# Integration of Multiple Bilingually-Learned Segmentation Schemes into Statistical Machine Translation

**Michael Paul and Andrew Finch and Eiichiro Sumita**
MASTAR Project
National Institute of Information and Communications Technology
Hikaridai 2-2-2, Keihanna Science City
619-0288 Kyoto, Japan
michael.paul@nict.go.jp

## Abstract

This paper proposes an unsupervised word segmentation algorithm that identifies word boundaries in continuous source language text in order to improve the translation quality of statistical machine translation (SMT) approaches. The method can be applied to any language pair where the source language is unsegmented and the target language segmentation is known. First, an iterative bootstrap method is applied to learn multiple segmentation schemes that are consistent with the phrasal segmentations of an SMT system trained on the resegmented bitext. In the second step, multiple segmentation schemes are integrated into a single SMT system by characterizing the source language side and merging identical translation pairs of differently segmented SMT models. Experimental results translating five Asian languages into English revealed that the method of integrating multiple segmentation schemes outperforms SMT models trained on any of the learned word segmentations and performs comparably to available state-of-the-art monolingually-built segmentation tools.

## 1 Introduction

The task of *word segmentation*, i.e., identifying word boundaries in continuous text, is one of the fundamental preprocessing steps of data-driven NLP applications like *Machine Translation* (MT). In contrast to Indo-European languages like English, many Asian languages like Chinese do not use a whitespace character to separate meaningful word units. The problems of word segmentation are:

(1) *ambiguity*, e.g., for Chinese, a single character can be a word component in one context, but a word by itself in another context.

(2) *unknown words*, i.e., existing words can be combined into new words such as proper nouns, e.g. "*White House*".

Purely dictionary-based approaches like (Cheng et al., 1999) addressed these problems by maximum matching heuristics. Recent research on unsupervised word segmentation focuses on approaches based on probabilistic methods. For example, (Brent, 1999) proposes a probabilistic segmentation model based on unigram word distributions, whereas (Venkataraman, 2001) uses standard n-gram language models. An alternative non-parametric Bayesian inference approach based on the Dirichlet process incorporating unigram and bigram word dependencies is introduced in (Goldwater et al., 2006).

The focus of this paper, however, is to learn word segmentations that are *consistent with phrasal segmentations of SMT translation models*. In case of small translation units, e.g. single Chinese or Japanese characters, it is likely that such tokens have been seen in the training corpus, thus these tokens can be translated by an SMT engine. However, the contextual information provided by these tokens might not be enough to obtain a good translation. For example, a Japanese-English SMT engine might translate the two successive characters "白" ("white") and "鳥" ("bird") as "*white bird*", while a human would translate "白鳥" as "*swan*". Therefore, the longer the translation unit, the more context can be exploited to find a meaningful translation. On the other hand, the longer the translation unit, the less likely it is that such a token will occur in the training data due to *data sparseness* of the language resources utilized to train the statistical translation models. Therefore, a word segmentation that is



*Proceedings of the Joint 5th Workshop on Statistical Machine Translation and MetricsMATR*, pages 400–408,
Uppsala, Sweden, 15-16 July 2010. 

"consistent with SMT models" is one that identifies translation units that are small enough to be translatable, but large enough to be meaningful in the context of the given input sentence, achieving a trade-off between the *coverage* and the *translation task complexity* of the statistical models in order to improve translation quality.

The use of monolingual probabilistic models does not necessarily yield a better MT performance (Chang et al., 2008). However, improvements have been reported for approaches taking into account not only monolingual, but also bilingual information, to derive a word segmentation suitable for SMT. Due to the availability of language resources, most recent research has focused on optimizing Chinese word segmentation (CWS) for Chinese-to-English SMT. For example, (Xu et al., 2008) proposes a Bayesian Semi-Supervised approach for CWS that builds on (Goldwater et al., 2006). The generative model first segments Chinese text using an off-the-shelf segmenter and then learns new word types and word distributions suitable for SMT. Similarly, a dynamic programming-based variational Bayes approach using bilingual information to improve MT is proposed in (Chung and Gildea, 2009). Concerning other languages, for example, (Kikui and Yamamoto, 2002) extended Hidden-Markov-Models, where hidden n-gram probabilities were affected by co-occurring words in the target language part for Japanese word segmentation.

Recent research on SMT is also focusing on the usage of multiple word segmentation schemes for the source language to improve translation quality. For example, (Zhang et al., 2008) combines dictionary-based and CRF-based approaches for Chinese word segmentation in order to avoid *out-of-vocabulary* (OOV) words. Moreover, the combination of different morphological decomposition of highly inflected languages like Arabic or Finnish is proposed in (de Gispert et al., 2009) to reduce the data sparseness problem of SMT approaches. Similarly, (Nakov et al., 2009) utilizes SMT engines trained on different word segmentation schemes and combines the translation outputs using system combination techniques as a post-process to SMT decoding.

