TALP Phrase-Based System and TALP System Combination for IWSLT 2006

Marta R. Costa-jussà, Josep M. Crego, Adrià de Gispert,
Patrik Lambert, Maxim Khalilov,
José A.R. Fonollosa, José B. Mariño and Rafael Banchs

TALP Research Center
Universitat Politècnica de Catalunya, Barcelona
{mruiz|jmcrego|agispert|lambert|khalilov|adrian|canton|rbanchs}@gps.tsc.upc.edu

Abstract

This paper describes the TALP phrase-based statistical machine translation system, enriched with the statistical machine reordering technique. We also report the combination of this system and the TALP-tuple, the n-gram-based statistical machine translation system. We report the results for all the tasks (Chinese, Arabic, Italian and Japanese to English) in the framework of the third evaluation campaign of the International Workshop on Spoken Language Translation.

1. Introduction

This paper describes the TALP-phrase system for the IWSLT 2006, which is an enhanced version of the system reported in the 2005 evaluation [1]. The main difference is the integration of a new reordering technique called statistical machine reordering, which was presented in [2] in a different framework.

Additionally, we report the results of combining the outputs of the two statistical machine translation TALP systems: phrase-based and n-gram-based. The latter of the two also participated in the 2005 evaluation and is described in [3].

Statistical machine translation systems are now usually modelled through a log-linear maximum entropy framework.

$$\tilde{e} = \underset{e}{argmax} \left\{ \sum_{m=1}^{M} \lambda_m h_m(e, f) \right\} \quad (1)$$

The feature functions, h_m, and weights, λ_i, are typically optimized to maximize the scoring function [4].

Two basic issues differentiate the n-gram-based system from the phrase-based system: the bilingual units are extracted from a monotonic segmentation of the training data; the unit probabilities are based on a standard back-off language model rather than directly on relative frequencies.

In both systems, the introduction of reordering capabilities is crucial for certain language pairs.

This paper is organized as follows. Section 2 describes the TALP-phrase system, with particular emphasis on a new reordering technique: the statistical machine reordering approach. In Section 3, we combine the TALP-phrase and the TALP-tuple. Finally, in Section 4, we report the results obtained for all the tasks of the evaluation, which include the translations from Chinese, Arabic, Italian and Japanese to English.

2. Description of the TALP-phrase System

2.1. Phrase-based Model

The basic idea of phrase-based translation is to segment the given source sentence into units (here called phrases), then translate each phrase and finally compose the target sentence from these phrase translations.

Given a sentence pair and a corresponding word alignment, phrases are extracted following the criterion in [5]. A phrase (or bilingual phrase) is any pair of m source words and n target words that satisfies two basic constraints (1) words are consecutive along both sides of the bilingual phrase and (2) no word on either side of the phrase is aligned to a word outside the phrase.

2.2. Feature functions

The baseline phrase-based system implements a log-linear combination of four feature functions, which are described as follows.

- The **translation** model is estimated with relative frequencies. Given the collected phrase pairs, we estimate the phrase translation probability distribution by relative frequency in both directions.
- The **target language** model consists of an n-gram model, in which the probability of a translation hypothesis is approximated by the product of word n-gram probabilities.
- The **forward and backwards lexicon** models. These provide lexicon translation probabilities for each phrase based on the word IBM Model 1 probabilities. For computing the forward lexicon model, IBM Model 1 probabilities from GIZA++ source-to-target alignments are used. In the case of the backwards lexicon model, target-to-source alignments are used.

- The **word bonus** model introduces a sentence length bonus in order to compensate the system preference for short output sentences.
- The **phrase bonus** model introduces a constant bonus per produced phrase.

All of these models are combined in the decoder. Additionally, the decoder allows for a non-monotonic search with the following distortion model.

- A word distance-based **distortion model**.

$$P(t_1^K) = exp(-\sum_{k=1}^{K} d_k)$$

where d_k is the distance between the first word of the k^{th} phrase (unit), and the last word +1 of the $(k-1)^{th}$ phrase. Distance is measured in words referring to the units source side.

To reduce the computational cost we place limits on the search using two parameters: the distortion limit (the maximum distance measured in words that a phrase may be reordered, m) and the reordering limit (the maximum number of reordering jumps in a sentence, j). This feature is independent of the reordering approach presented in this paper, so the two can be used simultaneously.

In order to combine the models in the decoder suitably, an optimization tool is needed to compute log-linear weights for each model.

