

Example-based Decoding for Statistical Machine Translation

Taro Watanabe and Eiichiro Sumita

ATR Spoken Language Translation Research Laboratories
2-2-2 Keihanna Science City
Kyoto 619-0288 Japan
{taro.watanabe, eiichiro.sumita}@atr.co.jp

Abstract

This paper presents a decoder for statistical machine translation that can take advantage of the example-based machine translation framework. The decoder presented here is based on the greedy approach to the decoding problem, but the search is initiated from a similar translation extracted from a bilingual corpus. The experiments on multilingual translations showed that the proposed method was far superior to a word-by-word generation beam search algorithm.

1 Introduction

The framework of statistical machine translation formulates the problem of translating a sentence in a language J into another language E as the maximization problem of the conditional probability $\hat{E} = \operatorname{argmax}_E P(E|J)$ (Brown et al., 1993). The application of the Bayes Rule resulted in $\hat{E} = \operatorname{argmax}_E P(E)P(J|E)$. The former term $P(E)$ is called a language model, representing the likelihood of E . The latter term $P(J|E)$ is called a translation model, representing the generation probability from E into J .

Under this concept, Brown et al. (1993) presented a translation model where a source sentence is mapped to a target sentence with the notion of word alignment¹. Although it has been successfully applied to similar language pairs, such as French–English and German–English, little success has been achieved for drastically different language pairs, such as Japanese–English. The problem lies in the huge search space by the frequently occurring insertion/deletion, the larger numbers of fertility for each word and the complicated word alignments. Due to its complexity, a beam search decoding algorithm was often stuck into sub-optimal solutions.

This paper presents an example-based decoding algorithm, an approach to merging statistical and example-based machine translation (Nagao, 1984).

¹The source/target sentences are the channel model's source/target that correspond to the translation system's output/input.

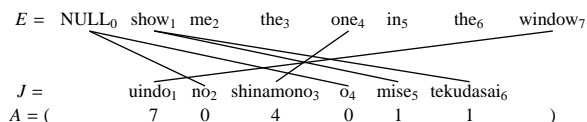


Figure 1: Example of word alignment

Given an input, the decoder first finds some translation examples whose source part is similar to the input. Second, it modifies the retrieved translation using the greedy search algorithm, a hill-climbing approach to find a (hopefully good) translation as introduced by Germann et al. (2001).

Translation experiments were carried out between four different languages pairs, Chinese, English, Japanese and Korean, and it was verified that in any directions, the proposed decoder was superior to the beam search based decoder.

The following section briefly reviews the word alignment based statistical machine translation, then proposes the example-based decoder. Section 4 shows experiments on various language pairs, followed by discussion.

2 Statistical Machine Translation

Word alignment based statistical translation represents bilingual correspondence by the notion of word alignment A , allowing one-to-many generation from each source word. A is an array for target words describing the indexes to the source words. For instance, Figure 1 illustrates an example of English and Japanese sentences, E and J , with sample

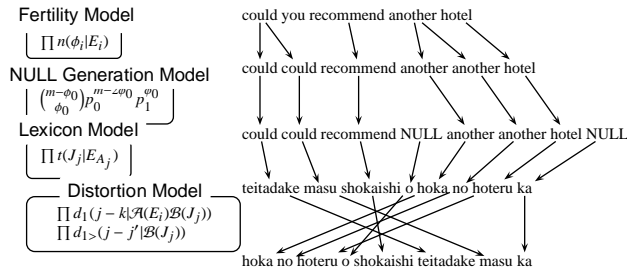


Figure 2: Translation Model $P(J, A|E)$ (IBM Model 4)

word alignments A . In this example, the “show₁” has generated two words, “mise₅” and “tekudasai₆”. Under this word alignment assumption, the translation model $P(J|E)$ can be further decomposed without approximation:

$$P(J|E) = \sum_A P(J, A|E)$$

2.1 Structure of Statistical Machine Translation

The term $P(J, A|E)$ is further decomposed into four components, which constitutes the whole process of transferring source sentence E into J with the word alignments A . For instance, IBM Model 4 defined by Brown et al. (1993) is structured as follows (refer to Figure 2):

