EBMT, SMT, Hybrid and More: ATR Spoken Language Translation System

Eiichiro SUMITA, Yasuhiro AKIBA, Takao DOI, Andrew FINCH, Kenji IMAMURA, Hideo OKUMA, Michael PAUL, Mitsuo SHIMOHATA, Taro WATANABE

ATR Spoken Language Translation Research Laboratories
2-2-2 Hikaridai, Keihanna Science City, Kyoto 619-0288, JAPAN
eiichiro.sumita@atr.jp

Abstract

This paper introduces ATR's project named Corpus-Centered Computation (C3), which aims at developing a translation technology suitable for spoken language translation.

C3 places *corpora* at the center of its technology. Translation knowledge is *extracted* from corpora, translation quality is *gauged* by referring to corpora, the best translation among multiple-engine outputs is *selected* based on corpora, and the corpora themselves are *paraphrased or filtered* by automated processes to improve the data quality on which translation engines are based.

In particular, this paper reports the hybridization architecture of different machine translation systems, our technologies, their performance on the IWSLT04 task, and paraphrasing methods.

1. Introduction

There are two main strategies used in corpus-based translation:

1. *Example-Based Machine Translation* (EBMT) [1]:

EBMT uses the corpus directly. EBMT retrieves the translation examples that are best matched to an input expression and then adjusts the examples to obtain the translation.

2. *Statistical Machine Translation* (SMT) [2]:

SMT learns statistical models for translation from corpora and dictionaries and then searches for the best translation at run-time according to the statistical models for language and translation.

By using the IWSLT04 task, this paper describes two endeavors that are independent at this moment: (a) a *hybridization of EBMT and statistical models*, and (b) a new approach for SMT, *phrase-based HMM*. (a) is used in the "unrestricted" Japanese-to-English track (Section 2), and (b) is used in "supplied" Japanese-to-English and Chinese-to-English tracks (Section 3). In addition, paraphrasing technologies, which are not used in the IWSLT04 task but boost translation performance, are also introduced in Section 4.

2. Hybrid MT System (Unrestricted J-to-E Track)

No complete translation system has emerged nor is likely to emerge in the foreseeable future. Every approach to translation has its own way of acquiring translation knowledge and using the knowledge. Each system generates its peculiar errors in attempting translation. As a result, translation performance differs sentence-by-sentence, system-by-system. There is the possibility of boosting translation performance through exploitation of multiple translations generated by different systems. Among several possible architectures to integrate multiple translation engines (Section 2.5), we demonstrate the acrchitecture below (Sections from 2.1 to 2.4) as one effective approach.

2.1. A Hybridization: Multiple EBMTs Followed By A Selector Based On SMT Models

It is important to integrate "different" types of element machine translation systems in order to boost the overall performance by having them compensate each other. We propose an architecture in which multiple EBMT engines work in parallel and their outputs are passed to a post-process that selects the best candidate according to SMT models.

Most EBMT systems employ phrases or sentences as the translation unit so that they can translate while taking a wider perspective in order to handle case relations, idiomatic expressions, sentence structure, and so on. However, when there is ambiguity in translation, EBMT selects the best translation mainly by the similarity between the input and the source part of the example. EBMT's validation of its translation is flawed.

On the other hand, SMT employing IBM models translates an input sentence by a combination of word transfer and word re-ordering. Therefore, when it is applied to a language pair in which the word order is much different (e.g. English and Japanese), it is difficult to find a globally optimal solution due to the enormous search space. However, SMT can sort translations in the order of their quality according to its statistical models.

We show two different EBMT systems here, briefly explain each system, and then compare them. Finally, we ex-

plain the selector used to determine the best from multiple translations based on SMT models.

2.2. Two EBMTs

2.2.1. D3, DP-based EBMT

Sumita [3] proposed D3 (Dp-match Driven transDucer), which exploits DP-matching between word sequences.

Let's illustrate the process with a simple sample below. Suppose we are translating a Japanese sentence into English. The Japanese input sentence (1-j) is translated into the English sentence (1-e) by utilizing the English sentence (2-e), whose source sentence (2-j) is similar to (1-j). The common parts are unchanged, and the different portions, shown in bold face, are substituted by consulting a bilingual dictionary.

;;; A Japanese input
(1-j) **iro**/ga/ki/ni/iri/masen
;;; the most similar example in corpus
(2-j) **dezain**/ga/ki/ni/iri/masen
(2-e) I do not like the **design**.
;;; the English output
(1-e) I do not like the **color**.