2.3. Statistical Machine Reordering

The aim of SMR consists of using an SMT system to deal with reordering problems. SMR is a first-pass translation performed on the source corpus, converting it into an intermediate representation, in which source-language words are presented in an order that more closely matches that of the target language (see Figure 1). Therefore, the SMR system can be seen as an SMT system which translates from an original source language (*S*) to a reordered source language (*S'*), given a target language (*T*). In this case, the translation task changes from *S2T* to *S'2T* (see Figure 2). The main difference between the two tasks is that the latter allows for (1) monotonized word alignment and (2) higher quality monotonized translation.

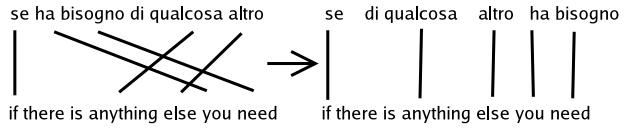

Figure 1: *Monotonization of the source language.*

For the reordering translation, we used an n-gram-based SMT system (and considered only the translation model, which is detailed below). Additionally, as for the input to the SMR system, in order to be able to infer new reorderings we use word classes instead of words themselves.

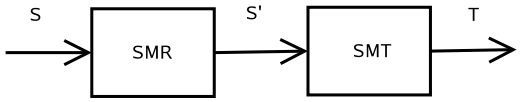

Figure 2: *SMR is applied before SMT.*

2.3.1. Description

Figure 3 shows the SMR block diagram. The input is the initial source sentence (*S*) and the output is the reordered source sentence (*S'*). There are three blocks in the SMR: (1) class replacing ; (2) the decoder, which requires the translation model; and (3) the block which reorders the original sentence using the indexes given by the decoder. The following example specifies the input and output of each block in the SMR.

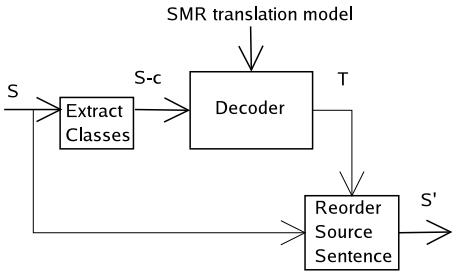

Figure 3: *SMR block diagram.*

1. Source sentence (*S*):

   se ha bisogno di qualcosa altro

2. Source sentence classes (*S-c*):

   49 137 160 189 176 75

3. Decoder output (translation, T):

   49 # 0 | 137 160 189 176 75 # 3 4 0 1 2

   where | indicates the segmentation into bilingual units and # indicates the limit between the source and the target part of each bilingual unit. The source part is composed of word classes and the target part is composed of the new positions of the source word classes, starting at 0.

4. SMR output (*S'*). The reordering information inside each translation unit of the decoder output (T) is applied to the original source sentence (S):

   se di qualcosa altro ha bisogno

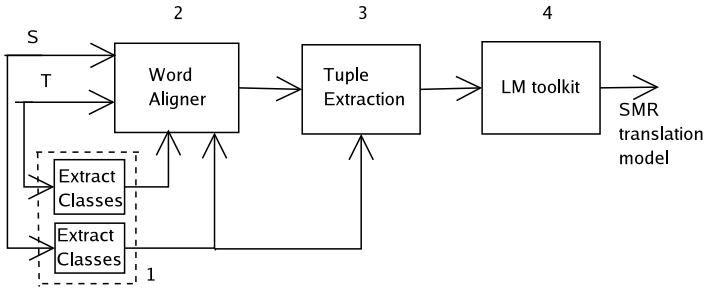

Figure 4: *Block diagram of the training process of the SMR translation model.*

2.3.2. Training

As explained, for the SMR system, we used an n-gram-based SMT system (and considered only the translation model). Figure 4

shows the block diagram of the training process of the SMR translation model, which is a bilingual n-gram-based model. The training process uses the training source and target corpora and consists of the following steps:

1. Determine source and target word classes.

2. Align parallel training sentences at the word level in both translation directions. Compute the union of the two alignments to obtain a symmetrized many-to-many word alignment.