In order to integrate multiple word segmentation schemes into the SMT decoder, (Dyer et al., 2008) proposed to generate word lattices covering all possible segmentations of the input sentence and to decode the lattice input. An extended version of the lattice approach that does not require the use (and existence) of monolingual segmentation tools was proposed in (Dyer, 2009) where a maximum entropy model is used to assign probabilities to the segmentations of an input word to generate diverse segmentation lattices from a single automatically learned model.

The method of (Ma and Way, 2009) also uses a word lattice decoding approach, but they iteratively extract multiple word segmentation schemes from the training bitext. This dictionary-based approach uses heuristics based on the maximum matching algorithm to obtain an agglomeration of segments that are covered by the dictionary. It uses all possible source segmentations that are consistent with the extracted dictionary to create a word lattice for decoding.

The method proposed in this papers differs from previous approaches in the following points:

- it works for any language pair where the source language is unsegmented and the target language segmentation is known.
- it can be applied for the translation of a source language where no linguistically motivated word segmentation tools are available.
- it applies machine learning techniques to identify segmentation schemes that improve translation quality for a given language pair.
- it decodes directly from unsegmented text using segmentation information implicit in the phrase-table to generate the target and thus avoids issues of consistency between phrase-table and input representation.
- it uses segmentations at all iterative levels of the bootstrap process, rather than only those from the final iteration allowing the consideration of segmentations from many levels of granularity.

Word segmentations are learned using a parallel corpus by aligning character-wise source language sentences to word units separated by a white-space in the target language. Successive characters aligned to the same target words are merged into a larger source language unit. Therefore, the granularity of the translation unit is defined in the given bitext context. In order to minimize the side effects of alignment errors and to achieve segmentation consistency, a Maximum-Entropy (ME) algorithm is applied to learn the source language word

segmentation that is consistent with the translation model of an SMT system trained on the resegmented bitext. The process is iterated until no further improvement in translation quality is achieved. In order to integrate multiple word segmentation into a single SMT system, the statistical translation models trained on differently segmented source language corpora are merged by characterizing the source side of each translation model, summing up the probabilities of identical phrase translation pairs, and rescoring the merged translation model (see Section 2).

The proposed segmentation method is applied to the translation of five Asian languages, i.e., Japanese, Korean, Thai, and two Chinese dialects (Standard Mandarin and Taiwanese Mandarin), into English. The utilized language resources and the outline of the experiments are summarized in Section 3. The experimental results revealed that the proposed method outperforms not only a baseline system that translates characterized source language sentences, but also all SMT models trained on any of the learned word segmentations. In addition, the proposed method achieves translation results comparable to SMT models trained on linguistically segmented bitext.

## 2 Word Segmentation

The word segmentation method proposed in this paper is an unsupervised, language-independent approach that treats the task of word segmentation as a *phrase-boundary tagging* task. This method uses a parallel text corpus consisting of initially unigram segmented source language character sequences and whitespace-separated target language words. The initial bitext is used to train a standard phrase-based SMT system ($SMT_{chr}$). The character-to-word alignment results of the SMT training procedure[1] are exploited to identify successive source language characters aligned to the same target language word in the respective bitext and to merge these characters into larger translation units, defining its granularity in the given bitext context.

The obtained translation units are then used to learn the word segmentation that is most consistent with the phrase alignments of the given SMT system. First, each character of the source language text is annotated with a word-boundary indicator where only two tags are used, i.e, *"E"* (end-of-word character tag) and *"I"* (in-word character tag). The annotations are derived from the SMT training corpus as described in Figure 1.

```
(1)  proc annotate-phrase-boundaries( Bitext ) ;
(2)  begin
(3)    for each (Src, Trg) in {Bitext} do
(4)      A ← align(Src, Trg) ;
(5)      for each i in {1, ..., len(Src)-1} do
(6)        Trg_i ← get-target(Src[i], A) ;
(7)        Trg_{i+1} ← get-target(Src[i+1], A) ;
(8)        if null(Trg_i) or Trg_i ≠ Trg_{i+1} then
(9)          (* aligned to none or different target *)
(10)         Src_ME ← assign-tag(Src[i],' E') ;
(11)       else
(12)         (* aligned to the same target *)
(13)         Src_ME ← assign-tag(Src[i],' I') ;
(14)       fi ;
(15)       Corpus_ME ← add(Src_ME) ;
(16)     od ;
(17)     (* last source token *)
(18)     LastSrc_ME ← assign-tag(Src[len(Src)],' E') ;
(19)     Corpus_ME ← add(LastSrc_ME) ;
(20)   od ;
(21)   return( Corpus_ME ) ;
(22) end ;
```

