Given Bilingual Terminology in Statistical Machine Translation: MWE-Sensitive Word Alignment and Hierarchical Pitman-Yor Process-Based Translation Model Smoothing

Tsuyoshi Okita and **Andy Way**
Dublin City University
Glasnevin, Dublin 9, Ireland

Abstract

This paper considers a scenario when we are given almost perfect knowledge about bilingual terminology in terms of a test corpus in Statistical Machine Translation (SMT). When the given terminology is part of a training corpus, one natural strategy in SMT is to use the trained translation model ignoring the given terminology. Then, two questions arises here. 1) Can a word aligner capture the given terminology? This is since even if the terminology is in a training corpus, it is often the case that a resulted translation model may not include these terminology. 2) Are probabilities in a translation model correctly calculated? In order to answer these questions, we did experiment introducing a Multi-Word Expression-sensitive (MWE-sensitive) word aligner and a hierarchical Pitman-Yor process-based translation model smoothing. Using 200k JP–EN NTCIR corpus, our experimental results show that if we introduce an MWE-sensitive word aligner and a new translation model smoothing, the overall improvement was 1.35 BLEU point absolute and 6.0% relative compared to the case we do not introduce these two.

Introduction

This paper considers a scenario in Statistical Machine Translation (SMT) (Brown et al. 1993; Marcu and Wong 2002; Chiang 2005; Koehn 2010) when we are given almost perfect knowledge about bilingual terminology in terms of a test corpus. The first motivation is that it is practically likely that most of the bilingual terminology are already given a priori before we translate a patent. For example, when a Japanese patent is applied to the Japanese Patent Office (JPO), a JP–EN translator has to pick up the English terminology which is often already provided by the bilingual terminology database maintained by the JPO unless it is a new term. Another example is a translation of scientific books. The candidate bilingual terminology is often chosen by a translator before he / she starts translating a book since this helps to use the same terminology consistently.

When the given terminology is not part of a training corpus, a conventional strategy is to substitute the given terminology in the pre-processing stage and to give the substituted sentences to a decoder with a small trick not to translate this substituted parts further. When the given terminology is part of a training corpus, which is our exact interest, we need to choose either the strategy which is identical with above ignoring trained translation model or the strategy that we use the trained translation model ignoring the given terminology. Now, we would like to pose two questions: 1) Can a word aligner capture the given terminology? and 2) Are probabilities in a translation model correctly calculated?

Copyright © 2011, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

The first question is valid since even if the terminology is in a training corpus, a resulted translation model may not include these terminology depending on the case. This is since the conventional word aligner (Och and Ney 2003) first obtains the 1-to-n mappings in bidirectional ways separately, and then obtains n-to-m mapping object by symmetrizing these bidirectional word alignments using the phrase extraction heuristics (Koehn, Och, and Marcu 2003). By the computational costs required for the principled approaches which need to handle the co-occurrence of words in one side of the languages (Sumita 2000; Marcu and Wong 2002), we prefer to solve the n-to-m mapping object problem in word alignment. Among few approaches in this direction, one approach in this line is an MWE-sensitive word aligner which gives information about alignment links as prior knowledge (Okita et al. 2010). It is expected that the deployment of this MWE-sensitive word aligner, in the context of the given bilingual terminology, will result in the recognition with higher precision of the terminology in a translation model. Furthermore, although Okita et al. (2010) only discusses MWEs, this word aligner has potential to incorporate larger category of frequent and less frequent linguistic knowledge such as paraphrases and Out-Of-Vocabulary words (OOV words). It is also expected that those linguistic knowledge may further help the identification of the correct terminology in a translation model (Okita 2009; Okita, Graham, and Way 2010).

The second question is related to the outcomes of word alignment, together with the following phrase extraction, as a form of probabilities in a translation model. It is a well known fact that the relative frequency (or maximum likelihood estimate) is not an accurate statistics for less frequent phrase pairs since it does not consider zero frequencies. Foster et al. applied various smoothing technique to translation model (Foster, Kuhn, and Johnson 2006). It is likely that our

consideration of three kinds of prior knowledge about bilingual terminology will emphasize this phenomenon further since the resulted phrase pairs become less frequent.

