Fully Syntactic EBMT System of KYOTO Team in NTCIR-8

Toshiaki Nakazawa Sadao Kurohashi
Graduate School of Informatics, Kyoto University
Yoshida-honmachi, Sakyo-ku
Kyoto, 606-8501, Japan
{nakazawa, kuro}@nlp.kuee.kyoto-u.ac.jp

ABSTRACT

This paper describes "KYOTO" EBMT system that attended the patent translation task at NTCIR-8. When translating very different language pairs, it is very important to handle sentences in tree structures to overcome the difference. Most of the recent translation methods consider a sentence as just a sequence of words. Some works incorporate tree structures in some parts of whole translation process, but not all the way from model training (parallel sentence alignment) to decoding. "KYOTO" is a fully tree-based translation system where we use the statistical tree-based phrase alignment and example-based translation.

Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Machine Translation*

General Terms

Design, Theory, Performance

Keywords

Syntactic EBMT, Tree-based, Statistical Phrase Alignment

1. INTRODUCTION

We consider that it is quite important to use linguistic information in translation process when tackling on very different language pairs such as Japanese and English, and one of the most important information is a sentence structure. Most of the state-of-the-art translation methods handle a sentence as just a sequence of words such as Phrase-based SMT [10] and Hiero[4]. On the other hand, some work incorporated structural information in the parallel sentence alignment [3, 17, 5, 19], but they did not mention tree-based translation. Another work proposed tree-based translation [13], but its word alignment comes from the conventional IBM models [1], which is a totally sequential model, with some heuristic symmetrization rules for combining the alignment results of both directions. To our knowledge, there is no framework which uses tree structures from the beginning of alignment to the end of translation, but such framework is actually desirable.

In this paper, we propose a fully tree-based translation framework based on dependency tree structures. In the alignment, we use statistical phrase alignment method which models phrase translations and phrase dependency relations. The details are shown in Section 2. It is a kind of tree-based reordering model, and can capture non-local reorderings which sequential word-based models cannot often handle properly. The model is also capable of estimating phrase correspondences automatically without heuristic rules. In the translation, we adopt an example-based machine translation (EBMT) system [15] which is very conformable to the tree structures[1]. EBMT can handle examples which are discontinuous as a word sequence, but continuous structurally. Accordingly, EBMT can quite naturally handle syntactic information. It also considers similarities of neighboring nodes, which is useful for chooseing suitable examples matching the context.

Figure 1 shows the overview of our EBMT system on Japanese-English translation.

Using the example database, new input sentence is translated. The input sentence is parsed and transformed into dependency structure. For all the arbitrary sub-trees, available examples are searched. Translation examples are also parsed in both source and target sides. Of course there are many available examples for one sub-tree, so we give some scores to the examples and use the highest scored example. Also there are many types of sub-tree combinations. We search the best combination by beam-search based on the calculated scores.

In the example, four examples are used. They are combined and finally we can get the output dependency tree. We call the outside nodes of the actually used nodes as "bond" nodes. The bond nodes of one example are replaced by the other example, and thus two examples can be combined. Using the bond information, we don't need to consider word or phrase orders. Bond information naturally resolve the reordering problem.

2. TREE-BASED PHRASE ALIGNMENT

[1] [15] used linguistic phrases as nodes of the dependency trees. Compared to this, the proposed model uses words as nodes.

.