3. Extract reordering tuples, see Figure 5.

(a) bilingual S2T tuple

ha bisogno di qualcosa altro # anything else you need # 0-2 1-3 2-0 3-0 3-1 4-1
(source) (target) (word alignment)
(wrd_src-wrd_trg)

(b) many-to-many word alignment → many-to-one word alignment
P_ibm (qualcosa, anything)>P_ibm(qualcosa, else)

ha bisogno di qualcosa altro # anything you need # 0-2 1-3 2-0 3-0 4-1

(c) bilingual S2S' tuple

ha bisogno di qualcosa altro # 3 4 0 1 2
(source) (new order)

(e) classes substitution

C137 C160 C189 C176 C75 # 1 2 0

Figure 5: *Example of the extraction of reordering tuples (step 3).*

   (a) From union word alignment, extract bilingual *S2T* tuples (i.e. source and target fragments) while maintaining the alignment inside the tuple. As an example of a bilingual *S2T* tuple consider: *ha bisogno di qualcosa altro # anything else you need # 0-2 1-3 2-0 3-0 3-1 4-1*, as shown in Figure 5, where the different fields are separated by # and correspond to: (1) the target fragment; (2) the source fragment; and (3) the word alignment (in this case, the fields that respectively correspond to a target and source word are separated by $-$).

   (b) Modify the many-to-many word alignment from each tuple to many-to-one. If one source word is aligned to two or more target words, the most probable link given IBM Model 1 is chosen, while the other are omitted (i.e. the number of source words is the same before and after the reordering translation). In the above example, the tuple would be changed to: *ha bisogno di qualcosa altro # anything you need # 0-2 1-3 2-0 3-0 4-1*, as P_{ibm1}(qualcosa, anything) is higher than P_{ibm1}(qualcosa, else).

   (c) From bilingual *S2T* tuples (with many-to-one inside alignment), extract bilingual *S2S'* tuples (i.e. the source fragment and its reordering). As in the example: *ha bisogno di qualcosa altro # 3 4 0 1 2*, where the first field is the source fragment, and the second is the reordering of these source words.

   (d) Eliminate tuples whose source fragment consists of the NULL word.

   (e) Replace the words of each tuple source fragment with the classes determined in Step 1.

4. Compute the bilingual language model of the bilingual *S2S'* tuple sequence composed of the source fragment (in classes) and its reorder.

Once the translation model is built, the original source corpus *S* is translated into the reordered source corpus *S'* with the SMR system, see Figure 3. The reordered training source corpus and the original training target corpus are used to train the SMT system (as explained earlier in this same section). Finally, with this system, the reordered test source corpus is translated.

3. Phrase-based and N-gram-based Combination

The aim of the system combination is to select the better translation given the 1-best output of each system: phrase-based and n-gram-based.

We perform a log-linear combination, which is computed using the following models:

- IBM Model 1 for the sentence in the source to target direction.
- IBM Model 1 for the sentence in the target to source direction.
- Target language models: 2gram, 3gram and 5gram.
- Word bonus.

The weights of each model are optimized with the simplex algorithm [6].

4. Evaluation Framework

4.1. Tools

- Word alignments were computed using the GIZA++ tool [7]. During word alignment, we used 50 classes per language. We aligned both translation directions and combined the two alignments with the union operation.
- Word classes (which were used to help the aligner and to perform the SMR process) were determined using "mkcls", a tool freely-available with GIZA++.
- The language model was estimated using the SRILM toolkit [8].
- The decoder was MARIE [9].
- The optimization tool used for computing log-linear weights was based on the simplex method [6]. Following the consensus strategy proposed in [10], the objective function was set to $100 \cdot BLEU + 4 \cdot NIST$.

4.2. Data

Experiments were carried out for all tasks of the IWSLT06 evaluation (Zh2En, Jp2En, Ar2En and It2En) using the BTEC Corpus provided for the open data track[1].

4.3. Description of tasks

For internal development work, true case and punctuation marks were removed from all parallel corpora (train, develop, test and references), thereby optimizing according to the 'additional' scoring scheme as defined in IWSLT 2006. For the final evaluation test set, punctuation marks and true case were included by using the SRILM 'disambig' tool as suggested by IWSLT organizers.

Given the availability of up to four development sets for all language pairs, our strategy was to use development 4 as the internal development set (**dev4**), while randomly selecting 500 sentences from developments 1, 2 and 3 (around 160 sentences from each) to build an internal test set (**dev123**). Finally, the approximately 1k remaining development sentences were included in the training corpus by selecting the first English manual reference.

