NTT Statistical Machine Translation for IWSLT 2006

Taro Watanabe, Jun Suzuki, Hajime Tsukada and Hideki Isozaki

NTT Communication Science Laboratories
2-4 Hikaridai, Seika-cho, Soraku-gun,
Kyoto, Japan 619-0237
{taro, jun, tsukada, isozaki}@cslab.kecl.ntt.co.jp

Abstract

We present the NTT translation system that is experimented for the evaluation campaign of "International Workshop on Spoken Language Translation (IWSLT)." The system consists of two primary components: a hierarchical phrase-based statistical machine translation system and a reranking system. The former is conceptualized as a synchronous-CFG in which phrases are hierarchically combined using non-terminals. The latter uses a modified voted perceptron approach with large number of features. Experiments showed that our hierarchical phrase-based model outperformed a conventional phrase-based model. In addition, our reranking algorithm further boosted the performance.

1. Introduction

This paper describes the NTT statistical machine translation system which is experimented in the evaluation campaign of the International Workshop on Spoken Language Translation (IWSLT) 2006.

Our system consists of two parts. A hierarchical phrase-based translation system that generates a large n-best list. The n-best list is further reranked using a variant of voted perceptron algorithm with additional feature functions.

This paper is organized as follows: first, we will review the framework of statistical machine translation followed by our hierarchical phrase-based approach. In Section 3, a n-best reranking algorithm will be presented. The reranking algorithm is based on a voted perceptron algorithm with a modified training procedure. Finally, we will discuss the detail of the task description and condition, followed by experimental results in Section 5.

2. Hierarchical Phrase-based Translation

2.1. Statistical Machine Translation

We use a log-linear approach [1] in which a foreign language sentence $f_1^J = f_1, f_2, ...f_J$ is translated into another language, i.e. English, $e_1^I = e_1, e_2, ..., e_I$ by seeking a maximum likelihood solution:

$$\hat{e}_1^I = \operatorname*{argmax}_{e_1^I} Pr(e_1^I|f_1^J) \quad (1)$$

$$= \operatorname*{argmax}_{e_1^I} \frac{\exp\left(\sum_{m=1}^M \lambda_m h_m(e_1^I, f_1^J)\right)}{\sum_{e'^{I'}_1} \exp\left(\sum_{m=1}^M \lambda_m h_m(e'^{I'}_1, f_1^J)\right)} \quad (2)$$

In this framework, the posterior probability $Pr(e_1^I|f_1^J)$ is directly maximized using a log-linear combination of feature functions $h_m(e_1^I, f_1^J)$, such as a ngram language model or a translation model. When decoding, the denominator is dropped since it depends only on f_1^J. Feature function scaling factors λ_m are optimized based on a maximum likelihood approach [1] or on a direct error minimization approach [2]. This modeling allows the integration of various feature functions depending on the scenario of how a translation is constituted.

2.2. Hierarchical Phrase-based Approach

In the phrase-based translation approach [3], the input foreign sentence is segmented into phrases, $\bar{f}_1^K$, mapped into corresponding English-side $\bar{e}_1^K$, then, reordered to form the output English sentence. The approach is able to capture phrase-wise local-reordering, or possibly neighboring phrase reordering, but does not account for long-distance reordering of phrases.

In the hierarchical phrase-based translation approach [4], translation is constituted by hierarchically combining phrases with the help of non-terminals embedded in phrases themselves. Each non-terminal represented in each phrase can capture reordering of phrases.

Based on the hierarchical phrase-based modeling, we adopted the left-to-right target generation method described in [5]. The method is able to generate translations efficiently, first, by simplifying the grammar so that the target-side takes a phrase-prefixed form, namely target normalized form. Our simplified grammar drastically reduces the number of rules extracted from a bilingual corpus empirically presented in [5]. Second, translation is generated in a left-to-right manner, similar to a phrase-based approach, using an Earley-style top-down parsing on the source-side. Coupled with the tar-

get normalized form, ngram language models are efficiently integrated during the search even with a higher order.

2.3. Simplified Grammar: Target Normalized Form

In [4], each production rule is restricted to a rank-2 or binarized form in which each rule contains at most two non-terminals. Under this restriction, enormous number of rules are still extracted from a bilingual corpus using the algorithm described in Section 2.4.