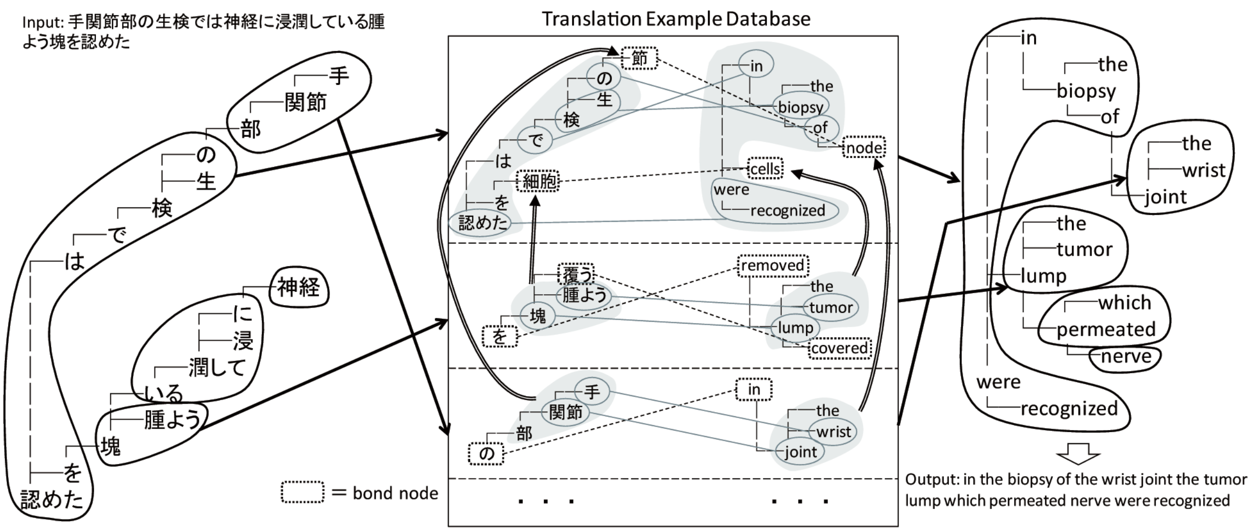

Figure 1: An example of Japanese-English translation.

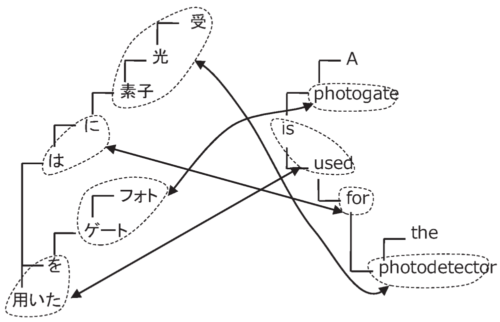

Figure 2: Example of a dependency tree and alignment.

We suppose that Japanese is the source language and English is the target language in the description of our model. Note that the model is not specialized for this language pair, and it can be applied to any language pair.

Because our model uses dependency tree structures, both source and target sentences are parsed beforehand. Japanese sentences are converted into word dependency structures using the morphological analyzer JUMAN [11], and the dependency analyzer KNP [8]. Charniak's nlparser[2] is used to convert English sentences into phrase structures, and then they are transformed into word dependency structures by rules defining head words. Figure 2 shows an example of dependency structures. The root of a tree is placed at the extreme left and words are placed from top to bottom. Note that sentences are not previously segmented into phrases. Phrases are automatically acquired during the model training.

2.1 Overview

This section outlines our proposed model in comparison to the IBM models, which are the conventional statistical alignment models.

In the IBM models [1], the best alignment $\hat{\mathbf{a}}$ between a given source sentence $\mathbf{f}$ and its target sentence $\mathbf{e}$ is acquired by the following equation:

$$\begin{aligned}\hat{\mathbf{a}} &= \operatorname*{argmax}_{\mathbf{a}} p(\mathbf{a}, \mathbf{f}|\mathbf{e}) \\ &= \operatorname*{argmax}_{\mathbf{a}} p(\mathbf{f}|\mathbf{a}, \mathbf{e}) \cdot p(\mathbf{a}|\mathbf{e}) \qquad (1)\end{aligned}$$

$p(\mathbf{f}|\mathbf{a}, \mathbf{e})$ is called *lexicon probability*, and it is in charge of word translations. $p(\mathbf{a}|\mathbf{e})$ is called *alignment probability*, and it is mainly in charge of word reorderings.

In the proposed model, we refine the IBM models in three ways. First, as for lexical probability, we consider phrases instead of words. Second, as for alignment probability, we consider dependencies of words instead of their positions in a sentence.

Finally, the proposed model can find the best alignment $\hat{\mathbf{a}}$ by not using $\mathbf{f}$-to-$\mathbf{e}$ alone, but simultaneously with $\mathbf{e}$-to-$\mathbf{f}$. That is, Equation 1 is modified as follows:

$$\begin{aligned}\hat{\mathbf{a}} = \operatorname*{argmax}_{\mathbf{a}} \quad & p(\mathbf{f}|\mathbf{a}, \mathbf{e}) \cdot p(\mathbf{a}|\mathbf{e}) \cdot \\ & p(\mathbf{e}|\mathbf{a}, \mathbf{f}) \cdot p(\mathbf{a}|\mathbf{f}) \qquad (2)\end{aligned}$$

Since our model regards a phrase as a basic unit, the above formula is calculated in a straightforward way. In contrast, the IBM models can consider a many-to-one alignment by combining one-to-one alignments, but they cannot consider a one-to-many or many-to-many alignment.

The models are estimated by EM-like algorithm which is very similar to [12]. The important difference is that we are using tree structures, and the model is not directed.

