Nonparametric Word Segmentation for Machine Translation

ThuyLinh Nguyen Stephan Vogel Noah A. Smith
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA 15213, USA
{thuylinh,vogel,nasmith}@cs.cmu.edu

Abstract

We present an unsupervised word segmentation model for machine translation. The model uses existing monolingual segmentation techniques and models the joint distribution over source sentence segmentations and alignments to the target sentence. During inference, the monolingual segmentation model and the bilingual word alignment model are coupled so that the alignments to the target sentence guide the segmentation of the source sentence. The experiments show improvements on Arabic-English and Chinese-English translation tasks.

1 Introduction

In statistical machine translation, the smallest unit is usually the *word*, defined as a token delimited by spaces. Given a parallel corpus of source and target text, the training procedure first builds a word alignment, then extracts phrase pairs from this word alignment. However, in some languages (e.g., Chinese) there are no spaces between words.

The same problem arises when translating between two very different languages, such as from a language with rich morphology like Hungarian or Arabic to a language with poor morphology like English or Chinese. A single word in a morphologically rich language is often the composition of several morphemes, which correspond to separate words in English.[1]

Often some preprocessing is applied involving word segmentation or morphological analysis of the source and/or target text. Such preprocessing tokenizes the text into morphemes or words, which linguists consider the smallest meaning-bearing units of the language. Take as an example the Arabic word "fktbwha" and its English translation "so they wrote it". The preferred segmentation of "fktbwha" would be "f-ktb-w-ha (*so-wrote-they-it*)," which would allow for a one-to-one mapping between tokens in the two languages. However, the translation of the phrase in Hebrew is "wktbw ath". Now the best segmentation of the Arabic words would be "fktbw-ha," corresponding to the two Hebrew words. This example shows that there may not be one correct segmentation that can be established in a preprocessing step. Rather, tokenization depends on the language we want to translate into and needs to be tied in with the alignment process. In short, we want to find the tokenization yielding the best alignment, and thereby the best translation system.

We propose an unsupervised tokenization method for machine translation by formulating a generative Bayesian model to "explain" the bilingual training data. Generation of a sentence pair is described as follows: first a monolingual tokenization model generates the source sentence, then the alignment model generates the target sentence through the alignments with the source sentence. Breaking this generation process into two steps provides flexibility to incorporate existing monolingual morphological segmentation models such as those of Mochihashi et al. (2009) or Creutz and Lagus (2007). Using nonparametric models and the Bayesian framework makes it possible to incorporate linguistic knowledge as prior

[1] We will use the terms *word segmentation*, *morphological analysis*, and *tokenization* more or less interchangeably.

distributions and obtain the posterior distribution through inference techniques such as MCMC or variational inference.

As new test source sentences do not have translations which can help to infer the best segmentation, we *decode* the source string according to the posterior distribution from the inference step.

In summary, our segmentation technique consists of the following steps:

- A joint model of segmented source text and its target translation.
- Inference of the posterior distribution of the model given the training data.
- A decoding algorithm for segmenting source text.
- Experiments in translation using the preprocessed source text.

Our experiments show that the proposed segmentation method leads to improvements on Arabic-English and Chinese-English translation tasks.

In the next section we will discuss related work. Section 3 will describe our model in detail. The inference will be covered in Section 4, and decoding in Section 5. Experiments and results will be presented in Section 6.

2 Related Work

The problem of segmentation for machine translation has been studied extensively in recent literature. Most of the work used some linguistic knowledge about the source and the target languages (Nießen and Ney, 2004; Goldwater and McClosky, 2005). Sadat and Habash (2006) experimented with a wide range of tokenization schemes for Arabic-English translation. These experiments further show that even for a single language pair, different tokenizations are needed depending on the training corpus size. The experiments are very expensive to conduct and do not generalize to other language pairs. Recently, Dyer (2009) created manually crafted lattices for a subset of source words as references for segmentation when translating into English, and then learned the segmentation of the source words to optimize the translation with respect to these references. He showed that the parameters of the model can be applied to similar languages when translating into English. However, manually creating these lattices is time-consuming and requires a bilingual person with some knowledge of the underlying statistical machine translation system.

There have been some attempts to apply unsupervised methods for tokenization in machine translation (Chung and Gildea, 2009; Xu et al., 2008). The alignment model of Chung and Gildea (2009) forces every source word to align with a target word. Xu et al. (2008) modeled the source-to-null alignment as in the source word to target word model. Their models are special cases of our proposed model when the source model[2] is a unigram model. Like Xu et al. (2008), we use Gibbs sampling for inference. Chung and Gildea (2009) applied efficient dynamic programming-based variational inference algorithms.

