LIG Statistical Machine Translation Systems for IWSLT 2010

Laurent BESACIER, Haithem AFLI, Do Thi Ngoc DIEP , Hervé BLANCHON, Marion POTET

LIG Laboratory
University of Grenoble, France
name.surname@imag.fr

Abstract

This paper describes the systems developed by the LIG laboratory for the 2010 IWSLT evaluation. We participated to the AE BTEC task and to the new TALK task.
For AE BTEC task we developed two different systems: a statistical phrase-based system and a hierarchical phrase-based system using the Moses toolkit. The combination of these systems, which improves the results on different development sets, makes our final submission.
This year, we concentrated on the new TALK task. The development of a reference translation system, as well as an ASR output translation system, is presented. For this latter task, re-punctuating the ASR output, before translation, seems to be very useful, while segmenting the ASR flow, which is also discussed in this paper, has shown to be less useful. Unsuccessful attempts to exploit ASR lattices instead of ASR 1best are also presented at the end of this article.

1. Introduction

In the framework of the evaluation campaign for the 2010 International Workshop on Spoken Language Translation (IWSLT-2010) the LIG participated to the AE BTEC task and to the new TALK task.

For the BTEC task, we aimed at building a hierarchical phrase-based translation system from Arabic to English. The developed system outperformed our last year Phrase-based system and a combination of both systems was submitted. This year, we concentrated on the new TALK task. For the translation from ASR output, re-punctuation of the ASR output, before translation, was investigated as well as the segmentation of the ASR flow.

The remainder of the paper is structured as follows. Section 2 gives an overview of the Arabic to English translation system and the associated results. Then, we describe chronologically the work done for the TALK task : in section 3, we present the system built for the translation of references. The best system obtained in section 3 is used for the real speech translation task (from ASR output) which is described in more details in section 4. In this section, we give an in depth account of the re-punctuation and re-segmentation of the ASR output, as well as the use of ASR lattices instead of the 1best. Finally, in section 5 we sum up our work.

2. BTEC AE System

2.1. Task, data and tools

Since 2007, the LIG laboratory participates yearly to the IWSLT evaluation campaign (Arabic – English BTEC task). In the experiments reported here, we have used the data provided by the IWSLT10 organizers and a few publicly available additional data.

For training the translation models, the train part of the IWSLT10 data was used (a training corpus of 19972 sentence pairs). As for development data, we used several subsets provided: the dev4 subset, made up of 489 sentences, which corresponds to the IWSLT06 development data (we will refer, in the rest of the paper, to dev06 for this data set); the dev5 subset, made up of 500 sentences, which corresponds to the IWSLT06 evaluation data (we will refer, in the rest of the paper, to tst06 for this data set); the dev6 subset, made up of 489 sentences, which corresponds to the IWSLT07 evaluation data (we will refer, in the rest of the paper, to tst07 for this data set); and the dev7 subset, made up of 509 sentences, which corresponds to the IWSLT08 evaluation data (we will refer, in the rest of the paper, to tst08 for this data set).

The tuning of the MT model parameters (Minimum Error Rate Training) was systematically done on the dev06 subset.
As additional data, we used an Arabic-English bilingual dictionary of around 84k entries. This dictionary can be found online[1]. For training the English LM, we also used out-of-domain corpora taken from the LDC's Gigaword corpus[2].

Our baseline SMT system was built using tools available in the MT community:

-GIZA++ [12] was used for the alignments,
-The Moses decoder [9] (and the training / testing scripts associated) was used (2010-04-26 release),
-SRILM [14] was used to train the LMs and to deal with ASR word graphs,
-The ASVM morphological analyzer [7] was used for Arabic word segmentation,
-All the performances reported in this paper are BLEU [13].

2.2. Overview of the 2010 system

More details on the LIG AE Phrase-based system can be found in [2] and [3]. It is trained on the provided 20k train bitext, concatenated to the bilingual dictionary of 84K entries. The Moses training script is used to build a phrase translation table. The Arabic part of the bitext is systematically segmented using ASVM segmentation (more deeply described on the LIG former IWSLT papers [2], [3]). On the English side, we removed punctuation and case (both pieces of information are further restored after translation using hidden-ngram and disambiguation from the SRILM toolkit). For English, we used both in-domain (English part of the train bitext) and out-of-domain (LDC's Gigaword corpus) to train the English LM.