Figure 1: ME Training Data Annotation

Using these alignment-based word boundary annotations, a Maximum-Entropy (ME) method is applied to learn the word segmentation consistent with the SMT translation model (see Section 2.1), to resegment the original source language corpus, and to retrain a phrase-based SMT engine that will hopefully achieve a better translation performance than the initial SMT engine. This process should be repeated as long as an improvement in translation quality is achieved. Eventually, the concatenation of succeeding translation units will result in overfitting, i.e., the newly created token can only be translated in the context of rare training data examples. Therefore, a lower translation quality due to an increase of untranslatable source language phrases is to be expected (see Section 2.2).

However, in order to increase the *coverage* and to reduce the *translation task complexity* of the statistical models, the proposed method integrates multiple segmentation schemes into the statistical translation models of a single SMT engine so that longer translation units are preferred for translation, if available, and smaller translation units can be used otherwise (see Section 2.3).

### 2.1 Maximum-Entropy Tagging Model

ME models provide a general purpose machine learning technique for classification and predic-

[1] For the experiments presented in Section 3, the GIZA++ toolkit was used.

| | |
|---|---|
| Lexical Context Features | $< t_0, w_{-2} >$ $< t_0, w_{-1} >$ $< t_0, w_0 >$ $< t_0, w_{+1} >$ $< t_0, w_{+2} >$ |
| Tag Context Features | $< t_0, t_{-1} >$ $< t_0, t_{-1}, t_{-2} >$ |

Table 1: Feature Set of ME Tagging Model

tion. They are versatile tools that can handle large numbers of features, and have shown themselves to be highly effective in a broad range of NLP tasks including sentence boundary detection or part-of-speech tagging (Berger et al., 1996).

A *maximum entropy classifier* is an exponential model consisting of a number of binary feature functions and their weights (Pietra et al., 1997). The model is trained by adjusting the weights to maximize the entropy of the probabilistic model given constraints imposed by the training data. In our experiments, we use a *conditional maximum entropy* model, where the conditional probability of the outcome given the set of features is modeled (Ratnaparkhi, 1996). The model has the form:

$$p(t, c) = \gamma \prod_{k=0}^{K} \alpha_k^{f_k(c,t)} \cdot p_0$$

where:

- $t$ is the tag being predicted;
- $c$ is the context of $t$;
- $\gamma$ is a normalization coefficient;
- $K$ is the number of features in the model;
- $f_k$ are binary feature functions;
- $a_k$ is the weight of feature function $f_k$;
- $p_0$ is the default model.

The feature set is given in Table 1. The *lexical context features* consist of target words annotated with a tag $t$. $w_0$ denotes the word being tagged and $w_{-2}, \ldots, w_{+2}$ the surrounding words. $t_0$ denotes the current tag, $t_{-1}$ the previous tag, etc. The *tag context features* supply information about the context of previous tag sequences. This conditional model can be used as a classifier. The model is trained iteratively, and we used the improved iterative scaling algorithm (IIS) (Berger et al., 1996) for the experiments presented in Section 3.

### 2.2 Iterative Bootstrap Method

The proposed iterative bootstrap method to learn the word segmentation that is consistent with an SMT engine is summarized in Figure 2. After the ME tagging model is learned from the initial character-to-word alignments of the respective bitext (`(1)-(4)`), the obtained ME tagger is

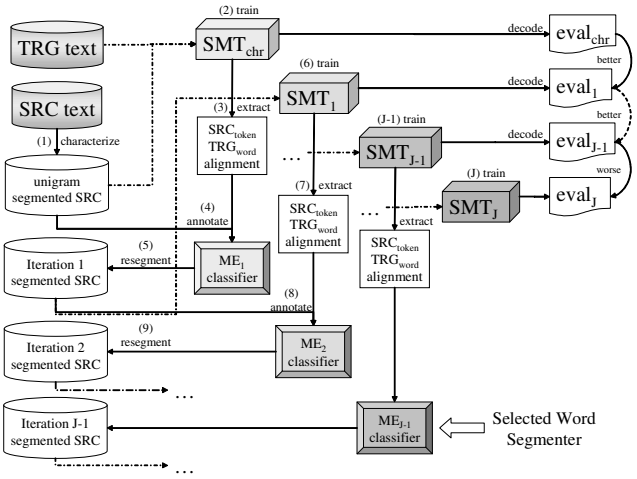


Figure 2: Iterative Bootstrap Method

applied to resegment the source language side of the unsegmented parallel text corpus (`(5)`). This results in a resegmented bitext that can be used to retrain and reevaluate another engine $SMT_1$ (`(6)`), achieving what is hoped to be a better translation performance than the initial SMT engine ($SMT_{chr}$).

