Machine Translation without Words through Substring Alignment

Graham Neubig[1,2], **Taro Watanabe**[2], **Shinsuke Mori**[1], **Tatsuya Kawahara**[1]
[1]Graduate School of Informatics, Kyoto University
Yoshida Honmachi, Sakyo-ku, Kyoto, Japan
[2]National Institute of Information and Communication Technology
3-5 Hikari-dai, Seika-cho, Soraku-gun, Kyoto, Japan

Abstract

In this paper, we demonstrate that accurate machine translation is possible without the concept of "words," treating MT as a problem of transformation between character strings. We achieve this result by applying phrasal inversion transduction grammar alignment techniques to character strings to train a character-based translation model, and using this in the phrase-based MT framework. We also propose a look-ahead parsing algorithm and substring-informed prior probabilities to achieve more effective and efficient alignment. In an evaluation, we demonstrate that character-based translation can achieve results that compare to word-based systems while effectively translating unknown and uncommon words over several language pairs.

1 Introduction

Traditionally, the task of statistical machine translation (SMT) is defined as translating a source sentence $\boldsymbol{f}_1^J = \{f_1, \ldots, f_J\}$ to a target sentence $\boldsymbol{e}_1^I = \{e_1, \ldots, e_I\}$, where each element of $\boldsymbol{f}_1^J$ and $\boldsymbol{e}_1^I$ is assumed to be a word in the source and target languages. However, the definition of a "word" is often problematic. The most obvious example of this lies in languages that do not separate words with white space such as Chinese, Japanese, or Thai, in which the choice of a segmentation standard has a large effect on translation accuracy (Chang et al., 2008). Even for languages with explicit word boundaries, all machine translation systems perform at least some precursory form of tokenization, splitting punctuation and words to prevent the sparsity that would occur if punctuated and non-punctuated words were treated as different entities. Sparsity also manifests itself in other forms, including the large vocabularies produced by morphological productivity, word compounding, numbers, and proper names. A myriad of methods have been proposed to handle each of these phenomena individually, including morphological analysis, stemming, compound breaking, number regularization, optimizing word segmentation, and transliteration, which we outline in more detail in Section 2.

These difficulties occur because we are translating sequences of *words* as our basic unit. On the other hand, Vilar et al. (2007) examine the possibility of instead treating each sentence as sequences of *characters* to be translated. This method is attractive, as it is theoretically able to handle all sparsity phenomena in a single unified framework, but has only been shown feasible between similar language pairs such as Spanish-Catalan (Vilar et al., 2007), Swedish-Norwegian (Tiedemann, 2009), and Thai-Lao (Sornlertlamvanich et al., 2008), which have a strong co-occurrence between single characters. As Vilar et al. (2007) state and we confirm, accurate translations cannot be achieved when applying traditional translation techniques to character-based translation for less similar language pairs.

In this paper, we propose improvements to the alignment process tailored to character-based machine translation, and demonstrate that it is, in fact, possible to achieve translation accuracies that ap-

The first author is now affiliated with the Nara Institute of Science and Technology.

Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pages 165–174,
Jeju, Republic of Korea, 8-14 July 2012.

proach those of traditional word-based systems using only character strings. We draw upon recent advances in many-to-many alignment, which allows for the automatic choice of the length of units to be aligned. As these units may be at the character, subword, word, or multi-word phrase level, we conjecture that this will allow for better character alignments than one-to-many alignment techniques, and will allow for better translation of uncommon words than traditional word-based models by breaking down words into their component parts.

We also propose two improvements to the many-to-many alignment method of Neubig et al. (2011). One barrier to applying many-to-many alignment models to character strings is training cost. In the inversion transduction grammar (ITG) framework (Wu, 1997), which is widely used in many-to-many alignment, search is cumbersome for longer sentences, a problem that is further exacerbated when using characters instead of words as the basic unit. As a step towards overcoming this difficulty, we increase the efficiency of the beam-search technique of Saers et al. (2009) by augmenting it with look-ahead probabilities in the spirit of A* search. Secondly, we describe a method to seed the search process using counts of all substring pairs in the corpus to bias the phrase alignment model. We do this by defining prior probabilities based on these substring counts within the Bayesian phrasal ITG framework.

An evaluation on four language pairs with differing morphological properties shows that for distant language pairs, character-based SMT can achieve translation accuracy comparable to word-based systems. In addition, we perform ablation studies, showing that these results were not possible without the proposed enhancements to the model. Finally, we perform a qualitative analysis, which finds that character-based translation can handle unsegmented text, conjugation, and proper names in a unified framework with no additional processing.

2 Related Work on Data Sparsity in SMT

As traditional SMT systems treat all words as single tokens without considering their internal structure, major problems of data sparsity occur for less frequent tokens. In fact, it has been shown that there is a direct negative correlation between vocabulary size (and thus sparsity) of a language and translation accuracy (Koehn, 2005). Sparsity causes trouble for alignment models, both in the form of incorrectly aligned uncommon words, and in the form of garbage collection, where uncommon words in one language are incorrectly aligned to large segments of the sentence in the other language (Och and Ney, 2003). Unknown words are also a problem during the translation process, and the default approach is to map them as-is into the target sentence.