We introduce a target normalized form in which the target-side of the aligned right-hand side is restricted to a Greibach Normal Form like structure:

$$X \leftarrow \langle \gamma, \bar{b}\beta, \sim \rangle \tag{3}$$

where X is a non-terminal, γ is a source-side string of arbitrary terminals and/or non-terminals. $\bar{b}\beta$ is a corresponding target-side where $\bar{b}$ is a string of terminals, or a phrase, and β is a (possibly empty) string of non-terminals. The use of phrase $\bar{b}$ as a prefix keeps the strength of the phrase-base framework. A contiguous English side coupled with a (possibly) discontiguous foreign language side preserves a phrase-bounded local word reordering. At the same time, the target-normalized framework still combines phrases hierarchically in a restricted manner. For instance, it can capture "ne ... pas" and "not ..." translating from French into English, but cannot directly handle the other direction.

The target-normalized form can be regarded as a type of rule in which certain non-terminals are always instantiated with phrase translation pairs. Thus, we will be able to reduce the number of rules induced from a bilingual corpus, which, in turn, help reducing the decoding complexity. Note that we do not imply arbitrary synchronous-CFGs are transformed into the target normalized form. The form simply restricts the grammar extracted from a bilingual corpus explained in Section 2.4.

2.4. Training

The phrase extraction algorithm is based on those presented by [3]. First, many-to-many word alignments are induced by running a one-to-many word alignment model, such as GIZA++ [6], in both directions and by combining the results based on a heuristic [7]. Second, phrase translation pairs are extracted from the word aligned corpus [3]. This method exhaustively extracts phrase pairs (f_j^{j+m}, e_i^{i+n}) from a sentence pair (f_1^J, e_1^I) that do not violate the word alignment constraints a.

In the hierarchical phrase-based model, production rules are accumulated by computing "holes" for extracted contiguous phrases [4]:

1. A phrase pair $(\bar{f}, \bar{e})$ constitutes a rule:

$$X \rightarrow \langle \bar{f}, \bar{e} \rangle$$

2. A rule $X \rightarrow \langle \gamma, \alpha \rangle$ and a phrase pair $(\bar{f}, \bar{e})$ s.t. $\gamma = \gamma' \bar{f} \gamma''$ and $\alpha = \bar{e}' \bar{e} \beta$ constitutes a rule:

$$X \rightarrow \langle \gamma' \, X_{\boxed{k}} \, \gamma'', \bar{e}' \, X_{\boxed{k}} \, \beta \rangle$$

where the boxed indices indicate non-terminal alignment. One of the major differences to the algorithm presented in [4] is the restriction of the target normalized form in the last step.

2.5. Decoding by Top-down Parsing

Decoding is performed by parsing on the source-side and by combining the projected target-side. A conventional method of parsing is a CKY-based method in which ordering is governed by the span-size of the source words [4]. One of the problem is the high computational complexity when integrated with ngram language model of the target-side especially when the ngram's order is quite high [8]. The complexity lies on the possible "holes" in the target-side. One of the solution is to perform a binarization so that the target-side will not contain holes [9].

We applied an Earley-style top-down parsing approach described in [5] that is similar to [10]. The basic idea is to perform a top-down parsing in order so that the projected target-side is generated in a left-to-right manner. The search is guided with a push-down automaton which keeps track of the span-size of uncovered source word positions. Combined with the rest-cost estimation aggregated in a bottom-up way, our decoder efficiently searches for the most-likely translation. Our decoding algorithm can be regarded as an instance of Earley algorithm, but the predicted rule's "dot" is moved synchronized with the left-to-right ordering of the projected target-side, not the left-to-right ordering on the source-side.

The use of target normalized form further simplify the decodig procedure. Since the rule form does not allow any holes for the target-side, the integration with ngram language model is straightforward: the prefixed phrases are simply concatenated and intersected.

Our decoder is based on an in-house developed phrase-based decoder which uses a bit vector to represent uncovered foreign word positions for each hypothesis [14]. We basically replaced the bit vector structure to the stack structure: Almost no modification was required for the word graph structure and the beam search strategy implemented for a phrase-based modeling, since the target-side's prefixed phrases are simply concatenated. The use of a stack structure directly models a synchronous-CFG formalism realized as a push-down automation, while the bit vector implementation is conceptualized as a finite state transducer.