This paper is organized as follows. Section 2 reviews Statistical Machine Translation. Section 3 introduces an MWE-sensitive word aligner and three kinds of prior knowledge. Section 4 mentions the smoothing technique applied to a translation model based on the hierarchical Pitman-Yor process. Experimental results are presented in Section 5. Section 6 concludes and provides avenues for further research.

Review of Statistical Machine Translation

Let e be an English word, f be a foreign word, $\bar{e}$ be an English phrase, $\hat{e}$ be an English sentence, $P(e|f)$ be a lexical translation probability for word e over word f, $P(\bar{e}|\bar{f})$ be a translation model for phrase $\bar{e}$ over $\bar{f}$, $P_{LM}(e)$ be a language model for e, and a be an alignment function.

Statistical Machine Translation consists of two steps. In the first step, for a given sentence aligned parallel corpus, we obtain three components: a translation model $P(\bar{f}|\bar{e})$, a reordering model $d(start_i - end_i - 1)$, and a language model $P_{LM}(e)$. In the second step, for a given test sentence, we obtains the best translation using these three components via the noisy-channel model as in (1):

$$\begin{aligned}\bar{\hat{e}}_{BEST} &= \arg\max_{e\in E} P(\hat{f}|\hat{e})P_{LM}(e) \qquad (1)\\ &= \arg\max_{e\in E}\{\Pi_{i=1}^{I} P(\bar{f}|\bar{e})d(start_i - end_i - 1)\}P_{LM}(e).\end{aligned}$$

The widely accepted procedure to obtain a translation model consists of two steps: a word alignment step (Brown et al. 1993; Och and Ney 2003) and a phrase extraction step (Koehn, Och, and Marcu 2003). Brown et al. (1993) introduced a generative model which uses an alignment function as latent variables, and an inference procedure based on an EM algorithm as in (2):

$$\begin{aligned}\mathbf{E^{EXH}}: \quad & q(z;x) = p(z|x;t)\\ \mathbf{M^{MLE}}: \quad & t' = \arg\max_t Q(t, t^{old}) =\\ & \arg\max_t \sum_{x,z} q(z|x)\log p(x,z;t) \qquad (2)\end{aligned}$$

where t denotes a lexical translation probability $t(e|f)$ (t is a parameter), and z denotes a latent variable; note that often t is omitted in word alignment literature but for our purposes in the next section this needs to be explicit. The generative models are called IBM Models 1 to 5 whose assumptions on alignment function and independence assumptions differ. Further details can be obtained by (Brown et al. 1993; Och and Ney 2003; Koehn 2010).

Translation Modeling with MWE-sensitive Word Aligner

After introducing an MWE-sensitive word aligner, we will mention three types of linguistic knowledge, MWEs, paraphrases, and OOV words, which we intend to incorporate to this word aligner.

Algorithm 1 Prior Knowledge about Paraphrases without Pivot (Callison-Burch, Koehn, and Osborne 2006)

Given: Results of word alignment $S = \{(\bar{f}_1, \bar{e}_1, a_1), \ldots, (\bar{f}_k, \bar{e}_k, a_k)\}$ where k indicates the size of possible alignment pairs. A set of target side paraphrases $P = \{(\bar{e}_1|\bar{e}_2), \ldots, (\bar{e}_{r-1}|\bar{e}_r)\}$. Note that a feature function $h(e, f)$ denotes an entry in the translation model (The procedure for the source side paraphrase can be obtained reversing 'target' and 'source'.)

Step 1: Augment the baseline translation model with the entry

$$h(\bar{e}, \bar{f}_1) = \begin{cases} p(\bar{f}_2|\bar{f}_1) & \text{If phrase table entry } (\bar{e}, \bar{f}_1) \\ & \text{is generated from } (\bar{e}, \bar{f}_2) \\ 1 & (otherwise)\end{cases}$$

An MWE-sensitive Word Aligner

The word aligner which can incorporate prior knowledge is called an MWE-sensitive word aligner (Okita et al. 2010). This method replaces the maximum likelihood estimate (shown in Equation (2)) with the MAP (Maximum A Posteriori) estimate (shown in Equation (3)).

$$\begin{aligned}\mathbf{M^{MAP}}: \quad & t' = \arg\max_t Q(t, t^{old}) + \log p(t) =\\ & \arg\max_t \sum_{x,z} q(z|x)\log p(x,z;t)\\ & + \log p(t) \qquad (3)\end{aligned}$$

Then, the prior $\log p(t)$, a probability used to reflect the degree of prior belief about the occurrences of the events, can embed prior knowledge about MWEs.