We retrieve the most similar source sentence of examples from a bilingual corpus. For this, we use *DP-matching*, which tells us the *edit distance* between word sequences while giving us the matched portions between the input and the example.

The edit distance is calculated as follows. The count of the inserted words, the count of the deleted words, and the semantic distance of the substituted words are summed. Then, this total is normalized by the sum of the lengths of the input and the source part of translation example. The semantic distance between two substituted words is calculated by using the hierachy of a thesaurus[4].

Our language resources in addition to a bilingual corpus are a bilingual dictionary, which is used for generating target sentences, and thesauri of both languages, which are used for incorporating the semantic distance between words into the distance between word sequences. Furthermore, lexical resources are also used for word alignment.

2.2.2. HPAT, Grammar-based EBMT

The second EBMT is different from the first EBMT in that it parses bitexts of a parallel coupus with grammars for both source and target languages.

Imamura [5] proposed a new phrase alignment approach called Hierarchical Phrase Alignment (HPA). First, two sentences are tagged and parsed independently. This operation obtains two syntactic trees. Next, words are linked by the word alignment program. Then, HPA retrieves equivalent phrases that satisfy two conditions: 1) words in the pair correspond with no deficiency and no excess; 2) the phrases are of the same syntactic category.

Imamura [6] subsequently proposed HPA-based translation (HPAT). HPAed bilingual trees include all information necessary to automatically generate transfer patterns. Translation is done according to transfer patterns using the TDMT engine [7]. First, the source part of transfer patterns are utilized, and source structure is obtained. Second, structural changes are performed by mapping source patterns to target patterns. Finally, lexical items are inserted by referring to a bilingual dictionary, and then a conventional generation is performed.

Finally, Imamura [8] proposed a *feedback cleaning* method that utilizes automatic evaluation to remove incorrect/redundant translation rules. BLEU was utilized to measure translation quality for the feedback process, and the hill-climbing algorithm was applied in searching for the combinatorial optimization. Utilizing the features of this task, incorrect/redundant rules were removed from the initial solution, which contains all rules acquired from the training corpus. Our experiments showed a considerable improvement in MT quality.

2.2.3. Comparison of Two EBMTs

As can be seen in Section 2.2.1, Section 2.2.2, and Table 1, the main difference between the two EBMT systems is in their use of grammars.

Table 1: *Resources used for two EBMTs in IWSLT04 unresticted Japanese-to-English track.*

	D3	HPAT
bilingual corpus	travel domain (20K)	travel domain (20K)
bilingual dictionary	in-house	in-house
thesaurus	in-house	in-house
grammar	N.A.	in-house

D3 achieves a good quality, when there is a similar translation example in the parallel corpus, otherwise D3 may fail to produce a good translation. On the contrary, HPAT produces a modest quality translation for most of the inputs (Table 2).

Table 2: *Features of the two EBMTs.*

	D3	HPAT
Unit	sentence	grammatical unit
Coverage	narrow	wide
Quality	good	modest

This is confirmed by the subjective evaluation of quality in Table 3. Here, we show MT's quality by using five ranks, S, A, B, C, and D [1], from good quality to poor qual-

[1]The five grades are defined as follows: (S) Splendid: fluent like a naitive speaker; (A) Perfect: no problem with either information or grammar; (B)

ity. This is judged by English native-speakers who are also familiar with Japanese. The evaluator investigates bilingual information, i.e., the source sentence and its MT output. This is an overall score that considers both adequacy and fluency, which are particular scores used in the IWSLT evaluation campaign. The IWSLT evaluator makes a monolingual evaluation, i.e., a reference translation made in advance by a professional translator and MT output, and judges the adequacy and fluency of the MT translation.

Table 3: *ATR's Overall Subjective Evaluation - percentages of S, A, B, C, and D ranks.*

	D3	HPAT
S	57.00	38.60
A	13.00	21.20
B	7.60	17.60
C	5.80	6.00
D	16.60	16.60

The portion of translations with rank "S" for D3 is very large, while the portions of translations with ranks "A," "B," and "C" are relatively small. Thus, the slope is very steep, while the slope of HPAT is gentle.

2.3. SMT-based Selector

We proposed an SMT-based method of automatically selecting the best translation among outputs generated by multiple machine translation (MT) systems [9].

Conventional approaches to the selection problem include a method that automatically selects the output to which the highest probability is assigned according to a language model (LM). [10] These existing methods have two problems. First, they do not check whether information on source sentences is adequately translated into MT outputs, although they do check the fluency of MT outputs. Second, they do not take the statistical behavior of assigned scores into consideration.

