV rate | 5.5% | 4.6% | 6.9% | 6.8% |

**Table 4.1** Data characteristics

independent units. In the end our MTUs are defined quite simply as pairs of source and target word sets that follow the given constraints:

(1) there are no word alignment links between distinct MTUs, and
(2) no smaller MTUs can be extracted without violating the previous constraint.

Since our word alignment algorithm is able to produce one-to-one, one-to-many, many-to-one, one-to-zero, and zero-to-one translations, these act as our basic units. As an example, let us consider example (1) once again. Using this new algorithm, the MTUs would be:

$m_1$ = ‹ *the, l'* ›
$m_2$ = ‹ *following, suivant* ›
$m_3$ = ‹ *example, exemple* ›
$m_4$ = ‹ *renames, change le nom* ›
$m_5$ = ‹ NULL, *de* ›
$m_6$ = ‹ *the, la* ›
$m_7$ = ‹ *table, table* ›

A finer grained partitioning into MTUs further reduces the data sparsity and partitioning issues associated with phrases. Yet it poses issues in modeling translation: given a sequence of MTUs that does not have a monotone segmentation, how do we model the probability of an aligned translation pair? We propose several solutions, and use each in a log-linear combination of models.

First, one may walk the MTUs in source order, ignoring insertion MTUs altogether. Such a model is completely agnostic of the target word order; instead of generating an aligned source target pair, it generates a source sentence along with a bag of target phrases. This approach expends a great deal of modeling effort in regenerating the source sentence, which may not be altogether desirable, though it does condition on surrounding translations. Also, it can be evaluated on candidates before orderings are considered. This latter property may be useful in two-stage decoding strategies where translations are considered before orderings.

Secondly, one may walk the MTUs in target order, ignoring deletion MTUs. Where the source-order MTU channel model expends probability mass generating the source sentence, this model expends a probability mass generating the target sentence and therefore may be somewhat redundant with the target language model.

Finally, one may walk the MTUs in dependency tree order. Let us assume that in addition to an aligned source-target candidate pair, we have a dependency parse of the source side. Where the past models conditioned on surface adjacent MTUs, this model conditions on tree adjacent MTUs. Currently we condition only on the ancestor chain, where $parent_1(m)$ is the parent MTU of $m$, $parent_2(m)$ is the grandparent of m, and so on:

$$P(S,T,A) = P(M) \approx \prod_{m \in M} P(m \mid parent_1^{n-1}(m))$$

This model hopes to capture information completely distinct from the other two models, such as translational preferences contingent on the head, even in the presence of long distance dependencies. Note that it generates unordered dependency tree pairs.

All of these models can be trained from a parallel corpus that has been word aligned and the source side dependency parsed. We walk through each sentence extracting MTUs in source, target, and tree order. Standard n-gram language modeling tools can be used to train MTU language models.

### 3.2. Decoding

We employ a dependency tree-based beam search decoder to search the space of translations. First the input is parsed into a dependency tree

structure. For each input node in the dependency tree, an n-best list of candidates is produced. Candidates consist of a target dependency tree along with a treelet and word alignment. The decoder generally assumes phrasal cohesion: candidates covering a substring (not subsequence) of the input sentence produce a potential substring (not subsequence) of the final translation. In addition to allowing a DP / beam decoder, this allows us to evaluate string-based models (such as the target language model and the source and target order MTU n-gram models) on partial candidates. This decoder is unchanged from our previous work: the MTU n-gram models are simply incorporated as feature functions in the log-linear combination. In the experiments section the MTU models are referred to as model set (1).