1. Choose the number of words to generate for each source word according to the Fertility Model. For example, “could” was increased to 2 words, while “you” was deleted.
2. Insert NULLs at appropriate positions by the NULL Generation Model. Two NULLs were inserted after “recommend” and “hotel”.
3. Translate word-by-word for each generated word by looking up the Lexicon Model. “recommend” was translated to “shokaishi”.
4. Reorder the translated words by referring to the Distortion Model. The “teitadake” was reordered to the 6th position, and “hoteru” was reordered to the 3rd position. Positioning is determined by the previous word’s alignment to capture phrasal constraints.

Please refer to (Brown et al., 1993) for the details of the symbols in Figure 2.

2.2 Decoding Problem in Statistical Machine Translation

The word alignment based statistical translation model was originally intended for similar language pairs, such as French and English. When applied to Japanese and English, for instance, the resulting word alignments are very complicated, as seen in Figure 1. The complexity is directly reflected by the structural differences: i.e., English takes an SVO structure while Japanese usually takes the form of SOV. In addition, insertion and deletion occur very frequently as seen in the example. For instance, there exists no corresponding Japanese morphemes for “the₃” and “the₆”. Therefore, they should be inserted when translating from Japanese. Similarly, Japanese morphemes “no₂” and “o₄” should be deleted.

Both the intricate alignments and the insertion/deletion of words lead to a computationally expensive process when decoding by a word-by-word beam search algorithm as presented by Tillmann and Ney (2000). Due to its complexity, many pruning strategies have to be introduced, such as beam pruning (Och et al., 2001), fertility pruning (Watanabe and Sumita, 2002) or word-for-word translation pruning (García-Vaera et al., 1998), so that the search system can output results in a reasonable time. However, search errors become inevitable under the restricted search space. As Akiba et al. (2002) pointed out, though there exist some correlations between translation quality and the probabilities assigned by the translation model, the beam search was often unable to find good translations.

3 Example-based Decoder

Instead of decoding word-by-word and generating an output string word-by-word, as seen in beam search strategies, this paper presents an alternative strategy taken after the framework of example-based machine translation: Retrieve a translation example from a parallel corpus whose source part is similar to the input sentence, then slightly modify the target part of the example so that the pair becomes the actual translation (refer to Figure 3).

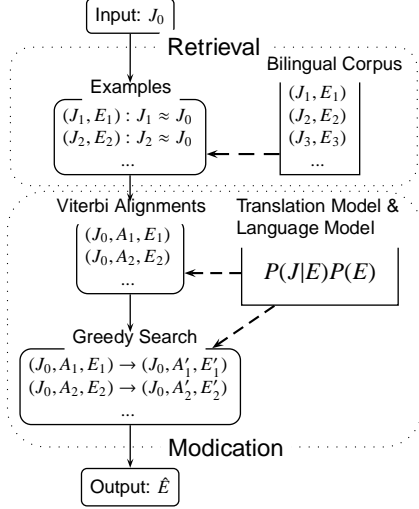


Figure 3: Example-based Decoding

3.1 Retrieval of Translation Examples

Given an input sentence J_0 , the retrieval process looks up a collection of translation examples $\{(J_1, E_1), (J_2, E_2), \dots\}$, where J_k is similar to J_0 using the edit distance criteria, penalizing an insertion/deletion/substitution by one. A simple implementation of the multiple alignment problem resulted in an NP-hard problem where dynamic programming (DP) algorithms should be applied to all the examples in a bilingual corpus. In addition, there exists another problem where the DP matching criteria solely does not always measure the closeness of two sentences. For instance, the sentence “I’m a computer system engineer.” can match many examples, such as “I’m a graduate student.” or “I’m an engineer.” with the same edit distance of 3.