The proposed approach scores MT outputs by using not only the language but also a translation model (TM). To conduct a statistical test later, this scoring is done by using each of multiple pairs of language and translation models. The method, then, checks whether the average TM$*$LM score of an MT output is significantly higher than that of another MT output. This check uses a multiple comparison test based on the Kruskal-Wallis test [11].

2.4. Results

2.4.1. Selecting Effect

As shown in Table 4, all of the metrics taken together show that the proposed selector outperforms both element translation systems; for example, mWER is decreased by 2.55 (about 7.5% reduction) from 28.86 to 26.31.

Table 4: *Objective Evaluation.*

	D3	HPAT	SELECT	DIFF.
BLEU	60.36	49.33	63.06	+3.00
NIST	10.35	9.78	10.72	+0.37
GTM	77.70	76.88	79.67	+1.97
mWER	28.86	37.18	26.31	-2.55
mPER	26.07	31.06	23.33	-2.97

Table 5: *ATR's Overall Subjective Evaluation - cumulative percentages of S, A, B, C, and D ranks.*

	D3	HPAT	SELECT	DIFF.
S	57.00	38.60	59.80	+2.80
S,A	70.00	59.80	73.00	+3.00
S,A,B	77.60	77.40	82.40	+4.80
S,A,B,C	83.40	83.40	87.80	+4.40
D	16.60	16.60	12.20	-4.40

Next, the relationship between translation quality of element systems and gain by the selector was analyzed. Table 5 shows that the proposed selector reduces the number of low-quality translations (ranked "D") while it increases the number of high-quality translations (ranked "S" to "B").

2.4.2. Performance vs. Corpus Size

Since the methods are corpus-based, the quantity of the corpus determines the system performance.

Table 6: *mWER vs. Corpus size.*

Training corpus	D3	HPAT
IWSLT-supplied (2K)	45.71	47.28
(20K)	28.86	37.18
DIFF.	-16.85	-10.10

The corpus used in this experiment is ten times larger than the supplied corpus, and the drastic reduction in mWER has been demonstrated (Table 6).

However, the quality with the small corpus is not so bad in the subjective evaluation shown in Table 7. We conjecture that adequacy is not low even with the supplied corpus, and the translation become similar to native English, that is, its fluency improves as the size of corpus increases.

2.5. Discussion

Related works have proposed ways to merge MT outputs from multiple MT systems [12] in order to output better translations. When the source language and the target language have similar sentence structures, this merging ap-

Good: easy to understand, with either some unimportant information missing or flawed grammar; (C) Fair: broken, but understandable with effort; (D) Unacceptable: important information has been translated incorrectly.

Table 7: *ATR's Overall Subjective Evaluation - IWSLT supplied corpus.*

	D3	HPAT	SELECT
S	34.80	25.20	34.00
S,A	47.40	44.20	50.60
S,A,B	62.60	70.40	72.20
S,A,B,C	73.40	80.40	81.80
D	26.60	19.60	18.20

proach is very attractive. On the other hand, when the source language and the target language have different sentence structures, such as English and Japanese, we often have translations whose structures are different from each other for a single input sentences. Thus, the authors regard the merging approach as less suitable than the approach of selecting.

Hybridization can be implemented in several arichitectures, for example, *SMT followed by EBMT*, *SMT and EBMT in parallel*, and so on. Which archtecture is best is still an interesting open question.

In addition to the merging and selecting approaches, a modification approach can be taken. For example, Marcu [14] proposed a method in which initial translations are constructed by combining bilingual phrases from translation memory, which is followed by modifying the translations by greedy decoding [15]. Watanabe et al. [16] proposed a decoding algorithm in which translations that are similar to the input sentence are retrieved from bilingual corpora and then modified by greedy decoding.

3. Phrase-based HMM SMT System (Supplied J-to-E and C-to-E Tracks)

This section describes an innovative approach to statistical translation modeling, namely the phrase-based HMM translation model. The model directly structures the phrase-based translation approach in a Hidden Markov structure and proposes an efficient way to estimate and induce phrase translation pairs in a uniform fashion.