[1] www.slt.atr.jp/IWSLT2006

		sent.	wrds	voc.	slen.	refs.
train	ar	24.0k	183k	10.5k	7.6	1
	en		166k	7.3k	6.9	
dev4	ar	489	5,889	1,237	12	7
dev123	ar	500	3,329	1,037	6.7	16
test	ar	500	6,570	1,480	13.1	7
ASRtest	ar	500	6,659	1,532	13.3	7

Table 1: *Arabic→English corpus statistics.*

		sent.	wrds	voc.	slen.	refs.
train	zh	46.9k	314k	9.7k	6.7	1
	en		326k	9.6k	7.0	
dev4	zh	489	5,478	1,096	11.2	7
dev123	zh	500	3,005	909	6.0	16
test	zh	500	5,846	1,292	11.7	7
ASRtest	zh	500	5,825	1,311	11.6	7

Table 2: *Chinese→English corpus statistics.*

Corpus statistics for all language pairs can be found in Tables 1, 2, 3 and 4, respectively, where number of sentences, running words, vocabulary, sentence length and human references are shown.

		sent.	wrds	voc.	slen.	refs.
train	it	24.6k	155k	10.2k	6.3	1
	en		166k	7.3k	6.8	
dev4	it	489	5,193	1,192	10.6	7
dev123	it	500	2,807	969	5.6	16
test	it	500	5,978	1,429	12.0	7
ASRtest	it	500	5,767	1,517	11.5	7

Table 3: *Italian→English corpus statistics.*

4.4. Language-dependent preprocessing

For all language pairs, training sentences were split by using full stops on both sides of the bilingual text (when the number of stops was equal), increasing the number of sentences and reducing their length. Specific preprocessing for each language is detailed in the respective section below.

4.4.1. English

English preprocessing includes part-of-speech tagging using the freely-available *TnT* tagger [11] and lemmatization using *wnmorph*, included in the WordNet package [12].

		sent.	wrds	voc.	slen.	refs.
train	jp	45.2k	390k	10.6k	8.6	1
	en		325k	9.6k	7.2	
dev4	jp	489	6,758	1,169	13.8	7
dev123	jp	500	3,818	936	7.6	16
test	jp	500	7,367	1,301	14.7	7
ASRtest	jp	500	7,494	1,331	15.0	7

Table 4: *Japanese→English corpus statistics.*

4.4.2. Arabic

Following a similar approach to that taken in [13], we use the Buckwalter Arabic Morphological Analyzer[2] to obtain possible word analyses for Arabic, and disambiguate them using the Morphological Analysis and Disambiguation for Arabic (MADA) tool [14], kindly provided by the University of Columbia.

Once analyzed, Arabic words are segmented by separating all prefixes (prepositions, conjunctions, the article and the future marker) and suffixes (pronominal clitics). The tool also provides POS tags for the resultant tokens.

4.4.3. Chinese

Chinese preprocessing included re-segmentation and POS-tagging. These tasks were performed using ICTCLAS [15].

4.4.4. Italian

Italian was POS-tagged and lemmatized using the freely-available FreeLing morpho-syntactic analysis package [16]. Additionally, Italian contracted prepositions were separated into preposition + article, for example 'alla'→'a la', 'degli'→'di gli' or 'dallo'→'da lo'.

4.4.5. Japanese

When dealing with Japanese, one has to come up with new methods for overcoming the absence of delimiters between words. We addressed this issue by word segmentation using the freely available JUMAN tool [17] version 5.1. This tool was also used for POS-tagging of the Japanese text.

4.5. Results

In Table 6 we show the results for all the TALP systems that participated in the IWSLT 2006: the TALP-phrase, the TALP-tuple and the combination of the two (TALP-comb). Here, the results correspond to the additional evaluation specification, i.e. case-insensitive and without punctuation marks. There are several tuns for each system. The runs on the TALP-tuple are explained in [18].

4.6. TALP Phrase-based System

Two TALP-phrase systems were used, the main difference being the inclusion of the SMR technique. A non-monotonic search (with $m = 5$ and $j = 3$) for all tasks and for all systems (with or without SMR technique); except for the Italian to English task where a monotonic search was used.

The primary system of each task is that which had the best performance in the internal test. In all tasks, the SMR improved the results in the internal test (see column "test" in Table 6), except for Italian to English.

The final evaluation suggests that, these conclusions cannot be generalized. In two tasks in particular, Arabic and Japanese to English, the best results from the internal test do not correlate with the results in the final evaluation, where the best performance was achieved by those systems that did not use the SMR technique. This bad generalization of the SMR might be explained by Table 5 which shows the number of unknown words in each test set. Notice that in the development and in the evaluation sets of most tasks the number of unknown words is higher than for the internal test set (specially, for the Arabic and Japanese tasks). The higher the number of unknown words, the worse the SMR output and, consequently, the quality of translation. Here, a possible solution would be to predict word classes for unknown words in order to avoid their bad influence in the SMR output.