2.6. Feature Functions

Feature functions evaluated during the decoding procedure is summarized as count-based models, lexicon-based models, language model, reordering models and length-based models.

2.6.1. Count-based Models

Main feature functions $h_\phi(f_1^J|e_1^I, \mathcal{D})$ and $h_\phi(e_1^I|f_1^J, \mathcal{D})$ estimate the likelihood of two sentences f_1^J and e_1^I over a derivation tree $\mathcal{D}$. We assume that the production rules in $\mathcal{D}$ are independent of each other:

$$h_\phi(f_1^J|e_1^I, \mathcal{D}) = \log \prod_{\langle\gamma,\alpha\rangle\in\mathcal{D}} \phi(\gamma|\alpha) \quad (4)$$

$\phi(\gamma|\alpha)$ is estimated through the relative frequency on a given bilingual corpus.

$$\phi(\gamma|\alpha) = \frac{count(\gamma,\alpha)}{\sum_\gamma count(\gamma,\alpha)} \quad (5)$$

where $count(\cdot)$ represents the cooccurrence frequency of rules γ and α.

2.6.2. Lexicon-based Models

We define lexically weighted feature functions $h_w(f_1^J|e_1^I, \mathcal{D})$ and $h_w(e_1^I|f_1^J, \mathcal{D})$ by applying the independence assumption of production rules as in Equation 4.

$$h_w(f_1^J|e_1^I, \mathcal{D}) = \log \prod_{\langle\gamma,\alpha\rangle\in\mathcal{D}} p_w(\gamma|\alpha) \quad (6)$$

The lexical weight $p_w(\gamma|\alpha)$ is computed from word alignments a inside γ and α [3]:

$$p_w(\gamma|\alpha, a) = \prod_{i=1}^{|\alpha|} \frac{1}{|\{j|(i,j)\in a\}|} \sum_{\forall(i,j)\in a} t(\gamma_j|\alpha_i) \quad (7)$$

where $t(\cdot)$ is a lexicon model trained from the word alignment annotated bilingual corpus discussed in Section 2.4. The alignment a also includes non-terminal correspondence with $t(X_{\boxed{k}}|X_{\boxed{k}}) = 1$. If we observed multiple alignment instances for γ and α, then, we take the maximum of the weights.

$$p_w(\gamma|\alpha) = \max_a p_w(\gamma|\alpha, a) \quad (8)$$

A deletion model penalizes missed foreign words that do not constitute a translation:

$$h_{del}(e_1^I, f_1^J) = \sum_{j=1}^{J} \left[\max_{0\le i\le I} t(f_j|e_i) < \tau_{del}\right] \quad (9)$$

The deletion model simply counts the number of words whose lexicon model probability is lower than a threshold τ_{del}. Likewise, an insertion model is integrated that penalizes the inserted English words that do not account for any foreign words in an input:

$$h_{ins}(e_1^I, f_1^J) = \sum_{i=1}^{I} \left[\max_{0\le j\le J} t(e_i|f_j) < \tau_{ins}\right] \quad (10)$$

2.6.3. Language Model

We used mixed-cased 5-gram language model estimated with modified Kneser-Ney smoothing [11]:

$$h_{lm}(e_1^I) = \log \prod_i p_n(e_i|e_{i-4}e_{i-3}e_{i-2}e_{i-1}) \quad (11)$$

2.6.4. Reordering Models

In order to limit the reorderings, two feature functions are employed:

$$h_{height}(e_1^I, f_1^J, \mathcal{D}) = \sum_{\mathcal{D}_i\in back(\mathcal{D})} height(\mathcal{D}_i) \quad (12)$$

$$h_{width}(e_1^I, f_1^J, \mathcal{D}) = \sum_{\mathcal{D}_i\in back(\mathcal{D})} width(\mathcal{D}_i) \quad (13)$$

where $back(\mathcal{D})$ is a set of subtrees backtracked during the derivation of $\mathcal{D}$, and $height(\mathcal{D}_i)$ and $width(\mathcal{D}_i)$ refer to the height and width of subtree $\mathcal{D}_i$, respectively. The basic idea is similar to a skip-based penalty usually applied in a phrase-based model [3], but differ in that the penalties are associated with the tree structure.

