Large-scale multiple language translation accelerator at the United Nations

Bruno Pouliquen[†], Cecilia Elizalde[‡], Marcin Junczys-Dowmunt[†], Christophe Mazenc[†], José García-Verdugo[‡]

[†] World Intellectual Property Organization
Global Databases Service
34, chemin des Colombettes
CH-1211 Geneva 20
Bruno.Pouliquen@wipo.int, Christophe.Mazenc@wipo.int, Marcin.Junczys-Dowmunt@wipo.int

[‡] United Nations
DGACM, Documentation Division
Spanish Translation Service
405 42nd Street
New York NY 10017
elizalde@un.org, garcia-verdugo@un.org

Abstract

Described is a large-scale implementation of a Moses-based machine translation system in the United Nations aiming at accelerating the work of translators. The system (called TAPTA4UN) has been trained on extensive parallel corpora in 6 languages. Both automatic and human evaluations are provided. The system is now used in production by professional translators. The technical challenges of scalability and the final evaluation by users are also described.

1 Introduction and related work

The introduction of machine translation (MT) at the United Nations (UN) is quite unusual in the sense that it was professional translators in New York, not the management, who promoted the development and implementation of a statistical machine translation (SMT) system that would facilitate their work, in particular for certain categories of recurrent documents. Some UN translators in New York had used Google Translate and Bing Translator either through their web interfaces or through computer-assisted translation tools, and had found that, in some specific cases, these systems were able to produce rough drafts that were deemed to be good enough for post-editing. However, these systems could not be customized to reflect the UN style and terminology guidelines, which are mandatory.

For this reason, a pro-bono collaboration was established with another international organization (World Intellectual Property Organization – WIPO) to use their existing SMT system (called TAPTA described in Pouliquen et al 2011), based on Moses (Koehn et al. 2007), and to train it with UN documents. This first experimental system, described in Pouliquen et al. 2012, was trained with a corpus of 11 years of English-Spanish UN documents, translated and revised by a large team of highly-skilled translators. The results of this system, which were reflected in automatic and human evaluations, elicited a very positive reaction from Spanish translators, in particular revisers (senior translators), who do not adopt new technologies easily, but who found that MT could be useful, as it eliminated the need for typing and it presented highly consistent terminology (hence the name: translation accelerator).

As the collaboration project with WIPO was coming to an end, the Spanish translators requested the management to implement the new system on servers administered by the UN. The enthusiastic feedback from the Spanish translators prompted colleagues in other translation services to request the inclusion of their language combinations to develop additional systems.

Following-up on this request and the momentum gathered by the English-Spanish prototype, UN decided to proceed and set up a system whose architecture is capable of handling all language pairs from and into English. As the UN has 6 official languages (namely Arabic, Chinese[1], English, French, Spanish and Russian),[2] our challenge was to train 10 SMT models, to set-up an architecture allowing UN translators to use the models intensively, and with a quality that should at the very least be better than pub-

[1] UN uses simplified Chinese

[2] Some documents are also available in German but we have not included them in our system yet

Sima'an, K., Forcada, M.L., Grasmick, D., Depraetere, H., Way, A. (eds.)
Proceedings of the XIV Machine Translation Summit (Nice, September 2–6, 2013), p. 345–352.

licly available MT engines (namely Google Translate and Microsoft Bing Translator).

For various reasons (budget, flexibility, accessibility, etc.) we also wanted to train and use the models in the cloud.

Similar work has already been done on processing extensive parallel corpora such as Koehn et al. (2009), with an impressive number of 462 language pairs. Other SMT systems running in production are using large corpora (e.g. Pouliquen et al. 2011, Zhechev 2012). Training an SMT system in the cloud is a known option as it allows any user to get a well scaled server on demand (see e.g. Khalilov & Choudhury, 2011).

2 System description

Our approach relies purely on phrase-based SMT, using the open-source toolkit Moses. The main advantage (over other techniques like rule-based systems) is that we can easily train new models as far as we have parallel data (in the future, we could in theory train 21 systems if we wanted to have all language combinations, and even 28 if we include German).

2.1 Data

Every UN document, regardless of the source language, is translated into the 5 remaining languages. Therefore, we have similar sizes of corpora across languages. However, for some technical/historical reasons not every single text is electronically available in all 6 languages, which forced us to export our data as 5 different bitext collections[3]. See table 1 for a detailed description of the corpora size (after filtering).

Table 1: Bitext corpora size

Language pair	*English Million words*	*Million segments*	*# documents*
en--ru	269	12	68993
en--fr	266	13	72324
en--ar	224	10	64708
en--es	212	10	57933
en--zh	190	8	59840

Independently from our work, though very similar, a new version of the UN corpus (Franz, 2013) has been made available recently by LDC[4].

2.2 Preprocessing steps

In any data-driven approach, the cleaner the corpus the better. Therefore, we also concentrated on preparing clean data. This is done through various steps, detailed here:

1) Tokenization

The basic step before going any further with text analysis is tokenization, as Moses is based on word translations. We built our own tokenization engine so that we could adapt it to our specific SMT needs (it is based on the Lucene framework, McCandless et al., 2012).

Certain common rules are applied to English, Spanish and French (acronyms, Greek letters, references), and we use some language-specific rules (e.g. apostrophes).