[1] http://freedict.cvs.sourceforge.net/freedict/eng-ara/
[2] http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2002L49

2.2.1 Improvements of the 2009 PB-SMT system

While studying the results of IWSLT09, we noticed that our system was tuned to optimize BLEU(no_case+no_punct) whereas translation output must be re-punctuated and recased before IWSLT evaluation. Based on this observation, we tried to improve our system by using our re-punctuation and recaser during the MERT (Minimum Error Rate Training) tuning. This new process slightly but consistently increased the BLEU score on our different development sets (detailed results not reported here).

2.2.2 Hierarchical Phrase-based system

Hierarchical phrase-based translation systems are an interesting alternative to the standard phrase-based systems. Recently, Moses provided tools for building hierarchical systems based on the David Chiang approach [8].

The entire statistical system based on the hierarchical tools of Moses was built like the Phrase-based system. First, we used GIZA++ for alignments. Grammar rules were extracted from these alignments, following the rules extraction process described in [8]. The LM of the phrase-based system was used in the hierarchical system. Weights were optimized on the development set (dev06) using the provided MERT procedure. *Figure 1* shows a few example rules obtained with this process (associated scores have been removed).

[X][X] تحتفظ ب امتعت ي [X] ||| [X][X] keep my baggage [X]
[X][X] تحجز [X][X] غرفة [X] ||| [X][X] get [X][X] room [X]
احجز [X][X] رحلة [X][X] بعد [X] ||| reserved [X][X] [X][X] flight yet [X]

Figure 1: Example of rules for the hierarchical phrase-based SMT system

For the tuning, we applied the same improvement we used for the phrase-based system (section 2.2.1). Moreover, we noticed that, with the Moses chart decoder, unknown words were copied verbatim to the output. We thus decided, for the hierarchical system, to drop unknown words (as done for the PB-SMT) in order to optimize BLEU even if it is not clear, from human judgments point of view, if this might help or not. Dropping unknown words resulted in significantly better BLEU scores (2 points in average, detailed results not reported here).

2.2.3 System combination

In order to take advantage of the strengths of the various modeling and decoding techniques of our systems, we implemented a system combination step based on confusion network decoding using the MANY toolkit [1].

2.3. Experimental results

Table 1 gives an overview of the experiments made with the combination of our SMT systems. In this table, we show results for the best hierarchical and phrase-based systems described previously. We give as well the results of the combined system. This later system is our official system for the 2010 evaluation campaign.

Table 1: Experiments (BLEU) for system combination (the final system submitted to IWSLT 2010 is put in bold)

	dev06	**tst06**	**tst07**	**tst08**
Best Hierarchical phrase based system	37,85	29,19	54,03	52,52
Best Phrase Based system	36,56	29,16	51,49	50,77
System combination	**38,91**	**30,94**	**55,35**	**53,27**

Table 2 shows the official results obtained by LIG this year. It is interesting to notice that while the BLEU (no_case+no_punct) is better for the IWSLT09 test set, it is not the case for the IWSLT10 test set. Further investigation on this is part of future work.

Table 2: Official automatic evaluation results obtained by the LIG at IWSLT10 (BLEU score)

	case+punc	**no case+no punc**
IWSLT09_testset	46,47	46,54
IWSLT10_testset	37,69	36,51

2.4. Progresses made along the years

Figure 2 presents performance evolution of the LIG AE system measured by running and evaluating our 2007, 2008, 2009 and 2010 systems on the same data sets (dev06, tst06 and tst07). The results show a yearly improvement of our AE MT system. In the mean time, we also noticed that our relative ranking, compared to other participants, has been slightly decreasing over the years.