Let us give information about alignment link between e and f by $T = \{(sentID, t_i, t_j, pos_i, pos_j), \ldots, \}$. We use this information to calculate the prior $p(t) = p(t; e, f, T)$ for the given word e and f: this is 1 if e and f have alignment link, 0 if they are not connected, and uniform if their link is not known. This is shown in (4):

$$p(t; e_i, f_i, T) = \begin{cases} 1 & (e_i = t_i, f_j = t_j) \\ 0 & (e_i = t_i, f_j \neq t_j) \\ 0 & (e_i \neq t_i, f_j = t_j) \\ \text{uniform} & (e_i \neq t_i, f_j \neq t_j)\end{cases} \qquad (4)$$

Then we embed this prior in the M-step of EM algorithm where we replaced its likelihood estimate with MAP estimate. Although this is for the case of IBM Model 1, IBM Models 3 and 4 are essentially the same.

First Prior Knowledge: MWEs The first type of prior knowledge is MWEs with their exact bilingual correspondences. Such correspondences in the training sentences can be incorporated by the prior which we shown in the previous paragraph.

Second Prior Knowledge: Paraphrases The second type of prior knowledge is paraphrases. Recently, various statistical methods are developed which extract paraphrases (Zhao

and Wang 2010). Paraphrases here is assumed to be extracted by the method of (Bannard and Callison-Burch 2005) where one of the pivot should be identical with the pivot language. Note that (Bannard and Callison-Burch 2005) extract the most likely alternative phrase $\bar{e}_2$ for a given phrase $\bar{e}_1$ by pivoting foreign phrase f, as is shown in (5):

$$\begin{aligned}\hat{\bar{e}}_2 &= \arg\max_{\bar{e}_2:\bar{e}_2\neq\bar{e}_1} P(\bar{e}_2|\bar{e}_1) \\ &= \arg\max_{\bar{e}_2:\bar{e}_2\neq\bar{e}_1} \sum_{\bar{f}} P(\bar{f}|\bar{e}_1)P(\bar{e}_2|\bar{f},\bar{e}_1) \\ &\approx \arg\max_{\bar{e}_2:\bar{e}_2\neq\bar{e}_1} \sum_{\bar{f}} P(\bar{f}|\bar{e}_1)P(\bar{e}_2|\bar{f}) \qquad (5)\end{aligned}$$

where $P(\bar{f}|\bar{e}) = count(\bar{e},\bar{f})/\sum_{\bar{f}} count(\bar{e},\bar{f})$. Then, this information is converted into the known correspondence in the training corpus, which is plugged into the MWE-sensitive word aligner. Note that Algorithm 1 intend to incorporate paraphrases directly into a translation model, which is the difference here.

Third Prior Knowledge: OOV Words The third type of prior knowledge has quite different nature compared to the above two. We will know the lists of OOV words after the construction and the execution of a MT system. Several categories of OOV words are shown below.

- Transliteration related terms.
- Proper nouns: Proper nouns represent unique entities, such as person's name, organization's name, locations name, signal name (electronics), chemical name (chemistry), and so forth.[1]
- Localization ("l10n") or internationalization ("i18n") terminology: these are the matters in computer systems which support multiple languages.[2]
- Equations and algorithms: these are often embedded in a sentence without definite boundary.
- Symbols and Encoding related characters.
- Noise: Noise elements are yielded due to the non-existence in training corpus or the fault in word / phrasal alignment.

Firstly, these tend to appear less frequently in training corpus. Hence, by the statistical methods it is often not easy to learn their correspondences by statistics. Secondly, their correspondences are often not affected by the surrounding context. Hence, once we know the correspondences, it is fairly easy to be retrieved. Thirdly, it is often difficult to enumerate whole the possible forms, for example the day / time format. In sum, while the detection algorithm are in general not easy to write which performs very well for general corpora, it is fairly easy to write an algorithm which can perform relatively well for the given corpus. For the proper nouns, a named-entity recognizer aims to detect such entities (Finkel, Grenager, and Manning 2005). For the localization terminology, it is fairly easy to construct using the well-developed rule-based software for localization / internationalization industries.

After detecting OOV words together with their counterparts, we could incorporate such prior knowledge into the MWE-sensitive word aligner.