We benefit from existing unsupervised monolingual segmentation. The source model uses the nested Pitman-Yor model as described by Mochihashi et al. (2009). When sampling each potential word boundary, our inference technique is a bilingual extension of what is described by Goldwater et al. (2006) for monolingual segmentation.

Nonparametric models have received attention in machine translation recently. For example, DeNero et al. (2008) proposed a hierarchical Dirichlet process model to learn the weights of phrase pairs to address the degeneration in phrase extraction. Teh (2006) used a hierarchical Pitman-Yor process as a smoothing method for language models.

Recent work on multilingual language learning successfully used nonparametric models for language induction tasks such as grammar induction (Snyder et al., 2009; Cohen et al., 2010), morphological segmentation (Goldwater et al., 2006; Snyder and Barzilay, 2008), and part-of-speech tagging (Goldwater and Griffiths, 2007; Snyder et al., 2008).

[2] Note that "source model" here means a model of source text, not a source model in the noisy channel paradigm.

3 Models

We start with the generative process for a source sentence and its alignment with a target sentence. Then we describe individual models employed by this generation scheme.

3.1 Generative Story

A source sentence is a sequence of word tokens, and each word is either aligned or not aligned. We focus only on the segmentation problem and not reordering source words; therefore, the model will not generate the order of the target word tokens. A sentence pair and its alignment are captured by four components:

- a sequence of words in the source sentence,
- a set of null-aligned source tokens,
- a set of null-aligned target tokens, and
- a set of (source word to target word) alignment pairs.

We will start with a high-level story of how the segmentation of the source sentence and the alignment are generated.

1. A source language monolingual segmentation model generates the source sentence.
2. Generate alignments:
   (a) Given the sequence of words of the source sentence already generated in step 1, the alignment model marks each source word as either aligned or unaligned. If a source word is aligned, the model also generates the target word.
   (b) Unaligned target words are generated.

The model defines the joint probability of a segmented source language sentence and its alignment. During inference, the two parts are coupled, so that the alignment will influence which segmentation is selected. However, there are several advantages in breaking the generation process into two steps.

First of all, in principle the model can incorporate any existing probabilistic monolingual segmentation to generate the source sentence. For example, the source model can be the nested Pitman-Yor process as described by Mochihashi et al. (2009), the minimum description length model presented by Creutz and Lagus (2007), or something else. Also the source model can incorporate linguistic knowledge from a rule-based or statistical morphological disambiguator.

The model generates the alignment after the source sentence with word boundaries already generated. Therefore, the alignment model can be any existing word alignment model (Brown et al., 1993; Vogel et al., 1996). Even though the choices of source model or alignment model can lead to different inference methods, the model we propose here is highly extensible. Note that we assume that the alignment consists of at most one-to-one mappings between source and target words, with null alignments possible on both sides.

Another advantage of a separate source model lies in the segmentation of an unseen test set. In section 5 we will show how to apply the source model distribution learned from training data to find the best segmentation of an unseen test set.

Notation and Parameters

We will use bold font for a sequence or bags of words and regular font for an individual word. A source sentence $\mathbf{s}$ is a sequence of $|\mathbf{s}|$ words s_i: $\left(s_1, \ldots, s_{|\mathbf{s}|}\right)$; the translation of sentence $\mathbf{s}$ is the target sentence $\mathbf{t}$ of $|\mathbf{t}|$ words $\left(t_1, \ldots, t_{|\mathbf{t}|}\right)$. In sentence $\mathbf{s}$ the list of unaligned words is $\mathbf{s}_{\mathrm{nal}}$ and the list of aligned source words is $\mathbf{s}_{\mathrm{al}}$. In the target sentence $\mathbf{t}$ the list of unaligned words is $\mathbf{t}_{\mathrm{nal}}$ and the list of target words having one-to-one alignment with source words $\mathbf{s}_{\mathrm{al}}$ is $\mathbf{t}_{\mathrm{al}}$. The alignment $\mathbf{a}$ of $\mathbf{s}$ and $\mathbf{t}$ is represented by $\{\langle s_i, \mathsf{null}\rangle \mid s_i \in \mathbf{s}_{\mathrm{nal}}\} \cup \{\langle s_i, t_{a_i}\rangle \mid s_i \in \mathbf{s}_{\mathrm{al}}; t_{a_i} \in \mathbf{t}_{\mathrm{al}}\} \cup \{\langle \mathsf{null}, t_j\rangle \mid t_j \in \mathbf{t}_{\mathrm{nal}}\}$ where a_i denotes the index in $\mathbf{t}$ of the word aligned to s_i.