### 3.3. Other translation models

**Phrasal channel models**

We can estimate traditional channel models using maximum likelihood or lexical weighting:

$$f_{\text{DirectMLE}}(S,T,A) = \prod_{(\sigma,\tau)\in treelets(A)} \frac{c(\sigma,\tau)}{c(\sigma,*)}$$

$$f_{\text{InverseMLE}}(S,T,A) = \prod_{(\sigma,\tau)\in treelets(A)} \frac{c(\sigma,\tau)}{c(*,\tau)}$$

$$f_{\text{DirectM1}}(S,T,A) = \prod_{(\sigma,\tau)\in treelets(A)} \prod_{t\in\tau} \sum_{s\in\sigma} p(t \mid s)$$

$$f_{\text{InverseM1}}(S,T,A) = \prod_{(\sigma,\tau)\in treelets(A)} \prod_{s\in\sigma} \sum_{t\in\tau} p(s \mid t)$$

We use word probability tables $p(t \mid s)$ and $p(s \mid t)$ estimated by IBM Model 1 (Brown et al. 1993). Such models can be built over phrases if used in a phrasal decoder or over treelets if used in a treelet decoder. These models are referred to as set (2).

**Word-based models**

A target language model using modified Kneser-Ney smoothing captures fluency; a word count feature offsets the target LM preference for shorter selections; and a treelet/phrase count helps bias toward translations using fewer phrases. These models are referred to as set (3).

$$f_{\text{targetLM}}(S,T,A) = \prod_{i=1}^{|T|} P(t_i \mid t_{i-n}^{i-1})$$

$$f_{\text{wordcount}}(S,T,A) = |T|$$

$$f_{\text{phrasecount}}(S,T,A) = |treelets(A)|$$

**Syntactic models**

As in Quirk and Menezes (2005), we include a linguistically-informed order model that predicts the head-relative position of each node independently, and a tree-based bigram target language model; these models are referred to as set (4).

$$f_{\text{order}}(S,T,A) = \prod_{t\in T} P(position(t) \mid S,T,A)$$

$$f_{\text{treeLM}}(S,T,A) = \prod_{t\in T} P(t \mid parent(t))$$

## 4. Experimental setup

We evaluate the translation quality of the system using the BLEU metric (Papineni et al., 02) under a variety of configurations. As an additional baseline, we compare against a phrasal SMT decoder, Pharaoh (Koehn et al. 2003).

### 4.1. Data

Two language pairs were used for this comparison: English to French, and English to Japanese. The data was selected from technical software documentation including software manuals and product support articles; Table 4.1 presents the major characteristics of this data.

### 4.2. Training

We parsed the source (English) side of the corpora using NLPWIN, a broad-coverage rule-based parser able to produce syntactic analyses at varying levels of depth (Heidorn 2002). For the purposes of these experiments we used a dependency tree output with part-of-speech tags and unstemmed surface words. Word alignments were produced by GIZA++ (Och and Ney 2003) with a standard training regimen of five iterations of Model 1, five iterations of the HMM Model, and five iterations of Model 4, in both directions. These alignments were combined heuristically as described in our previous work.

We then projected the dependency trees and used the aligned dependency tree pairs to extract treelet translation pairs, train the order model, and train MTU models. The target language models were trained using only the target side of the corpus. Finally we trained model weights by maximizing BLEU (Och 2003) and set decoder optimization parameters (*n*-best list size, timeouts

| | EF | EJ |
|---|---|---|
| *Phrasal decoder (Pharaoh)* | | |
| Model sets (2),(3) | 45.8±2.0 | 32.9±0.9 |
| *Treelet decoder, without discontiguous mappings* | | |
| Model sets (2),(3) | 45.1±2.1 | 33.2±0.9 |
| Model sets (2),(3),(4) | 48.4±2.0 | 34.8±0.9 |
| *Treelet decoder, with discontiguous mappings* | | |
| Model sets (2),(3) | 46.4±2.1 | 34.3±0.9 |
| Model sets (2),(3),(4) | 48.7±2.1 | 34.9±0.9 |
| Model sets (1),(3),(4) | 49.6±2.1 | 33.9±0.8 |
| Model sets (1)-(4) | 50.5±2.1 | 36.2±0.9 |