In the statistical approach to machine translation, originally proposed in [2], the problem of translating a source text in a foreign language, $\mathbf{f}$, into a target language, for instance English, $\mathbf{e}$ is formulated as the maximization problem of

$$\hat{\mathbf{e}} = \underset{\mathbf{e}}{\operatorname{argmax}} P(\mathbf{e}|\mathbf{f}) \tag{1}$$

The noisy channel modeling of the above problem resulted in

$$\hat{\mathbf{e}} = \underset{\mathbf{e}}{\operatorname{argmax}} P(\mathbf{f}|\mathbf{e})P(\mathbf{e}) \tag{2}$$

Many previous efforts in the phrase-based approach to statistical machine translation basically approximated the former term, $P(\mathbf{f}|\mathbf{e})$, as the products of sequence of phrase translations with additional constraints [17, 18, 19]:

$$P(\mathbf{f}|\mathbf{e}) \approx \prod_i P(\bar{\mathbf{f}}_i|\bar{\mathbf{e}}_{\mathbf{a}_i}) \tag{3}$$

where $\bar{\mathbf{f}}_i$ is the ith phrase of the phrase-segmented sentence $\bar{\mathbf{f}}_1^m$ for $\mathbf{f}$, and $\mathbf{a}_i$ is the phrase alignment for the phrase-segmented texts. [2]

Instead, we introduced two new hidden variables, $\bar{\mathbf{f}}$ and $\bar{\mathbf{e}}$, to explicitly capture the phrase translation relationship:

$$P(\mathbf{f}|\mathbf{e}) = \sum_{\bar{\mathbf{f}},\bar{\mathbf{e}}} P(\mathbf{f},\bar{\mathbf{f}},\bar{\mathbf{e}}|\mathbf{e}) \tag{4}$$

The term $P(\mathbf{f},\bar{\mathbf{f}},\bar{\mathbf{e}}|\mathbf{e})$ is further decomposed into three terms:

$$P(\mathbf{f},\bar{\mathbf{f}},\bar{\mathbf{e}}|\mathbf{e}) = P(\mathbf{f}|\bar{\mathbf{f}},\bar{\mathbf{e}},\mathbf{e})P(\bar{\mathbf{f}}|\bar{\mathbf{e}},\mathbf{e})P(\bar{\mathbf{e}}|\mathbf{e}) \tag{5}$$

The first term of Equation 5 represents the probability that a segmented input sentence $\bar{\mathbf{f}}$ can be reordered and generated as the input text of $\mathbf{f}$. The second term indicates the translation probability of the two phrase sequences of $\bar{\mathbf{e}}$ and $\bar{\mathbf{f}}$. The last term is the likelihood of the phrase-segmented text $\bar{\mathbf{e}}$ generated from $\mathbf{e}$. We call these terms the Phrase Segmentation Model, the Phrase Translation Model, and the Phrase Ngram Model, respectively.

3.1. Phrase Ngram Model

The phrase ngram model is approximated as:

$$P(\bar{\mathbf{e}}|\mathbf{e}) \approx \prod_i P(\bar{\mathbf{e}}_i|\bar{\mathbf{e}}_{i-1}) \tag{6}$$

$P(\bar{\mathbf{e}}_i|\bar{\mathbf{e}}_{i-1})$ is treated as the bigram constraints of adjacent translated phrases $\bar{\mathbf{e}}_i$ and $\bar{\mathbf{e}}_{i-1}$.

The phrase ngram model can be easily estimated with the Forward-Backward algorithm by expanding all possible phrase segmentations of $\mathbf{e}$ into a lattice structure $\bar{\mathbf{E}}$ as shown in Figure 1. Each node in the lattice represents a particular phrase $\bar{\mathbf{E}}_i$ in a sentence $\mathbf{e}$ connected by edges with associated probability of $P(\bar{\mathbf{E}}_i|\bar{\mathbf{E}}_{i'})$.

The estimation procedure can be roughly summarized as follows.

1. Initialize the probability table.
2. For each sentence $\mathbf{e}$ in the training corpus, estimate the posterior probabilities $P(\bar{\mathbf{E}}_i,\bar{\mathbf{E}}_{i'}|\mathbf{e})$ on the lattice using the Forward-Backward algorithm.
3. Estimate the prior probabilities based on the maximum likelihood estimation by using the estimated posterior probabilities as the frequency of the occurrence of words:
$$P(\bar{\mathbf{E}}_i|\bar{\mathbf{E}}_{i'}) = \frac{\sum_{\mathbf{e}} P(\bar{\mathbf{E}}_i,\bar{\mathbf{E}}_{i'}|\mathbf{e})}{\sum_{\mathbf{e}}\sum_{\bar{\mathbf{E}}_i} P(\bar{\mathbf{E}}_i,\bar{\mathbf{E}}_{i'}|\mathbf{e})} \tag{7}$$
4. Iterate steps 2 and 3 until a termination condition is satisfied.

[2] A phrase is simply a consecutive sequence of words and is not always linguistically coherent.