This is a major problem in agglutinative languages such as Finnish or compounding languages such as German. Previous works have attempted to handle morphology, decompounding and regularization through lemmatization, morphological analysis, or unsupervised techniques (Nießen and Ney, 2000; Brown, 2002; Lee, 2004; Goldwater and McClosky, 2005; Talbot and Osborne, 2006; Mermer and Akın, 2010; Macherey et al., 2011). It has also been noted that it is more difficult to translate into morphologically rich languages, and methods for modeling target-side morphology have attracted interest in recent years (Bojar, 2007; Subotin, 2011).

Another source of data sparsity that occurs in all languages is proper names, which have been handled by using cognates or transliteration to improve translation (Knight and Graehl, 1998; Kondrak et al., 2003; Finch and Sumita, 2007), and more sophisticated methods for named entity translation that combine translation and transliteration have also been proposed (Al-Onaizan and Knight, 2002).

Choosing word units is also essential for creating good translation results for languages that do not explicitly mark word boundaries, such as Chinese, Japanese, and Thai. A number of works have dealt with this word segmentation problem in translation, mainly focusing on Chinese-to-English translation (Bai et al., 2008; Chang et al., 2008; Zhang et al., 2008b; Chung and Gildea, 2009; Nguyen et al., 2010), although these works generally assume that a word segmentation exists in one language (English) and attempt to optimize the word segmentation in the other language (Chinese).

We have enumerated these related works to demonstrate the myriad of data sparsity problems and proposed solutions. Character-based translation has the potential to handle all of the phenomena in the previously mentioned research in a single

unified framework, requiring no language specific tools such as morphological analyzers or word segmenters. However, while the approach is attractive conceptually, previous research has only been shown effective for closely related language pairs (Vilar et al., 2007; Tiedemann, 2009; Sornlertlamvanich et al., 2008). In this work, we propose effective alignment techniques that allow character-based translation to achieve accurate translation results for both close and distant language pairs.

3 Alignment Methods

SMT systems are generally constructed from a parallel corpus consisting of target language sentences $\mathcal{E}$ and source language sentences $\mathcal{F}$. The first step of training is to find alignments $\mathcal{A}$ for the words in each sentence pair.

We represent our target and source sentences as $\boldsymbol{e}_1^I$ and $\boldsymbol{f}_1^J$. e_i and f_j represent single elements of the target and source sentences respectively. These may be words in word-based alignment models or single characters in character-based alignment models.[1] We define our alignment as $\boldsymbol{a}_1^K$, where each element is a span $a_k = \langle s, t, u, v \rangle$ indicating that the target string $e_s, \ldots, e_t$ and source string $f_u, \ldots, f_v$ are aligned to each-other.

3.1 One-to-Many Alignment

The most well-known and widely-used models for bitext alignment are for one-to-many alignment, including the IBM models (Brown et al., 1993) and HMM alignment model (Vogel et al., 1996). These models are by nature directional, attempting to find the alignments that maximize the conditional probability of the target sentence $P(\boldsymbol{e}_1^I | \boldsymbol{f}_1^J, \boldsymbol{a}_1^K)$. For computational reasons, the IBM models are restricted to aligning each word on the target side to a single word on the source side. In the formalism presented above, this means that each e_i must be included in at most one span, and for each span $u = v$. Traditionally, these models are run in both directions and combined using heuristics to create many-to-many alignments (Koehn et al., 2003).

However, in order for one-to-many alignment methods to be effective, each f_j must contain enough information to allow for effective alignment with its corresponding elements in $\boldsymbol{e}_1^I$. While this is often the case in word-based models, for character-based models this assumption breaks down, as there is often no clear correspondence between characters.

3.2 Many-to-Many Alignment

On the other hand, in recent years, there have been advances in many-to-many alignment techniques that are able to align multi-element chunks on both sides of the translation (Marcu and Wong, 2002; DeNero et al., 2008; Blunsom et al., 2009; Neubig et al., 2011). Many-to-many methods can be expected to achieve superior results on character-based alignment, as the aligner can use information about substrings, which may correspond to letters, morphemes, words, or short phrases.

Here, we focus on the model presented by Neubig et al. (2011), which uses Bayesian inference in the phrasal inversion transduction grammar (ITG, Wu (1997)) framework. ITGs are a variety of synchronous context free grammar (SCFG) that allows for many-to-many alignment to be achieved in polynomial time through the process of biparsing, which we explain more in the following section. Phrasal ITGs are ITGs that allow for non-terminals that can emit phrase pairs with multiple elements on both the source and target sides. It should be noted that there are other many-to-many alignment methods that have been used for simultaneously discovering morphological boundaries over multiple languages (Snyder and Barzilay, 2008; Naradowsky and Toutanova, 2011), but these have generally been applied to single words or short phrases, and it is not immediately clear that they will scale to aligning full sentences.