Set	Chinese	Arabic	Italian	Japanese
development	71	165	138	66
test	50	55	79	25
evaluation	106	220	186	202

Table 5: *Number of unknown words in development, test and evaluation sets.*

The SMR technique obtained fairly good results for the n-gram-based system, as is shown in [2]. However, we can say that the improvement of the SMR technique is not clear for the phrase-based system in these tasks. SMR could be expected to produce greater improvement in an n-gram-based system than in a phrase-based system. For instance, the extraction of units in the former system is monotonous. This is why the monotonocity of the alignment produces a greater increase in smaller units, which tends to benefit from translation.

4.7. TALP System Combination

In the combined approach, a general improvement of the BLEU score is observed, whereas the NIST score seems to decrease.

This behaviour can be seen in almost all tasks and for the development, the internal test and the evaluation.

This can be explained by the particular features that have been used. Both the IBM Model 1 and the language model tend to benefit shorter outputs. Although, a word bonus was used, we have seen that the outputs produced by the TALP-comb system are shorter than those outputs produced by the TALP-phrase or the TALP-tuple systems, which is why the NIST did not improve.

[2]Version 2.0. Linguistic Data Consortium Catalog: LDC2004L02.

Language	System	Dev		Test		Eval	
		BLEU	NIST	BLEU	NIST	BLEU	NIST
zh2en	TALP-phrase primary (SMR)	19.29	6.57	46.33	8.95	20.08	6.42
	TALP-phrase contrast1	20.36	6.75	44.87	8.56	20.06	6.26
	TALP-tuple primary	19.75	6.64	44.63	8.99	20.34	6.22
	TALP-tuple contrast1	19.69	6.59	44.87	8.96	19.80	6.39
	TALP-comb	21.19	6.69	49.72	8.36	**20.21**	**5.97**
ar2en	TALP-phrase primary (SMR)	27.07	7.15	55.34	10.28	22.20	6.54
	TALP-phrase contrast1	25.95	7.07	54.06	10.24	23.66	6.70
	TALP-tuple primary	29.27	7.52	55.11	10.45	23.83	6.80
	TALP-tuple contrast1	29.48	7.46	54.71	10.41	23.60	6.72
	TALP-tuple contrast2	28.75	7.40	56.39	10.53	23.40	6.65
	TALP-tuple contrast3	29.09	7.41	53.31	10.30	23.10	6.67
	TALP-comb	30.29	7.41	57.34	10.46	**23.95**	**6.60**
it2en	TALP-phrase primary	41.66	9.08	62.68	10.69	35.55	8.32
	TALP-phrase contrast1 (SMR)	41.65	8.92	61.45	10.46	35.55	8.32
	TALP-tuple primary	43.05	9.21	63.40	10.76	37.38	8.59
	TALP-tuple contrast1	43.05	9.20	62.52	10.65	31.13	8.46
	TALP-tuple contrast2	43.63	9.24	63.73	10.79	37.55	8.49
	TALP-tuple contrast3	41.60	9.15	60.98	10.56	36.21	8.35
	TALP-comb	44.13	9.04	63.38	10.43	**37.74**	**8.41**
jp2en	TALP-phrase primary (SMR)	15.37	6.01	48.93	9.54	14.51	5.58
	TALP-phrase contrast1	17.04	6.40	47.52	9.82	15.09	5.82
	TALP-tuple primary	16.59	6.34	47.14	9.42	14.61	5.27
	TALP-tuple contrast1	18.20	6.37	45.45	8.97	15.17	5.18
	TALP-comb	19.36	6.42	51.73	8.8	**15.66**	**5.51**

Table 6: *Results obtained using the TALP-phrase, TALP-tuple and the combination of the two for in all the tasks of the IWSLT 2006. The evaluations are case-insensitive and without punctuation marks.*

5. Conclusions

This paper has presented the TALP-phrase and the TALP-comb for the IWSLT 2006 evaluation.

The TALP-phrase uses the SMR reordering technique, which was expected to coherently improve the quality of translation in the evaluation set as it had in the internal set. The high number of unknown words in the evaluation set may have caused a detriment of the SMR behaviour. We are currently studying improvements for the SMR technique.