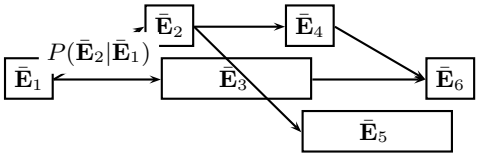

Figure 1: Phrase Ngram Model

3.2. Phrase Segmentation Model

According to the generative modeling represented in Equation 5, the term $P(\mathbf{f}|\bar{\mathbf{f}}, \bar{\mathbf{e}}, \mathbf{e})$ can be regarded as the distortion probability of how a phrase segmented sentence $\bar{\mathbf{f}}$ will be reordered to form the source sentence $\mathbf{f}$.

Instead, we model this as the likelihood of a particular phrase segment $\bar{\mathbf{f}}_j$ observed in $\mathbf{f}$:

$$P(\mathbf{f}|\bar{\mathbf{f}}, \bar{\mathbf{e}}, \mathbf{e}) \propto P(\bar{\mathbf{f}}|\mathbf{f}) \quad (8)$$

$$\approx \prod_j P(\bar{\mathbf{f}}_j|\mathbf{f}) \quad (9)$$

The segmentation model is realized as the unigram posterior probability of the phrase ngram model presented in Section 3.1. To briefly summarize, the unigram posterior probability can be efficiently computed by the Forward-Backward algorithm using the lattice structure $\bar{\mathbf{F}}$ for $\mathbf{f}$:

$$P(\bar{\mathbf{F}}_j|\mathbf{f}) = \frac{P(\bar{\mathbf{F}}_j, \mathbf{f})}{\sum_{\bar{\mathbf{F}}_j} P(\bar{\mathbf{F}}_j, \mathbf{f})} \quad (10)$$

The phrase segmentation model can be viewed as the prior term to assign a certain weight to a particular phrase given a source text. If we restrict the phrase length to 1, i.e. each phrase consisting of only one word, then the phrase segmentation model will assign 1 to all phrases.

3.3. Phrase Translation Model

The phrase translation model is approximated so that the phrase translation can be captured as the product of the individual phrase translations.

$$P(\bar{\mathbf{f}}|\bar{\mathbf{e}}, \mathbf{e}) \approx \prod_j P(\bar{\mathbf{f}}_j|\bar{\mathbf{e}}_{\mathbf{a}_j}) \quad (11)$$

where the $\mathbf{a}_i$ represents phrase alignment as seen in word alignment based translation model, such as the IBM Models.

3.4. Phrase-based HMM Statistical Translation

Combining all of the submodels – the phrase ngram model, the phrase segmentation model, and the phrase translation model – Equation 4 can be rewritten as

$$P(\mathbf{f}|\mathbf{e}) \approx \sum_{\bar{\mathbf{e}}, \bar{\mathbf{f}}} \prod_{j,i} P(\bar{\mathbf{f}}_j|\mathbf{f}) P(\bar{\mathbf{f}}_j|\bar{\mathbf{e}}_i) P(\bar{\mathbf{e}}_i|\bar{\mathbf{e}}_{i'}) \quad (12)$$

If the phrase segmented sentences $\bar{\mathbf{e}}$ and $\bar{\mathbf{f}}$ are expanded into the corresponding lattice structures of $\bar{\mathbf{E}}$ and $\bar{\mathbf{F}}$, then

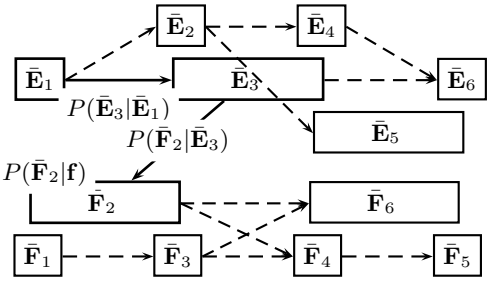

Figure 2: Phrase-based HMM Statistical Translation Model

Equation 12 can be regarded as a Hidden Markov Model in which each source phrase $\bar{\mathbf{F}}_j$ in the lattice $\bar{\mathbf{F}}$ is treated as an observation emitted from a state $\bar{\mathbf{E}}_i$, a target phrase, in the lattice $\bar{\mathbf{E}}$, as shown in Figure 2.

The use of the phrase-based HMM structure has already been proposed in [20] in the context of aligning documents and abstracts. In their approach, jump probabilities were explicitly encoded as the state transitions that roughly corresponded to the alignment probabilities in the context of the word-based statistical translation model. The use of the explicit jump or alignment probabilities served for the completeness of the translation modeling at the cost of the enormous search space needed to train the phrase-based HMM structure.