The probability of a sequence or a set is denoted by $\mathbb{P}\left(.\right)$, probability at the word level is $\mathsf{p}\left(.\right)$. For example, the probability of sentence $\mathbf{s}$ is $\mathbb{P}\left(\mathbf{s}\right)$, the probability of a word s is $\mathsf{p}\left(s\right)$, the probability that the target word t aligns to an aligned source word s is $\mathsf{p}\left(t \,|\, s\right)$.

A sentence pair and its alignment are generated from the following models:

- The *source* model generates sentence $\mathbf{s}$ with probability $\mathbb{P}(\mathbf{s})$.

- The *source-to-null* alignment model decides independently for each word s whether it is unaligned with probability $\mathsf{p}(\mathsf{null} \mid s_i)$ or aligned with probability: $1 - \mathsf{p}(\mathsf{null} \mid s_i)$. The probability of this step, for all source words, is: $\mathbb{P}(\mathbf{s}_{\text{nal}}, \mathbf{s}_{\text{al}} \mid \mathbf{s}) = \prod_{s_i \in \mathbf{s}_{\text{nal}}} \mathsf{p}(\mathsf{null} \mid s_i) \times \prod_{s_i \in \mathbf{s}_{\text{al}}} (1 - \mathsf{p}(\mathsf{null} \mid s_i))$.

  We will also refer to the source-to-null model as the *deletion model*, since words in $\mathbf{s}_{\text{nal}}$ are effectively deleted for the purposes of alignment.

- The *source-to-target* alignment model generates a bag of target words $\mathbf{t}_{\text{al}}$ aligned to the source words $\mathbf{s}_{\text{al}}$ with probability: $\mathbb{P}(\mathbf{t}_{\text{al}} \mid \mathbf{s}_{\text{al}}) = \prod_{s_i \in \mathbf{s}_{\text{al}}; t_{a_i} \in \mathbf{t}_{\text{al}}} \mathsf{p}(t_{a_i} \mid s_i)$. Note that we do not need to be concerned with generating $\mathbf{a}$ explicitly, since we do not model word order on the target side.

- The *null-to-target* alignment model generates the list of unaligned target words $\mathbf{t}_{\text{nal}}$ given aligned target words $\mathbf{t}_{\text{al}}$ with $\mathbb{P}(\mathbf{t}_{\text{nal}} \mid \mathbf{t}_{\text{al}})$ as follows:

  - Generate the number of unaligned target words $|\mathbf{t}_{\text{nal}}|$ given the number of aligned target words $|\mathbf{t}_{\text{al}}|$ with probability $\mathbb{P}(|\mathbf{t}_{\text{nal}}| \mid |\mathbf{t}_{\text{al}}|)$.
  - Generate $|\mathbf{t}_{\text{nal}}|$ unaligned words $t \in \mathbf{t}_{\text{nal}}$ independently, each with probability $\mathsf{p}(t \mid \mathsf{null})$.

  The resulting null-to-target probability is therefore: $\mathbb{P}(\mathbf{t}_{\text{nal}} \mid \mathbf{t}_{\text{al}}) = \mathbb{P}(|\mathbf{t}_{\text{nal}}| \mid |\mathbf{t}_{\text{al}}|) \prod_{t \in \mathbf{t}_{\text{nal}}} \mathsf{p}(t \mid \mathsf{null})$.

  We also call the null-to-target model the *insertion* model.