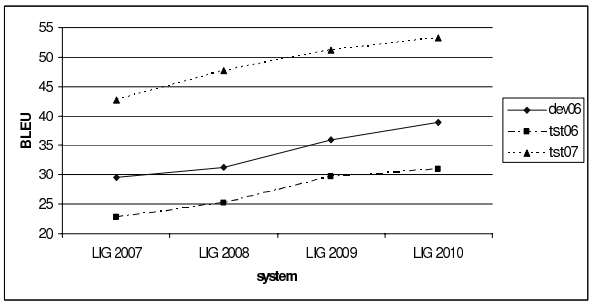

Figure 2: Evaluation of the verbatim text translation performance (BLEU) for LIG systems from 2007.

3. TALK Task : Translation of References

This year, a new task was dedicated to the translation of the TED Talks corpus, a collection of public speeches on a variety of topics for which video, transcripts and translations are available on the Web. Training data for this exercise was limited to a supplied collection of freely available parallel texts, including a parallel corpus of TED Talks. The translation input conditions of the TALK task consisted of (1) automatic speech recognition (ASR) outputs, i.e., word lattices (SLF), N-best lists (NBEST) and 1-best (1BEST) speech recognition results, and (2) correct recognition results

(CRR), i.e., text input without speech recognition errors. Participants of the TALK task had to submit MT runs for both input conditions.

3.1. Used Resources

We used the TED Talks collection plus other parallel corpora distributed by the ACL 2010 Workshop on Statistical Machine Translation (WMT).

For the training of the translation models, we used the provided Europarl and News parallel corpora (total 1,767,780 sentences) and the TED training corpus (total 47,652 sentences). The UN data (total 7,230,217 sentences) was used in one experiment described here but finally eliminated in our final system. For the language model training, in addition to the French side of the bitexts described above, the News monolingual corpus in French was used (total 15,234,997 sentences).

The TED dev set (1307 sentences) was used for tuning and evaluation purpose. This corpus will be referred to as Dev in the rest of this paper.

3.2. Preprocessing / Post-processing

As far as preprocessing is concerned, we lowercased and tokenized all the data but kept punctuation for the LM and TM models training. Before translation, a source English sentence is thus lowercased and tokenized. The translated output in French needs to be detokenized and recased. We tried two different techniques to recase the translated output:

– the first one used a target language model trained on cased French data (Europarl+News+UN+Newsmono: 24M sentences in total) and the disambig command of SRILM (similar to what we did for the BTEC AE task),

– the second one used a SMT-like approach where a phrase table was trained from a parallel French no-case/case corpus (trained on the News monolingual corpus in French of 15M sentences). This second approach systematically outperformed the first one and was used in our final system.

For the Reference translation task, the punctuation of the translated output was refined using the punctuation of the source sentence (practically, the ending punctuation mark of the source sentence was put at the end of the translated sentence). The contrastive results with/without this post-processing step can be found in *Table 4*.

3.3. Language modeling

The target language model is a standard 3-gram language model trained using the SRI language modeling toolkit [14]. The smoothing technique we applied is the modified Kneser-Ney discounting with interpolation.

As a first language model, we interpolated a LM trained on the TED training data (47k sentences) with a LM trained on the News monolingual corpus in French (15M sentences). A perplexity test, reported in *Table 3*[3] was used to optimize the interpolation weight (on the Dev of TED corpus). Based on the results of *Table 3*, the LM selected for the experiments of this section, is the one corresponding to an interpolation weight equal to 0.5. This LM will be referred to as LM1 in the rest of this section. The second LM used in this section (LM2), was obtained by interpolating a LM trained on more data (Europarl+News+UN+Newsmono: 24M sentences in total) with the TED LM using the same weight (0.5). The results of this section will be given with these two models: LM1 and LM2.

Table 3: Optimizing the LM interpolation weights (TED LM interpolated with bigger out-of-domain LM), the second line numbers are Perplexity

Weight (TED)	0	0.3	0.4	**0.5**	0.6	0.7	1
on Dev	198	172	170	**169**	169	172	202

3.4. Translation modeling and tuning

For the translation model training, the uncased (but punctuated) corpus was word aligned and then, the pairs of source and corresponding target phrases were extracted from the word-aligned bilingual training corpus using the scripts provided with the Moses decoder [9]. The result is a phrase-table containing all the aligned phrases. This phrase-table, produced by the translation modeling, is used to extract several translations models. In our experiments, we used thirteen standard translation models: six distortion models, a lexicon word-based and a phrase-based translation model for both direction, and a phrase, word and distortion penalty.

