Context-dependent SMT Model using Bilingual Verb-Noun Collocation

Young-Sook Hwang
ATR SLT Research Labs
2-2-2 Hikaridai Seika-cho
Soraku-gun Kyoto, 619-0288, JAPAN
youngsook.hwang@atr.jp

Yutaka Sasaki
ATR SLT Research Labs
2-2-2 Hikaridai Seika-cho
Soraku-gun Kyoto, 619-0288, JAPAN
yutaka.sasaki@atr.jp

Abstract

In this paper, we propose a new context-dependent SMT model that is tightly coupled with a language model. It is designed to decrease the translation ambiguities and efficiently search for an optimal hypothesis by reducing the hypothesis search space. It works through reciprocal incorporation between source and target context: a source word is determined by the context of previous and corresponding target words and the next target word is predicted by the pair consisting of the previous target word and its corresponding source word. In order to alleviate the data sparseness in chunk-based translation, we take a stepwise back-off translation strategy. Moreover, in order to obtain more semantically plausible translation results, we use bilingual verb-noun collocations; these are automatically extracted by using chunk alignment and a monolingual dependency parser. As a case study, we experimented on the language pair of Japanese and Korean. As a result, we could not only reduce the search space but also improve the performance.

1 Introduction

For decades, many research efforts have contributed to the advance of statistical machine translation. Recently, various works have improved the quality of statistical machine translation systems by using phrase translation (Koehn et al., 2003; Marcu et al., 2002; Och et al., 1999; Och and Ney, 2000; Zens et al., 2004). Most of the phrase-based translation models have adopted the noisy-channel based IBM style models (Brown et al., 1993):

$$\hat{e}_1^I = argmax_{e_1^I} Pr(f_1^J|e_1^I)Pr(e_1^I) \qquad (1)$$

In these model, we have two types of knowledge: translation model, $Pr(f_1^J|e_1^I)$ and language model, $Pr(e_1^I)$. The translation model links the source language sentence to the target language sentence. The language model describes the well-formedness of the target language sentence and might play a role in restricting hypothesis expansion during decoding. To recover the word order difference between two languages, it also allows modeling the reordering by introducing a relative distortion probability distribution. However, in spite of using such a language model and a distortion model, the translation outputs may not be fluent or in fact may produce nonsense.

To make things worse, the huge hypothesis search space is much too large for an exhaustive search. If arbitrary reorderings are allowed, the search problem is NP-complete (Knight, 1999). According to a previous analysis (Koehn et al., 2004) of how many hypotheses are generated during an exhaustive search using the IBM models, the upper bound for the number of states is estimated by $N \simeq 2^J |V_e|^2 J$, where J is the number of source words and $|V_e|$ is the size of the target vocabulary. Even though the number of possible translations of the last two words is much smaller than $|V_e|^2$, we still need to make further improvement. The main concern is the ex-

Proceedings of the 43rd Annual Meeting of the ACL, pages 549–556,
Ann Arbor, June 2005.

ponential explosion from the possible configurations of source words covered by a hypothesis. In order to reduce the number of possible configurations of source words, decoding algorithms based on A^* as well as the beam search algorithm have been proposed (Koehn et al., 2004; Och et al., 2001). (Koehn et al., 2004; Och et al., 2001) used heuristics for pruning implausible hypotheses.

Our approach to this problem examines the possibility of utilizing context information in a given language pair. Under a given target context, the corresponding source word of a given target word is almost deterministic. Conversely, if a translation pair is given, then the related target or source context is predictable. This implies that if we considered bilingual context information in a given language pair during decoding, we can reduce the computational complexity of the hypothesis search; specifically, we could reduce the possible configurations of source words as well as the number of possible target translations.

In this study, we present a statistical machine translation model as an alternative to the classical IBM-style model. This model is tightly coupled with target language model and utilizes bilingual context information. It is designed to not only reduce the hypothesis search space by decreasing the translation ambiguities but also improve translation performance. It works through reciprocal incorporation between source and target context: source words are determined by the context of previous and corresponding target words, and the next target words are predicted by the current translation pair. Accordingly, we do not need to consider any distortion model or language model as is the case with IBM-style models.