The above generation process defines the joint probability of source sentence $\mathbf{s}$ and its alignment $\mathbf{a}$ as follows:

$$\mathbb{P}(\mathbf{s}, \mathbf{a}) = \underbrace{\mathbb{P}(\mathbf{s})}_{\text{source model}} \times \underbrace{\mathbb{P}(\mathbf{a} \mid \mathbf{s})}_{\text{alignment model}} \quad (1)$$

$$\mathbb{P}(\mathbf{a} \mid \mathbf{s}) = \mathbb{P}(\mathbf{t}_{\text{al}} \mid \mathbf{s}_{\text{al}}) \times \mathbb{P}(\mathbf{t}_{\text{nal}} \mid \mathbf{t}_{\text{al}}) \quad (2)$$
$$\times \prod_{s_i \in \mathbf{s}_{\text{nal}}} \mathsf{p}(\mathsf{null} \mid s_i) \times \prod_{s_i \in \mathbf{s}_{\text{al}}} (1 - \mathsf{p}(\mathsf{null} \mid s_i))$$

3.2 Source Model

Our generative process provides the flexibility of incorporating different monolingual models into the probability distribution of a sentence pair. In particular we use the existing state-of-the-art nested Pitman-Yor n-gram language model as described by Mochihashi et al. (2009). The probability of $\mathbf{s}$ is given by

$$\mathbb{P}(\mathbf{s}) = \mathbb{P}(|\mathbf{s}|) \prod_{i=1}^{|\mathbf{s}|} \mathsf{p}(s_i \mid s_{i-n}, \ldots, s_{i-1}) \quad (3)$$

where the n-gram probability is a hierarchical Pitman-Yor language model using $(n-1)$-gram as the base distribution.

At the unigram level, the model uses the base distribution $\mathsf{p}(s)$ as the infinite-gram character-level Pitman-Yor language model.

3.3 Modeling Null-Aligned Source Words

The probability that a source word aligns to null $\mathsf{p}(\mathsf{null} \mid s)$ is defined by a binomial distribution with Beta prior $\text{Beta}(\alpha p, \alpha(1-p))$, where α and p are model parameters. When $p \to 0$ and $\alpha \to \infty$ the probability $\mathsf{p}(\mathsf{null} \mid s)$ converges to 0 forcing each source words align to a target word. We fixed $p = 0.1$ and $\alpha = 20$ in our experiment.

Xu et al. (2008) view the null word as another target word, hence in their model the probability that a source word aligns to null can only depend on itself.

By modeling the source-to-null alignment separately, our model lets the distribution depend on the word's n-gram context as in the source model. $\mathsf{p}(\mathsf{null} \mid s_{i-n}, \ldots, s_i)$ stands for the probability that the word s_i is not aligned given its context $(s_{i-n}, \ldots, s_{i-1})$.

The n-gram source-to-null distribution $\mathsf{p}(\mathsf{null} \mid s_{i-n}, \ldots, s_i)$ is defined similarly to $\mathsf{p}(\mathsf{null} \mid s_i)$ definition above in which the base distribution p now becomes the $(n-1)$-gram: $\mathsf{p}(\mathsf{null} \mid s_{i-n+1}, \ldots, s_i)$.[3]

[3] We also might have conditioned this decision on words *following* s_i, since those have all been generated already at this stage.

3.4 Source-Target Alignment Model

The probability $\mathsf{p}\,(t\,|\,s)$ that a target word t aligns to a source word s is a Pitman-Yor process:

$$t \mid s \sim \mathrm{PY}\,(d, \alpha, \mathsf{p}_0\,(t\,|\,s))$$

here d and α are the input parameters, and $\mathsf{p}_0\,(t\,|\,s)$ is the base distribution.

Let $|s,\cdot|$ denote the number of times s is aligned to any t in the corpus and let $|s,t|$ denote the number of times s is aligned to t anywhere in the corpus. And let $\mathsf{ty}(s)$ denote the number of different target words t the word s is aligned to anywhere in the corpus. In the Chinese Restaurant Process metaphor, there is one restaurant for each source word s, the s restaurant has $\mathsf{ty}(s)$ tables and total $|s,\cdot|$ customers; table t has $|s,t|$ customers.

Then, at a given time in the generative process for the corpus, we can write the probability that t is generated by the word s as:

- if $|s,t| > 0$:

$$\mathsf{p}\,(t\,|\,s) = \frac{|s,t| - d + [\alpha + d\mathsf{ty}(s)]\mathsf{p}_0\,(t\,|\,s)}{|s,\cdot| + \alpha}$$

- if $|s,t| = 0$:

$$\mathsf{p}\,(t\,|\,s) = \frac{[\alpha + d\mathsf{ty}(s)]\mathsf{p}_0\,(t\,|\,s)}{|s,\cdot| + \alpha}$$

For language pairs with similar character sets such as English and French, words with similar surface form are often translations of each other. The base distribution can be defined based on the edit distance between two words (Snyder and Barzilay, 2008).