The unsupervised ME tagging method can also be applied to the token-to-word alignments extracted during the training of the $SMT_1$ engine to obtain an ME tagging model $ME_1$ capable of handling longer translation units (`(7)-(8)`). Such a bootstrap method iteratively creates a sequence of SMT engines $SMT_i$ (`(9)-(J)`), each of which reduces the translation complexity, because larger chunks can be translated in a single step leading to fewer word order or word disambiguation errors. However, at some point, the increased length of translation units learned from the training corpus will lead to overfitting, resulting in reduced translation performance when translating unseen sentences. Therefore, the bootstrap method stops when the $J^{th}$ resegmentation of the training corpus results in a lower automatic evaluation score for the unseen sentences than the one for the previous iteration. The ME tagging model $ME_{J-1}$ that achieved the highest automatic translation scores is then selected as the best single-iteration word segmenter.

### 2.3 Integration of Multiple Segmentations

The integration of multiple word segmentation schemes is carried out by merging the translation models of the SMT engines trained on the characterized and iteratively learned segmentation schemes. This process is performed by linearly interpolating the model probabilities of each of the

models. In our experiments, equal weights were used; however, it might be interesting to investigate varying the weights according to iteration number, as the latter iterations may contain more useful segmentations.

In addition, we also remove the internal segmentation of the source phrases. The advantages are twofold. Primarily it allows decoding directly from unsegmented text. Moreover, the segmentation of the source phrase can differ between models at differing iterations; removing the source segmentation at this stage makes the phrase pairs in the translations models at various stages in the iterative process consistent with one another. Consequently, duplicate bilingual phrase pairs appear in the phrase table. These duplicates are combined by normalizing their model probabilities prior to model interpolation.

The rescored translation model covers all translation pairs that were learned by any of the iterative models. Therefore, the selection of longer translation units during decoding can reduce the complexity of the translation task. On the other hand, overfitting problems of single-iteration models can be avoided because multiple smaller source language translation units can be exploited to cover the given input parts and to generate translation hypotheses based on the concatenation of associated target phrase expressions. Moreover, the merging process increases the translation probabilities of the source/target translation parts that cover the same surface string but differ only in the segmentation of the source language phrase. Therefore, the more often such a translation pair is learned by different iterative models, the more often the respective target language expression will be exploited by the SMT decoder.

The translation of unseen data using the merged translation models is carried out by (1) characterizing the input text and (2) applying the SMT decoding in a standard way.

## 3 Experiments

The effects of using different word segmentations and integrating them into an SMT engine are investigated using the multilingual *Basic Travel Expressions Corpus* (BTEC), which is a collection of sentences that bilingual travel experts consider useful for people going to or coming from other countries (Kikui et al., 2006). For the word segmentation experiments, we selected five Asian languages that do not naturally separate word units, i.e., Japanese (ja), Korean (ko), Thai (th), and two dialects of Chinese (Standard Mandarin (zh) and Taiwanese Mandarin (tw)).

| BTEC | | train set | dev set | test set |
|---|---|---|---|---|
| # of sen | | 160,000 | 1,000 | 1,000 |
| en | voc | 15,390 | 1,262 | 1,292 |
| | len | 7.5 | 7.1 | 7.2 |
| ja | voc | 17,168 | 1,407 | 1,408 |
| | len | 8.5 | 8.2 | 8.2 |
| ko | voc | 17,246 | 1,366 | 1,365 |
| | len | 8.0 | 7.7 | 7.8 |
| th | voc | 7,354 | 1,081 | 1,053 |
| | len | 7.8 | 7.3 | 7.4 |
| zh | voc | 11,084 | 1,312 | 1,301 |
| | len | 7.1 | 6.4 | 6.5 |

Table 2: Language Resources

Table 2 summarizes the characteristics of the BTEC corpus used for the training (*train*) of the SMT models, the tuning of model weights and stop conditions of the iterative bootstrap method (*dev*), and the evaluation of translation quality (*test*). Besides the number of sentences (*sen*) and the vocabulary (*voc*), the sentence length (*len*) is also given as the average number of words per sentence. The given statistics are obtained using commonly-used linguistic segmentation tools available for the respective language, i.e., CHASEN (ja), WORDCUT (th), ICTCLAS (zh), HanTagger (ko). No segmentation was available for Taiwanese Mandarin and therefore no meaningful statistics could be obtained.