2.6.5. Length-based Models

Two trivial length-based feature functions are included that count the number of target words and the number of production rules that constitute a translation.

$$h_l(e_1^I) = I \quad (14)$$

$$h_r(\mathcal{D}) = rule(\mathcal{D}) \quad (15)$$

3. Reranking by Voted Perceptron

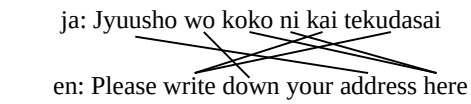
This section explains our discriminative reranking method which further improves the quality of baseline MT system.

Our reranking method basically follows the parse reranking method explained in [12]. We first generate a n-best list of candidate outputs (translations) from a baseline MT system, the hierarchical phrase-based translation described in Section 2. Then, a reranking model is trained by a ranking voted perceptron on a development set. Finally, in the process of decoding, we re-rank the n-best list of test data fed from the baseline MT using the trained reranking model. We adopted the above method, [12], with a BLEU-score-based weight update scheme. The reranking setting of MT is an ordinal regression procedure in each output pairs, similar to the parse reranking task, which can be generally reduced to a classification setting in each sample.

3.1. Features

The feature functions described in Section 2.6 are locally decidable and are devised mainly for the efficiency of a DP-based search procedure presented in Section 2.5. One advantage of discriminative reranking is that a wide variety of fea-

(a) IBM-model1 alignments



(b) Hierarchical Phrase Pairs (Rules)

< X_1, Please X_1 >
< X_1 kai tekudasai, write X_1 >
<X_1 wo X_2, down X_1 X_2>
<jyuusyo, your address>
<koko ni, here>

Figure 1: Example of IBM-model1 alignments and hierarchical phrase pairs

tures can be integrated, including sentence-wise global features.

We employed three feature types for our reranking:

1. SC: Scores of feature functions used in the baseline system with additional feature functions, namely, sentence-wise IBM-model1 alignment scores of both source-target and target-source alignments, and n-gram scores of target sentences where n equals 1 to 4,

2. AL: word pairs obtained by IBM-model1 alignment,

3. RU: hierarchical phrase pairs and shapes of rules.

Figure 1 shows an example of IBM model1 alignment. From the example in Figure 1, eight word pairs, [Jyuusho - address], [wo-down], [koko-here], [ni-here], [kai-write], [tekudasai-write], [NULL-Please], and [NULL-your] are extracted as AL features.

As shown in Figure 1, five hierarchical phrase pairs, namely, $\langle X_1$, Please $X_1\rangle$, $\langle X_1$ kai tekudasai, write $X_1\rangle$, $\langle X_1$ wo X_2, down X_1 $X_2\rangle$, $\langle$ Juusyo, your address $\rangle$, and $\langle$ koko ni, here $\rangle$ are obtained as RU features. In addition, we also handle the rule patterns for RU features. We abstract all consecutive words in one special symbol 'W'. For example, $\langle X_1$, Please $X_1\rangle$ and $\langle X_1$ kai tekudasai, write $X_1\rangle$ become $\langle X_1$, W $X_1\rangle$ and $\langle X_1$ W, W $X_1\rangle$, respectively. Therefore, $\langle X_1$, W $X_1\rangle$, $\langle X_1$ W, W $X_1\rangle$, $\langle X_1$ W X_2, W X_1 $X_2\rangle$, and $\langle W, W\rangle$ are also obtained as RU features. Note that AL and RU are sets of sparse features which generally amount to more than ten thousand features.

3.2. Reranking Algorithm

Algorithm in Figure 2 shows a ranking voted perceptron algorithm extended for BLEU-score based weight updates. Our extension updates the weight for candidate pairs x_i^m and x_j^m where $i < j$ if they are not ordered correctly in the current ranking X^m in terms of BLEU-score (line 6-12 in Figure 2).