We maximize the data likelihood:

$$\max_{\theta} \sum_{(\mathbf{e},\mathbf{f})} \log p(\mathbf{e}, \mathbf{f}; \theta) \qquad (3)$$

In the E-step, we compute the posterior distribution of the alignments with the current parameter θ:

$$q(\mathbf{a}; \mathbf{e}, \mathbf{f}) := p(\mathbf{a}|\mathbf{e}, \mathbf{f}; \theta) \qquad (4)$$

Table 1: A probability calculation example.

f	e	Phrase alignment probability
受 光 素子	photodetector	p(受 光 素子 |photodetector) $\cdot$ p(photodetector| 受 光 素子)
に は	for	p(に は |for) $\cdot$ p(for| に は)
フォト ゲート	photogate	p(フォト ゲート |a photogate) $\cdot$ p(a photogate| フォト ゲート)
を 用いた	is used	p(を 用いた |is used) $\cdot$ p(is used| を 用いた)
NULL	the	p(the|NULL)

e		e → f dependency
e_c	e_p	relation probability
A	photogate	$p_{\mathbf{ef}}$(SAME)
photogate	is	$p_{\mathbf{ef}}$(c)
used	is	$p_{\mathbf{ef}}$(SAME)
for	used	$p_{\mathbf{ef}}$(c)
the	photodetector	$p_{\mathbf{ef}}$(NULL_c)
photodetector	for	$p_{\mathbf{ef}}$(c)

f		f → e dependency
f_c	f_p	relation probability
受	光	$p_{\mathbf{fe}}$(SAME)
光	素子	$p_{\mathbf{fe}}$(SAME)
素子	に	$p_{\mathbf{fe}}$(c)
に	は	$p_{\mathbf{fe}}$(SAME)
は	用いた	$p_{\mathbf{fe}}$(c)
フォト	ゲート	$p_{\mathbf{fe}}$(SAME)
ゲート	を	$p_{\mathbf{fe}}$(c)
を	用いた	$p_{\mathbf{fe}}$(SAME)

In the M-step, we update the parameter θ:

$$\theta' := \underset{\theta}{\operatorname{argmax}} \sum_{\mathbf{a},\mathbf{e},\mathbf{f}} q(\mathbf{a};\mathbf{e},\mathbf{f}) \log p(\mathbf{a},\mathbf{e},\mathbf{f};\theta) \tag{5}$$

In the following sections, we decompose the lexicon probability and alignment probability.

2.2 Phrase Translation Probability

Suppose $\mathbf{f}$ consists of N phrases $F_1, F_2, ..., F_N$ and NULL(F_0), and $\mathbf{e}$ consists of M phrases $E_1, E_2, ..., E_M$ and NULL(E_0). The alignment mapping $\mathbf{A^{fe}}$ consists of associations $j \rightarrow i = A_j^{fe}$ from source phrase j to target phrase $i = A_j^{fe}$.

We consider *phrase translation probability* $p(F_j|E_i)$ instead of word translation probability. There is one restriction: that phrases composed of more than one word cannot be aligned to NULL. Only a single word can be aligned to NULL. Using the phrase translation probability, we decompose lexical probability as follows:

$$p(\mathbf{f}|\mathbf{a},\mathbf{e}) = \prod_{j=1}^{N} p(F_j|E_{A_j^{fe}}) \tag{6}$$

Suppose phrase F_j and E_i are aligned, the probability mass related to this alignment in Equation 6 is as follows:

$$p(F_j|E_i) \cdot p(E_i|F_j) \tag{7}$$

We call this probability for the link between F_j and E_i *phrase alignment probability*. The upper part of Table 1 shows phrase alignment probabilities for the alignment in Figure 2.

2.3 Dependency Relation Probability

Getting a rough idea of the reordering model in the IBM Models, it is defined on the relative position between an alignment and its previous alignment. Our model, on the other hand, considers dependencies of words instead of positional relations.

We start with a dependency relation where a word e_c depends on e_p in $\mathbf{e}$. In a possible alignment, e_c belongs to a phrase E_C, e_p belongs to E_P, so E_C depends on E_P. In this situation, we consider the relation between corresponding phrases in $\mathbf{f}$, $F_{A_P^{ef}}$ and $F_{A_C^{ef}}$. Even if two languages have different word order, their dependency structures are similar in many cases, i.e. $F_{A_C^{ef}}$ tends to depend on $F_{A_P^{ef}}$. Our model takes this tendency into consideration. In order to denote the relationship between phrases, we introduce $rel(e_p, e_c)$, which is defined as the path from $F_{A_P^{ef}}$ to $F_{A_C^{ef}}$. It is represented by applying the notations below:

- 'c' if going down to the *c*hild node
- 'p' if going up to the *p*arent node

For example, in Figure 2, the path from "for" to "photodetector" is 'c', from "the" to "for" is 'p;p' because it travels across two nodes. All the phrases are considered as a single node, so the path from "photogate" to "the" is 'p;c;c;c' with the alignment in Figure 2.