For the training of the SMT models, standard word alignment (Och and Ney, 2003) and language modeling (Stolcke, 2002) tools were used. Minimum error rate training (MERT) was used to tune the decoder's parameters and performed on the *dev* set using the technique proposed in (Och and Ney, 2003). For the translation, a multi-stack phrase-based decoder was used.

For the evaluation of translation quality, we applied standard automatic metrics, i.e., BLEU (Papineni et al., 2002) and METEOR (Lavie and Agarwal, 2007). We have tested the statistical significance of our results[2] using the bootstrap method reported in (Zhang et al., 2004) that (1) performs a random sampling with replacement from the evaluation data set, (2) calculates the evaluation metric score of each engine for the sampled test sentences and the difference between the two MT system scores, (3) repeats the sampling/scoring step itera-

[2] 2000 iterations were used for the analysis of the automatic evaluation results in this paper. All reported differences in evaluation scores are statistically significant.

tively, and (4) applies the *Student's t-test* at a significance level of 95% confidence to test whether the score differences are significant.

In addition, human assessment of translation quality was carried out using the *Ranking* metrics. For the *Ranking* evaluation, a human grader was asked to *"rank each whole sentence translation from Best to Worst relative to the other choices (ties are allowed)"* (Callison-Burch et al., 2007). The *Ranking* scores were obtained as the average number of times that a system was judged better than any other system and the normalized ranks (*NormRank*) were calculated on a per-judge basis for each translation task using the method of (Blatz et al., 2003).

Section 3.1 compares the proposed method to the baseline system that translates characterized source language sentences and to the SMT engines that are trained on iteratively learned as well as language-dependent linguistic word segmentations. The effects of the iterative learning method are summarized in Section 3.2.

### 3.1 Effects of Word Segmentation

The automatic evaluation scores of the SMT engines trained on the differently segmented source language resources are given in Table 3, where "*character*" refers to the baseline system of using character-segmented source text; "*single-best*"[3] is the SMT engine that is trained on the corpus segmented by the best-performing iteration of the bootstrap approach; "*proposed*" is the SMT engine whose models integrate multiple word segmentation schemes; and "*linguistic*" uses language-dependent linguistically motivated word segmentation tools. The reported scores are calculated as the mean score of all metric scores obtained for the iterative sampling method used for statistical significance testing and listed as percentage figures.

The results show that the proposed method outperforms the *character* (*single-best*) system for each of the involved languages achieving gains of 2.0 to 9.1 (0.4 to 1.6) BLEU points and 2.0 to 5.9 (0.7 to 4.6) METEOR points, respectively. However, the improvements depend on the source language. For example, the smallest gains were obtained for Standard Mandarin, because single characters frequently form words of their own, thus resulting in more ambiguity than Japanese, where consecutive *hiragana* or *katakana* characters can form larger meaningful units.

Comparing the proposed method towards linguistically motivated segmenters, the results show that the proposed method outperforms the SMT engines using linguistic segmentation tools for tasks such as translating Korean and Standard Mandarin into English. Slightly lower evaluation scores were achieved for the automatically learned word segmentation for Japanese, although the results of the proposed method are quite similar. This is a suprisingly strong result, given the maturity of the linguistically motivated segmenters, and given that our segmenters use only the bilingual corpus used to train the SMT systems.

The Thai-English experiments expose some issues that are related to the definition of what a "character" is. Our segmentation schemes are learned directly from the bitext without any language-specific information, and can cope well with most languages. However, Thai seems to be an exceptional case in our experiments, because (1) the Thai script is a segmental writing system which is based on consonants but in which vowel notation is obligatory, so that the characterization of the baseline system affects vowel dependencies, (2) it uses tone markers that are placed above the consonant, but are treated as a single character in our approach, and (3) vowels sounding after a consonant are non-sequential and can occur before, after, above, or below a consonant increasing the number of word form variations in the training corpus and reducing the accuracy of the learned ME tagging models. This is an interesting result that motivates further study on how to incorporate features on language scripts into our machine learning framework. For example, Japanese is written in three different scripts (*kanji*, *hiragana*, *katakana*). Therefore, the script class of each character could be used as an additional feature to obtain the initial segmentation of the training corpus.

Finally, the results for Taiwanese Mandarin, where no linguistic tool was available to segment the source language text, shows that the proposed method can be applied successfully for the translation of any language where no linguistically-motivated segmentation tools are available.

Table 4 summarizes the subjective evaluation results which were carried out by a paid evaluation expert who is a native speaker of English. The *NormRank* results confirm the findings of the au-

[3] This approximates the approach of (Ma and Way, 2009) and is given as a way of showing the effect of segmentation at multiple levels of granularity.