For the decoding, the system uses a log-linear combination of the previous target language model and the thirteen translation models extracted from the phrase table. As the system can be beforehand tuned by adjusting log-linear combination weights on a development corpus, we used the Minimum Error Rate Training (MERT) method. MERT was applied on the TED Dev corpus (1307 sentences). We decided to not split this corpus into a Dev and a Test part. So all the results reported on this section are given on Dev after systematic tuning on the same Dev corpus. Finally, it is also important to note that, during tuning, punctuation was systematically removed from the Nbest lists and BLEU was calculated using un-punctuated references. While such tuning procedure might be sub-optimal to optimize BLEU (cased), we did this to anticipate the ASR output translation task for which decoding (and tuning) is also done without punctuation.

3.5. Improvements over our baseline

The baseline system described in the previous section is referred to as *Baseline* in *Table 4*. This table presents also incremental improvements obtained during the development of this translation system (from references). The system improvements proposed are the following:

– do not reorder over punctuation (+mp) during decoding: +0.09 BLEU,

– refine the punctuation after translation using the source sentence punctuation (+postp, see section 3.2): +1.22 BLEU,

– add UN data to train the translation model: -0.11 BLEU,

– apply phrase-table pruning with a technique similar to [10] (+PTprune): +0.84 BLEU (retuning with MERT needed after pruning),

– use of LM2 (more data) instead of LM1: +0.08 BLEU,

– give more weight to the TED data during training by duplicating it 10 times before training the phrase table (then

[3] This ppl test was done without case and without punctuation on the LM training data and on the Dev data

PT pruning is applied); this method is similar to the work of [16] (+resample) : - 0.04 BLEU,

– glue full sentences before decoding: we tried to glue back the full source sentences from the initial 1307 Dev segments ; this was done using the punctuation information ; then the full sentences were sent to the translation system and the output was re-segmented before scoring using the MWER tool provided by RWTH [11]: -0.10 BLEU.

Table 4: Improvements of the Reference translation system: BLEU (without case) given on Dev data after tuning ; system in bold corresponds to LIG_P for translation of references

System	LM	BLEU no-punct	BLEU punct
Baseline	LM1	0.2410	0.2461
+mp	LM1	0.2414	0.2470
+mp+postp	LM1	0.2414	0.2592
+mp+postp+UN	LM1	0.2393	0.2581
+mp+postp+PTprune(no mert)	LM1	0.2407	0.2582
+mp+postp+PTprune(mert)	LM1	**0.2507**	0.2676
+mp+postp+PTprune(mert)	**LM2**	0.2495	**0.2684**
+mp+postp+PTprune(mert)+resamp.	LM2	0.2483	0.2680
+mp+postp+PTprune(mert) + gluing full sentences before	LM1	0.2502	0.2674

3.6. Discussion

The best improvements came from the phrase table pruning, while adding UN data did not help. The refinement of the punctuation using the source data is necessary to optimize the BLEU calculated on punctuated output. The use of a different segmentation (sentences instead of segments) did not help either. Our official system submitted to IWSLT 2010 (Reference translation) is the one put in bold in *Table 4*. This performance placed LIG at the 3rd rank among 9 participants to the TALK Reference translation task. *Table 2* reports LIG official results for the TALK task.

Table 2: Official automatic evaluation results obtained by the LIG at IWSLT10 (BLEU score) - Reference translation

	case+punc	**no case+no punc**
TED_Dev	0.2534	0.2495
TED_Tst	0.2742	0.2623

4. TALK task : Translation of ASR output

For this system, the translation model used is the pruned phrase table while the language model is LM2. This corresponds to the system in bold in *table 4* but the re-punctuation process is obviously different since no "source punctuation" can be used in that case (the ASR output does not contain any punctuation mark).

4.1. Preprocessing of the 1best ASR output

In order to be consistent with our translation model, the ASR output is lowercased and tokenized before translation. Different additional pre-processing steps were investigated:

– re-punctuating the (source) English ASR output,

– re-segmenting the (source) English ASR output.