[1]In English, proper nouns are usually capitalized. In Japanese, there is no particular rules.

[2]This includes day / time format, time zones, formatting of numbers (decimal separator, digit grouping), currency (including its symbols and its variation within the same country), weights and measures, paper sizes, telephone numbers, addresses, postal codes, titles, government assigned numbers (social security number in US, national insurance number in UK).

Translation Model Smoothing

A translation model smoothing method intends to examine the outcomes of word alignment and phrase extraction in terms of their probabilities. This is since it is in general believed that the relative frequencies are better smoothed due to the data sparseness and ignorance of zero probabilities (Foster, Kuhn, and Johnson 2006). We think that there are at least two cases that smoothing will be fairly effective: a case when a corpus size is relatively small and a case when a corpus includes n-grams whose order are unbalanced. The case which we examine in this paper is likely to fall among the latter case. This is since it is likely that the given terminology may include higher n-grams and a language model may not even include their back-offs.

We consider the statistical smoothing method based on the hierarchical Pitman-Yor process, which is a nonparametric generalization of the Dirichlet distribution that produces power-law distributions (Goldwater, Griffiths, and Johnson 2006). First we review the hierarchical Pitman-Yor process-based language model, and then, we introduce a hierarchical Pitman-Yor process-based translation model.

Review of Hierarchical Pitman-Yor LM

HPYLM: Generative Model Hierarchical Pitman-Yor Language Model (HPYLM) (Goldwater, Griffiths, and Johnson 2006; Teh 2006; Mochihashi, Yamada, and Ueda 2009; Okita and Way 2010) is constructed encoding the property of the power-law distribution.

Let $PY(d,\theta,G_0)$ denotes a Pitman-Yor process (Pitman 1995), d denotes a discount parameter, θ denotes a strength parameter, and G_0 a base distribution. We define $\pi(u)$ as the suffix of u consisting of all but the earliest word in Equation (6) as in (Teh 2006): we see u as n-gram words and $\pi(u)$ as (n-1)-gram words. Then, we place a Pitman-Yor process prior *recursively* over $G_{\pi(u)}$ in the generative model, as is shown in (6):

$$\begin{cases} G_u|d_{|u|},\theta_{|u|},G_{\pi(u)} \sim PY(d_{|u|},\theta_{|u|},G_{\pi(u)}) \\ \dots \\ G_\emptyset|d_0,\theta_0,G_0 \sim PY(d_0,\theta_0,G_0) \end{cases} \qquad (6)$$

Note that the discount and strength parameters are functions of the length $|u|$ of the context, while the mean vector is $G_{\pi(u)}$, and the vector of probabilities of the current word given all but the earliest word in the context.

HPYLM: Inference One procedure to do an inference in order to generate words drawn from G is called Chinese restaurant process, which iteratively marginalizes out G.

Let h be an n-gram context; for example in 3-gram, this is $h = \{w_1, w_2\}$. A Chinese restaurant contains an infinite number of tables t, each with infinite seating capacity. Customers, which are the n-gram counts $c(w|h)$, enter the restaurant and seat themselves over the tables $1, \ldots, t_{hw\cdot}$. The first customer sits at the first available table, while each of the subsequent customers sits at an occupied table with probability proportional to the number of customers already sitting there $c_{hwk} - d$, or at a new unoccupied table with probability proportional to $\theta + d \cdot t_{h\cdot}$ as is shown in (7):

$$w|h \sim \begin{cases} c_{hwk} - d & (1 \leq k \leq t_{hw}). \\ \theta + d \cdot t_{h\cdot} & (k = new). \end{cases} \quad (7)$$

where c_{hwk} is the number of customers seated at table k until now, and $t_{h\cdot} = \sum_w t_{hw}$ is the total number of tables in h.

Hence, the predictive distribution of n-gram probability in HPYLM is recursively calculated as in (8):

$$p(w|h) = \frac{c(w|h) - d \cdot t_{hw}}{\theta + c(h)} + \frac{\theta + d \cdot t_{h\cdot}}{\theta + c(h)} p(w|h') \quad (8)$$

where $p(w|h')$ is the same probability using a $(n$-1)-gram context h'. Implementation of this inference procedure relates to the Markov chain Monte Carlo sampling. The simplest way is to build a Gibbs sampler while more efficient way is to build a blocked Gibbs sampler (Mochihashi, Yamada, and Ueda 2009).