Under this framework, we propose a chunk-based translation model for more grammatical, fluent and accurate output. In order to alleviate the data sparseness problem in chunk-based translation, we use a stepwise back-off method in the order of a chunk, sub-parts of the chunk, and word level. Moreover, we utilize verb-noun collocations in dealing with long-distance dependency which are automatically extracted by using chunk alignment and a monolingual dependency parser.

As a case study, we developed a Japanese-to-Korean translation model and performed some experiments on the BTEC corpus.

2 Overview of Translation Model

The goal of machine translation is to transfer the meaning of a source language sentence, $f_1^J = f_1 \dots f_J$, into a target language sentence, $e_1^I = e_1 \dots e_I$. In most types of statistical machine translation, conditional probability $Pr(e_1^I|f_1^J)$ is used to describe the correspondence between two sentences. This model is used directly for translation by solving the following maximization problem:

$$\hat{e}_1^I = argmax_{e_1^I} Pr(e_1^I|f_1^J) \quad (2)$$

$$= argmax_{e_1^I} \frac{Pr(e_1^I, f_1^J)}{Pr(f_1^J)} \quad (3)$$

$$= argmax_{e_1^I} Pr(e_1^I, f_1^J) \quad (4)$$

Since a source language sentence is given and the $Pr(f_1^J)$ probability is applied to all possible corresponding target sentences, we can ignore the denominator in equation (3). As a result, the joint probability model can be used to describe the correspondence between two sentences. We apply Markov chain rules to the joint probability model and obtain the following decomposed model:

$$Pr(e_1^I, f_1^J) \simeq \prod_{i=1}^{I} Pr(f_{a_i}|e_i, e_{i-1}) Pr(e_i|e_{i-1} f_{a_{i-1}}) \quad (5)$$

where a_i is the index of the source word that is aligned to the word e_i under the assumption of the fixed one-to-one alignment. In this model, we have two probabilities:

- source word prediction probability under a given target language context, $Pr(f_{a_i}|e_{i-1}, e_i)$
- target word prediction probability under the preceding translation pair, $Pr(e_i|e_{i-1}, f_{a_{i-1}})$

The probability of target word prediction is used for selecting the target word that follows the previous target words. In order to make this more deterministic, we use bilingual context, i.e. the translation pair of the preceding target word. For a given target word, the corresponding source word is predicted by source word prediction probability based on the current and preceding target words.

Since a target and a source word are predicted through reciprocal incorporation between source and target context from the beginning of a target sentence, the word order in the target sentence is automatically determined and the number of possible configurations of source words is decreased. Thus, we do not need to perform any computation for word re-ordering. Moreover, since correspondences are provided based on bilingual contextual evidence, translation ambiguities can be decreased. As a result, the proposed model is expected to reduce computational complexity during the decoding as well as improve performance.

Furthermore, since a word-based translation approach is often incapable of handling complicated expressions such as an idiomatic expressions or complicated verb phrases, it often outputs nonsense translations. To avoid nonsense translations and to increase explanatory power, we incorporate structural aspects of the language into the chunk-based translation model. In our model, one source chunk is translated by exactly one target chunk, i.e., one-to-one chunk alignment. Thus we obtain:

$$\tilde{e}_1^K = argmax_{\tilde{e}_1^K} Pr(\tilde{e}_1^K, \tilde{f}_1^K) \tag{6}$$

$$Pr(\tilde{e}_1^K, \tilde{f}_1^K) \simeq \prod_{i=1}^{K} Pr(\tilde{f}_{a_i}|\tilde{e}_i, \tilde{e}_{i-1}) Pr(\tilde{e}_i|\tilde{e}_{i-1}, \tilde{f}_{a_{i-1}}) \tag{7}$$

where K is the number of chunks in a source and a target sentence.

3 Chunk-based J/K Translation Model with Back-Off

With the translation framework described above, we built a chunk-based J/K translation model as a case study. Since a chunk-based translation model causes severe data sparseness, it is often impossible to obtain any translation of a given source chunk. In order to alleviate this problem, we apply back-off translation models while giving the consideration to linguistic characteristics.

Japanese and Korean is a very close language pair. Both are agglutinative and inflected languages in the word formation of a *bunsetsu* and an *eojeol*. A *bunsetsu/eojeol* consists of two sub parts: the head part composed of content words and the tail part composed of functional words agglutinated at the end of the head part. The head part is related to the meaning of a given segment, while the tail part indicates a grammatical role of the head in a given sentence.