**Table 5.1**. Broad system comparison.

| | EF | EJ |
|---|---|---|
| *Treelet decoder, model sets (1),(3),(4)* | | |
| MTU unigram | 47.8±2.1 | 33.2±0.9 |
| MTU bigram | 49.6±2.1 | 33.9±0.8 |
| MTU trigram | 49.9±2.0 | 34.0±0.9 |
| MTU 4-gram | 49.6±2.1 | 34.1±0.9 |
| *Treelet decoder, model sets (1)-(4)* | | |
| MTU unigram | 48.6±2.1 | 34.3±1.0 |
| MTU bigram | 50.5±2.1 | 36.2±0.9 |
| MTU trigram | 48.9±2.0 | 36.1±0.9 |
| MTU 4-gram | 50.4±2.0 | 36.2±1.0 |

**Table 5.2**. Varying MTU n-gram model order.

etc) on a development test set of 200 held-out sentences each with a single reference translation. Parameters were individually estimated for each distinct configuration.

**Pharaoh**

The same GIZA++ alignments as above were used in the Pharaoh decoder (Koehn 2004). We used the heuristic combination described in (Och and Ney 2003) and extracted phrasal translation pairs from this combined alignment as described in (Koehn et al., 2003). Aside from MTU models and syntactic models (Pharaoh uses its own ordering approach), the same models were used: MLE and lexical weighting channel models, target LM, and phrase and word count. Model weights were also trained following Och (2003).

## 5. Results

We begin with a broad brush comparison of systems in Table 5.1. Throughout this section, treelet and phrase sizes are measured in terms of MTUs, not words. By default, all systems (including Pharaoh) use treelets or phrases of up to four MTUs, and MTU bigram models. The first results reiterate that the introduction of discontiguous mappings and especially a linguistically motivated order model (model set (4)) can improve translation quality. Replacing the standard channel models (model set (2)) with MTU bigram models (model set (1)) does not appear to degrade quality; it even seems to boost quality on EF. Furthermore, the information in the MTU models appears somewhat orthogonal to the phrasal models; a combination results in improvements for both language pairs.

The experiments in Table 5.2 compare quality using different orders of MTU n-gram models. (Treelets containing up to four MTUs were still used as the basis for decoding; only the order of the MTU n-gram models was adjusted.) A unigram model performs surprisingly well. This supports our intuition that atomic handling of non-compositional multi-word translations is a major contribution of phrasal SMT. Furthermore bigram models increase translation quality supporting the claim that local context is another contribution. Models beyond bigrams had little impact presumably due to sparsity and smoothing.

Table 5.3 explores the impact of using different phrase/treelet sizes in decoding. We see that adding MTU models makes translation more resilient given smaller phrases. The poor performance at size 1 is not particularly surprising: both systems require insertions to be lexically anchored: the only decoding operation allowed is translation of some visible source phrase, and insertions have no visible trace.

## 6. Conclusions

In this paper we have teased apart the role of

**Table 5.3**. Varying phrase / treelet size.

| | *Phrasal decoder model sets (2),(3)* | | *Treelet decoder: MTU bigram model sets (1),(3),(4)* | | *Treelet decoder: MTU bigram model sets (1)-(4)* | |
|---|---|---|---|---|---|---|
| *Size* | **EF** | **EJ** | **EF** | **EJ** | **EF** | **EJ** |
| 1 | 32.6±1.8 | 20.5±0.7 | 26.3±1.3 | 15.4±0.7 | 29.8±1.4 | 16.7±0.7 |
| 2 | 40.4±1.9 | 29.7±0.7 | 48.7±2.1 | 32.4±0.9 | 47.7±2.1 | 33.8±0.8 |
| 3 | 44.3±2.1 | 30.7±0.9 | 48.5±2.0 | 34.6±0.9 | 48.5±2.0 | 35.1±0.9 |
| 4 | 45.8±2.0 | 32.9±0.9 | 49.6±2.1 | 33.9±0.8 | 50.5±2.1 | 36.2±0.9 |

phrases and handled each contribution via a distinct model best suited to the task. Non-compositional translations stay as MTU phrases. Context and robust estimation is provided by MTU-based n-gram models. Local and global ordering is handled by a tree-based model.