We decompose alignment probability using rel as follows:

$$p(\mathbf{a}|\mathbf{e}) = \prod_{(e_p,e_c)\in D_{\mathbf{e}\text{-}pc}} p_{\mathbf{ef}}(rel(e_p, e_c)) \tag{8}$$

where $D_{\mathbf{e}\text{-}pc}$ denotes a set of parent-child word pairs in $\mathbf{e}$. We call $p_{\mathbf{ef}}(rel(e_p, e_c))$ $\mathbf{e} \rightarrow \mathbf{f}$ *dependency relation probability*. $p_{\mathbf{ef}}$ is a kind of tree-based reordering model.

There are some special cases for rel. When E_C and E_P are the same, that is, e_c and e_p belong to the same phrase, rel is represented as 'SAME'. When e_p is aligned to NULL, e_c is aligned to NULL, and both of them are aligned to NULL, rel is represented as 'NULL_p', 'NULL_c', and 'NULL_b', respectively. The lower part of Table 1 shows dependency relation probabilities corresponding to Figure 2.

3. MODEL TRAINING

Our model is trained in two steps. In Step 1, word translation probability is estimated. Then, in Step 2, phrases are acquired, and both phrase translation probability and dependency relation probability are estimated. In both steps, parameter estimation is done with the EM algorithm.

3.1 Step 1

In Step 1, word translation probability in each direction is estimated independently. This is done in exactly the same way as in IBM Model 1.

3.2 Step 2

Both phrase translation probability and dependency relation probability are estimated, and one undirected alignment is found using the **e**-to-**f** and **f**-to-**e** probabilities simultaneously in this step. In contrast to Step 1, it is impossible to enumerate all the possible alignments. To find the best alignment, we first create an initial alignment based on only phrase translation probability, and then gradually revise it by considering the dependency relation probability with a hill-climbing algorithm.

The initial parameters of Step 2 are calculated as follows. The dependency relation probability is calculated using the final alignment result of Step 1, and we use the word translation probability estimated in Step 1 as the initial phrase translation probability.

3.2.1 Initial Alignment

We first create an initial alignment based on the phrase translation probability without considering the dependency relation probabilities.

For all the combinations of possible phrases (including NULL), phrase alignment probabilities are calculated (equation 7). Correspondences are adopted one by one in descending order of geometric mean of the phrase alignment probabilities. All the words should be aligned only once, that is, the correspondences are adopted exclusively. Generation of possible phrases is explained in Section 3.2.3.

3.2.2 Hill-climbing

To find better alignments, the initial alignment is gradually revised with a hill-climbing algorithm. We use four kinds of revising operations:

Swap: Focusing on any two correspondences, the partners are swapped.

Extend: Focusing on one correspondence, the source or target phrase is extended to include its neighboring (parent or child) NULL-aligned word.

Add: A new correspondence is added between a source word and a target word both of which are aligned to NULL.

Reject: A correspondence is rejected and the source and target phrase are aligned to NULL.

The alignment is revised only if the alignment probability gets increased. It is repeated until no operation can improve the alignment probability, and the final state is the best approximate alignment. As a by-product of hill-climbing, pseudo n-best alignment can be acquired. It is used in collecting fractional counts. Figure 3 shows an example of hill-climbing process.

3.2.3 Phrase Generation

If there is a word which is aligned to NULL in the best approximate alignment, a new possible phrase is generated by merging the word into a neighboring phrase which is not aligned to NULL. In the last alignment result in Figure 3, for example, “素子” is treated as being included in the correspondence between “受 光” and “photodetector” and the correspondence between “に” and “for”. As a result, we consider the correspondence between “受 光 素子” and “photodetector” and the correspondence between “素子 に” and “for” existing in parallel sentences. The new possible phrase is taken into consideration from the next iteration.

3.2.4 Model Estimation

Collecting all the alignment results, we estimate phrase alignment probabilities and dependency relation probabilities.

Similarly to the common EM algorithm, we estimate the parameters of phrase alignment probabilities as follows:

$$p(F_j|E_i) = \frac{C(F_j, E_i)}{\sum_k C(F_k, E_i)}$$
$$p(E_i|F_j) = \frac{C(E_i, F_j)}{\sum_k C(E_k, F_j)} \qquad (9)$$

where $C(F_j, E_i)$ is a frequency of F_j and E_i is aligned.

Using the estimated phrase alignment probabilities and dependency relation probabilities, we go back to the initial alignment described in Section 3.2.1 iteratively.

4. EXAMPLE DATABASE CONSTRUCTION

For the sake of using in EBMT system, translation example database is constructed. Once we detect the phrase correspondences in the parallel sentences, all of each correspondence and all the combinations of adjoining correspondences are registered into the database. This is because the basic idea of example-based machine translation is preferring to use larger translation example, which considers larger context and could provide an appropriate translation. In Figure 4, not only the single correspondence “生 検 $\leftrightarrow$ biopsy”, but the combinations of adjoining correspondence such as “生 検 で $\leftrightarrow$ in biopsy”, “の 生検 $\leftrightarrow$ biopsy of” and “の 生 検 で $\leftrightarrow$ in biopsy of” are registered.