We are working with diverse language pairs (Arabic-English and Chinese-English), so we use the base distribution as the flat distribution $\mathsf{p}_0\,(t\,|\,s) = \frac{1}{T}$; T is the number of distinct target words in the training set. In our experiment, the model parameters are $\alpha = 20$ and $d = .5$.

3.5 Modeling Null-Aligned Target Words

The null-aligned target words are modeled conditioned on previously generated target words as:

$$\mathbb{P}\,(\mathbf{t}_{\mathrm{nal}}\,|\,\mathbf{t}_{\mathrm{al}}) = \mathbb{P}\,(|\mathbf{t}_{\mathrm{nal}}| \mid |\mathbf{t}_{\mathrm{al}}|) \prod_{t \in \mathbf{t}_{\mathrm{nal}}} \mathsf{p}\,(t\,|\,\mathsf{null})$$

This model uses two probability distributions:

- the number of unaligned target words: $\mathbb{P}\,(|\mathbf{t}_{\mathrm{nal}}| \mid |\mathbf{t}_{\mathrm{al}}|)$, and
- the probability that each word in $\mathbf{t}_{\mathrm{nal}}$ is generated by null: $\mathsf{p}\,(t\,|\,\mathsf{null})$.

We model the number of unaligned target words similarly to the distribution in the IBM3 word alignment model (Brown et al., 1993). IBM3 assumes that each aligned target words generates a null-aligned target word with probability p_0 and fails to generate a target word with probability $1 - p_0$. So the parameter p_0 can be used to control the number of unaligned target words. In our experiments, we fix $p_0 = .05$. Following this assumption, the probability of $|\mathbf{t}_{\mathrm{nal}}|$ unaligned target words generated from $|\mathbf{t}_{\mathrm{al}}|$ words is: $\mathbb{P}\,(|\mathbf{t}_{\mathrm{nal}}| \mid |\mathbf{t}_{\mathrm{al}}|) = \binom{|\mathbf{t}_{\mathrm{al}}|}{|\mathbf{t}_{\mathrm{nal}}|} p_0^{|\mathbf{t}_{\mathrm{nal}}|}(1 - p_0)^{|\mathbf{t}_{\mathrm{al}}| - |\mathbf{t}_{\mathrm{nal}}|}$.

The probability that a target word t aligns to null, $\mathsf{p}\,(t\,|\,\mathsf{null})$, also has a Pitman-Yor process prior. The base distribution of the model is similar to the source-to-target model's base distribution which is the flat distribution over target words.

4 Inference

We have defined a probabilistic generative model to describe how a corpus of alignments and segmentations can be generated jointly. In this section we discuss how to obtain the posterior distributions of the missing alignments and segmentations given the training corpus, using Gibbs sampling.

Suppose we are provided a morphological disambiguator for the source language such as MADA morphology tokenization toolkit (Sadat and Habash, 2006) for Arabic.[4] The morphological disambiguator segments a source word to morphemes of smallest meaning-bearing units of the source language. Therefore, a target word is equivalent to one or several morphemes. Given a morphological disambiguation toolkit, we use its output to bias our inference by not considering word boundaries after every character but only

[4] MADA provides several segmentation schemes; among them the MADA-D3 scheme seeks to separate all morphemes of each word.

considering potential word boundaries as a subset of the morpheme boundaries set. In this way, the inference uses the morphological disambiguation toolkit to limit its search space.

The inference starts with an initial segmentation of the source corpus and also its alignment to the target corpus. The Gibbs sampler considers one potential word boundary at a time. There are two hypotheses at any given boundary position of a sentence pair $(\mathbf{s}, \mathbf{t})$: the *merge* hypothesis stands for no word boundary and the resulting source sentence $\mathbf{s}_{\mathrm{merge}}$ has a word s spanning over the sample point; the *split* hypothesis indicates the resulting source sentence $\mathbf{s}_{\mathrm{split}}$ has a word boundary at the sample point separating two words $s_1 s_2$. Similar to Goldwater et al. (2006) for monolingual segmentation, the sampler randomly chooses the boundary according to the relative probabilities of the merge hypothesis and the split hypothesis.

The model consists of source and alignment model variables; given the training corpora size of a machine translation system, the number of variables is large. So if the Gibbs sampler samples both source variables and alignment variables, the inference requires many iterations until the sampler mixes. Xu et al. (2008) fixed this by repeatedly applying GIZA++ word alignment after each sampling iteration through the training corpora.