These additional pre-processing steps are described and evaluated in section 4.3 and 4.4 respectively.

4.2. Post processing of the MT output

While re-punctuation of the French MT output was done in a straightforward way for the Reference translation (see section 3.2), it was necessary to develop a true re-punctuation system for French in the case of ASR output translation. This was done by building a French language model trained on punctuated and uncased French data (Europarl +News+UN+Newsmono: 24M sentences in total). The punctuation was restored after translation using this LM and the hidden-ngram command from SRILM toolkit. After re-punctuation, we used the SMT-based recaser presented in section 3.2.

4.3. Re-punctuating the English 1best ASR output

To re-punctuate the English 1best ASR output we used a conventional approach (LM+hidden-ngram). *Table 5* reports results obtained using (1) no re-punctuation (2) re-punctuation using a LM trained on punctuated and uncased TED data only (3) re-punctuation using a LM trained on punctuated and uncased TED data interpolated (weight=0.5) with a LM trained on punctuated and uncased Europarl+News English data (total 1,767,780 sentences). Whereas all these results were obtained without retuning the log-linear weights, (4) is an attempt to retune these weights using the 1best ASR output instead of the Reference in English (MT tuning was done without punctuation and without case on the MT Nbest lists).

Table 5: Effect of re-punctuating the English 1best ASR before translation (results obtained with the IWSLT eval. server on Dev data) ; system (3) corresponds to LIG_C1

Repunct.	bleu(p+c)	bleu(c)	bleu(x)[4]
1: no	0.1402	0.1465	0.1657
2: LM (TED)	0.1445	0.1475	0.1670
3: LM (TED+Europarl+News)	**0.1475**	0.1504	0.1698
4: (3) + MERT(1best)	0.1462	**0.1521**	**0.1720**

Those results show that re-punctuating the ASR output is useful because there is actually punctuation in our translation model. The best re-punctuation LM is the one which interpolates in-domain data (TED) with a large amount of out-of-domain data. The re-tuning of the log-linear weights did improve BLEU(nopunct+nocase) while it slightly decreased BLEU(punct+case). System (3) was submitted as our "contrastive 1" (LIG_C1) system. As an additional experiment, we evaluated the performances of a "verbatim-driven" ASR output translation system. Such a system actually translates the 1best ASR output using the correct ending punctuation from the English Reference (this punctuation information being used to re-punctuate the translated French, as done in section 3). This experiment gives an upper bound of the performance that could be obtained if the re-punctuation of the ASR output was perfect. The BLEU(case+punct) obtained in that case was 0.1625. Such a system was obviously not included in the LIG submission.

[4] No punct+no case

4.4. Re- segmenting the English 1best ASR output

For the ASR data (1BEST for instance), the segmentation is different from the Reference since it is segmented into longer segments. For instance, the TED Dev data has 259 ASR 1best segments instead of 1307 for the Reference translation task. Considering the ASR output as a single flow of words, we decided to investigate the effect of ASR output segmentation on the speech translation performance. This was done by building an English segmenter that infers the most likely segmentation (location of segment boundaries) from the ASR flow, based on a segment language model. The segment language model is a standard backoff 3-gram modeling segmentation using the boundary tags <s> and </s>. This segment LM was trained on the TED data where the symbols "." and "?" were used to position the boundary tags. We used a bias *b* to make a segment boundary a priori more likely by a factor of *b*. For instance, a bias equal to 1 lead to 450 segments while a bias equal to 0.1 lead to 59 segments. The re-segmented ASR data was sent (without re-punctuation) to the MT system. For scoring, we used the MWER tool provided by RWTH [11] to re-segment the MT output. The results are reported in *table 6*.