In order to overcome those problems, a tf/idf criteria was introduced to search for the relevant examples by treating each example as a document. Particularly, when given an input J_0 , the decoder first retrieves $N_r (\leq N)$ relevant translation samples, $\{(J_1, E_1), (J_2, E_2), \dots\}$ using the tf/idf criteria as seen in the information retrieval framework (Manning and Schütze, 1999):

$$P_{tf/idf}(J_k, J_0) = \sum_{i: J_{0,i} \in J_k} \frac{\log(N/df(J_{0,i}))/\log N}{|J_0|}$$

where $J_{0,i}$ is the i th word of J_0 , $df(J_{0,i})$ is the document frequency for the word $J_{0,i}$, and N is the total

number of examples in a bilingual corpus. Note that the term frequency is 1 if the word exists in J_k , otherwise 0, and tf/idf scores are summed and normalized by the input sentence length.

Then, for each sample (J_k, E_k) , DP matching is performed against J_0 to compute the edit distance:

$$dis(J_k, J_0) = I(J_k, J_0) + D(J_k, J_0) + S(J_k, J_0)$$

where $k \leq N_r$ and $I(J_k, J_0)$, $D(J_k, J_0)$ and $S(J_k, J_0)$ are the number of insertions/deletions/substitutions respectively. All of the samples are scored by the following criteria:

$$score = \begin{cases} (1.0 - \alpha)(1.0 - \frac{dis(J_k, J_0)}{|J_0|}) + \alpha P_{tf/idf}(J_k, J_0) & dis(J_k, J_0) > 0 \\ 1.0 & otherwise \end{cases}$$

In this scoring, $dis(J_k, J_0)$ is transformed into the word error rate by normalization with the input length $|J_0|$, then subtracted from 1 to derive the correction rate. The correction rate is linearly interpolated with the normalized tf/idf score with a variable α . α is a tuning parameter and was set to 0.2 in our experiments. Note that when the distance of the input sentence and the source part of an example is zero, the example is treated as exactly matched and is scored as one.

3.2 Modification of Translation Examples

After the retrieval of similar examples $\{(J_1, E_1), (J_2, E_2), \dots\}$, the modification step tweaks the sample translations according to a statistical translation model. In this step, the greedy algorithm was applied, which originated from Germann et al. (2001). However, it differs in that the search starts from the retrieved translation example, not from a guessed translation.

For each translation example (J_k, E_k) ,

1. Compute the viterbi alignment A_k for the pair (J_0, E_k)
2. Perform the hill-climbing algorithm for (J_0, A_k, E_k) to obtain (J_0, A'_k, E'_k) by modifying A_k and E_k .

A_k is computed through hill-climbing by moving/swapping particular word alignments as proposed by (Brown et al., 1993). When the retrieved

samples contain an exact match scored as one, the search terminates and returns the retrieved examples with the highest probability together with the viterbi alignment.

When the samples are not an exact match, the decoder performs hill-climbing, modifying the output and alignment for a given example (J_0, A, E) where A is the word alignment initially assigned by the viterbi alignment computation procedure and E is the target part of the example. In this greedy strategy, the operators applied to each hill-climbing step are:

- Translate words: Modify the output word E_{A_j} to e aligned from J_{0_j} . If $e = \text{NULL}$ then J_{0_j} is aligned to NULL and $A_j = 0$. When the fertility of E_{A_j} becomes zero, then the word E_{A_j} is removed. e is selected from among the translation candidates, computed from the inverse of the Lexicon Model (Germann et al., 2001).
- Translate and insert words: Perform the translation of a word, and insert a sequence of zero fertility words at appropriate positions. The candidate sequence of zero fertility words is selected from the viterbi alignment of the training corpus (Watanabe and Sumita, 2002).
- Translate and align words: Move the alignment of A_j to i and modifies the output word from E_i to e .
- Move alignments: This operator does not alter the output word sequence, but modify the alignment A through moving/swapping (Brown et al., 1993).
- Swap segments: Swap non-overlapping subsets of E , by swapping a segment from i_0 to i_1 and from i_2 to i_3 . Note that $i_1 < i_2$.
- Remove words: Remove a sequence of zero fertility words from E
- Join words: Join words of E_i and $E_{i'}$ when the fertility of both of the words is more than or equal to one.