Our inference technique is not precisely Gibbs sampling. Rather than sampling the alignment or attempting to collapse it out (by summing over all possible alignments when calculating the relative probabilities of the merge and split hypotheses), we seek the *best* alignment for each hypothesis. In other words, for each hypothesis, we perform a local search for a high-probability alignment of the merged word or split words, given the rest of alignment for the sentence. Up to one word may be displaced and realigned. This “local-best” alignment is used to score the hypothesis, and after sampling merge or split, we keep that best alignment.

This inference technique is motivated by runtime demands, but we do not yet know of a theoretical justification for combining random steps with maximization over some variables. A more complete analysis is left to future work.

5 Decoding for Unseen Test Sentences

Section 4 described how to get the model’s posterior distribution and the segmentation and alignment of the training data under the model. We are left with the problem of *decoding* or finding the segmentation of test sentences where the translations are not available. This is needed when we want to translate new sentences. Here, tokenization is performed as a preprocessing step, decoupled from the subsequent translation steps.

The decoding step uses the model’s posterior distribution for the training data to segment unseen source sentences. Because of the clear separation of the source model and the alignment model, the source model distribution learned from the Gibbs sampling directly represents the distribution over the source language and can therefore also handle the segmentation of unknown words in new test sentences. Only the source model is used in preprocessing.

The best segmentation $\mathbf{s}^*$ of a string of characters $\mathbf{c} = \left(c_1, \ldots, c_{|\mathbf{c}|}\right)$ according to the n-gram source model is:

$$\mathbf{s}^* = \underset{\mathbf{s}\text{ from }\mathbf{c}}{\operatorname{argmax}} \, \mathsf{p}\left(|\mathbf{s}|\right) \prod_{i=1}^{i=|\mathbf{s}|} \mathsf{p}\left(s_i \,|\, s_{i-n}, \ldots, s_{i-1}\right)$$

We use a stochastic finite-state machine for decoding. This is possible by composition of the following two finite state machines:

- Acceptor $\mathcal{A}_{\mathbf{c}}$. The string of characters $\mathbf{c}$ is represented as an finite state acceptor machine where any path through the machine represents an unweighted segmentation of $\mathbf{c}$.

- Source model weighted finite state transducer $\mathcal{L}_{\mathbf{c}}$. Knight and Al-Onaizan (1998) show how to build an n-gram language model by a weighted finite state machine. The states of the transducer are $(n-1)$-gram history, the edges are words from the language. The arc s_i coming from state $(s_{i-n}, \ldots, s_{i-1})$ to state $(s_{i-n+1}, \ldots, s_i)$ has weight $\mathsf{p}\left(s_i \,|\, s_{i-n}, \ldots, s_{i-1}\right)$.

The best segmentation $\mathbf{s}^*$ is given as $\mathbf{s}^* = \mathrm{BestPath}(\mathcal{A}_{\mathbf{c}} \circ \mathcal{L}_{\mathbf{c}})$.

6 Experiments

This section presents experimental results on Arabic-English and Chinese-English translation tasks using the proposed segmentation technique.

6.1 Arabic-English

As a training set we use the BTEC corpus distributed by the International Workshop on Spoken Language Translation (IWSLT) (Matthias and Chiori, 2005). The corpus is a collection of conversation transcripts from the travel domain. The "Supplied Data" track consists of nearly 20K Arabic-English sentence pairs. The development set consists of 506 sentences from the IWSLT04 evaluation test set and the unseen set consists of 500 sentences from the IWSLT05 evaluation test set. Both development set and test set have 16 references per Arabic sentence.

6.2 Chinese-English

The training set for Chinese-English translation task is also distributed by the IWSLT evaluation campaign. It consists of 67K Chinese-English sentence pairs. The development set and the test set each have 489 Chinese sentences and each sentence has 7 English references.

6.3 Results

We will report the translation results where the preprocessing of the source text are our unigram, bigram, and trigram source models and source-to-null model.

The MCMC inference algorithm starts with an initial segmentation of the source text into full word forms. For Chinese, we use the original word segmentation as distributed by IWSLT. To get an initial alignment, we generate the IBM4 Viterbi alignments in both directions using the GIZA++ toolkit (Och and Ney, 2003) and combine them using the "grow-diag-final-and" heuristic. The output of combining GIZA++ alignment for a sentence pair is a sequence of s_i-t_j entries where i is an index of the source sentence and j is an index of the target sentence. As our model allows only one-to-one mappings between the words in the source and target sentences, we remove s_i-t_j from the sequence if either the source word s_i or target word t_j is already in a previous entry of the combined alignment sequence. The resulting alignment is our initial alignment for the inference.