In our approach, the state transitions are governed by the phrase ngram model, bigram of phrase connection probabilities, but this method ignores phrase alignment probabilities. Therefore, the phrase-based HMM translation model is a deficient model. However its simplicity contributes to the faster estimation of parameters.

3.5. Parameter Estimation

The parameters for the phrase-based HMM translation model can be efficiently estimated by using the Forward-Backward algorithm briefly described in Section 3.1.

For the Forward-Backward procedure, we define two auxiliary variables, $\alpha(\mathbf{e}_{i_1}^{i_2}, \mathbf{f}_{j_1}^{j_2})$ and $\beta(\mathbf{e}_{i_1}^{i_2}, \mathbf{f}_{j_1}^{j_2})$. $\alpha(\mathbf{e}_{i_1}^{i_2}, \mathbf{f}_{j_1}^{j_2})$ represents the forward estimates of the probability of the phrase $\mathbf{e}_{i_1}^{i_2}$ translated into $\mathbf{f}_{j_1}^{j_2}$ after the emission of the all phrase combinations presented in $\mathbf{e}_1^{i_1-1}$. Similarly, $\beta(\mathbf{e}_{i_1}^{i_2}, \mathbf{f}_{j_1}^{j_2})$ represents the backward estimates of the probability of the phrase $\mathbf{e}_{i_1}^{i_2}$ translated into $\mathbf{f}_{j_1}^{j_2}$ considering the all right phrase combinations of $\mathbf{e}_{i_2+1}^{l}$.

Therefore, the Forward-Backward algorithm can be for-

mulated to solve the recursions

$$\alpha(\mathbf{e}_{i_1}^{i_2}, \mathbf{f}_{j_1}^{j_2}) = \sum_{i'=1}^{i_1-2} \sum_{\mathbf{f}_{j_1'}^{j_2'} \cap \mathbf{f}_{j_1}^{j_2} = \varnothing} \alpha(\mathbf{e}_{i'}^{i_1-1}, \mathbf{f}_{j_1'}^{j_2'}) \times P(\mathbf{e}_{i_1}^{i_2} | \mathbf{e}_{i'}^{i_1-1}) P(\mathbf{f}_{j_1}^{j_2} | \mathbf{e}_{i_1}^{i_2}) P(\mathbf{f}_{j_1}^{j_2} | \mathbf{f}) \quad (13)$$

$$\beta(\mathbf{e}_{i_1}^{i_2}, \mathbf{f}_{j_1}^{j_2}) = \sum_{i'=i_2+2}^{l} \sum_{\mathbf{f}_{j_1'}^{j_2'} \cap \mathbf{f}_{j_1}^{j_2} = \varnothing} \beta(\mathbf{e}_{i_2+1}^{i'}, \mathbf{f}_{j_1'}^{j_2'}) \times P(\mathbf{e}_{i_2+1}^{i'} | \mathbf{e}_{i_1}^{i^2}) P(\mathbf{f}_{j_1'}^{j_2'} | \mathbf{e}_{i_2+1}^{i'}) P(\mathbf{f}_{j_1'}^{j_2'} | \mathbf{f}) \quad (14)$$

To overcome the problem of local convergence often observed in the EM algorithm [21], we use the lexicon model from the GIZA++ [22] training as the initial parameters for the phrase translation model. In addition, the phrase ngram model and the phrase segmentation models are individually trained over the monolingual corpus and remained fixed during the HMM iterations.

3.6. Phrase Segment Induction

Equations 13 and 14 involve summation over all possible contexts, either in its left-hand-side or right-hand-side on the lattice structure of $\bar{\mathbf{E}}$, and the summation over all possible segmentation over $\bar{\mathbf{F}}$. Since the computation is still enormous, even with the help of dynamic programming, we restrict the possible segmentation to those phrase translation pairs induced before the estimation.

The phrase pairs are induced by first considering all possible bilingual phrase pairs in a training corpus using the product of two phrase translation probabilities:

$$P(\bar{e}|\bar{f})P(\bar{f}|\bar{e}) = \frac{\mathrm{count}(\bar{e}, \bar{f})^2}{\sum_{\bar{f}} \mathrm{count}(\bar{e}, \bar{f}) \sum_{\bar{e}} \mathrm{count}(\bar{e}, \bar{f})} \quad (15)$$

where $\mathrm{count}(\bar{e}, \bar{f})$ is the cooccurrence frequency of the two phrases $\bar{e}$ and $\bar{f}$. The basic idea of Equation 15 is to capture the bilingual correspondence while considering two directions.