Algorithm extended ranking voted perceptron: training

$D = \{D^1, ..., D^M\}$: Development set
$C^m = \{c_1^m, ..., c_N^m\}$: the original N-best list of D^m
c_n^m: n-th candidate in C^m

$X^m = \{x_1^m, ...x_N^m\}$: (reordered) N-best list of D^m
x_i^m: i-th candidate in the (reordered) N-best list X^m
$Ranking(W, C^m)$: returns N-best list of C^m reordered based on the score, $s_n^m =< W, \phi(c_n^m) >$
$\phi(x_n^m)$: the feature vector of x_n^m

W: weight vector
$V = \{V_1, ...V_T\}$: set of weight vectors
T: Number of pre-defined iteration

1: **For** $t = 1, \ldots, T$
2: **For** $m = 1, \ldots, M$;; for each sample in dev-set
3: $X^m \leftarrow Ranking(W, C^m)$
4: **For** $i = 1, \ldots, |X^m|$
5: **For** $j = i + 1, \ldots, |X^m|$
6: **If** $(BLEU(x_j^m) > BLEU(x_i^m)$
7: & $WER(x_j^m) <= WER(x_i^m))$
8: $s = (BLEU(x_j^m) - BLEU(x_i^m))$
9: $W = W + s * (\phi(x_j^m) - \phi(x_i^m))$
10: **End_If**
11: **End_For**
12: **End_For**
13: $V_t = W$
14: **End_For**
15: **End_For**
16: **Return** V

Figure 2: Reranking Algorithm for training

The decoding scheme for our voting reranking model is presented in Figure 3.

3.3. Correct Ranking Score

Our reranking algorithm involves online supervised learning as shown in Figure 2. Under this situation, calculating BLEU-score is rather costly since it requires a document-wise computation, not a sample-wise computation. This means that we have to re-calculate BLEU-score for every iteration inside the second for-loop (line 6 in Figure 2). To reduce the calculation cost, we employed approximated BLEU-score for a ranking score.

Let O_n^m be an output set, where $O_n^m = \{c_1^1, \cdots, c_1^{m-1}, c_n^m, c_1^{m+1}, \cdots, c_1^M\}$. O_n^m contains all 1-best outputs of the baseline MT system except a sample D^m, whose output is the n-th candidates from the baseline MT system. We calculate BLEU-score using output set O_n^m as the approximated BLEU-score for c_n^m. As a result, approximated BLEU-score is independent for the first loop; We are only required to calculate BLEU-score once for

Algorithm decoding algorithm for reranking model

$G = \{G^1, \ldots, G^K\}$: Test set
$E^m = \{e_1^m, \ldots, e_N^m\}$: the original N-best list of G^k
$I = \{I_1, \ldots, I_N\}$: votes for ith candidate

1: **For** $k = 1, \ldots, K$
2: $I = \mathbf{0}$
3: **For** $t = 1, \ldots, T$
4: $i = \arg\max_i < V^t, \phi(e_i^k) >$
5: $I_i = I_i + 1$
6: **End_For**
7: **Output** e_i^k where $i = \arg\max_i I_i$
8: **End_For**

Figure 3: Decoding algorithm for reranking model

all candidates in development set during pre-processing. Alternatively, we may use a segment-wise BLEU score for an approximation. However, our internal studies indicated that the segment-wise BLEU-score resulted in a wrong objective as an approximation for document BLEU score.

Additionally, we used word error rate (WER) as a weak constraint of updating weight to reduce over-fitting BLEU score on development set (line 7 in Figure 2).

4. Tasks

The experiments were carried out on the Basic Travel Expression Corpus (BTEC) task [13]. BTEC is a multilingual corpus in traveling domain which was collected from phrase books for tourists. In the IWSLT 2006 open data track, the subsets of BTEC consists of training set and three development sets (Dev1 through Dev3) indicated in Table 1. Another development set (Dev4) and the final test set were provided in this track[1]. The translation pairs set up for the task are: Arabic-to-English, Italian-to-English, Japanese-to-English and Chinese-to-English.

The task description for the IWSLT 2006 evaluation campaign can be summarized as follows:

1. Spoken language, instead of written texts, are used as inputs to our translation system.

2. The last development set (Dev4) and test set are not part of BTEC, but collected from "simple conversations in travel domain"[2].

3. Since ASR output is used for translation source, the source language side is lower-cased and without punctuations.

4. However, translation is evaluated case-sensitive with punctuation mark.

[1]We used 1-best ASR output for those sets.

[2]The details will be available from `http://www.slc.atr.jp/IWSLT2006/archives/2005/11/evaluation_camp.html`

Although the spoken language translation specific problem, i.e. illformed input, is still unresolved, we mainly investigated the last two task-specific problems.