We also apply the MADA morphology segmentation toolkit (Habash and Rambow, 2005) to preprocess the Arabic corpus. We use the D3 scheme (each Arabic word is segmented into morphemes in sequence [CONJ+ [PART+ [Al+ BASE +PRON]]]), mark the morpheme boundaries, and then combine the morphemes again to have words in their original full word form. During inference, we only sample over these morpheme boundaries as potential word boundaries. In this way, we limit the search space, allowing only segmentations consistent with MADA-D3.

The inference samples 150 iterations through the whole training set and uses the posterior probability distribution from the last iteration for decoding. The decoding process is then applied to the entire training set as well as to the development and test sets to generate a consistent tokenization across all three data sets. We used the OpenFST toolkit (Allauzen et al., 2007) for finite-state machine implementation and operations. The output of the decoding is the preprocessed data for translation. We use the open source Moses phrase-based MT system (Koehn et al., 2007) to test the impact of the preprocessing technique on translation quality.[5]

6.3.1 Arabic-English Translation Results

	Dev.	Test
Original	59.21	54.00
MADA-D3	58.28	54.92
Unigram	**59.44**	56.18
Bigram	58.88	56.18
Trigram	58.76	**56.82**

Table 1: Arabic-English translation results (BLEU).

We consider the Arabic-English setting. We use two baselines: original full word form and MADA-D3 tokenization scheme for Arabic-English translation. Table 1 compares the translation results of our segmentation methods with

[5] The Moses translation alignment is the output of GIZA++, not from our MCMC inference.

these baselines. Our segmentation method shows improvement over the two baselines on both the development and test sets. According to Sadat and Habash (2006), the MADA-D3 scheme performs best for their Arabic-English translation especially for small and moderate data sizes. In our experiments, we see an improvement when using the MADA-D3 preprocessing over using the original Arabic corpus on the unseen test set, but not on the development set.

The Gibbs sampler only samples on the morphology boundary points of MADA-D3, so the improvement resulting from our segmentation technique does not come from removing unknown words. It is due to a better matching between the source and target sentences by integrating segmentation and alignment. We therefore expect the same impact on a larger training data set in future experiments.

6.3.2 Chinese-English Translation Results

	Dev.	Test
Whole word	23.75	**29.02**
Character	23.39	27.74
Unigram	**24.90**	**28.97**
Trigram	23.98	28.20

Table 2: Chinese-English translation result in BLEU score metric.

We next consider the Chinese-English setting. The translation performance using our word segmentation technique is shown in Table 2. There are two baselines for Chinese-English translation: (a) the source text in the full word form distributed by the IWSLT evaluation and (b) no segmentation of the source text, which is equivalent to interpreting each Chinese character as a single word.

Taking development and test sets into account, the best Chinese-English translation system results from our unigram model. It is significantly better than other systems on the development set and performs almost equally well with the IWSLT segmentation on the test set. Note that the segmentation distributed by IWSLT is a manual segmentation for the translation task.

Chung and Gildea (2009) and Xu et al. (2008) also showed improvement over a simple monolingual segmentation for Chinese-English translation. Our character-based translation result is comparable to their monolingual segmentations. Both trigram and unigram translation results outperform the character-based translation.

We also observe that there are no additional gains for Chinese-English translation when using a higher n-gram model. Our Gibbs sampler has the advantage that the samples are guaranteed to converge eventually to the model's posterior distributions, but in each step the modification to the current hypothesis is small and local. In iterations 100–150, the average number of boundary changes for the unigram model is 14K boundaries versus only 1.5K boundary changes for the trigram model. With 150 iterations, the inference output of trigram model might not yet represent its posterior distribution. We leave a more detailed investigation of convergence behavior to future work.

Conclusion and Future Work

We presented an unsupervised segmentation method for machine translation and presented experiments for Arabic-English and Chinese-English translation tasks. The model can incorporate existing monolingual segmentation models and seeks to learn a segmenter appropriate for a particular translation task (target language and dataset).

Acknowledgements

We thank Kevin Gimpel for interesting discussions and technical advice. We also thank the anonymous reviewers for useful feedback. This work was supported by DARPA Gale project, NSF grants 0844507 and 0915187.