However, there is a big problem concerning the function words. Function words do not often have exactly corresponding words in the opposite language, so they are often NULL-aligned. Japanese case markers such as “が (ga)” (subjective case), “を (wo)” (objective case) and so on, and English articles are typical words, which do not have corresponding parts. We also create examples which include such NULL-aligned words. In Figure 4, we create “を 認めた $\leftrightarrow$ were recognized”, “細胞 を 認めた $\leftrightarrow$ cells were recognized”, “は を 認めた $\leftrightarrow$ were recognized” and so on.

This example database construction step is quite similar to the phrase extraction step in Phrase-based SMT, but there are two big differences derived from using tree structures. One is that the created examples have linguistic meanings (we do not create meaningless examples such as “ は 大型 $\leftrightarrow$ large”). The other is that we can create examples which are discontiguous in word sequence. Hiero [4] also can do this with some restrictions on creating translation rules. In our case, the only one restriction is that the example should be contiguous in tree structure.

Note that the examples are stored in tree expressions, not in sequence of words.

5. TREE-BASED TRANSLATION

As a tree-based translation method, we adopt example-based machine translation system [15]. In this section, we briefly introduce the translation procedure in the EBMT system.

5.1 Retrieval of Translation Examples

The input sentence is converted into the dependency structure as in the parallel sentence alignment. Then, for each

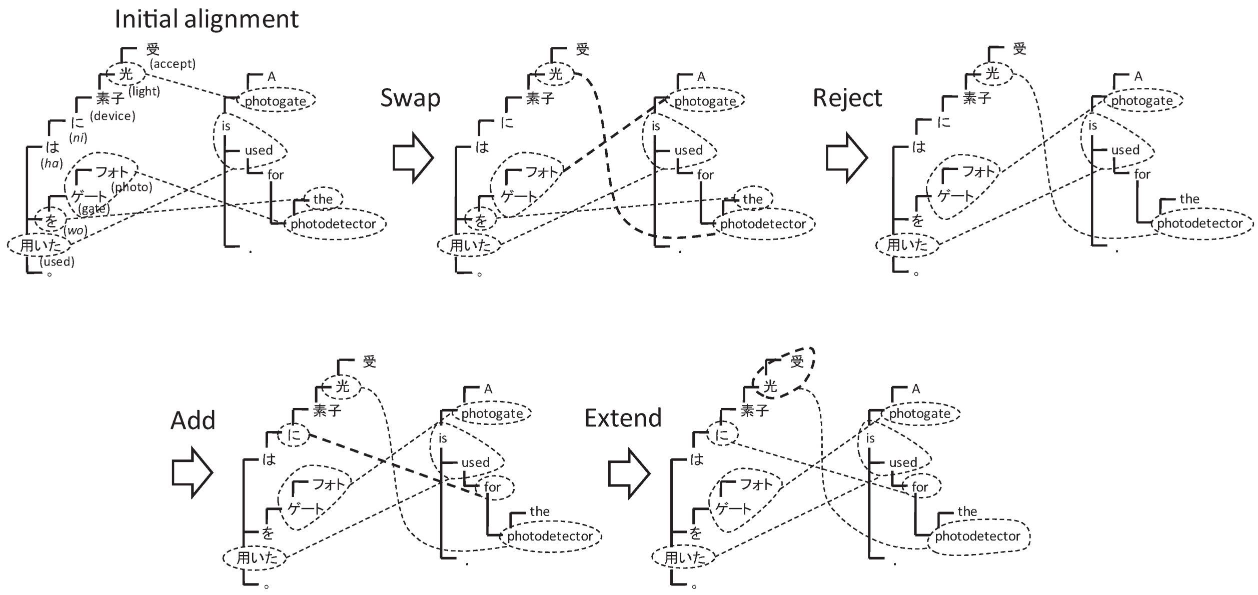

Figure 3: An example of hill-climbing.

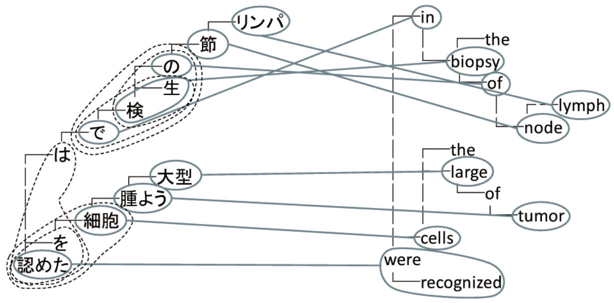

Figure 4: An example of creating translation example.

sub-tree, available translation examples are retrieved from the example database. Here the word "available" means that all the words in the focusing input sub-tree appear in the source tree of the example, and the dependency relations between the words are same. For a input node for which no translation example is found, the bilingual dictionary is looked up.