Table 6: Effect of re-segmenting the English 1best ASR before translation
(results obtained with the IWSLT eval. server on Dev data)

Segmentation	bleu(p+c)	bleu(c)	bleu(x)
ASR (259 seg)	0.1402	0.1465	0.1657
SegmentLM (29 seg)	0.1409	0.1477	0.1675
SegmentLM (59 seg)	0.1409	**0.1480**	**0.1677**
SegmentLM (117 seg)	0.1403	0.1473	0.1671
SegmentLM (223 seg)	**0.1411**	0.1475	0.1676
SegmentLM (305 seg)	0.1409	0.1475	0.1675
SegmentLM (390 seg)	0.1402	0.1466	0.1668
SegmentLM (450 seg)	0.1402	0.1466	0.1666

The results obtained are not conclusive. The performance remains rather stable for different segmentation granularities. Based on this, we decided to not use, in our submitted systems, any segmentation of the ASR output before translation.

4.5. (Unsuccessful) attempts to exploit ASR graphs

In spoken language translation, one problem we sometimes face is that the word graphs provided by the ASR system do not have necessarily a word representation (tokenization, case) compatible with the one used to train the MT models. It is actually the case in the framework of the TALK task where the English ASR system used to generate the lattices was unknown to the participants. Moreover, we quickly realized that the density of the provided lattices was too high. Lastly, even if we were aware that Moses toolkit is able to decode lattices, we decided to perform confusion network (CN) decoding because we thought that handling punctuation might be easier with confusion nets than with lattices. Based on these observations and assumptions, we applied the following treatments to the ASR lattices provided for the TALK task:

– we pruned lattices nodes with posteriors inferior to 0.001 times the highest posterior path,

– we lowercased all the words inside the ASR word graph,

– we tokenized the lattices,

– we converted the obtained lattices to confusion networks[5].

Table 7 summarizes some results obtained.

The first line reports the best result obtained in section 4.3 (*table 5*) and corresponds to LIG_C1 official system.

The second line corresponds to the same system that was applied to the consensus hypotheses obtained from each CN (coming from the four steps explained above), instead of the ASR 1best. It is interesting to note that doing this decreases BLEU by approximately 0.5 points. It means that working with CN, we start with a disadvantage of 0.5 BLEU! This gap might be due to a too drastic pruning as well as to the introduction of new (and possibly incorrect) hypotheses paths within the CN.

The third line uses an approach similar to [5]: we generate multiple hypotheses of punctuation marks during the re-punctuation of the source English (this was done from the consensus hypotheses) and a CN is generated with punctuation marks. At this point, the obtained CN contains punctuation ambiguity but not yet word ambiguity.

On the contrary, the fourth line deals with word ambiguity (we tried to directly translate the CN obtained from ASR lattices) while no punctuation is added. We did not have time to experiment both punctuation and word ambiguity by merging the punctuated CN (line 3) with the original CN (line 4), this idea (suggested and experimented in [5]) is part of future work.

Finally, it is important to note that all the parameters of the log-linear model used for the CN decoder were retuned (for line 3 and line 4 results) since an additional parameter, corresponding to the CN posterior probability is added in the case of CN decoding.

Table 7: Attempts to decode graphs for the TED task
(results obtained with the IWSLT eval. server on Dev data)

System.	bleu(p+c)	bleu(c)	bleu(x)
(1) 1 best	**0.1475**	**0.1504**	**0.1698**
(2) consensus	0.1419	0.1454	0.1645
(3) CN decoding : consensus + punct. ambig.	0.1429	0.1437	0.1628
(4) CN decoding : word ambig.	0.1401	0.1492	0.1674

The performance obtained with these methods is disappointing. The use of punctuation or word ambiguity via CN decoding did not improve our ASR 1best translation system.

4.6. Discussion

Our best improvements came from the use of a module to re-punctuate the English 1best ASR output, while CN decoding did not lead to further improvements.

As a final submission for the translation of ASR output task, we decided to combine the LIG_C1 system (referred to as system (3) in *table 5*) with system (4) of *table 5* and system (3) of *table 7*. The combination was very basic since a confusion net was built from the three hypotheses and the

[5] These 4 steps can be done with the following command : **lattice-tool –in-lattice lattice_file –read-htk –posterior-prune 0.001 –tolower –split-multiwords –multi-char "'" –write-mesh CN_file**

word sequence corresponding to the highest probability path was printed out. Such a system combination scheme slightly increased the performance, as illustrated in *Table 8*.

This combined system was submitted as our "primary" (LIG_P) system. With this system, LIG obtained the best performance evaluated with BLEU(punct+case) among 9 participants.