For each hill-climbing step, the decoder tries all the possible operators, then selects the best step for the next iteration. The hill-climbing operators were

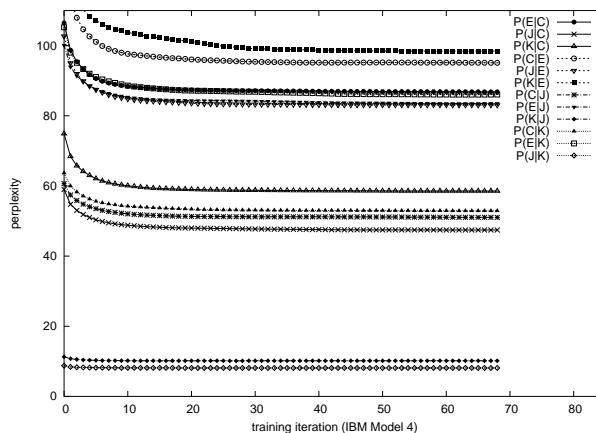


Figure 4: Test set perplexity of the translation models for multiple language pairs

taken from Germann et al. (2001), but two new operators were added, the translate-and-align-words, and the move-alignment. At the first step of computing the viterbi alignment for an input string and a retrieved translation, if there exists an input word whose translations do not exist in the retrieved sample, the word will either be aligned to NULL or an irrelevant word by raising the fertility. Therefore, the translate-and-align operator can force it to move the alignment to another word and to choose the right word-for-word translation using the Lexicon Model. Similarly, the move-alignment operator can resolve the problem by simply alternating the existing word alignments.

4 Evaluation

4.1 Corpus

The corpus for this experiment was extracted from the Basic Travel Expression Corpus (BTEC), a collection of travel conversational phrases for Japanese and English (Takezawa et al., 2002). The corpus was extended to other languages, Korean and Chinese, as illustrated in Table 1. The entire corpus was split into three parts, 152,169 sentences for training, 4,846 sentences for testing, and the remaining 10,148 sentences for parameter tuning, such as the termination criteria for the training iteration, and the parameter tuning for decoders.

Table 1: Basic Travel Expression Corpus (BTEC)

	Chinese	English	Japanese	Korean
# of sentences			167,163	
# of words	956,732	980,790	1,148,428	1,269,888
vocabulary size	16,411	15,641	21,896	13,395
# of singletons	5,207	5,547	9,220	4,191
3-gram perplexity	45.53	35.35	24.06	20.34

4.2 Models

Tri-gram language models for the four languages were trained and evaluated by the perplexity measure as shown in Table 1. For all the translation directions of the four languages, 12 translation models were trained toward IBM Model 4, initiating translation iterations from IBM Model 1 with intermediate HMM model iterations. Figure 4 shows the test set perplexity per target word for the IBM Model 4 iterations. For instance, $P(E|C)$ stands for the Chinese-to-English translation model, hence will be used for the English-to-Chinese translation together with the language model for Chinese. This plot indicates the complexity of the pair of languages, for instance, Japanese-to-Korean translation is far easier than English-to-Korean translation. Figure 4 indicates three groups of language pairings in terms of the statistical translation models. The English and other language pairs are very dissimilar, while the Japanese and Korean pair is close. The Chinese and Japanese/Korean pairs are somewhat in the middle. Combined with the tri-gram perplexity measure, it is possible to estimate that the Japanese-to-Korean translation will be one of the easiest problems, while the English-to-Chinese will be the hardest direction to translate.