By putting this linguistic knowledge to practical use, we build a head-tail based translation model as a back-off version of the chunk-based translation model. We place several constraints on this head-tail based translation model as follows:

- The head of a given source chunk corresponds to the head of a target chunk. The tail of the source chunk corresponds to the tail of a target chunk. If a chunk does not have a tail part, we assign *NUL* to the tail of the chunk.
- The head of a given chunk follows the tail of the preceding chunk and the tail follows the head of the given chunk.

The constraints are designed to maintain the structural consistency of a chunk. Under these constraints, the head-tail based translation can be formulated as the following equation:

$$\begin{aligned} Pr(\tilde{f}_{a_i}|\tilde{e}_i, \tilde{e}_{i-1}) Pr(\tilde{e}_i|\tilde{e}_{i-1}, \tilde{f}_{a_{i-1}}) = \\ Pr(\tilde{f}^h_{a_i}|\tilde{e}^h_i, \tilde{e}^t_{i-1}) Pr(\tilde{e}^h_i|\tilde{e}^t_{i-1}\tilde{f}^t_{a_{i-1}}) \\ Pr(\tilde{f}^t_{a_i}|\tilde{e}^t_i, \tilde{e}^h_i) Pr(\tilde{e}^t_i|\tilde{e}^h_i\tilde{f}^h_{a_i}) \end{aligned} \tag{8}$$

where $\tilde{e}^h_i$ denotes the head of the i^{th} chunk and $\tilde{e}^t_i$ means the tail of the chunk.

In the worst case, even the head-tail based model may fail to obtain translations. In this case, we back it off into a word-based translation model. In the word-based translation model, the constraints on the head-tail based translation model are not applied. The concept of the chunk-based J/K translation framework with back-off scheme can be summarized as follows:

1. Input a dependency-parsed sentence at the chunk level,
2. Apply the chunk-based translation model to the given sentence,
3. If one of chunks does not have any corresponding translation:
   - divide the failed chunk into a head and a tail part,

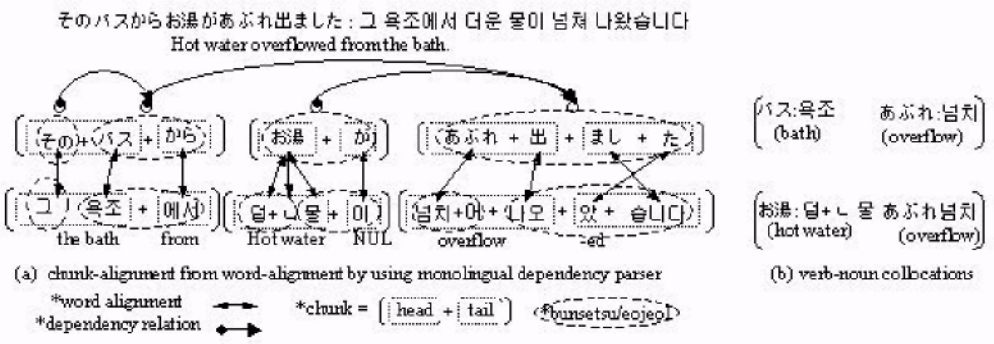

Figure 1: An example of (a) chunk alignment for chunk-based, head-tail based translation and (b) bilingual verb-noun collocation by using the chunk alignment and a monolingual dependency parser

- back-off the translation into the head-tail based translation model,
- if the head or tail does not have any corresponding translation, apply a word-based translation model to the chunk.

Here, the back-off model is applied only to the part that failed to get translation candidates.

3.1 Learning Chunk-based Translation

We learn chunk alignments from a corpus that has been word-aligned by a training toolkit for word-based translation models: the Giza++ (Och and Ney, 2000) toolkit for the IBM models (Brown et al., 1993). For aligning chunk pairs, we consider word(bunsetsu/eojeol) sequences to be chunks if they are in an immediate dependency relationship in a dependency tree. To identify chunks, we use a word-aligned corpus, in which source language sentences are annotated with dependency parse trees by a dependency parser (Kudo et al., 2002) and target language sentences are annotated with POS tags by a part-of-speech tagger (Rim, 2003). If a sequence of target words is aligned with the words in a single source chunk, the target word sequence is regarded as one chunk corresponding to the given source chunk. By applying this method to the corpus, we obtain a word- and chunk-aligned corpus (see Figure 1).