**BLEU**

| source language | word segmentation | | | |
|---|---|---|---|---|
| | character | single-best | proposed | linguistic |
| ja | 36.93 | 39.65 | 41.25 | **41.46** |
| ko | 34.72 | 37.32 | **38.51** | 37.19 |
| th | 41.42 | 50.16 | 50.53 | **56.68** |
| zh | 36.59 | 37.02 | **38.61** | 38.13 |
| tw | 45.71 | 50.95 | **52.21** | – |

**METEOR**

| source language | word segmentation | | | |
|---|---|---|---|---|
| | character | single-best | proposed | linguistic |
| ja | 59.78 | 60.95 | 65.45 | **66.03** |
| ko | 58.45 | 60.06 | **64.31** | 63.04 |
| th | 67.22 | 71.22 | 72.58 | **79.02** |
| zh | 61.77 | 62.38 | **63.80** | 62.72 |
| tw | 70.14 | 73.64 | **74.38** | – |

Table 3: Automatic Evaluation

**NormRank**

| source language | word segmentation | | | |
|---|---|---|---|---|
| | character | single-best | proposed | linguistic |
| ja | 2.76 | 2.85 | **3.18** | 3.12 |
| ko | 2.68 | 2.90 | **3.17** | 3.09 |
| th | 2.65 | 2.95 | 3.05 | **3.43** |
| zh | 2.87 | 3.01 | **3.07** | 3.04 |
| tw | 2.83 | 2.86 | **3.24** | – |

Table 4: Subjective Evaluation

tomatic evaluation. In addition, for Japanese, the translation outputs of the proposed method were judged better than those of the linguistically segmented SMT model.

### 3.2 Effects of Bootstrap Iteration

In order to get an idea of the robustness of the proposed method, the changes in system performance for each source language during the iterative bootstrap method is given in Figure 3. The results for BLEU and METEOR show that all languages reach their best performance after the first or second iteration and then slightly, but consistently decrease with the increased number of iterations. The reason for this is the effect of overfitting caused by the concatenation of source tokens that are aligned to longer target phrases, resulting in the segmentation of longer translation units.

The changes in the vocabulary size and the word length are summarized in Figure 4. The amount of words extracted by the proposed method is much larger than the one of the baseline system, increasing the vocabulary size by a factor of 10 for Standard Mandarin and Taiwanese Mandarin, 30 for Japanese and Korean, and 100 for Thai. It is also larger than the vocabulary obtained for the linguistic tools by a factor of 1.5 to 2.5 for all investigated

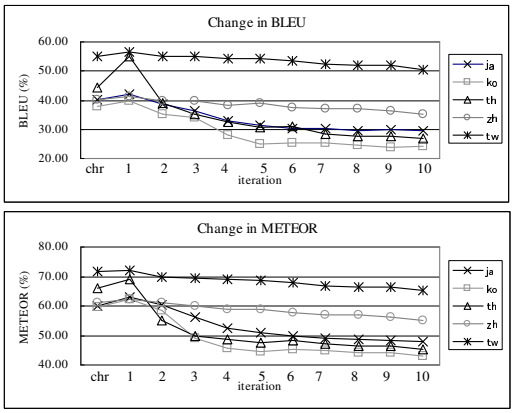


Figure 3: Change in System Performance

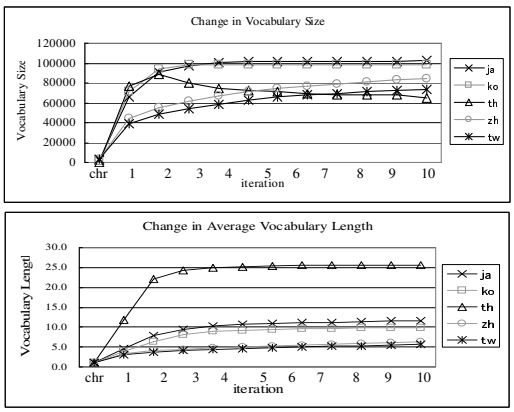


Figure 4: Change in Vocabulary Size and Length

languages. The average vocabulary length also increased for each iteration whereby the length of the translation units learned after 10 iterations almost doubles the word size of the initial iteration.

The overfitting problem of the iterative bootstrap method is illustrated in the increase of *out-of-vocabulary* words, i.e. source language words contained in the unseen evaluation data set that cannot be translated by the respective SMT. The results given in Figure 5 show a large increase in OOV for the first three iterations, resulting in lower translation qualities as listed in Figure 3.