4 Look-Ahead Biparsing

In this work, we experiment with the alignment method of Neubig et al. (2011), which can achieve competitive accuracy with a much smaller phrase table than traditional methods. This is important in the character-based translation context, as we would like to use phrases that contain large numbers of characters without creating a phrase table so large that it cannot be used in actual decoding. In this framework, training is performed using sentence-

[1] Some previous work has also performed alignment using morphological analyzers to normalize or split the sentence into morpheme streams (Corston-Oliver and Gamon, 2004).

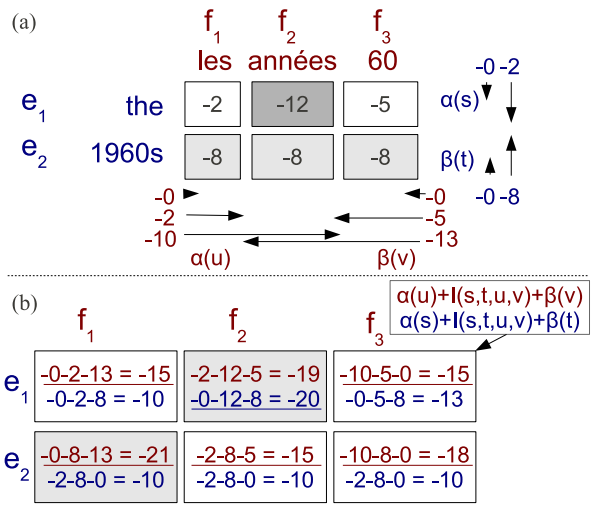

Figure 1: (a) A chart with inside probabilities in boxes and forward/backward probabilities marking the surrounding arrows. (b) Spans with corresponding look-aheads added, and the minimum probability underlined. Lightly and darkly shaded spans will be trimmed when the beam is $\log(P) \geq -3$ and $\log(P) \geq -6$ respectively.

wise block sampling, acquiring a sample for each sentence by first performing bottom-up biparsing to create a chart of probabilities, then performing top-down sampling of a new tree based on the probabilities in this chart.

An example of a chart used in this parsing can be found in Figure 1 (a). Within each cell of the chart spanning $\boldsymbol{e}_s^t$ and $\boldsymbol{f}_u^v$ is an "inside" probability $I(a_{s,t,u,v})$. This probability is the combination of the generative probability of each phrase pair $P_t(\boldsymbol{e}_s^t, \boldsymbol{f}_u^v)$ as well as the sum the probabilities over all shorter spans in straight and inverted order[2]

$$
\begin{aligned}
I(a_{s,t,u,v}) &= P_t(e_s^t, f_u^v) \\
&+ \sum_{s \leq S \leq t} \sum_{u \leq U \leq v} P_x(\text{str}) I(a_{s,S,u,U}) I(a_{S,t,U,v}) \\
&+ \sum_{s \leq S \leq t} \sum_{u \leq U \leq v} P_x(\text{inv}) I(a_{s,S,U,v}) I(a_{S,t,u,U})
\end{aligned}
$$

where $P_x(\text{str})$ and $P_x(\text{inv})$ are the probability of straight and inverted ITG productions.

While the exact calculation of these probabilities can be performed in $O(n^6)$ time, where n is the length of the sentence, this is impractical for all but the shortest sentences. Thus it is necessary to use methods to reduce the search space such as beam-search based chart parsing (Saers et al., 2009) or slice sampling (Blunsom and Cohn, 2010).[3]

In this section we propose the use of a look-ahead probability to increase the efficiency of this chart parsing. Taking the example of Saers et al. (2009), spans are pushed onto a different queue based on their size, and queues are processed in ascending order of size. Agendas can further be trimmed based on a histogram beam (Saers et al., 2009) or probability beam (Neubig et al., 2011) compared to the best hypothesis $\hat{a}$. In other words, we have a queue discipline based on the inside probability, and all spans a_k where $I(a_k) < cI(\hat{a})$ are pruned. c is a constant describing the width of the beam, and a smaller constant probability will indicate a wider beam.

This method is insensitive to the existence of competing hypotheses when performing pruning. Figure 1 (a) provides an example of why it is unwise to ignore competing hypotheses during beam pruning. Particularly, the alignment "les/1960s" competes with the high-probability alignment "les/the," so intuitively should be a good candidate for pruning. However its probability is only slightly higher than "années/1960s," which has no competing hypotheses and thus should not be trimmed.

In order to take into account competing hypotheses, we can use for our queue discipline not only the inside probability $I(a_k)$, but also the outside probability $O(a_k)$, the probability of generating all spans other than a_k, as in A* search for CFGs (Klein and Manning, 2003), and tic-tac-toe pruning for word-based ITGs (Zhang and Gildea, 2005). As the calculation of the actual outside probability $O(a_k)$ is just as expensive as parsing itself, it is necessary to approximate this with heuristic function O^* that can be calculated efficiently.