4.3 Evaluation

The translation experiments were carried out on 510 sentences selected randomly from the test set. Two decoding algorithms were tested. One is the example-based decoder as explained in Section 3, and the other is a left-to-right output generation beam search algorithm as presented by Watanabe and Sumita (2002).

The metrics for this evaluation were as follows:

WER: Word-error-rate, which penalizes the edit distance (insertion/deletion/substitution) against reference translations (Och and Ney, 2002).

PER: Position independent WER, which penalizes without considering positional disfluencies (Och and Ney, 2002).

BLEU: BLEU score, which computes the ratio of the n-gram for the translation results found in reference translations (Papineni et al., 2002). Contrary to the above error metrics, the higher scores indicates better translations.

SE: Subjective evaluation ranks ranging from A to D (A : perfect, B : fair, C : acceptable and D : nonsense), judged by a native speaker. The scores are evaluated by the ratio of A ranked sentences, A+B for either A or B ranks, and A+B+C for either A, B or C ranks. We have evaluated only a language-to-English and a language-to-Japanese assuming that they are translations for Japanese-to-English and English-to-Japanese, respectively².

For all the languages, 16 reference translations were created for the non-subjective evaluation criteria, WER, PER and BLEU.

Table 2 summarizes all the results. The values in bold font are for the example-based decoder. It also presents the ratio of the exact matching of inputs to the bilingual corpus (refer to the column for Exact), and the perplexity for the translation model (the column for PP). For all the language pairs and directions, a reduction of WER/PER and an improvement of BLEU scores and SE scores are observed with the proposed method.

Tables 3 and 4 summarize the SE scores for the inputs which were exactly or not exactly matched against any translation examples in the training corpus. Even if exact matching could not be found, as presented in Table 4, the example-based decoder outperforms the beam search algorithm.

²Input sentences are assumed to be either Japanese or English.

Table 2: Translation results for multiple languages with the beam search decoder and the example-based decoder. (The values in bold font are for the example-based decoder, otherwise for the beam search decoder.)

	PP	Exact [%]	WER [%]	PER [%]	BLEU [%]	SE [%]			
						A	A+B	A+B+C	D
C-E	95.1		45.0/ 34.3	39.8/ 30.3	43.6/ 56.7	48.4/ 65.3	65.9/ 76.9	71.4/ 81.0	28.6/ 19.0
C-J	51.0	52.7	35.7/ 25.5	31.3/ 22.6	56.9/ 67.8	50.8/ 69.0	59.4/ 74.3	66.9/ 80.2	33.1/ 19.8
C-K	52.9		38.4/ 29.1	34.2/ 26.2	56.1/ 65.0	-	-	-	-
E-C	86.8		45.0/ 38.0	39.7/ 33.4	42.1/ 51.9	-	-	-	-
E-J	83.5	40.8	34.2/ 29.0	30.5/ 26.1	59.2/ 65.7	55.8/ 65.1	62.4/ 71.6	70.2/ 77.8	29.8/ 22.2
E-K	86.1		38.7/ 35.6	34.3/ 31.6	57.3/ 61.5	-	-	-	-
J-C	47.4		46.8/ 33.0	38.9/ 27.8	39.7/ 57.1	-	-	-	-
J-E	83.1	33.7	42.9/ 35.0	37.4/ 30.3	47.6/ 57.4	50.8/ 63.7	65.7/ 74.5	70.2/ 77.6	29.4/ 22.0
J-K	8.1		27.7/ 20.8	25.4/ 19.2	67.2/ 73.5	-	-	-	-
K-C	58.6		41.9/ 32.9	34.4/ 27.6	45.1/ 55.5	-	-	-	-
K-E	98.4	39.2	45.1/ 36.4	38.5/ 32.1	44.3/ 56.8	49.2/ 61.6	65.7/ 72.9	72.2/ 78.4	27.8/ 21.6
K-J	10.2		26.8/ 20.8	24.6/ 19.3	64.3/ 70.8	56.5/ 69.2	66.5/ 77.5	78.4/ 84.7	21.6/ 15.3