The first interesting result is that at normal phrase sizes, augmenting an SMT system with MTU n-gram models improves quality; whereas replacing the standard phrasal channel models by the more theoretically sound MTU n-gram channel models leads to very similar performance.

Even more interesting are the results on smaller phrases. A system using very small phrases (size 2) and MTU bigram models matches (English-French) or at least approaches (English-Japanese) the performance of the baseline system using large phrases (size 4). While this work does not yet obviate the need for phrases, we consider it a promising step in that direction.

An immediate practical benefit is that it allows systems to use much smaller phrases (and hence smaller phrase tables) with little or no loss in quality. This result is particularly important for syntax-based systems, or any system that allows discontiguous phrases. Given a fixed length limit, the number of surface phrases extracted from any sentence pair of length $n$ where all words are uniquely aligned is $O(n)$, but the number of treelets is potentially exponential in the number of children; and the number of rules with two gaps extracted by Chiang (2005) is potentially $O(n^3)$. Our results using MTUs suggest that such systems can avoid unwieldy, poorly estimated long phrases and instead anchor decoding on shorter, more tractable knowledge units such as MTUs, incorporating channel model information and contextual knowledge with an MTU n-gram model.

Much future work does remain. From inspecting the model weights of the best systems, we note that only the source order MTU n-gram model has a major contribution to the overall score of a given candidate. This suggests that the three distinct models, despite their different walk orders, are somewhat redundant. We plan to consider other approaches for conditioning on context. Furthermore phrasal channel models, in spite of the laundry list of problems presented here, have a significant impact on translation quality. We hope to replace them with effective models without the brittleness and sparsity issues of heavy lexicalization.

## References


Banchs, Rafael, Josep Crego, Adrià de Gispert, Patrik Lambert, and Jose Mariño. 2005. Statistical machine translation of Euparl data by using bilingual n-grams. In *Proceedings of ACL Workshop on Building and Using Parallel Texts.*

Brown, Peter, Vincent Della Pietra, Stephen Della Pietra, and Robert Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics* 19(2): 263-311.

Callison-Burch, Chris, Colin Bannard, and Josh Schroeder. 2005. Scaling phrase-based machine translation to larger corpora and longer phrases. In *Proceedings of ACL*.

Chiang, David. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of ACL*.

Heidorn, George. 2000. "Intelligent writing assistance". In Dale et al. *Handbook of Natural Language Processing*, Marcel Dekker.

Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase based translation. In *Proceedings of NAACL*.

Koehn, Philipp. 2004. Pharaoh: A beam search decoder for phrase-based statistical machine translation models. In *Proceedings of AMTA.*

Och, Franz Josef and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1): 19-51.

Och, Franz Josef and Hermann Ney. 2004. The Alignment Template approach to statistical machine translation, *Computational Linguistics*, 30(4):417-450.

Och, Franz Josef. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of ACL*.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL*.

Quirk, Chris and Arul Menezes. 2005. Dependency tree translation: syntactically-informed phrasal SMT. In *Proceedings of ACL*.

Stolcke, Andreas. 1998. Entropy-based pruning of backoff language models. In *Proceedings of DARPA Broadcast News Transcription and Understanding*.

Vogel, Stephan, Ying Zhang, Fei Huang, Alicia Tribble, Ashish Venugopal, Bing Zhao, Alex Waibel. 2003. The CMU statistical machine translation system. In *Proceedings of MT Summit.*

Zens, Richard, and Hermann Ney. 2003. A comparative study on reordering constraints in statistical machine translation. In *Proceedings of ACL*.

Zhang, Ying and Stephan Vogel. 2005. An efficient phrase-to-phrase alignment model for arbitrarily long phrase and large corpora. In *Proceedings of EAMT*.