4.1. Preprocessing

Since Dev4 and the final test set were drawn from a differently created corpus, we used the whole corpora from BTEC (Train through Dev3) as a training set, and the parameter tuning was performed on Dev4.

All the corpora were preprocessed according to the standard defined within the IWSLT 2006 evaluation campaign: English-side of the parallel corpora were simply punctuation isolated, but the casing were preserved. The punctuations were removed from the source-sides, and lower-cased. Numeral characters were isolated for Japanese and Chinese.

4.2. Rule Extraction from Multiple Alignments

After a simple in-house experiment, we found that the above simple approach resulted in many errors in word alignments. This is due to the punctuation mismatch between the source-side and English-side. Therefore, we follow the idea of [14] when extracting rules from a word alignment annotated corpus.

First, three kinds of corpora are prepared to differentiate punctuation removal strategies: *nopunct-with-nopunct* where punctuation marks are removed in both languages, *punct-with-punct* where punctuations are kept, and *nopunct-with-punct* in which punctuations are removed in the source-side but remained for the target-side. Those three corpora are merged into one large corpus.

Second, the merged corpus is preprocessed with three different strategies: lower-cased, stemmed and prefix4 where only the prefix of 4-letter are preserved. Word alignment is obtained for each differently preprocessed corpus by running GIZA++ in both directions, and by refining word alignment with a heuristic.

Third, from three distinctly preprocessed corpora, rules are extracted using the algorithm presented in 2.4. In this step, preprocessed corpora are recovered into their original form. When recovered, punctuation marks on the source-side were removed together with corresponding word alignments.

The idea is to induce better word alignments by considering non-punctuation corpus, together with punctuation preserved corpus.

5. Results

5.1. Official Results

Our official results for the IWSLT 2006 Open Data Track are summarized in Table 2. The primary system submitted for the track is the combination of the hierarchical phrase-based translation and the reranking algorithm presented in Section 3 with 'SC' and 'AL' features, excluding 'RU' features. Translation results on the spoken input are slightly lower

Table 1: Corpus statistics for IWSLT 2006 evaluation campaign (open track)

		Arabic	English	Italian	English	Japanese	English	Chinese	English
Train	sentences	19,972		19,972		39,953		39,953	
	words	130,643	184,789	144,281	184,789	353,630	369,540	303,801	369,540
Dev1	sentences	506	8,096	506	8,096	506	8,096	506	8,096
	words	2,555	66,130	2,871	66,130	3,586	66,130	2,910	66,130
Dev2	sentences	500	8,000	500	8,000	500	8,000	500	8,000
	words	2,659	65,568	2,759	65,568	3,588	65,568	2,996	65,568
Dev3	sentences	506	8,096	506	8,096	506	8,096	506	8,096
	words	2,566	66,686	2,846	66,686	3,632	66,686	3,292	66,686
	vocabulary	17,864	9,308	10,864	9,308	12,293	11,690	11,099	11,690
Dev4	sentences	489	3,423	489	3,423	489	3,423	489	3,423
Test	sentences	500		500		500		500	

spoken for read speech/*text* for correct recognition results.

Table 2: Official results for IWSLT 2006 open data track

		BLEU	NIST	METEOR	mWER	mPER
Arabic-English	spoken	20.71 (5th)	4.84	43.97	64.67	56.65
	text	22.65 (5th)	5.33	47.76	62.79	54.15
Italian-English	spoken	27.69 (7th)	6.70	56.07	57.00	48.13
	text	34.49 (5th)	7.83	64.31	50.79	41.57
Japanese-English	spoken	19.84 (2nd)	5.48	45.00	71.08	55.12
	text	22.03 (2nd)	5.91	48.77	69.02	52.17
Chinese-English	spontaneous	15.59 (6th)	4.18	39.46	70.20	59.72
	spoken	18.34 (5th)	4.53	42.15	68.44	57.71
	text	21.35 (5th)	5.13	47.43	65.47	53.70

when compared against correct recognition inputs. The system performed around average for most of the language pairs, but performed quite well for the Japanese-English task. Since Japanese-to-English translation requires longer reordering of phrases, our hierarchically combined phrases can capture those reorderings.