Table 3: Translation results for exactly matched inputs

	SE [%]			
	A	A+B	A+B+C	D
C-E	65.4/ 92.6	78.4/ 97.0	82.5/ 97.8	17.5/ 2.2
J-E	72.1/ 97.1	80.8/ 99.4	83.7/ 99.4	16.3/ 0.6
K-E	69.0/ 92.0	81.0/ 97.5	85.0/ 98.0	15.0/ 2.0
C-J	65.4/ 97.0	73.6/ 98.1	78.4/ 98.9	21.6/ 1.1
E-J	79.3/ 95.2	81.7/ 98.1	84.6/ 98.6	15.4/ 1.4
K-J	74.5/ 98.5	80.0/ 99.5	87.5/ 99.5	12.5/ 0.5

Table 4: Translation results for non-exactly matched inputs

	SE [%]			
	A	A+B	A+B+C	D
C-E	29.5/ 34.9	51.9/ 54.4	58.9/ 62.2	41.1/ 37.8
J-E	39.9/ 46.7	58.0/ 61.8	63.3/ 66.6	36.1/ 32.8
K-E	36.5/ 41.9	55.8/ 57.1	63.9/ 65.8	36.1/ 34.2
C-J	34.4/ 37.8	43.6/ 47.7	53.9/ 59.3	46.1/ 40.7
E-J	39.4/ 44.4	49.0/ 53.3	60.3/ 63.6	39.7/ 36.4
K-J	44.8/ 50.3	57.7/ 63.2	72.6/ 75.2	27.4/ 24.8

The translation results are also evaluated according to the search error rate by differentiating exactly matched inputs (matched) and non-exactly matched ones (non-matched) as presented in Table 5. The search error rate was computed by measuring whether a system produced the best scored translation among the two systems, the beam search decoder and the example-based decoder. In terms of this criteria, the example-based decoder is worse than the beam search decoder due to the results by the exactly matched inputs.

Figure 5 shows some translation examples extracted from the Japanese-to-English experiments, presenting input sentences, corresponding reference translations, translations by two decoders and retrieved translations by the example-based decoder.

5 Discussion

In terms of a decoder for statistical machine translation, the example-based decoder is basically identical to the greedy method proposed by Germann et al. (2001), but differs in that the initial condition is derived from the examples of transla-

tions, not from merely guessed sentences by an input string. In general, the greedy method is strongly influenced by the initial state of the search, but our method provides the strong bias especially required for long distance language pairs, such as Japanese and English.

Marcu (2001) proposed a slightly different approach in which translation examples are extracted phrase-by-phrase into a translation memory and the search process is initiated by concatenating the phrases found in the memory. Both share similar

	matched	non-matched	total
	C-E	15.2/ 58.0	28.2/ 38.6
C-J	16.7/ 46.8	30.3/ 24.9	23.1/ 24.9
C-K	14.9/ 45.0	25.7/ 29.5	20.0/ 37.6
E-C	19.2/ 46.2	39.1/ 26.8	31.0/ 34.1
E-J	19.7/ 34.1	33.4/ 26.2	27.8/ 29.4
E-K	14.4/ 42.8	31.8/ 32.8	24.7/ 36.9
J-C	10.5/ 63.4	37.6/ 31.4	28.4/ 42.2
J-E	16.3/ 51.7	35.8/ 28.1	29.2/ 36.1
J-K	14.5/ 39.5	50.3/ 10.9	38.2/ 20.6
K-C	14.0/ 51.5	31.9/ 28.7	24.9/ 37.6
K-E	19.5/ 42.5	31.3/ 31.9	26.7/ 36.1
K-J	15.0/ 32.5	27.7/ 12.3	22.7/ 20.2