Translation Model Smoothing

An n-gram is often defined as a subsequence of n items from a given sequence where items can be phonemes, syllables, letters, words or base pairs. Although we can extend this definition of n-gram to the one which includes 'phrases', let us use the different term 'n-phrase-gram' instead in this paper, in order not to mix up with the n-gram for words. Fig. 1 shows a typical example of phrase extraction process. In this process, under the consistency constrained, phrase pairs are extracted which is depicted in the center. Note that this figure is depicted separating the source and the target sides.

Fig. 2 shows the same figure if we depict them in pairs. The lowest column includes only 1-phrase-grams, the second lowest column includes 2-phrase-grams, and so on. The line connecting two nodes indicates parent-child relations. Then, this becomes the lattice structure of the generated phrase pairs. These generated phrase pairs may have several paths to yield the whole sentences. As is similar with HPYLM, we can limit this by considering the suffix of a sequence, meaning that we can process a sequence always from left-to-right. Hence, although the natural lattice would include the dashed lines, the dashed lines can be eliminated if we impose constraint that we should always read the suffix of this sequence from left-to-right. This constraint makes the resulted structure a tree. If the resulted structure is a tree, we can employ the same strategy with HPYLM. The predictive distribution can be calculated by Equation (8) with the replacement of n-grams with n-phrase-grams.

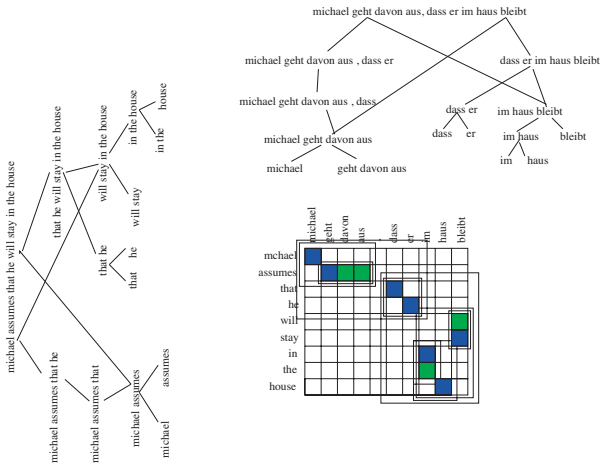


Figure 1: A toy example of phrase extraction process. Resulted phrase pairs can be described as a lattice structure.

Experimental Results

Our baseline was a standard log-linear PB-SMT system based on Moses. The GIZA++ implementation (Och and Ney 2003) of IBM Model 4 was used for word alignment. For phrase extraction the grow-diag-final heuristics described in (Och and Ney 2003) was used to derive the refined alignment. We then performed MERT process (Och 2003) which optimizes the BLEU metric, while a 5-gram language model was derived with Kneser-Ney smoothing trained with SRILM (Stolcke 2002) on the English side of the training data. We used Moses (Koehn et al. 2007) for decoding.

We used NTCIR-8 patent corpus for JP–EN (Fujii et al. 2010). We randomly selected 200k sentence pairs as training corpus. For JP–EN patent corpus, we used 1.2k sentence for development set while we used a test set prepared for NTCIR-8 evaluation campaign. Japanese side was segmented by Cabocha (Kudo and Matsumoto 2003). Table 1 shows the statistics of each prior knowledge. We prepared terminology without using external resources but with the interference of human beings. For the first prior knowledge, MWEs are extracted by the heuristic MWE-extraction strategy similar to (Kupiec 1993), and then corrected these extracted terminology by hand inspecting the corpora. For the second prior knowledge, paraphrases are extracted by the method described in (Bannard and Callison-Burch 2005). For the third prior knowledge, OOV word lists were created in this way. We constructed a PB-SMT decoder, decoded all the training corpus as well as test corpus, and collected whole the OOV words from the translation outputs. Then, we supply the translation counterparts by human beings.[3]

Table 2 shows our results. Without translation model smoothing, the improvement of BLEU by the prior 1 was 0.80 BLEU point absolute, the prior 2 was 0.65 BLEU point absolute, the prior 3 was 0.58 BLEU point absolute, and the

[3]Due to the way of segmentation, around 20% of the transliteration terms was not possible to find their counterparts.