Here we propose a heuristic function that is designed specifically for phrasal ITGs and is computable with worst-case complexity of n^2, compared with the n^3 amortized time of the tic-tac-toe pruning

[2] P_t can be specified according to Bayesian statistics as described by Neubig et al. (2011).

[3] Applying beam-search before sampling will sample from an improper distribution, although Metropolis-in-Gibbs sampling (Johnson et al., 2007) can be used to compensate. However, we found that this had no significant effect on results, so we omit the Metropolis-in-Gibbs step for experiments.

algorithm described by (Zhang et al., 2008a). During the calculation of the phrase generation probabilities P_t, we save the best inside probability I^* for each monolingual span.

$$I_e^*(s,t) = \max_{\{\tilde{a}=\langle \tilde{s},\tilde{t},\tilde{u},\tilde{v}\rangle;\tilde{s}=s,\tilde{t}=t\}} P_t(\tilde{a})$$
$$I_f^*(u,v) = \max_{\{\tilde{a}=\langle \tilde{s},\tilde{t},\tilde{u},\tilde{v}\rangle;\tilde{u}=u,\tilde{v}=v\}} P_t(\tilde{a})$$

For each language independently, we calculate forward probabilities α and backward probabilities β. For example, $\alpha_e(s)$ is the maximum probability of the span $(0, s)$ of e that can be created by concatenating together consecutive values of I_e^*:

$$\alpha_e(s) = \max_{\{S_1,\ldots,S_x\}} I_e^*(0,S_1)I_e^*(S_1,S_2)\ldots I_e^*(S_x,s).$$

Backwards probabilities and probabilities over f can be defined similarly. These probabilities are calculated for e and f independently, and can be calculated in n^2 time by processing each α in ascending order, and each β in descending order in a fashion similar to that of the forward-backward algorithm. Finally, for any span, we define the outside heuristic as the minimum of the two independent look-ahead probabilities over each language

$$O^*(a_{s,t,u,v}) = \min(\alpha_e(s) * \beta_e(t), \alpha_f(u) * \beta_f(v)).$$

Looking again at Figure 1 (b), it can be seen that the relative probability difference between the highest probability span "les/the" and the spans "années/1960s" and "60/1960s" decreases, allowing for tighter beam pruning without losing these good hypotheses. In contrast, the relative probability of "les/1960s" remains low as it is in conflict with a high-probability alignment, allowing it to be discarded.

5 Substring Prior Probabilities

While the Bayesian phrasal ITG framework uses the previously mentioned phrase distribution P_t during search, it also allows for definition of a phrase pair prior probability $P_{prior}(\boldsymbol{e}_s^t, \boldsymbol{f}_u^v)$, which can efficiently seed the search process with a bias towards phrase pairs that satisfy certain properties. In this section, we overview an existing method used to calculate these prior probabilities, and also propose a new way to calculate priors based on substring co-occurrence statistics.

5.1 Word-based Priors

Previous research on many-to-many translation has used IBM model 1 probabilities to bias phrasal alignments so that phrases whose member words are good translations are also aligned. As a representative of this existing method, we adopt a base measure similar to that used by DeNero et al. (2008):

$$P_{m1}(\boldsymbol{e},\boldsymbol{f}) = M_0(\boldsymbol{e},\boldsymbol{f})P_{pois}(|\boldsymbol{e}|;\lambda)P_{pois}(|\boldsymbol{f}|;\lambda)$$
$$M_0(\boldsymbol{e},\boldsymbol{f}) = (P_{m1}(\boldsymbol{f}|\boldsymbol{e})P_{uni}(\boldsymbol{e})P_{m1}(\boldsymbol{e}|\boldsymbol{f})P_{uni}(\boldsymbol{f}))^{\frac{1}{2}}.$$

P_{pois} is the Poisson distribution with the average length parameter λ, which we set to 0.01. P_{m1} is the word-based (or character-based) Model 1 probability, which can be efficiently calculated using the dynamic programming algorithm described by Brown et al. (1993). However, for reasons previously stated in Section 3, these methods are less satisfactory when performing character-based alignment, as the amount of information contained in a character does not allow for proper alignment.

5.2 Substring Co-occurrence Priors

Instead, we propose a method for using raw substring co-occurrence statistics to bias alignments towards substrings that often co-occur in the entire training corpus. This is similar to the method of Cromieres (2006), but instead of using these co-occurrence statistics as a heuristic alignment criterion, we incorporate them as a prior probability in a statistical model that can take into account mutual exclusivity of overlapping substrings in a sentence.