input:	銀行の前でバッグをひったくられました
reference:	i was robbed of my bag in front of the bank
beam:	my bag was stolen in the front of the bank
example:	i was robbed of my bag in front of the bank
retrieved:	(i was robbed of my wallet bank on the subway)
input:	今調べておりますのでしばらくお待ちください
reference:	would you wait for a moment while i check
beam:	i 'm out so please wait a moment
example:	would you wait a moment while we check
retrieved:	(would you wait for a moment while we check)
input:	お金を入れたのに機械が動きませんお金を返してもらいたいのですが
reference:	even though i put in some money the machine doesn't work i 'd like a refund please
beam:	i 'd like a refund for this machine doesn't money got heading
example:	although i put in some coins the machine didn't work i 'd like a refund
retrieved:	(although i put in some coins the machine did't work i 'd like a refund)
input:	十時にここで待っていてくれるんですね
reference:	you will wait for me here at ten right
beam:	here is your pay wait ten o'clock
example:	are you waiting here at ten o'clock
retrieved:	(why are you waiting here)
input:	いいえ赤いバッグの隣にあるものです
reference:	no the one next to the red bag
beam:	no red bag in the room next door is something
example:	no it 's next to the red one
retrieved:	(it 's next to that building)
input:	ええそれと似た手帳が届いています
reference:	yes someone did turn in a notebook like that
beam:	yes it is similar there any messages for me
example:	i have a notebook come yes it is similar
retrieved:	(yes we have a japanese speaking guide)

Figure 5: Sample translations from Japanese-to-English experiments

principles of exploiting examples to bias the search, but differ by the unit of examples: our approach uses whole sentences as examples while the other utilizes the phrase unit. During the search process, the phrase unit approach has to perform swapping operations together with insertions of zero fertility words for language pairs with twisted word alignments and frequent insertions/deletions, like those found in Figure 1. The replacements and the insertions heavily rely on the constraints of the language model due to the weak representation of those phenomena in IBM Model 4. However, the n-gram based language model cannot take into account the long distance context, hence the phrase unit method will also stick in sub-optimal solutions as seen in the word-by-word generation beam search. On the other hand, the whole sentence approach is able to bias the search space by feeding the sentence with already re-ordered and inserted word sequences. Therefore, the search is expected to modify examples locally with the help of local context language models. One of the disadvantages of the whole sentence method is the availability of similar examples. We are in the

process of investigating this problem by combining two different unit sizes, by allowing a decoder to initiate the greedy search process from the combined phrasal examples if the similarity scores of any examples are below a certain threshold.

The example-based decoder can share an alternative view point: it is an example-based translation system, but uses statistically acquired knowledge to generate translations. Conventional example-based MT systems consist of three parts: the extraction of examples into storage, retrieval and the modification of examples when given an input. They can store examples either by sentence (Sumita, 2001), by fragment or by phrase (Nagao, 1984; Watanabe and Maruyama, 1994; Way, 1999; Brown, 2000; Richardson et al., 2001), and adjust fetched similar translation samples while translating. The difference with the proposed method lies in the process of transforming examples to match the input sentence. A conventional example-based MT system basically uses bilingual dictionaries, while the example-based decoder uses statistical translation models to adjust examples. The adaptation of the statistical model is justified by the correlation between the quality of translations and the probability assigned by the model (Akiba et al., 2002). Therefore, the more accurate the translation model is, such as the syntax-based translation model (Yamada and Knight, 2001) or the phrase-based translation model (Marcu and Wong, 2002), the more quality improvement will be expected. Furthermore, the example-based decoder can be adapted to the error correction framework with more accurate translation models. The greedy decoding process can be initiated from the translations from other systems, such as a rule-based MT system or an example-based MT system, instead of from translation examples extracted solely from a bilingual corpus.