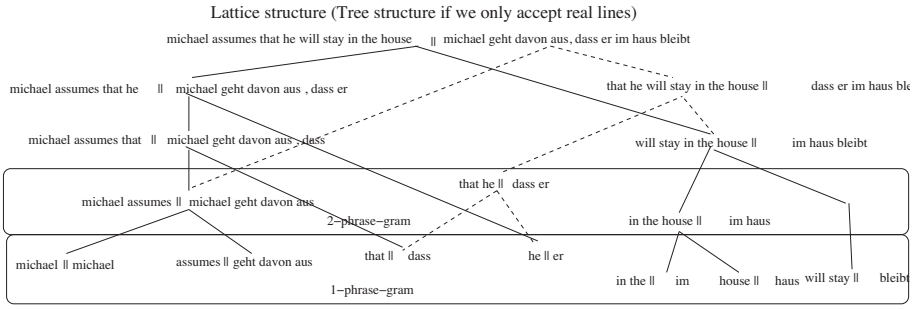


Figure 2: Figure shows a lattice structure of translation model for a toy example.

JP–EN		training	test
prior knowledge 1	MWEs	120070	3865
prior knowledge 2	paraphrases	432135	—
prior knowledge 3	transliteration	25928	284
	proper nouns	3408	2
	localization	207	2
	equations	103	1
	symbols	13842	684
	noise	19007	175

Table 1: Statistics of prior knowledge.

JP–EN	without TM smoothing	with TM smoothing
baseline	21.68	22.44
prior 1	22.48	22.78
prior 2	22.43	22.64
prior 3	22.26	22.52
all 1-3	22.95	23.03
heuristics	21.90	22.49

Table 2: Results for 200k JP–EN sentences. Heuristics in the last row shows the result when prior knowledge 1 was added at the bottom of the translation model.

prior 1 to 3 was 1.27 BLEU point absolute. With translation model smoothing, the improvement of BLEU compared to the baseline with no TM smoothing by the prior 1 was 1.10 BLEU point absolute, the prior 2 was 0.96 BLEU point absolute, the prior 3 was 0.85 BLEU point absolute, and the prior 1 to 3 was 1.35 BLEU point absolute. With translation model smoothing, the improvement of BLEU compared to the baseline with TM smoothing by the prior 1, 2 and 3 was rather very small. This shows that an MWE-sensitive aligner and the translation model smoothing improved the results if we applied them separately, but the combined effect was not much observed unless we incorporate MWEs, paraphrases, and OOVs together.

Conclusion and Further Studies

This paper considered a scenario when we are given almost perfect knowledge about bilingual terminology in terms of a test corpus in Statistical Machine Translation (SMT). The focus is on the effect of bilingual terminology to the training step: the performance of an MWE-senstive word aligner and the translation model smoothing method. When we use the both of these, we obtained the improvement of 1.35 BLEU point absolute and 6.0% relative for this settings. We obtained the positive results by only the translation model smoothing as well. Note that considering the fact that there are various interventions of human beings including giving the exact fragments of answer in test sentences, this improvement is rather too small than we expected.

There are several avenues for further research. Firstly, this paper does not consider syntactical issues. It would be interesting to see the same results on factored translation model (Koehn and Hoang 2007) incorporating the deep parsing results, while it is also interesting to extend the MWE-sensitive word aligner in order that it can incorporate syntactical features as prior knowledge. Secondly, this paper uses the in-domain linguistic knowledge, i.e. MWEs and paraphrases are trained within training corpus. We would like to see the results for an out-of-domain corpus. Similarly, it would be interesting to see whether the approach of the hierarchical Pitman-Yor process may work as well for the out-of-domain prior knowledge.

Acknowledgments

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (`http://www.cngl.ie`) at Dublin City University. We would also like to thank the Irish Centre for High-End Computing and Machine Translation Marathons.

References

Bannard, C., and Callison-Burch, C. 2005. Paraphrasing with biingual parallel corpora. *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)* 597–604.

Brown, F.; Pietra, V.; Pietra, A.; ; and Mercer, R. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics, Vol.19, Issue 2* 263–311.

Callison-Burch, C.; Koehn, P.; and Osborne, M. 2006. Improved statistical machine translation using paraphrases. *In*

Proceedings of the North American Chapter of the Association for Computationa Linguistics (NAACL 2006) 17–24.