The TALP-comb is the combination of the TALP-phrase and the TALP-tuple, using several n-gram language models, a word bonus and the IBM Model 1 for the whole sentence. The combination seems to obtain clear improvements in BLEU score but not in NIST, since the features that operate in the combination generally benefit shorter outputs.

6. Acknowledgments

This work was partially funded by the European Union under the integrated project TC-STAR (IST-2002-FP6-506738, `http://www.tc-star.org`), by the Spanish government under an FPU grant, by the Autonomous Government of Catalonia, the European Social Fund and the Technical University of Catalonia.

The authors wish to thank Nizar Habash for making MADA available for the Arabic experiments.

7. References

[1] M. Costa-jussà and J. Fonollosa, "Tuning a phrase-based statistical translation system for the iwslt 2005 chinese to english and arabic to english tasks," in *IWSLT*, Pittsburgh, 2005.

[2] ——, "Statistical machine reordering," in *Empirical Methods in Natural Language Processing (EMNLP)*, Sydney, 2006.

[3] J. Crego, A. de Gispert, P. Lambert, M. Costa-jussà, M. Khalilov, J. Mariño, J. Fonollosa, and R. Banchs, "Ngram-based smt system enhanced with reordering patterns," in *HLT-NAACL06 Workshop on Building and Using Parallel Texts: Data-Driven Machine Translation and Beyond*, New York, June 2006.

[4] F. Och and H. Ney, "Discriminative training and maximum entropy models for statistical machine translation," in *40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, USA, July 2002, pp. 295–302.

[5] ——, "The alignment template approach to statistical machine translation," *Computational Linguistics*, vol. 30, no. 4, pp. 417–449, December 2004.

[6] J. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, pp. 308–313, 1965.

[7] F. Och, "Giza++ software. http://www-i6.informatik.rwth-aachen.de/~och/ software/giza++.html," 2003.

[8] A. Stolcke, "Srilm - an extensible language modeling toolkit," in *Proc. of the 7th Int. Conf. on Spoken Language Processing, ICSLP'02*, Denver, USA, September 2002.

[9] J. Crego, J. Mariño, and A. de Gispert, "An Ngram-based statistical machine translation decoder," in *Proc. of the 9th Int. Conf. on Spoken Language Processing, ICSLP'05*, Lisboa, April 2005.

[10] B. Chen, R. Cattoni, N. Bertoldi, M. Cettolo, and M. Federico, "The ITC-irst statistical machine translation system for IWSLT-2005," in *Proc. of the Int. Workshop on Spoken Language Translation, IWSLT'05*, Pittsburgh, October 2005, pp. 98–104.

[11] T. Brants, "Tnt - a statistical part-of-speech tagger," in *Proceedings of the Sixth Applied Natural Language Processing, ANLP*, Seattle, 2000.

[12] G. Miller, R. Beckwith, C. Fellbaum, D. Gross, K. Miller, and R. Tengi, "Five papers on WordNet," *Special Issue of International Journal of Lexicography*, vol. 3, no. 4, pp. 235–312, 1991.

[13] N. Habash and F. Sadat, "Arabic preprocessing schemes for statistical machine translation," in *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, New York, 2006, pp. 49–52.

[14] N. Habash and O. Rambow, "Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop," in *43rd Annual Meeting of the Association for Computational Linguistics*, Michigan, 2005, pp. 573–580.

[15] H. Zhang, H. Yu, D. Xiong, and Q. Liu, "Hmm-based chinese lexical analyzer ictclas," in *Proc. of the 2nd SIGHAN Workshop on Chinese language processing*, Sapporo,Japan, 2003, pp. 184–187.

[16] J. Atserias, B. Casas, E. Comelles, M. González, L. Padró, and M. Padró, "Freeling 1.3: Syntactic and semantic services in an open-source nlp library," in *5th Int. Conf. on Language Resource and Evaluation (LREC)*, 2006, pp. 184–187.

[17] Y. Matsumoto and M. Nagao, "Improvements of japanese morphological analyzer juman," in *Proc. of the Int. Workshop on Sharable Natutal Language Resources*, 1994, pp. 22–28.

[18] J. M. Crego, A. de Gispert, P. Lambert, M. Khalilov, M. Costa-jussà, J. Mariño, R. Banchs, and J. A. Fonollosa, "The talp ngram-based system for the iwslt2006," in *IWSLT*, Kyoto, November 2006.