5.2 Selection of Translation Examples

All the retrieved examples are scored by the equation:

$$(S_{size} + 0.2 \sum S_{sim}) \cdot P_{tr} \tag{10}$$

where S_{size} is a size of the example, S_{sim} is a similarity of neighboring node, P_{tr} is a translation probability. The basic idea of EBMT is preferring to use larger translation example, which considers larger context and could provide an appropriate translation. According to this idea, our system also selects larger examples. Another advantage of EBMT is considering the similarities of the neighboring outside nodes. The similarities are ranging from 0.0 to 1.0 calculated based on a thesaurus.

Then, the best combination of examples which covers all the input tree without overlaps and makes the sum of the scores the highest is searched by beam-search algorithm.

5.3 Combination of Translation Examples

When combining the examples, in most cases, *bond node* is available outside of the example, to which the adjoining example is attached. Figure 1 is an example of combining translation examples. The combination process starts from the example used for the root node of the input tree (the first one in Figure 1). Then the example for the child node of the sub-tree covered by the initial example is combined (the second and third examples). When combining the second example to the first one, "細胞 ↔ cells" is used as bond node, and for the third example, "節 ↔ node" is used as bond node. The combination repeated until all the examples are combined into one target tree. Finally, output target sentence is generated from the tree structure.

Note that there are NULL-aligned nodes in the examples (the nodes which are not circled, such as 'は', 'を', '部 (*part*)' and articles in English). As explained in Section 4, such NULL-aligned nodes are included in the larger examples, thus we can translate a sentence with small number of examples.

6. EXPERIMENTAL RESULTS AND DISCUSSIONS

We conducted alignment and translation experiments. A JST[2] Japanese-English paper abstract corpus consisting of 1M parallel sentences was used for the model training. This corpus was constructed from a 2M Japanese-English paper abstract corpus by NICT[3] using the method of Uchiyama and Isahara [18]. Trainings were run on the original forms of words for both the proposed model and the models used

[2] http://www.jst.go.jp/
[3] http://www.nict.go.jp/

Table 2: Results of alignment and translation.

Alignment & Trans. method	Alignment Pre.	Alignment Rec.	Alignment F	Trans. BLEU
none & Moses	83.74	38.66	52.89	18.37
dep. tree only & Moses	89.29	40.77	55.98	18.32
phrase only & Moses	83.07	48.29	61.08	18.60
proposed & Moses	87.75	50.27	63.92	**19.13**
proposed+rule & Moses	87.83	**58.40**	**70.16**	18.28
proposed+rule & EBMT				18.80
intersection	**90.34**	34.28	49.71	16.93
grow-final-and	81.32	48.85	61.04	18.67
grow-diag-final-and	79.39	51.15	62.22	19.12

for comparison.

6.1 Alignment Experiments

As gold-standard data for alignment experiment, we used 475 sentence pairs which were annotated by hand. The annotations were only sure (S) alignments (there were no possible (P) alignments) [16]. The unit of evaluation was word-base for both Japanese and English. We used precision, recall, and F-measure as evaluation criteria.

For comparison of alignment quality, we used GIZA++ [16] which implements the prominent sequential word-base statistical alignment model of IBM Models. We conducted word alignment bidirectionally with its default parameters and merged them using three types of symmetrization heuristics [10].

In addition, to confirm the effectiveness of using dependency trees and phrases, we conducted alignment experiments on the following four conditions:

- Neither dependency trees nor phrases are used (referred to as 'none').
- Using dependency trees only.
- Using phrases only.
- Using both dependency trees and phrases (referred to as 'proposed').

For the conditions which do not use dependency trees, we used positional relations of a sentence as a sequence of words instead of dependency tree relations. The results are shown in Table 2.

The proposed model could achieve a higher F-measure by 1.7 points compared with the sequential model (proposed vs grow-diag-final-and). 'Intersection' achieved the best Precision, but its Recall is quite low. 'grow-diag-final-and' achieved the best Recall, but its Precision is lower than our best result where the Recall is almost same. Thus, we can say our result is better than the sequential word alignment models.

As discussed in Section 4, function words are often aligned to NULL or misaligned. To resolve the problem, we made some rules which modifies the final alignment result:

- English articles are merged into its parent node (usually noun).
- Correspondences between Japanese particles and English 'be' or 'have' are rejected.
- Japanese 'する' and 'れる (passive voice)' or English 'be' and 'have' are merged into its parent verb or adjective if they are NULL-aligned.