Additional phrases were exhaustively induced based on the intersection/union of the viterbi word alignments of the two directional models, $P(\mathbf{e}|\mathbf{f})$ and $P(\mathbf{f}|\mathbf{e})$, computed by GIZA++ [17].

After the extraction of phrase translation pairs, their monolingual phrase lexicons were extracted and used as the possible segmentation for the source and target sentences.

3.7. Decoder

The decision rule to compute the best translation is based on the log-linear combinations of all subcomponents of translation models as presented in [23].

$$\hat{\mathbf{e}} = \underset{\mathbf{e}}{\mathrm{argmax}} \frac{1}{Z(\mathbf{f})} \sum_j \lambda_j \log Pr_j(\mathbf{e}, \mathbf{f}) \quad (16)$$

where $Pr_j(\mathbf{e}, \mathbf{f})$ are the subcomponents of translation models, such as the phrase ngram model or the language model, and λ_j is the weight for each model. The weighting parameters, λ_j, can be efficiently computed based either on the maximum likelihood criterion [23] by IIS or GIS algorithms or on the minimum error rate criterion [24] by some unconstrained optimization algorithms, such as the Downhill Simplex Method [25].

The decoder is taken after the word-graph-based decoder [26], which allows the multi-pass decoding strategies to incorporate complicated submodel structures. The first pass of the decoding procedure generates the word-graph, or the lattice, of translations for an input sentence by using a beam search. On the first pass, the submodels of all phrase-based HMM translation models were integrated with the word-based trigram language model and the class 5-gram model. The second pass uses A* strategy to search for the best path of translation on the generated word-graph.

3.8. Results

The results appear strange in two points: (1) Our proposal didn't work well for the Japanese-to-English track but did work well for the Chinese-to-English track; (2) Our proposal achieved high fluency but marked low adequacy.

The former was attributed to the fact that we had to narrow down the beamwidth for handling long Japanese input. The latter was attributed to the fact that we tuned our parameter to mWER and we exploited phrase models as well.

Table 8: *Evaluation - IWSLT Chinese-to-English supplied task.*

System	mWER	Fluency	Adequacy
Top	45.59	38.20	33.38
Our	**46.99**	**38.20**	**29.50**
Bottom	61.69	25.04	29.06

4. Other Features of C3

This section introduces another feature of C3: paraphrasing and filtering corpora, which are not used in the IWSLT04 task but are useful for boosting MT performance.

The large variety of possible translations in a corpus causes difficulty in building machine translation on the corpus. Specifically, theis variety makes it more difficult to find appropriate translation examples for D3, to extract good transfer patterns for HPAT, and to estimate the parameters for SMT.

We propose ways to overcome these problems by paraphrasing corpora through automated processes or filtering corpora by abandoning inappropriate expressions.

4.1. Paraphrasing

Three methods have been investigated for automatic paraphrasing. (1) Shimohata et al. [27] grouped sentences by the equivalence of the translation and extract rules of paraphrasing by *DP-matching*. (2) Finch et al. [28] clustered sentences in a paraphrase corpus to obtain pairs that are similar to each other for training *SMT models*. Then by using the models, the decoder generates a paraphrase. (3) Finch et al. [29] developed a paraphraser based on data-oriented parsing, which utilizes synatactic information within an example-based framework.

The experimental results indicate that the EBMT based on normalization of the source side had increased coverage [30] and that the SMT created on the normalized target sentences had a reduced word-error rate [31]. Finch et al. [32] demonstrated that the expansion of reference sentences by paraphrasing is effective for automatic machine translation evaluation.

In addition, longer sentences, which are inherent in spoken language, can be translated effectively by splitting them into short sentences and then concatenating the translated short sentences. Doi proposed a new splitting method based on N-gram and sentence similarity [33].

4.2. Filtering

Imamura et al. [34] proposed a calculation that measures the literalness of a translation pair and called it Translation Correspondece Rate (TCR). After the word alignment of a translation pair, TCR is calculated as the rate of the aligned word count over the count of words in the translation pair. After abandoning the non-literal parts of the corpus, the HPAT transfer patterns are acquired. The effect of this measure has been confirmed by the improvement in translation quality.

5. Conclusions

Our project, called C3, places corpora at the center of speech-to-speech technology.

In this paper, (1) a hybridization of multiple EBMTs followed by a statitical selector, (2) a new SMT, phrase-based HMM SMT, and (3) paraphrasing methods are introduced. Good performance by translation components is demonstrated through experiments, including the IWSLT04 task.