Chiang, D. 2005. A hierarchical phrase-based model for statistical machine translation. *In Proceedings of the 43th Annual Meeting of the Association for Computational Linguistics (ACL-05)* 263–270.

Finkel, J. R.; Grenager, T.; and Manning, C. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)* 363–370.

Foster, G.; Kuhn, R.; and Johnson, H. 2006. Phrasetable smoothing for statistical machine translation. *In Proceedings of the Empirical Methods in Natural Language Processing (EMNLP 2006)* 53–61.

Fujii, A.; Utiyama, M.; Yamamoto, M.; Utsuro, T.; Ehara, T.; Echizen-ya, H.; and Shimohata, S. 2010. Overview of the patent translation task at the NTCIR-8 workshop. *In Proceedings of the 8th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-lingual Information Access* 293–302.

Goldwater, S.; Griffiths, T. L.; and Johnson, M. 2006. Contextual dependencies in unsupervised word segmentation. *In Proceedings of Conference on Computational Linguistics / Association for Computational Linguistics (COLING-ACL06)* 673–680.

Koehn, P., and Hoang, H. 2007. Factored translation models. *In Proceedings of the Empirical Methods in Natural Language Processing (EMNLP 2007)* 868–876.

Koehn, P.; Hoang, H.; Birch, A.; Callison-Burch, C.; Federico, M.; Bertoldi, N.; Cowan, B.; Shen, W.; Moran, C.; Zens, R.; Dyer, C.; Bojar, O.; Constantin, A.; and Herbst, E. 2007. Moses: Open source toolkit for Statistical Machine Translation. *In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions* 177–180.

Koehn, P.; Och, F.; and Marcu, D. 2003. Statistical phrase-based translation. *In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computationa Linguistics (HLT / NAACL 2003)* 115–124.

Koehn, P. 2010. Statistical machine translation. *Cambridge University Press. Cambridge. UK*.

Kudo, T., and Matsumoto, Y. 2003. Fast methods for kernel-based text analysis. *In Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL 2003)* 24–31.

Kupiec, J. 1993. An algorithm for finding noun phrase correspondences in bilingual corpora. *In Proceedings of the 31st Annual Meeting of Association for Computational Linguistics* 17–22.

Marcu, D., and Wong, W. 2002. A phrase-based, joint probability model for statistical machine translation. *In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)* 133–139.

Mochihashi, D.; Yamada, T.; and Ueda, N. 2009. Bayesian unsupervised word segmentation with nested pitman-yor language modeling. *In Proceedings of Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP 2009)* 100–108.

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

Och, F. 2003. Minimum Error Rate Training in Statistical Machine Translation. *In Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics* 160–167.

Okita, T., and Way, A. 2010. Pitman-Yor process-based language model for Machine Translation. *International Journal on Asian Language Processing* 21(2):57–70.

Okita, T.; Guerra, A. M.; Graham, Y.; and Way, A. 2010. Multi-Word Expression-sensitive word alignment. *In Proceedings of the Fourth International Workshop On Cross Ling ual Information Access (CLIA2010, collocated with COLING2010)* 26–34.

Okita, T.; Graham, Y.; and Way, A. 2010. Gap between theory and practice: Noise sensitive word alignment in machine translation. *In Journal of Machine Learning Research Workshop and Conference Proceedings Volume 11: Workshop on Applications of Pattern Analysis (WAPA2010)* 119–126.

Okita, T. 2009. Data cleaning for word alignment. *In Proceedings of Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP 2009) Student Research Workshop* 72–80.

Pitman, J. 1995. Exchangeable and partially exchangeable random partitions. *Probability Theory and Related Fields* 102:145–158.

Stolcke, A. 2002. SRILM – An extensible language modeling toolkit. *In Proceedings of the International Conference on Spoken Language Processing* 901–904.

Sumita, E. 2000. Lexical transfer using a vector-space model. *In Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000)* 425–431.

Teh, Y. W. 2006. A hierarchical Bayesian language model based on Pitman-Yor processes. *In Proceedings of Joint Conference of the 44th Annual Meeting of the Association for Computational Linguistics* 985–992.

Zhao, S., and Wang, H. 2010. Paraphrases and applications. *Coling 2010: Paraphrases and Applications–Tutorial notes* 1–87.