Note that all the rules are concerning only the function words which are easy to cause alignment errors. By adapting these rules, the F-measure is improved to 70.16 which is higher by 7.9 points compared with the sequential model (the row indicated with 'proposed+rule' in Table 2). Even with simple, small rules, the alignment accuracy can be much improved.

Sequential statistical methods, which regard a sentence as a sequence of words, work well for language pairs that are not too different in their language structure. Japanese and English have significantly different structures. One of the issues is that Japanese sentences have an SOV word order, but in English, the word order is SVO, so the dependency relations are often turned over. For language pairs such as Japanese and English, deeper sentence analysis using NLP resources is necessary and useful. Therefore, our method is suitable for such language pairs.

By comparing the results on the four conditions, we can see the following points:

1. Phrasal alignment improves the recall, but lowers the precision.
2. By using dependency trees, precision can be improved.
3. We can find a balance point by using both phrasal alignment and dependency trees.

The causes of alignment errors in our model can be summarized into categories. The biggest one is parsing errors. Since our model is highly dependent on the parsing result, the alignments would easily turn out wrong if the parsing result was incorrect.

Sometimes the hill-climbing algorithm could not revise the initial alignment. Most of these cases would happen when one word occurred several times on one side, but some of those occurrences were omitted on the other side. Let's suppose there are two identical words on the source side, but the target side has only one corresponding word. Initial alignment is created without considering the dependencies at all, so it cannot judge which source word should be aligned to the corresponding target word. In this case, the best alignment searching sometimes gets the local solution. This problem could be resolved by considering local dependencies for ambiguous words. Another solutions are to use random restarts or annealing.

6.2 Translation Experiments

As the Japanese to English translation test set, we used 500 paper abstract sentences which are parts of JST corpus. As a decoder for sequential models, we used state-of-the-art phrase-based SMT toolkit Moses [9] with its default options except for distortion limit (6 → -1 means infinite).

At first, to see the effectiveness of alignment quality improvement to the translation quality, we used Moses for both sequential alignment and tree-based alignment. The results are shown in the last column in Table 2. Although the alignment quality was improved compared with the conventional alignment methods, it did not widely contribute to the improvement of translation quality. This phenomenon has been already discussed in some works [14, 7], but we

Table 3: Results of manual evaluation of translation (evaluated 500 sentences in view of Adequacy).

	BLEU	Average score of Adequacy				
		5	≥ 4	≥ 3	≥ 2	≥ 1
RBMT (J-SERVER)	11.49	27 (5.4)	**216 (43.2)**	**420 (84.0)**	490 (98.0)	500 (100.0)
EBMT (KYOTO)	18.80	**39 (7.8)**	210 (42.0)	411 (82.2)	**494 (98.8)**	500 (100.0)
SMT (Moses)	**19.12**	22 (4.4)	144 (28.8)	373 (74.6)	488 (97.6)	500 (100.0)

thought this is because the tree structures is not used in decoding process. It is quite natural to use tree structure even in translation for structurally different language pairs.

As an another experiment, we conducted the tree-based translation with the result of proposed alignment model. In addition, commercial rule-based system[4] is used for comparison. The translation results are evaluated with not only BLEU score, but subjective evaluation. The subjective evaluation is done by three valuators. They give scores to each output sentence according to the Adequacy[5] of the translations, and the average of the three scores is defined as the score of the translation. The score ranges 1 (worse) to 5 (good), and the translations which get 3 or grater scores can be almost acceptable translations. The results are shown in Table 3. We showed the number of sentences in each cell followed by its percentage between the parentheses.

In view of BLEU score, the best system is SMT, and the EBMT result which uses tree structure is lower than SMT. The RBMT result is much worse than the other systems. If we see only this result, someone may say EBMT or using tree structure in decoding is not good. However, in view of subjective evaluation, the order of translation quality is completely inverted. Focusing on the ratio of sentences which get 3 or greater score, RBMT and EBMT exceeds 80% where SMT could not. Moreover, EBMT could output the most number of sentences which get the score 5. This result supports the assumption that combining smaller number of larger examples leads to better translations. From all these results, the following arguments can be raised:

- BLEU score is not always correlated with the genuine translation quality. Especially, it should not be used when comparing the fundamentally different systems.
- Using tree structures even in translation step is quite effective for structurally different language pairs.

For example, for the same input in Figure 1, output of Moses[6] is worse than that of EBMT.

6.3 NTCIR-8 Patent Translation Task

We used same EBMT system described above for NTCIR-8 Patent Translation Task. Table 4 shows the evaluation result of our KYOTO system compared to the baseline system "Moses" (BLEU scores with parentheses are re-evaluated results with up-to-date system). The detail of the task is described in [6]. Although the BLEU scores of our system are quite lower than Moses, the translations are not pessimistically worse because the MAP score of extrinsic evaluation is very competitive to Moses. The human evaluation results shown in the previous section can also support this notion.