References

Allauzen, C., M. Riley, J. Schalkwyk, W. Skut, and M. Mohri. 2007. OpenFst: A General and Efficient Weighted Finite-State Transducer Library. In *Proceedings of the CIAA 2007*, volume 4783 of *Lecture Notes in Computer Science*, pages 11–23. Springer. `http://www.openfst.org`.

Brown, Peter F., Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The

Mathematics of Statistical Machine Translation: Parameter Estimation. *Comput. Linguist.*, 19(2):263–311.

Chung, T. and D. Gildea. 2009. Unsupervised Tokenization for Machine Translation. In *Proceedings of EMNLP 2009*, pages 718–726, Singapore, August. Association for Computational Linguistics.

Cohen, S. B., D. M. Blei, and N. A. Smith. 2010. Variational Inference for Adaptor Grammars. In *Proceedings of NAACL-HLT*, pages 564–572, June.

Creutz, Mathias and Krista Lagus. 2007. Unsupervised Models for Morpheme Segmentation and Morphology Learning. *ACM Trans. Speech Lang. Process.*, 4(1):1–34.

DeNero, J., A. Bouchard-Côté, and D. Klein. 2008. Sampling Alignment Structure under a Bayesian Translation Model. In *Proceedings of EMNLP 2008*, pages 314–323, Honolulu, Hawaii, October. Association for Computational Linguistics.

Dyer, C. 2009. Using a Maximum Entropy model to build segmentation lattices for MT. In *Proceedings of HLT 2009*, pages 406–414, Boulder, Colorado, June.

Goldwater, S. and T. L. Griffiths. 2007. A Fully Bayesian Approach to Unsupervised Part-of-Speech Tagging. In *Proceedings of ACL*.

Goldwater, S. and D. McClosky. 2005. Improving Statistical Machine Translation Through Morphological Analysis. In *Proc. of EMNLP*.

Goldwater, S., T. L. Griffiths, and M. Johnson. 2006. Contextual Dependencies in Unsupervised Word Segmentation. In *Proc. of COLING-ACL*.

Habash, N. and O. Rambow. 2005. Arabic Tokenization, Part-of-Speech Tagging, and Morphological Disambiguation in One Fell Swoop. In *Proc. of ACL*.

Knight, K. and Y. Al-Onaizan. 1998. Translation with Finite-State Devices. In *Proceedings of AMTA*, pages 421–437.

Koehn, P., H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proc. of ACL (demo session)*.

Matthias, E. and H. Chiori. 2005. Overview of the IWSLT 2005 Evaluation Campaign. In *Proceedings of IWSLT*.

Mochihashi, D., T. Yamada, and N. Ueda. 2009. Bayesian Unsupervised Word Segmentation with Nested Pitman-Yor Language Modeling. In *Proceedings of 47th ACL*, pages 100–108, Suntec, Singapore, August.

Nießen, S. and H. Ney. 2004. Statistical Machine Translation with Scarce Resources Using Morpho-Syntactic Information. *Computational Linguistics*, 30(2), June.

Och, F. and H. Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1).

Sadat, F. and N. Habash. 2006. Combination of Arabic Preprocessing Schemes for Statistical Machine Translation. In *Proceedings of the ACL*, pages 1–8.

Snyder, B. and R. Barzilay. 2008. Unsupervised Multilingual Learning for Morphological Segmentation. In *Proceedings of ACL-08: HLT*, pages 737–745, June.

Snyder, B., T. Naseem, J. Eisenstein, and R. Barzilay. 2008. Unsupervised Multilingual Learning for POS Tagging. In *Proceedings of EMNLP*.

Snyder, B., T. Naseem, and R. Barzilay. 2009. Unsupervised Multilingual Grammar Induction. In *Proceedings of ACL-09*, pages 73–81, August.

Teh, Y. W. 2006. A Hierarchical Bayesian Language Model Based On Pitman-Yor Processes. In *Proceedings of ACL*, pages 985–992, July.

Vogel, S., H. Ney, and C. Tillmann. 1996. HMM-Based Word Alignment in Statistical Translation. In *Proceedings of COLING*, pages 836–841.

Xu, J., J. Gao, K. Toutanova, and H. Ney. 2008. Bayesian Semi-Supervised Chinese Word Segmentation for Statistical Machine Translation. In *Proceedings of (Coling 2008)*, pages 1017–1024, Manchester, UK, August.