Furthermore, we plan to pursue a better blend of multiple processes, EBMT, SMT and other innovations such as paraphrasing.

6. Acknowledgements

The research reported here was supported in part by a contract with the National Institute of Information and Communications Technology (NICT) of Japan entitled, “A study of speech dialogue translation technology based on a large corpus”. The authors’ heartfelt thanks go to Kadokawa-Shoten for providing the Ruigo-Shin-Jiten.

7. References

[1] Nagao, M., “A Framework of a Mechanical Translation between Japanese and English by Analogy Principle”, Artificial and Human Intelligence, North Holland, 173–180, 1984.

[2] Brown, P. F., “A Statistical approach to machine translation,” Computational Linguistics, 16 (2), pp. 79–85, 1990.

[3] Sumita, E., “Example-based machine translation using DP-matching between word sequences”, Workshop on DDMT, ACL, pp. 1-8, 2001.

[4] Sumita, E., “Experiments and prospects of example-based machine translation”, ACL, pp. 185-192, 1991.

[5] Imamura, K., “Hierarchical phrase alignment harmonized with parsing”, NLPRS, pp. 377-384, 2001.

[6] Imamura, K., “Application of transfer knowledge acquired by hierarchical phrase alignmnet”, TMI, pp. 74–84, 2002.

[7] Furuse, O., “Constituent boundary parsing for example-based machine translation”, Coling, pp. 105-111, 1994.

[8] Imamura, K., “Feedback cleaning of machine translation rules using automatic evaluation”, ACL, pp. 447-454, 2003.

[9] Akiba, Y., “Using language and translation models to select the best from outputs from multiple MT systems”, Coling, pp. 8-14, 2002.

[10] Callison-Burch, C., “A program for automatically selecting the best output from multiple machine translation engines”, MTS, pp. 63-66, 2002.

[11] Hochberg, C., Multiple comparison procedure, Wiley, 1983.

[12] Hogan, C., “An evaluation of multi-engine MT”, AMTA, pp. 113-123, 1998.

[14] Marcu, D., “Toward unified approach to memory- and statistical-based machine translation”, ACL, pp. 378-385, 2001.

[15] Germann, U., “Fast decoding and optimal decoding for machine translation”, ACL, pp. 228-235, 2001.

[16] Watanabe, T., “Example-based decoding for statistical machine translation”, MTS, pp. 410-417, 2003.

[17] Koehn, P., “Statistical phrase-based translation”, HLT-NAACL, pp. 48-54, 2003.

[18] Vogel, S., “The CMU statistical translation system”, MTS, pp. 402-409, 2003.

[19] Tillmann, C., “A projection extension algorithm for statistical machine translation”, EMNLP, pp. 402-409, 2003.

[20] Daume III, H., “A phrase-based hmm approach to document/abstract alignment”, EMNLP, pp. 119-126, 2004.

[21] Dempster, A. P., “Maximum likelihood from imcomplete data via the em algorithm”, Journal of Royal Statistical Society, B(39), pp. 1–38, 1977.

[22] Och, F., “Systematical comparison of various statistical alignment models”, Computational Linguistics, 29(1), pp. 19–51, 2003.

[23] Och, F., “Discriminative training and maximum entropy model for statistical machine translation”, ACL, pp. 295–302, 2002.

[24] Och, F., “Minimum error rate training in statistical machine translation”, ACL, pp. 160–167, 2003.

[25] Press, W. H., ”Numerical Recipes in C++”, Cambridge University Press, 2002.

[26] Ueffing, N., “Generation of word graphs in statistical machine translation”, EMNLP, pp. 156–163, 2002.

[27] Shimohata, M., “Automatic paraphrasing based on parallel corpus for normalization”, LREC, 2002.

[28] Finch, A., “Paraphrasing by Statistical machine translation”, FIT, 2002.

[29] Finch, A., “Data-oriented Paraphrasing”, RANLP, 2003.

[30] Shimohata, M., “Identifying synonymous expressions from a bilingual corpus for exmaple-based machine translation”, Workshop on machine translation in Asia, Coling, 2002.

[31] Watanabe, T., “Statistical machine translation based on paraphrased corpus”, LREC, 2002.

[32] Finch, A., “Using paraphraser to improve machine translation evaluation”, IJCNLP, 2004.

[33] Doi, T., “Splitting input sentence for machine translation using language model with sentence similarity”, Coling, 2004.

[34] Imamura, K., “Automatic construction of machine translation knowledge using translation literalness”, EACL, pp. 155-162, 2003.