Table 5 illustrates translation examples using different segmentation schemes for the Japanese-English translation task. The SMT engines that output the best translations are marked with an asterisk. In the first example, the concatenation of “もう真夜中” (*already midnight*) by the *single-best* segmentation scheme leads to an OOV word, thus only a partial translation can be achieved. However, the problem can be resolved using the proposed method. The second example is best translated using the *single-best* word segmentation that correctly handles the sentence coordination. The

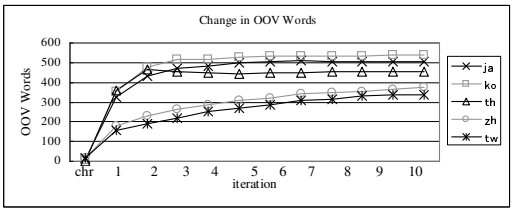


Figure 5: Change in Out-of-Vocabulary Size

baseline system omits the sentence coordination information, resulting in an unacceptable translation. The third examples illustrates that longer tokens reduce the translation complexity and thus can be translated better than the other segmentation that cause more ambiguities.

## 4 Conclusions

This paper proposes a new language-independent method to segment languages that do not use whitespace characters to separate meaningful word units in an unsupervised manner in order to improve the performance of a state-of-the-art SMT system. The proposed method does not need any linguistic information about the source language which is important when building SMT systems for the translation of relatively resource-poor languages which frequently lack morphological analysis tools. In addition, the development costs are far less than those for developing linguistic word segmentation tools or even paying humans to segment the data sets manually, since only the bilingual corpus used to train the SMT system is needed to train the segmenter.

The effectiveness of the proposed method was investigated for the translation of Japanese, Korean, Thai, and two Chinese dialects (Standard Mandarin and Taiwanese Mandarin) into English for the domain of travel conversations. The automatic evaluation of the translation results showed consistent improvements of 2.0 to 9.1 BLEU points and 2.0 to 5.9 METEOR points compared to a baseline system that translates characterized input. Moreover, it improves the best performing SMT engine of the iterative learning procedure by 0.4 to 1.6 BLEU points and 0.7 to 4.6 METEOR points.

In addition, the proposed method achieved translation results similar to SMT models trained on bitext segmented with linguistically motivated tools, even outperforming these for Korean, Chinese, and Japanese in the human evaluation, although no external information and only the given bitext was used to train the segmentation models.

| | | |
|---|---|---|
| linguistic | *seg:* | ええ/。/えーと/、/もう/真夜中/です/ね 。 |
| | *trans:* | Yes. Let's see. It's midnight. |
| character* | *seg:* | え/え/。/え/ー/と/、/も/う/真/夜/中/で/<br>す/ね/。 |
| | *trans:* | Yes. Well, it's already midnight. |
| single-best | *seg:* | ええ。/えーと/、/もう真夜中/で/すね。 |
| | *trans:* | Yes. Let 's see. |
| proposed* | *seg:* | え/え/。/え/ー/と/、/も/う/真/夜/中/で/<br>す/ね/。 |
| | *trans:* | Yes. Well, it's already midnight. |
| linguistic | *seg:* | ジーンズ/が/欲しい/の/です/が/、/<br>いい/店/を/教え/て/ください/。 |
| | *trans:* | I'd like a pair of jeans.<br>Could you recommend a good shop? |
| character | *seg:* | ジ/ー/ン/ズ/が/欲/し/い/の/で/す/が/、/<br>い/い/店/を/教/え/て/く/だ/さ/い/。 |
| | *trans:* | Could you recommend a good 'd like<br>a pair of jeans. |
| single-best* | *seg:* | ジーンズ/が欲し/い/のですが、/<br>いい/店/を教え/てください/。 |
| | *trans:* | I'd like some jeans.<br>Could you recommend a good shop? |
| proposed | *seg:* | ジ/ー/ン/ズ/が/欲/し/い/の/で/す/が/、/<br>い/い/店/を/教/え/て/く/だ/さ/い/。 |
| | *trans:* | I 'd like a pair of jeans and<br>could you recommend a good shop? |
| linguistic | *seg:* | 今日/の/午後/まで/に/でき/ます/か/。 |
| | *trans:* | Will it be ready by this afternoon? |
| character | *seg:* | 今/日/の/午/後/ま/で/に/で/き/ま/す/<br>か/。 |
| | *trans:* | It'll be ready by this afternoon? |
| single-best | *seg:* | 今日の/午後まで/に/できま/すか。 |
| | *trans:* | Will it be ready by this afternoon? |
| proposed* | *seg:* | 今/日/の/午/後/ま/で/に/で/き/ま/す/<br>か/。 |
| | *trans:* | Can you have these ready by this<br>afternoon? |

Table 5: Sample Translations

The experiments using Thai are interesting because the script is a segmental writing system using tone markers and vowel dependencies. This exposed some issues that are related to the definition of what a "character" is and motivates further study on how to incorporate features on language scripts into our machine learning framework.