We define this prior probability using three counts over substrings $c(\boldsymbol{e})$, $c(\boldsymbol{f})$, and $c(\boldsymbol{e},\boldsymbol{f})$. $c(\boldsymbol{e})$ and $c(\boldsymbol{f})$ count the total number of sentences in which the substrings $\boldsymbol{e}$ and $\boldsymbol{f}$ occur respectively. $c(\boldsymbol{e},\boldsymbol{f})$ is a count of the total number of sentences in which the substring $\boldsymbol{e}$ occurs on the target side, and $\boldsymbol{f}$ occurs on the source side. We perform the calculation of these statistics using enhanced suffix arrays, a data structure that can efficiently calculate all substrings in a corpus (Abouelhoda et al., 2004).[4]

While suffix arrays allow for efficient calculation of these statistics, storing all co-occurrence counts $c(\boldsymbol{e},\boldsymbol{f})$ is an unrealistic memory burden for larger

[4]Using the open-source implementation esaxx `http://code.google.com/p/esaxx/`

corpora. In order to reduce the amount of memory used, we discount every count by a constant d, which we set to 5. This has a dual effect of reducing the amount of memory needed to hold co-occurrence counts by removing values for which $c(\boldsymbol{e}, \boldsymbol{f}) < d$, as well as preventing over-fitting of the training data. In addition, we heuristically prune values for which the conditional probabilities $P(\boldsymbol{e}|\boldsymbol{f})$ or $P(\boldsymbol{f}|\boldsymbol{e})$ are less than some fixed value, which we set to 0.1 for the reported experiments.

To determine how to combine $c(\boldsymbol{e})$, $c(\boldsymbol{f})$, and $c(\boldsymbol{e}, \boldsymbol{f})$ into prior probabilities, we performed preliminary experiments testing methods proposed by previous research including plain co-occurrence counts, the Dice coefficient, and χ-squared statistics (Cromieres, 2006), as well as a new method of defining substring pair probabilities to be proportional to bidirectional conditional probabilities

$$\begin{aligned} P_{cooc}(\boldsymbol{e}, \boldsymbol{f}) &= P_{cooc}(\boldsymbol{e}|\boldsymbol{f}) P_{cooc}(\boldsymbol{f}|\boldsymbol{e}) / Z \\ &= \left(\frac{c(\boldsymbol{e}, \boldsymbol{f}) - d}{c(\boldsymbol{f}) - d} \right) \left(\frac{c(\boldsymbol{e}, \boldsymbol{f}) - d}{c(\boldsymbol{e}) - d} \right) / Z \end{aligned}$$

for all substring pairs where $c(\boldsymbol{e}, \boldsymbol{f}) > d$ and where Z is a normalization term equal to

$$Z = \sum_{\{\boldsymbol{e}, \boldsymbol{f}; c(\boldsymbol{e}, \boldsymbol{f}) > d\}} P_{cooc}(\boldsymbol{e}|\boldsymbol{f}) P_{cooc}(\boldsymbol{f}|\boldsymbol{e}).$$

The experiments showed that the bidirectional conditional probability method gave significantly better results than all other methods, so we adopt this for the remainder of our experiments.

It should be noted that as we are using discounting, many substring pairs will be given zero probability according to P_{cooc}. As the prior is only supposed to bias the model towards good solutions and not explicitly rule out any possibilities, we linearly interpolate the co-occurrence probability with the one-to-many Model 1 probability, which will give at least some probability mass to all substring pairs

$$P_{prior}(\boldsymbol{e}, \boldsymbol{f}) = \lambda P_{cooc}(\boldsymbol{e}, \boldsymbol{f}) + (1 - \lambda) P_{m1}(\boldsymbol{e}, \boldsymbol{f}).$$

We put a Dirichlet prior ($\alpha = 1$) on the interpolation coefficient λ and learn it during training.

6 Experiments

In order to test the effectiveness of character-based translation, we performed experiments over a variety of language pairs and experimental settings.

	de-en	fi-en	fr-en	ja-en
TM (en)	2.80M	3.10M	2.77M	2.13M
TM (other)	2.56M	2.23M	3.05M	2.34M
LM (en)	16.0M	15.5M	13.8M	11.5M
LM (other)	15.3M	11.3M	15.6M	11.9M
Tune (en)	58.7k	58.7k	58.7k	30.8k
Tune (other)	55.1k	42.0k	67.3k	34.4k
Test (en)	58.0k	58.0k	58.0k	26.6k
Test (other)	54.3k	41.4k	66.2k	28.5k

Table 1: The number of words in each corpus for TM and LM training, tuning, and testing.

6.1 Experimental Setup

We use a combination of four languages with English, using freely available data. We selected French-English, German-English, Finnish-English data from EuroParl (Koehn, 2005), with development and test sets designated for the 2005 ACL shared task on machine translation.[5] We also did experiments with Japanese-English Wikipedia articles from the Kyoto Free Translation Task (Neubig, 2011) using the designated training and tuning sets, and reporting results on the test set. These languages were chosen as they have a variety of interesting characteristics. French has some inflection, but among the test languages has the strongest one-to-one correspondence with English, and is generally considered easy to translate. German has many compound words, which must be broken apart to translate properly into English. Finnish is an agglutinative language with extremely rich morphology, resulting in long words and the largest vocabulary of the languages in EuroParl. Japanese does not have any clear word boundaries, and uses logographic characters, which contain more information than phonetic characters.