From the aligned corpus, we directly estimate the phrase translation probabilities, $Pr(\tilde{f}|\tilde{e})$, and the model parameters, $Pr(\tilde{f}_{a_i}|\tilde{e}_i, \tilde{e}_{i-1})$, $Pr(\tilde{e}_i|\tilde{e}_{i-1}, \tilde{f}_{a_{i-1}})$. These estimation are made based on relative frequencies.

3.2 Decoding

For efficient decoding, we implement a multi-stack decoder and a beam search with A^* algorithm. At each search level, the beam search moves through at most n-best translation candidates, and a multi-stack is used for partial translations according to the translation cardinality. The output sentence is generated from left to right in the form of partial translations.

Initially, we get n translation candidates for each source chunk with the beam size n. Every possible translation is sorted according to its translation probability. We start the decoding with the initialized beams and initial stack S_0, the top of which has the information of the initial hypothesis, $\langle \tilde{e}_0 = \$, \tilde{f}_0 = \$ \rangle$. The decoding algorithm is described in Table 1.

In the decoding algorithm, estimating the backward score is so complicated that the computational complexity becomes too high because of the context consideration. Thus, in order to simplify this problem, we assume the context-independence of only the backward score estimation. The backward score is estimated by the translation probability and language model score of the uncovered segments. For each uncovered segment, we select the best translation with the highest score by multiplying the translation probability of the segment by its language model score. The translation probability and language model score are computed without giving consideration to context.

After estimating the forward and backward score of each partial translation on stack S_i, we try to

1. Push the initial hypothesis $\langle \tilde{e}_0 = \$, \tilde{f}_0 = \$ \rangle$ on the initial stack S_0
2. for i=1 to K
   - Pop the previous state information of $\langle \tilde{e}_{i-1}, \tilde{f}_{a_{i-1}} \rangle$ from stack S_{i-1}
   - Get next target $\tilde{e}_i$ and corresponding source $\tilde{f}_{a_i}$
   - for all pairs of $\langle \tilde{e}_i, \tilde{f}_{a_i} \rangle$
     - Check the head-tail consistency
     - Mark the source segment as a covered one
     - Estimate forward and backward score
     - Push the state of pair $\langle \tilde{e}_i, \tilde{f}_{a_i} \rangle$ onto stack S_i
   - Sort all translations on stack S_i by the scores
   - Prune the hypotheses
3. while (stack S_K is not empty)
   - Pop the state of the pair $\langle \tilde{e}_K, \tilde{f}_{a_K} \rangle$
   - Compose translation output, $\langle \tilde{e}_1 \ldots \tilde{e}_K \rangle$
4. Output the best N translations

Table 1: A^* multi-stack decoding algorithm

prune the hypotheses. In pruning, we first sort the partial translations on stack S_i according to their scores. If the gradient of scores steeply decreases over the given threshold at the k^{th} translation, we prune the translations of lower scores than the k^{th} one. Moreover, if the number of filtered translations is larger than N, we only take the top N translations. As a final translation, we output the single best translation.

4 Resolving Long-distance Dependency

Since most of the current translation models take only the local context into account, they cannot account for long-distance dependency. This often causes syntactically or semantically incorrect translation to be output. In this section, we describe how this problem can be solved. For handling the long-distance dependency problem, we utilize bilingual verb-noun collocations that are automatically acquired from the chunk-aligned bilingual corpora.

4.1 Automatic Extraction of Bilingual Verb-Noun Collocation(BiVN)

To automatically extract the bilingual verb-noun collocations, we utilize a monolingual dependency parser and the chunk alignment result. The basic concept is the same as that used in (Hwang et al., 2004): bilingual dependency parses are obtained by sharing the dependency relations of a monolingual dependency parser among the aligned chunks. Then bilingual verb sub-categorization patterns are acquired by navigating the bilingual dependency trees. A verb sub-categorization is the collocation of a verb and all of its argument/adjunct nouns, i.e. verb-noun collocation(see Figure 1).