## References

Adam Berger, Stephen Della Pietra, and Vincent Della Pietra. 1996. A maximum entropy approach to NLP. *Computational Linguistics*, 22(1):39–71.

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2003. Confidence estimation for statistical machine translation. In *Final Report of the JHU Summer Workshop*.

Michael Brent. 1999. An efficient, probabilistically sound algorithm for segmentation and word discovery. *Machine Learning*, 34:71–105.

Chris Callison-Burch, Cameron Fordyce, Philipp Koehn, Christof Monz, and Jan Schroeder. 2007. (Meta-) Evaluation of Machine Translation. In *Proceedings of the 2nd Workshop on SMT*, pages 136–158, Prague, Czech Republic.

Pi-Chuan Chang, Michel Galley, and Christopher Manning. 2008. Optimizing Chinese Word Segmentation for Machine Translation Performance. In *Proc. of the 3rd Workshop on SMT*, pages 224–232, Columbus, USA.

Kwok-Shing Cheng, Gilbert Young, and Kam-Fai Wong. 1999. A study on word-based and integratbit Chinese text compression algorithms. *American Society of Information Science*, 50(3):218–228.

Tagyoung Chung and Daniel Gildea. 2009. Unsupervised Tokenization for Machine Translation. In *Proc. of the EMNLP*, pages 718–726, Singapore.

Adrian de Gispert, Sami Virpioja, Mikko Kurimo, and William Byrne. 2009. Minimum Bayes Risk Combination of Translation Hypotheses from Alternative Morphological Decompositions. In *Proc. of HLT, Companion Volume*, pages 73–76, Boulder, USA.

Christopher Dyer, Smaranda Muresan, and Philip Resnik. 2008. Generalizing Word Lattice Translation. In *Proc. of ACL*, pages 1012–1020, Columbus, USA.

Christopher Dyer. 2009. Using a maximum entropy model to build segmentation lattices for MT. In *Proc. of HLT*, pages 406–414, Boulder, USA.

Sharon Goldwater, Thomas Griffith, and Mark Johnson. 2006. Contextual Dependencies in Unsupervised Word Segmentation. In *Proc. of the ACL*, pages 673–680, Sydney, Australia.

Geninchiro Kikui and Hirofumi Yamamoto. 2002. Finding Translation Pairs from English-Japanese Untokenized Aligned Corpora. In *Proc. of the Workshop on Speech-to-Speech Translation*, pages 23–30, Philadephia, USA.

Geninchiro Kikui, Seiichi Yamamoto, Toshiyuki Takezawa, and Eiichiro Sumita. 2006. Comparative study on corpora for speech translation. *IEEE Transactions on Audio, Speech and Language*, 14(5):1674–1682.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proc. of the 2nd Workshop on SMT*, pages 228–231, Prague, Czech Republic.

Yanjun Ma and Andy Way. 2009. Bilingually Motivated Domain-Adapted Word Segmentation for Statistical Machine Translation. In *Proc. of the 12th EACL*, pages 549–557, Athens, Greece.

Preslav Nakov, Chang Liu, Wei Lu, and Hwee Tou Ng. 2009. The NUS SMT System for IWSLT 2009. In *Proc. of IWSLT*, pages 91–98, Tokyo, Japan.

Franz J. Och and Hermann Ney. 2003. A Systematic Comparison of 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. of the 40th ACL*, pages 311–318, Philadelphia, USA.

Stephen Della Pietra, Vincent Della Pietra, and John Lafferty. 1997. Inducing Features of Random Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):380–393.

Adwait Ratnaparkhi. 1996. A Maximum Entropy Model for Part-Of-Speech Tagging. In *Proc. of the EMNLP*, pages 133–142, Pennsylvania, USA.

Andreas Stolcke. 2002. SRILM an extensible language modeling toolkit. In *Proc. of ICSLP*, pages 901–904, Denver, USA.

Anand Venkataraman. 2001. A statistical model for word discovery in transcribed speech. *Computational Linguistics*, 27(3):351–372.

Jia Xu, Jianfeng Gao, Kristina Toutanova, and Hermann Ney. 2008. Bayesian Semi-Supervised Chinese Word Segmentation for SMT. In *Proc. of the COLING*, pages 1017–1024, Manchester, UK.

Ying Zhang, Stephan Vogel, and Alex Waibel. 2004. Interpreting Bleu/NIST Scores: How Much Improvement do We Need to Have a Better System? In *Proc. of the LREC*, pages 2051–2054, Lisbon, Portugal.

Ruiqiang Zhang, Keiji Yasuda, and Eiichiro Sumita. 2008. Improved Statistical Machine Translation by Multiple Chinese Word Segmentation. In *Proc. of the 3rd Workshop on SMT*, pages 216–223, Columbus, USA.