5.2. Results on Hierarchical Phrase-based Translation

We first compared our baseline hierarchical phrase-based translation against an in-house developed phrase-based translation that performed quite well for the shared task of "Workshop on Statistical Machine Translation" [14]. Table 3 shows the number of phrases and rules extracted from each task. The grammar size for our hierarchical phrase-based system is almost twice as large as the size of the phrase table for our phrase-based system. The phrase-based system employs a lexicalized reordering model to capture phrase-wise reordering [15]. For the hierarchical phrase-based system, span-size for each non-terminal was constrained to 7 for all tasks. Window-size constraints were set to 7 in the phrase-based system. As indicated in Table 4, our hierarchical phrase-based system outperforms the phrase-based system in all tasks.

5.3. Results on Reranking

Table 5 shows the reranking results for IWSLT2006. The rows of '1-best' in Table 5 show the performance of our baseline MT system, hierarchical phrase-based system (contrast-1 system). Then, the rows of 'SC' display the performance using only SC features for reranking (primary system). Finally, the rows of 'ALL' show the reranking performance with three feature types, SC, AL and RU (contrast-2 results). All results are obtained by $n = 1000$ of n-best list size.

As Table 5 indicates, we obtained large improvements over all tasks, except for the Japanese-English task. The Japanese-English task already achieved good performance even with our baseline MT system. Since we did not employ tree structure-based features in reranking, the improvement were subtle.

Note that our primal system for IWSLT2006 submission is reranking with only SC features because of the limited time of the evaluation schedule. We would like to emphasize that these results indicate that sparse features, such as AL and RU, can further improve the overall translation quality.

6. Conclusions

We experimented with the NTT statistical machine translation system for the evaluation campaign of IWSLT 2006

Table 3: Phrase/rule size for each task

	Arabic-English	Italian-English	Japanese-English	Chinese-English
Phrase	35,006,211	17,836,633	92,593,962	43,718,878
Rule	60,232,573	40,522,884	168,068,599	83,268,205

Table 4: Comparison of the phrase-based and hierarchical phrase-based system (BLEU [%])

	Arabic-English		Italian-English		Japanese-English		Chinese-English		
	spoken	text	spoken	text	spoken	text	spontaneous	spoken	text
Phrase-based	19.37	22.63	25.33	31.62	18.33	20.77	13.87	15.88	19.24
Hierarchical phrase	20.67	22.96	27.71	34.95	19.83	22.62	16.21	18.48	21.36

Open Data Track. Our system consisted of a baseline MT system of hierarchical phrase-based translation with a left-to-right target generation decoding method. The n-best list generated from our baseline system is reranked by a voted perceptron algorithm with a sparse feature functions trained with an approximated BLEU criterion. The experiments indicated that our hierarchical phrase-based system is far better than a conventional phrase-based system. In addition, the reranking algorithm can successfully improve the performance by incorporating diverse feature functions. As our future work, we are in the process of investigating more feature functions, especially useful for our hierarchical modeling.

7. Acknowledgements

We would like to thank the organizer of the IWSLT 2006 for their efforts to coordinate the campaign. We would also like to thank our colleagues for their useful advice in supporting our experiments.

8. References

[1] F. J. Och and H. Ney, "Discriminative training and maximum entropy models for statistical machine translation," in *Proc. of ACL 2002*, 2002, pp. 295–302.

[2] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proc. of ACL 2003*, 2003, pp. 160–167. [Online]. Available: http://www.aclweb.org/anthology/P03-1021.pdf

[3] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *Proc. of NAACL 2003*, Edmonton, Canada, 2003, pp. 48–54.

[4] D. Chiang, "A hierarchical phrase-based model for statistical machine translation," in *Proc. of ACL 2005*, Ann Arbor, Michigan, June 2005, pp. 263–270. [Online]. Available: http://www.aclweb.org/anthology/P/P05/P05-1033

[5] T. Watanabe, H. Tsukada, and H. Isozaki, "Left-to-right target generation for hierarchical phrase-based translation," in *Proc. of COLING-ACL 2006*, Sydney, Australia, July 2006, pp. 777–784.

[6] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, March 2003.

[7] ——, "The alignment template approach to statistical machine translation," *Comput. Linguist.*, vol. 30, no. 4, pp. 417–449, 2004.