To acquire more reliable and general knowledge, we apply the following filtering method with statistical χ^2 test and unification operation:

- step 1. Filter out the reliable translation correspondences from all of the alignment pairs by χ^2 test at a probability level of α_1
- step 2. Filter out reliable bilingual verb-noun collocations *BiVN* by a unification and χ^2 test at a probability level of α_2: Here, we assume that two bilingual pairs, $\langle v_f : v_e \rangle$ and $\langle n_f : n_e \rangle$ are unifiable into a frame $\langle v_f : v_e, n_f : n_e \rangle$ iff both of them are reliable pairs filtered in step 1. and they share the same verb pair $\langle v_f : v_e \rangle$.

4.2 Application of BiVN

The acquired BiVN is used to evaluate the bilingual correspondence of a verb-noun pair dependent on each other and to select the correct translation. It can be applied to any verb-noun pair regardless of the distance between them in a sentence. Moreover, since the verb-noun relation in BiVN is bilingual knowledge, the sense of each corresponding verb and noun can be almost completely disambiguated by each other.

In our translation system, we apply this **BiVN** during decoding as follows:

1. Pivot verbs and their dependents in a given dependency-parsed source sentence
2. When extending a hypothesis, if one of the pivoted verb and noun pairs is covered and its corresponding translation pair is in **BiVN**, we give positive weight $\beta > 1$ to the hypothesis.

$$\psi(BiVN_i) = \begin{cases} 1 & \text{if } BiVN_i \in \textbf{BiVN} \\ 0 & \text{otherwise} \end{cases}$$

where $BiVN_i = \langle v_f : v_e, n_f : n_e \rangle$ and $\psi(BiVN_i)$ is a function that indicates whether the bilingual translation pair is in **BiVN**. By adding the weight of the $\psi(BiVN_i)$ function, we refine our model as follows:

$$\tilde{e}_1^K \simeq argmax \prod_{i=1}^{K} Pr(\tilde{f}_{a_i}|\tilde{e}_i, \tilde{e}_{i-1}) Pr(\tilde{e}_i|\tilde{e}_{i-1}\tilde{f}_{a_{i-1}})\beta^{VN(f_{a_i})\psi(BiVN_i)} \quad (10)$$

where $VN(f_{a_i})$ is a function indicating whether the pair of a verb and its argument $\langle v_f, n_f \rangle$ is covered with $v_f = f_{a_i}$ or $n_f = f_{a_i}$ and $BiVN_i = \langle v_f : v_e, n_f : n_e \rangle$ is a bilingual translation pair in the hypothesis.

5 Experiments

5.1 Corpus

The corpus for the experiment was extracted from the Basic Travel Expression Corpus (BTEC), a collection of conversational travel phrases for Japanese and Korean (see Table 2). The entire corpus was split into two parts: 162,320 sentences in parallel for training and 10,150 sentences for test. The Japanese sentences were automatically dependency-parsed by CaboCha (Kudo et al., 2002) and the Korean sentences were automatically POS tagged by KUTagger (Rim, 2003)

5.2 Translation Systems

Four translation systems were implemented for evaluation: 1) Word based IBM-style SMT System(WBIBM), 2) Chunk based IBM-style SMT System(CBIBM), 3) Word based LM tightly Coupled SMT System(WBLMC), and 4) Chunk based LM tightly Coupled SMT System(CBLMC). To examine the effect of BiVN, BiVN was optionally used for each system.

The word-based IBM-style (WBIBM) system[1] consisted of a word translation model and a bi-gram language model. The bi-gram language model was generated by using CMU LM toolkit (Clarkson et al., 1997). Instead of using a fertility model, we allowed a multi-word target of a given source word if it aligned with more than one word. We didn't use any distortion model for word re-ordering. And we used a log-linear model $Pr(e|f) = exp(\sum_i \lambda_i h(e, f))$ for weighting the language model and the translation model. For decoding, we used a multi-stack decoder based on the A^* algorithm, which is almost the same as that described in Section 3. The difference is the use of the language model for controlling the generation of target translations.

The chunk-based IBM-style (CBIBM) system consisted of a chunk translation model and a bi-gram language model. To alleviate the data sparseness problem of the chunk translation model, we applied the back-off method at the head-tail or morpheme level. The remaining conditions are the same as those for WBIBM.