With regards to data preparation, the EuroParl data was pre-tokenized, so we simply used the tokenized data as-is for the training and evaluation of all models. For word-based translation in the Kyoto task, training was performed using the provided tokenization scripts. For character-based translation, no tokenization was performed, using the original text for both training and decoding. For both tasks, we selected as training data all sentences for which both

[5] http://statmt.org/wpt05/mt-shared-task

	de-en	fi-en	fr-en	ja-en
GIZA-word	**24.58** / 64.28 / 30.43	20.41 / 60.01 / 27.89	**30.23** / **68.79** / 34.20	**17.95** / 56.47 / **24.70**
ITG-word	23.87 / **64.89** / **30.71**	**20.83** / 61.04 / 28.46	**29.92** / 68.64 / **34.29**	17.14 / 56.60 / **24.89**
GIZA-char	08.05 / 45.01 / 15.35	06.91 / 41.62 / 14.39	11.05 / 48.23 / 17.80	09.46 / 49.02 / 18.34
ITG-char	21.79 / 64.47 / 30.12	18.38 / **62.44** / **28.94**	26.70 / 66.76 / 32.47	15.84 / **58.41** / 24.58

	en-de	en-fi	en-fr	en-ja
GIZA-word	**17.94** / 62.71 / **37.88**	**13.22** / 58.50 / **27.03**	**32.19** / 69.20 / **52.39**	**20.79** / 27.01 / **38.41**
ITG-word	17.47 / **63.18** / **37.79**	**13.12** / **59.27** / **27.09**	31.66 / **69.61** / 51.98	**20.26** / **28.34** / **38.34**
GIZA-char	06.17 / 41.04 / 19.90	04.58 / 35.09 / 11.76	10.31 / 42.84 / 25.06	01.48 / 00.72 / 06.67
ITG-char	15.35 / 61.95 / 35.45	12.14 / **59.02** / 25.31	27.74 / 67.44 / 48.56	17.90 / **28.46** / 35.71

Table 2: Translation results in word-based BLEU, character-based BLEU, and METEOR for the GIZA++ and phrasal ITG models for word and character-based translation, with bold numbers indicating a statistically insignificant difference from the best system according to the bootstrap resampling method at $p = 0.05$ (Koehn, 2004).

source and target were 100 characters or less,[6] the total size of which is shown in Table 1. In character-based translation, white spaces between words were treated as any other character and not given any special treatment. Evaluation was performed on tokenized and lower-cased data.

For alignment, we use the GIZA++ implementation of one-to-many alignment[7] and the pialign implementation of the phrasal ITG models[8] modified with the proposed improvements. For GIZA++, we used the default settings for word-based alignment, but used the HMM model for character-based alignment to allow for alignment of longer sentences. For pialign, default settings were used except for character-based ITG alignment, which used a probability beam of 10^{-4} instead 10^{-10}.[9] For decoding, we use the Moses decoder,[10] using the default settings except for the stack size, which we set to 1000 instead of 200. Minimum error rate training was performed to maximize word-based BLEU score for all systems.[11] For language models, word-based translation uses a word 5-gram model, and character-based translation uses a character 12-gram model, both smoothed using interpolated Kneser-Ney.

[6] 100 characters is an average of 18.8 English words

[7] `http://code.google.com/p/giza-pp/`

[8] `http://phontron.com/pialign/`

[9] Improvement by using a beam larger than 10^{-4} was marginal, especially with co-occurrence prior probabilities.

[10] `http://statmt.org/moses/`

[11] We chose this set-up to minimize the effect of tuning criterion on our experiments, although it does indicate that we must have access to tokenized data for the development set.

6.2 Quantitative Evaluation

Table 2 presents a quantitative analysis of the translation results for each of the proposed methods. As previous research has shown that it is more difficult to translate into morphologically rich languages than into English (Koehn, 2005), we perform experiments translating in both directions for all language pairs. We evaluate translation quality using BLEU score (Papineni et al., 2002), both on the word and character level (with $n = 4$), as well as METEOR (Denkowski and Lavie, 2011) on the word level.

It can be seen that character-based translation with all of the proposed alignment improvements greatly exceeds character-based translation using one-to-many alignment, confirming that substring-based information is necessary for accurate alignments. When compared with word-based translation, character-based translation achieves better, comparable, or inferior results on character-based BLEU, comparable or inferior results on METEOR, and inferior results on word-based BLEU. The differences between the evaluation metrics are due to the fact that character-based translation often gets words mostly correct other than one or two letters. These are given partial credit by character-based BLEU (and to a lesser extent METEOR), but marked entirely wrong by word-based BLEU.