6 Conclusion

This paper presented an example-based decoding algorithm for statistical machine translation, which can offer both of the benefits of example-based and statistical machine translation, by the retrieve-and-modify strategy. The retrieval process was modeled after the information retrieval framework, while the modification process was taken from the greedy algorithm for statistical machine translation, but using

a retrieved similar translation as the starting point, rather than mere guessed initial states. An evaluation on a multilingual corpus, Chinese, English, Japanese and Korean, indicated that in all of the language pairs, the proposed method was superior to a word-by-word beam search algorithm.

Acknowledgment

The research reported here was supported in part by a contract with the Telecommunications Advancement Organization of Japan entitled, “A study of speech dialogue translation technology based on a large corpus”.

References

- Yasuhiro Akiba, Taro Watanabe, and Eiichiro Sumita. (2002). Using language and translation models to select the best among outputs from multiple mt systems. In *Proc. of COLING 2002*, Taipei, Taiwan, August.
- Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. (1993). The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.
- Ralf D. Brown. (2000). Automated Generalization of Translation Examples. In *Proceedings of the Eighteenth International Conference on Computational Linguistics (COLING-2000)*, pages 125–131.
- Ismael García-Vaera, Francisco Casacuberta, and Hermann Ney. (1998). An iterative, dp-based search algorithm for statistical machine translation. In *Proc. of ICSLP’98*, volume 4, pages 1135–1138, Sydney.
- Ulrich Germann, Michael Jahr, Kevin Knight, Daniel Marcu, and Kenji Yamada. (2001). Fast decoding and optimal decoding for machine translation. In *Proc. of ACL 2001*, Toulouse, France.
- Christopher D. Manning and Hinrich Schütze. (1999). *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts.
- Daniel Marcu and William Wong. (2002). A phrase-based, joint probability model for statistical machine translation. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, Philadelphia, PA, July.
- Daniel Marcu. (2001). Towards a unified approach to memory- and statistical-based machine translation. In *Proc. of ACL 2001*, pages 378–385, Toulouse, France, July.
- Makoto Nagao. (1984). A framework of a mechanical translation between Japanese and English by analogy principle. In Elithorn and R. Banerji, editors, *Artificial and Human Intelligence*. NATO Publications.
- Franz Josef Och and Hermann Ney. (2002). Discriminative training and maximum entropy models for statistical machine translation. In *Proc. of ACL 2002*, pages 295–302, Philadelphia, PA.
- Franz Josef Och, Nicola Ueffing, and Hermann Ney. (2001). An efficient a* search algorithm for statistical machine translation. In *Proc. of the ACL-2001 Workshop on Data-Driven Machine Translation*, pages 55–62, Toulouse, France, July.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proc. of ACL 2002*, pages 311–318.
- Stephen D. Richardson, William B. Dolan, Arul Menezes, and Monica Corston-Oliver. (2001). Overcoming the customization bottleneck using example-based mt. In *Proceedings of the ACL 2001 Workshop on Data-Driven Methods in Machine Translation*, pages 9–16.
- Eiichiro Sumita. (2001). Example-based machine translation using dp-matching between word sequences. In *Proc. of the ACL-2001 Workshop on Data-Driven Machine Translation*, pages 1–8, Toulouse, France, July.
- 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. In *Proc. of LREC 2002*, pages 147–152, Las Palmas, Canary Islands, Spain, May.
- Christoph Tillmann and Hermann Ney. (2000). Word re-ordering and dp-based search in statistical machine translation. In *Proc. of the COLING 2000*, July–August.
- Hideo Watanabe and Hiroshi Maruyama. (1994). A transfer system using example-based approach. *IE-ICE Transactions on Information and Systems*, E77-D(2):247–257.
- Taro Watanabe and Eiichiro Sumita. (2002). Bidirectional decoding for statistical machine translation. In *Proc. of COLING 2002*, pages 1079–1085, Taipei, Taiwan, August.
- Andy Way. (1999). Translating with examples. In *Proc. of Workshop of example-based machine translation, MTSummit VIII*.
- Kenji Yamada and Kevin Knight. (2001). A syntax-based statistical translation model. In *Proc. of ACL 2001*, Toulouse, France.