The word-based LM tightly coupled (WBLMC) system was implemented for comparison with the chunk-based systems. Except for setting the translation unit as a morpheme, the other conditions are the same as those for the proposed chunk-based translation system.

The chunk-based LM tightly coupled (CBLMC) system is the proposed translation system. A bi-gram language model was used for estimating the backward score.

5.3 Evaluation

Translation evaluations were carried out on 510 sentences selected randomly from the test set. The metrics for the evaluations are as follows:

PER(Position independent WER), which penalizes without considering positional disfluencies(Niesen et al., 2000).

mWER(multi-reference Word Error Rate), which is based on the minimum edit distance between the target sentence and the sentences in the reference set (Niesen et al., 2000).

BLEU, which is the ratio of the n-gram for the translation results found in the reference translations with a penalty for too short sentences (Papineni et al., 2001).

NIST which is a weighted n-gram precision in combination with a penalty for too short sentences.

For this evaluation, we made 10 multiple references available. We computed all of the above criteria with respect to these multiple references.

[1] In this experiment, a word denotes a morpheme

	Training		Test	
	Japanese	Korean	Japanese	Korean
# of sentences	162,320		10,150	
# of total morphemes	1,153,954	1,179,753	74,366	76,540
# of bunsetsu/eojeol	448,438	587,503	28,882	38,386
vocabulary size	15,682	15,726	5,144	4,594

Table 2: Statistics of Basic Travel Expression Corpus

	PER	mWER	BLEU	NIST
WBIBM	0.3415 / 0.3318	0.3668 / 0.3591	0.5747 / 0.5837	6.9075 / 7.1110
WBLMC	0.2667 / 0.2666	0.2998 / 0.2994	0.5681 / 0.5690	9.0149 / 9.0360
CBIBM	0.2677 / 0.2383	0.2992 / 0.2700	0.6347 / 0.6741	8.0900 / 8.6981
CBLMC	0.1954 / 0.1896	0.2176 / 0.2129	0.7060 / 0.7166	9.9167 / 10.027

Table 3: Evaluation Results of Translation Systems: without BiVN/with BiVN

WBIBM	WBLMC	CBIBM	CBLMC
0.8110 / 0.8330	2.5585 / 2.5547	0.3345 / 0.3399	0.9039 / 0.9052

Table 4: Translation Speed of Each Translation Systems(sec./sentence): without BiVN/with BiVN

5.4 Analysis and Discussion

Table 3 shows the performance evaluation of each system. CBLMC outperformed CBIBM in overall evaluation criteria. WBLMC showed much better performance than WBIBM in most of the evaluation criteria except for *BLEU* score. The interesting point is that the performance of WBLMC is close to that of CBIBM in *PER* and *mWER*. The *BLEU* score of WBLMC is lower than that of CBIBM, but the *NIST* score of WBLMC is much better than that of CBIBM.

The reason the proposed model provided better performance than the IBM-style models is because the use of contextual information in CBLMC and WBLMC enabled the system to reduce the translation ambiguities, which not only reduced the computational complexity during decoding, but also made the translation accurate and deterministic. In addition, chunk-based translation systems outperformed word-based systems. This is also strong evidence of the advantage of contextual information.

To evaluate the effectiveness of bilingual verb-noun collocations, we used the BiVN filtered with $\alpha 1 = .05, \alpha 2 = .1$, where coverage is 64.86% on the test set and average ambiguity is 2.99. We suffered a slight loss in the speed by using the BiVN(see Table 4), but we could improve performance in all of the translation systems(see Table 3). In particular, the performance improvement in CBIBM with BiVN was remarkable. This is a positive sign that the BiVN is useful for handling the problem of long-distance dependency. From this result, we believe that if we increased the coverage of BiVN and its accuracy, we could improve the performance much more.