Interestingly, for translation into English, character-based translation achieves higher accuracy compared to word-based translation on Japanese and Finnish input, followed by German,

	fi-en	ja-en
ITG-word	2.851	2.085
ITG-char	2.826	2.154

Table 3: Human evaluation scores (0-5 scale).

Source Unk. (13/26)	Ref:	directive on equality
	Word:	tasa-arvodirektiivi
	Char:	equality directive
Target Unk. (5/26)	Ref:	yoshiwara-juku station
	Word:	yoshiwara no eki
	Char:	yoshiwara-juku station
Uncommon (5/26)	Ref:	world health organisation
	Word:	world health
	Char:	world health organisation

Table 4: The major gains of character-based translation, unknown, hyphenated, and uncommon words.

and finally French. This confirms that character-based translation is performing well on languages that have long words or ambiguous boundaries, and less well on language pairs with relatively strong one-to-one correspondence between words.

6.3 Qualitative Evaluation

In addition, we performed a subjective evaluation of Japanese-English and Finnish-English translations. Two raters evaluated 100 sentences each, assigning a score of 0-5 based on how well the translation conveys the information contained in the reference. We focus on shorter sentences of 8-16 English words to ease rating and interpretation. Table 3 shows that the results are comparable, with no significant difference in average scores for either language pair.

Table 4 shows a breakdown of the sentences for which character-based translation received a score of at 2+ points more than word-based. It can be seen that character-based translation is properly handling sparsity phenomena. On the other hand, word-based translation was generally stronger with reordering and lexical choice of more common words.

6.4 Effect of Alignment Method

In this section, we compare the translation accuracies for character-based translation using the phrasal ITG model with and without the proposed improvements of substring co-occurrence priors and look-ahead parsing as described in Sections 4 and 5.2.

	fi-en	en-fi	ja-en	en-ja
ITG +cooc +look	**28.94**	**25.31**	**24.58**	**35.71**
ITG +cooc -look	28.51	24.24	**24.32**	**35.74**
ITG -cooc +look	**28.65**	24.49	**24.36**	35.05
ITG -cooc -look	27.45	23.30	23.57	34.50

Table 5: METEOR scores for alignment with and without look-ahead and co-occurrence priors.

Figure 5 shows METEOR scores[12] for experiments translating Japanese and Finnish. It can be seen that the co-occurrence prior gives gains in all cases, indicating that substring statistics are effectively seeding the ITG aligner. The introduced look-ahead probabilities improve accuracy significantly when substring co-occurrence counts are not used, and slightly when co-occurrence counts are used. More importantly, they allow for more aggressive beam pruning, increasing sampling speed from 1.3 sent/s to 2.5 sent/s for Finnish, and 6.8 sent/s to 11.6 sent/s for Japanese.

7 Conclusion and Future Directions

This paper demonstrated that character-based translation can act as a unified framework for handling difficult problems in translation: morphology, compound words, transliteration, and segmentation.

One future challenge includes scaling training up to longer sentences, which can likely be achieved through methods such as the heuristic span pruning of Haghighi et al. (2009) or sentence splitting of Vilar et al. (2007). Monolingual data could also be used to improve estimates of our substring-based prior. In addition, error analysis showed that word-based translation performed better than character-based translation on reordering and lexical choice, indicating that improved decoding (or pre-ordering) and language modeling tailored to character-based translation will likely greatly improve accuracy. Finally, we plan to explore the middle ground between word-based and character based translation, allowing for the flexibility of character-based translation, while using word boundary information to increase efficiency and accuracy.

[12] Similar results were found for character and word-based BLEU, but are omitted for lack of space.

References

Mohamed I. Abouelhoda, Stefan Kurtz, and Enno Ohlebusch. 2004. Replacing suffix trees with enhanced suffix arrays. *Journal of Discrete Algorithms*, 2(1).

Yaser Al-Onaizan and Kevin Knight. 2002. Translating named entities using monolingual and bilingual resources. In *Proc. ACL*.

Ming-Hong Bai, Keh-Jiann Chen, and Jason S. Chang. 2008. Improving word alignment by adjusting Chinese word segmentation. In *Proc. IJCNLP*.

Phil Blunsom and Trevor Cohn. 2010. Inducing synchronous grammars with slice sampling. In *Proc. HLT-NAACL*, pages 238–241.

Phil Blunsom, Trevor Cohn, Chris Dyer, and Miles Osborne. 2009. A Gibbs sampler for phrasal synchronous grammar induction. In *Proc. ACL*.

Ondřej Bojar. 2007. English-to-Czech factored machine translation. In *Proc. WMT*.

Peter F. Brown, Vincent J.Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19.

Ralf D. Brown. 2002. Corpus-driven splitting of compound words. In *Proc. TMI*.