Table 4: NTCIR-8 Evaluation Result.

	Intrinsic (BLEU)		Extrinsic		
	JE	EJ	BLEU	MAP	Recall@100
KYOTO	21.23	24.13	17.25	0.1909	0.5258
	(22.22)	(24.29)	-	-	-
Moses	29.08	35.27	24.01	0.1943	0.5701

7. CONCLUSION

In this paper, we have proposed a totally tree-based translation framework which is composed of statistical phrase alignment model based on dependency tree structures, and example-based translation method where the examples are expressed in tree structures. The alignment model incorporates the tree-based reordering model. Experimental results show that the word sequential model does not work well for linguistically different language pairs, and this can be resolved by using syntactic information. We have conducted the experiments only on Japanese-English corpora. To firmly support our claim that syntactic information is important, it is necessary to do more investigation on other language pairs.

Most frequent alignment errors are derived from parsing errors. Because our method depends heavily on structural information, parsing errors easily make the alignment accuracy worse. Although the parsing accuracy is high in general for both Japanese and English, it sometimes outputs incorrect dependency structures because technical or unknown words often appears in scientific papers. This problem could be resolved by introducing parsing probabilities into our model using parsing tools which can output n-best parsing with their parsing probabilities. This will not only improve the alignment accuracy, it will allow revision of the parsing result.

8. REFERENCES

[1] P. F. Brown, S. A. D. Pietra, V. J. D. Pietra, and R. L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Association for Computational Linguistics*, 19(2):263–312, 1993.

[2] E. Charniak and M. Johnson. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 173–180, 2005.

[3] C. Cherry and D. Lin. A probability model to improve word alignment. In *Proceedings of the 41st Annual*

[4] J-SERVER(http://www.j-server.com/index.shtml) by KODENSHA

[5] The criteria is "How much of the meaning expressed in the gold-standard translation is also expressed in the target translation?"

[6] Moses: "in the wrist joint and in the biopsy of the nerve infiltrating tumor lump was recognized"
Reference: "in the biopsy of the wrist joint division the tumor lump which permeated to the nerve was recognized"

Meeting of the Association of Computational Linguistics, pages 88–95, 2003.

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

[5] B. Cowan, I. Kučerová, and M. Collins. A discriminative model for tree-to-tree translation. In *Proceedings of the 2006 Conference on EMNLP*, pages 232–241, Sydney, Australia, July 2006. Association for Computational Linguistics.

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

[7] K. Ganchev, J. a. V. Graça, and B. Taskar. Better alignments = better translations? In *Proceedings of ACL-08: HLT*, pages 986–993, Columbus, Ohio, June 2008. Association for Computational Linguistics.

[8] D. Kawahara and S. Kurohashi. A fully-lexicalized probabilistic model for japanese syntactic and case structure analysis. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 176–183, New York City, USA, June 2006. Association for Computational Linguistics.

[9] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. Moses: Open source toolkit for statistical machine translation. In *Annual Meeting of the Association for Computational Linguistics (ACL), demonstration session*, 2007.

[10] P. Koehn, F. J. Och, and D. Marcu. Statistical phrase-based translation. In *HLT-NAACL 2003: Main Proceedings*, pages 127–133, 2003.

[11] S. Kurohashi, T. Nakamura, Y. Matsumoto, and M. Nagao. Improvements of Japanese morphological analyzer JUMAN. In *Proceedings of The International Workshop on Sharable Natural Language*, pages 22–28, 1994.

[12] P. Liang, B. Taskar, and D. Klein. Alignment by agreement. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 104–111, New York City, USA, June 2006. Association for Computational Linguistics.

[13] A. Menezes and C. Quirk. Syntactic models for structural word insertion and deletion during translation. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 734–743, Honolulu, Hawaii, October 2008. Association for Computational Linguistics.

[14] R. C. Moore, W.-t. Yih, and A. Bode. Improved discriminative bilingual word alignment. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 513–520, Sydney, Australia, July 2006. Association for Computational Linguistics.

[15] T. Nakazawa and S. Kurohashi. Kyoto-U: Syntactical EBMT system for NTCIR-7 patent translation task. In *Proceedings of the 7th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering, and Cross-Lingual Information Access*, 2008.

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

[17] C. Quirk, A. Menezes, and C. Cherry. Dependency treelet translation: Syntactically informed phrasal SMT. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 271–279, 2005.

[18] M. Utiyama and H. Isahara. A japanese-english patent parallel corpus. In *MT summit XI*, pages 475–482, 2007.

[19] D. Wu and P. Fung. Inversion transduction grammar constraints for mining parallel sentences from quasi-comparable corpora. In *Second International Joint Conference on Natural Language Processing (IJCNLP-2005)*, pages 257–268, 10 2005.