Table 4 shows the translation speed of each system. For the evaluation of processing time, we used the same machine, with a Xeon 2.8 GHz CPU and 4GB memory , and checked the time of the best performance of each system. The chunk-based translation systems are much faster than the word-based systems. It may be because the translation ambiguities of the chunk-based models are lower than those of the word-based models. However, the processing speed of the IBM-style models is faster than the proposed model. This tendency can be analyzed from two viewpoints: decoding algorithm and DB system for parameter retrieval. Theoretically, the computational complexity of the proposed model is lower than that of the IBM models. The use of a

sorting and pruning algorithm for partial translations provides shorter search times in all system. Since the number of parameters for the proposed model is much more than for the IBM-style models, it took a longer time to retrieve parameters. To decrease the processing time, we need to construct a more efficient DB system.

6 Conclusion

In this paper, we proposed a new chunk-based statistical machine translation model that is tightly coupled with a language model. In order to alleviate the data sparseness in chunk-based translation, we applied the back-off translation method at the head-tail and morpheme levels. Moreover, in order to get more semantically plausible translation results by considering long-distance dependency, we utilized verb-noun collocations which were automatically extracted by using chunk alignment and a monolingual dependency parser. As a case study, we experimented on the language pair of Japanese and Korean. Experimental results showed that the proposed translation model is very effective in improving performance. The use of bilingual verb-noun collocations is also useful for improving the performance.

However, we still have some problems of the data sparseness and the low coverage of bilingual verb-noun collocation. In the near future, we will try to solve the data sparseness problem and to increase the coverage and accuracy of verb-noun collocations.

References

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and R. L. Mercer. 1993. *The mathematics of statistical machine translation: Parameter estimation*, *Computational Linguistics*, 19(2):263-311.

P.R. Clarkson and R. Rosenfeld. 1997. *Statistical Language Modeling Using the CMU-Cambridge Toolkit*, Proc. of ESCA Eurospeech.

Young-Sook Hwang, Kyonghee Paik, and Yutaka Sasaki. 2004. *Bilingual Knowledge Extraction Using Chunk Alignment*, Proc. of the 18th Pacific Asia Conference on Language, Information and Computation (PACLIC-18), pp. 127-137, Tokyo.

Kevin Knight. 1999. *Decoding Complexity in Word-Replacement Translation Models*, *Computational Linguistics, Squibs Discussion*, 25(4).

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003 *Statistical Phrase-Based Translation*, Proc. of the Human Language Technology Conference(HLT/NAACL)

Philipp Koehn. 2004 *Pharaoh: a Beam Search Decoder for Phrase-Based Statistical Machine Translation Models*, Proc. of AMTA'04

Taku Kudo, Yuji Matsumoto. 2002. *Japanese Dependency Analyisis using Cascaded Chunking*, Proc. of CoNLL-2002

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

Sonja Niesen, Franz Josef Och, Gregor Leusch, Hermann Ney. 2000. *An Evaluation Tool for Machine Translation: Fast Evaluation for MT Research*, Proc. of the 2nd International Conference on Language Resources and Evaluation, pp. 39-45, Athens, Greece.

Franz Josef Och, Christoph Tillmann, Hermann Ney. 1999. *Improved alignment models for statistical machine translation*, Proc. of EMNLP/WVLC.

Franz Josef Och and Hermann Ney. 2000. *Improved Statistical Alignment Models* , Proc. of the 38th Annual Meeting of the Association for Computational Linguistics, pp. 440-447, Hongkong, China.

Franz Josef Och, Nicola Ueffing, Hermann Ney. 2001. *An Efficient A* Search Algorithm for Statistical Machine Translation* , Data-Driven Machine Translation Workshop, pp. 55-62, Toulouse, France.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. *Bleu: a method for automatic evaluation of machine translation* , IBM Research Report, RC22176.

Toshiyuki Takezawa, Eiichiro Sumita, Fumiaki Sugaya, Hirofumi Yamamoto, and Seiichi Yamamoto. 2002. *Toward a broad-coverage bilingual corpus for speech translation of travel conversations in the real world*, *Proc. of LREC 2002*, pp. 147-152, Spain.

Richard Zens and Hermann Ney. 2004. *Improvements in Phrase-Based Statistical Machine Translation*, Proc. of the Human Language Technology Conference (HLT-NAACL) , Boston, MA, pp. 257-264.

Hae-Chang Rim. 2003. *Korean Morphological Analyzer and Part-of-Speech Tagger*, Technical Report, NLP Lab. Dept. of Computer Science and Engineering, Korea University