Pi-Chuan Chang, Michel Galley, and Christopher D. Manning. 2008. Optimizing Chinese word segmentation for machine translation performance. In *Proc. WMT*.

Tagyoung Chung and Daniel Gildea. 2009. Unsupervised tokenization for machine translation. In *Proc. EMNLP*.

Simon Corston-Oliver and Michael Gamon. 2004. Normalizing German and English inflectional morphology to improve statistical word alignment. *Machine Translation: From Real Users to Research*.

Fabien Cromieres. 2006. Sub-sentential alignment using substring co-occurrence counts. In *Proc. COLING/ACL 2006 Student Research Workshop*.

John DeNero, Alex Bouchard-Côté, and Dan Klein. 2008. Sampling alignment structure under a Bayesian translation model. In *Proc. EMNLP*.

Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. In *Proc. WMT*.

Andrew Finch and Eiichiro Sumita. 2007. Phrase-based machine transliteration. In *Proc. TCAST*.

Sharon Goldwater and David McClosky. 2005. Improving statistical MT through morphological analysis. In *Proc. EMNLP*.

Aria Haghighi, John Blitzer, John DeNero, and Dan Klein. 2009. Better word alignments with supervised ITG models. In *Proc. ACL*.

Mark Johnson, Thomas Griffiths, and Sharon Goldwater. 2007. Bayesian inference for PCFGs via Markov chain Monte Carlo. In *Proc. NAACL*.

Dan Klein and Christopher D. Manning. 2003. A* parsing: fast exact Viterbi parse selection. In *Proc. HLT*.

Kevin Knight and Jonathan Graehl. 1998. Machine transliteration. *Computational Linguistics*, 24(4).

Phillip Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proc. HLT*, pages 48–54.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proc. EMNLP*.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT Summit*.

Grzegorz Kondrak, Daniel Marcu, and Kevin Knight. 2003. Cognates can improve statistical translation models. In *Proc. HLT*.

Young-Suk Lee. 2004. Morphological analysis for statistical machine translation. In *Proc. HLT*.

Klaus Macherey, Andrew Dai, David Talbot, Ashok Popat, and Franz Och. 2011. Language-independent compound splitting with morphological operations. In *Proc. ACL*.

Daniel Marcu and William Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proc. EMNLP*.

Coşkun Mermer and Ahmet Afşın Akın. 2010. Unsupervised search for the optimal segmentation for statistical machine translation. In *Proc. ACL Student Research Workshop*.

Jason Naradowsky and Kristina Toutanova. 2011. Unsupervised bilingual morpheme segmentation and alignment with context-rich hidden semi-Markov models. In *Proc. ACL*.

Graham Neubig, Taro Watanabe, Eiichiro Sumita, Shinsuke Mori, and Tatsuya Kawahara. 2011. An unsupervised model for joint phrase alignment and extraction. In *Proc. ACL*, pages 632–641, Portland, USA, June.

Graham Neubig. 2011. The Kyoto free translation task. http://www.phontron.com/kftt.

ThuyLinh Nguyen, Stephan Vogel, and Noah A. Smith. 2010. Nonparametric word segmentation for machine translation. In *Proc. COLING*.

Sonja Nießen and Hermann Ney. 2000. Improving SMT quality with morpho-syntactic analysis. In *Proc. COLING*.

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

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

Markus Saers, Joakim Nivre, and Dekai Wu. 2009. Learning stochastic bracketing inversion transduction grammars with a cubic time biparsing algorithm. In *Proc. IWPT*, pages 29–32.

Benjamin Snyder and Regina Barzilay. 2008. Unsupervised multilingual learning for morphological segmentation. *Proc. ACL*.

Virach Sornlertlamvanich, Chumpol Mokarat, and Hitoshi Isahara. 2008. Thai-lao machine translation based on phoneme transfer. In *Proc. 14th Annual Meeting of the Association for Natural Language Processing*.

Michael Subotin. 2011. An exponential translation model for target language morphology. In *Proc. ACL*.

David Talbot and Miles Osborne. 2006. Modelling lexical redundancy for machine translation. In *Proc. ACL*.

Jörg Tiedemann. 2009. Character-based PSMT for closely related languages. In *Proc. 13th Annual Conference of the European Association for Machine Translation*.

David Vilar, Jan-T. Peter, and Hermann Ney. 2007. Can we translate letters. In *Proc. WMT*.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. HMM-based word alignment in statistical translation. In *Proc. COLING*.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3).

Hao Zhang and Daniel Gildea. 2005. Stochastic lexicalized inversion transduction grammar for alignment. In *Proc. ACL*.

Hao Zhang, Chris Quirk, Robert C. Moore, and Daniel Gildea. 2008a. Bayesian learning of non-compositional phrases with synchronous parsing. *Proc. ACL*.

Ruiqiang Zhang, Keiji Yasuda, and Eiichiro Sumita. 2008b. Improved statistical machine translation by multiple Chinese word segmentation. In *Proc. WMT*.