Table 8 : Official automatic evaluation results obtained by LIG at IWSLT10 (BLEU score) – ASR output translation

System.	bleu(p+c)	bleu(c)	bleu(x)
LIG_P (Dev)	**0.1478**	**0.1523**	**0.1719**
LIG_C1 (Dev)	0.1475	0.1504	0.1698
LIG_P (Tst)	**0,1634**	0.1733	0.1903
LIG_C1 (Tst)	0.1633	**0.1735**	**0.1905**

5. Conclusion

For the Arabic to English translation, we introduced, this year, a hierarchical system which outperformed our PB-SMT. This system was combined with an improved PB-SMT before submission. For the TALK task, we first optimized BLEU for the Reference translation task (the best improvement was obtained using PT pruning) and worked on the re-punctuation and re-segmentation of the ASR output, before translation. The final speech translation system submitted by LIG was ranked among the best sites that participated to IWSLT TALK task this year.

6. References

[1] Barrault L "MANY: Open Source MT System Combination at WMT'10". *ACL Workshop on Statistical Machine Translation (WMT'10)*, Uppsala (Sweden), 15-16 July.

[2] L. Besacier, A. Ben-Youcef, H. Blanchon "The LIG Arabic / English Speech Translation System à IWSLT08" *IWSLT08*. Hawai. USA. October 2008.

[3] Bougares F., Besacier L., & Blanchon H. (2009). "The LIG Arabic / English Speech translation System at IWSLT09". *IWSLT09*. Tokyo, Japan. December 2010.

[4] Brown Peter, F., Pietra, V. J., Pietra, S. A., & Mercer, R. L. (1993). "The Mathematics of Statistical Machine Translation: Parameter Estimation." *IBM T.J. Watson Research Center Report*, 264-311.

[5] R. Cattoni, N. Bertoldi, M. Federico "Punctuating Confusion Networks for Speech Translation". In *Interspeech 2007*. Antwerp, Belgium, August 2007.

[6] B. Chen, D. Xiong, M. Zhang, A. Aw, and H. Li, "I2r multi-pass machine translation system for iwslt 2008," in *IWSLT2008*. Hawai. USA. October 2008.

[7] Diab, M., Hacioglu, K., & Jurafsky, D. (2004). "Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks". *HLT-NAACL04*. USA.

[8] Chiang, D. (2007). « Hierarchical phrase-based translation ». 2007. *Computational Linguistics* 33(2):201–228.

[9] Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., et al. (2007). "Moses: Open Source Toolkit for Statistical Machine Translation". *ACL 2007*, demonstration session.

[10] H. Johnson & al (2007) "Improving Translation Quality by Discarding Most of the Phrasetable". In *proceedings of the EMNLP-CoNLL 2007*. pp 967-975.

[11] E. Matusov, G. Leusch, O. Bender, and H. Ney. "Evaluating Machine Translation Output with Automatic Sentence Segmentation". *In International Workshop on Spoken Language Translation (IWSLT)*, pages 148-154, Pittsburgh, PA, USA, October 2005.

[12] Och, F. J. and Ney, H., "A Systematic Comparison of Various Statistical Alignment Models", *Computational Linguistics*, vol. 29, no. 1, pp. 19-51, March 2003.

[13] Papineni, K., Roukos, S., Ward, T., and Zhu, W., "BLEU: A method for automatic evaluation of machine translation", *ACL'02*, pp. 311-318, Philadelphia, USA, July 2002.

[14] Stolcke, A., "SRILM - An Extensible Language Modeling Toolkit", *ICSLP'02*, vol. 2, pp. 901-904, Denver, Colorado, September 2002.

[15] W. Shen, B. Delaney, T. Anderson, and R. Slyh, "The MIT-LL/AFRL IWSLT-2008 MT system," in *IWSLT08*, Hawaii, U.S.A, 2008, pp.69–76.

[16] Kashif Shah, Loïc Barrault & Holger Schwenk, "Translation Model Adaptation by Resampling", *ACL Workshop on Statistical Machine Translation (WMT'10)*, Uppsala, Sweden, 2010.