[8] L. Huang, H. Zhang, and D. Gildea, "Machine translation as lexicalized parsing with hooks," in *Proceedings of the Ninth International Workshop on Parsing Technology*, Vancouver, British Columbia, October 2005, pp. 65–73. [Online]. Available: http://www.aclweb.org/anthology/W/W05/W05-1507

[9] H. Zhang, L. Huang, D. Gildea, and K. Knight, "Synchronous binarization for machine translation," in *Proc. of HLT-NAACL 2006*. New York City, USA: Association for Computational Linguistics, June 2006, pp. 256–263. [Online]. Available: http://www.aclweb.org/anthology/N/N06/N06-1033

[10] A. Zollmann and A. Venugopal, "Syntax augmented machine translation via chart parsing," in *Proceedings on the Workshop on Statistical Machine Translation*. New York City: Association for Computational Linguistics, June 2006, pp. 138–141.

[11] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," in *Proc. of ACL 1996*. Morristown, NJ, USA: Association for Computational Linguistics, 1996, pp. 310–318.

[12] M. Collins and N. Duffy, "New Ranking Algorithms for Parsing and Tagging: Kernels over Discrete Structures, and the Voted Perceptron," in *Proc. of ACL'2002*, 2002, pp. 263–270.

[13] T. Takezawa, E. Sumita, F. Sugaya, H. Yamamoto, and S. Yamamoto, "Toward a broad-coverage bilingual corpus for speech translation of travel conversations in the real world," in *Proc. of LREC 2002*, Las Palmas, Canary Islands, Spain, May 2002, pp. 147–152.

Table 5: Impact of reranking for IWSLT 2006 open data track (ALL: late submission)

			BLEU	NIST	METEOR	mWER	mPER
Arabic-English	spoken	1-best	20.57 (5th)	4.94	44.11	64.96	56.84
		SC	20.71 (5th)	4.84	43.97	64.67	56.65
		ALL	22.62 (2nd)	5.04	45.14	62.68	55.11
	text	1-best	22.41 (5th)	5.42	48.27	62.76	53.82
		SC	22.65 (5th)	5.33	47.76	62.79	54.15
		ALL	24.46 (2nd)	5.64	49.42	60.69	54.73
Italian-English	spoken	1-best	27.67 (7th)	6.67	55.91	56.96	48.37
		SC	27.69 (7th)	6.70	56.07	57.00	48.13
		ALL	30.01 (1st)	6.96	57.24	54.95	46.55
	text	1-best	34.86 (5th)	7.87	64.59	50.21	41.34
		SC	34.49 (5th)	7.83	64.31	50.79	41.57
		ALL	37.51 (2nd)	8.17	66.15	47.79	39.36
Japanese-English	spoken	1-best	20.00 (2nd)	5.49	44.63	70.57	55.14
		SC	19.84 (2nd)	5.48	45.00	71.08	55.12
		ALL	20.47 (2nd)	5.59	45.33	70.00	53.62
	text	1-best	22.25 (2nd)	5.89	48.85	67.90	51.72
		SC	22.03 (2nd)	5.91	48.77	69.02	52.17
		ALL	22.56 (2nd)	5.94	48.83	67.63	51.03
Chinese-English	spontaneous	1-best	15.98 (5th)	4.26	39.96	70.39	59.36
		SC	15.59 (6th)	4.18	39.46	70.20	59.72
		ALL	16.86 (3rd)	4.50	40.21	70.16	58.78
	spoken	1-best	18.57 (5th)	4.58	41.68	69.08	57.77
		SC	18.34 (5th)	4.53	42.15	68.44	57.71
		ALL	18.63 (3rd)	4.75	43.02	67.85	56.45
	text	1-best	21.37 (5th)	5.19	46.80	65.50	54.01
		SC	21.35 (5th)	5.13	47.43	65.47	53.70
		ALL	22.35 (3th)	5.48	48.38	64.73	52.23

[14] T. Watanabe, H. Tsukada, and H. Isozaki, "Ntt system description for the wmt2006 shared task," in *Proceedings on the Workshop on Statistical Machine Translation*. New York City: Association for Computational Linguistics, June 2006, pp. 122–125.

[15] P. Koehn, A. Axelrod, A. B. Mayne, C. Callison-Burch, M. Osborne, and D. Talbot, "Edinburgh system description for the 2005 IWSLT speech translation evaluation," in *Proc. of IWSLT 2005*, Pittsburgh, PA, USA, 2005.