For Chinese, we use SmartCn[5] with some adaptations (eg. recognize figure references as one single token: "部队由贝宁(300～304)、加纳" → "部队 | 由 | 贝宁 | (300–304) | , | 加纳").

Arabic is also quite challenging for SMT (see Soudi et al., 2012). We remove short vowels, and try to split the most common prefixes, which are usually not ambiguous when found at the beginning of a word:

al-"ال", wal- "وال", lil- "لـل", bi- "ب", lia- "ال", kal- "كـال“

In addition, for other prefixes which could be ambiguous (like "ka-"), we add a finite list of words where we force the split.

e.g.

"كدبلوماسي" *diplomat*, which is decomposed in the prefix "ka-" - *as* - ("ك") and "دبلوماسي" - *diplomat*

2) Sentence alignment

We use a home-made adaptation of Champollion (Ma, 2006) where the tool first aligns sentences and then tries to split at a lower level (segment level) when the sentence contains more than 80 characters (the further step of word alignment

[3] For technical reason the original language and the translated language are unknown when exporting the data, therefore we do not differentiate source and target direction (ie. the French texts translated into English from the English texts translated into French).

[4] http://www.ldc.upenn.edu/Catalog/catalogEntry.jsp?catalogId=LDC2013T06

[5] http://lucene.apache.org/core/old_versioned_docs/versions/3_0_0/api/contrib-smartcn/

will ignore long sentences, therefore smaller segments are generated at that stage).

3) Filtering

Texts that are not in the expected language are discarded. Special care is taken to clean the automatically aligned segments: after the Champollion process, we compute a lexical score between segments, using an automatically learned bilingual dictionary. Each aligned segment has a score between 0 and 100. If its average score is lower than 15, all the segments in a bitext are discarded. A filter is then applied to adjacent segments in order to "propagate" the alignment score to neighbors, so that a bad alignment between good segments is retained and, conversely, a good alignment between bad segments is discarded. Another specific filter discards groups of 3 consecutive segments with exactly the same source and target text in order to exclude untranslated paragraphs from a text.

2.3 Training and scalability

A tool that parallelizes all the preprocessing steps has been built. The input corpus is split into several parts and each step (tokenization, sentence alignment, filtering) is applied to each part in parallel.

We set apart a collection of recent documents from the resulting parallel corpora for all language pairs. Out of these documents, we randomly select 1000 segments for testing and 2000 as a development set used later for weight optimization.

As in a typical training procedure, the first step relies on computing word alignment models. For this we use MGIZA++ (Gao & Vogel 2008) to accelerate the process. Despite the multi-thread computations of MGIZA++, this step still takes about 2-3 days on a 8-core server, on average and for each language pair (Note that word alignment is slow because of generally long sentences).

In the newest version of Moses, training scripts have been extensively parallelized, reducing the time needed for the remaining standard training steps to about one day. Before, this was the longest part which could take several days.

Original binary versions of the phrase table and reordering model are huge; on average we observe sizes of around 20 Gb, counting both tables. We have chosen to prune the phrase table using the significance-based method described in Johnson et al. (2007). This is also done in parallel by splitting the phrase table and merging the filtered parts. Pruned single-word phrase pairs are reintroduced into the phrase table to avoid OOV-errors during translation. The reordering model is filtered according to the pruned phrase table. As a result, our phrase table and reordering model sizes are reduced by a factor of 3.

A further significant size reduction is achieved by applying the compact phrase table tool described in Junczys-Dowmunt (2012) to binarize phrase tables and the reordering models.

Language models are being computed with the IRSTLM toolkit (Federico et al., 2008). We use 5-gram language models but we first prune the model by setting a threshold that discards half of the less significant 5-grams and then we apply the prune-lm tool provided by IRSTLM. Pruning cuts the size to approximately half the original, while not greatly affecting quality (BLEU score, Papineni et al. (2002), falls from 50 to 49.8).

Table 2: Model size reduction (English→Chinese model).

	Phrase table		Reordering model		Language model	
	M rows	Gb	M rows	Gb	M rows	Gb
Basic	82	9.70	82	8.70	49	1.70
Pruned	19	2.20	19	1.90	31	1.00
Binarized		0.27		0.15		0.70

The size reduction has been systematically applied to all described language pairs and translation directions. An example for English into Chinese is given in Table 2. We can see that instead of the initial 20 Gb, we end up with only 1.12 Gb of disk space consumed for each model. This means that during translation a similar amount of memory is being used by each translation process. Several translation directions can now be easily loaded into memory on a single server avoiding IO-bottlenecks completely. Besides making it possible to use large translation models on average-size servers, there are significant speed gains.

Translation speed is further optimized by using the Cube-Pruning algorithm (Huang & Chiang, 2007) built into Moses. This (under-advertised) feature of Moses, in combination with in-memory storage, allows for increases in the average translation speed by an order of magnitude, compared to standard settings. Small losses in translation quality (fractions of one BLEU point) due to Cube-pruning are acceptable.

2.4 Architecture

The training and decoding services are currently hosted on two cloud servers running instances of

Linux. We have chosen to employ two Amazon Elastic Clouds servers with the so-called "High-Memory Quadruple Extra Large Instance" configuration.[6]

We first trained our models on one server and used the other one as a decoding server. However, even after optimization of the memory footprint, one server alone was not enough to host all the language pairs[7]. Therefore, we decided to run some decoders on one server (en-es, en-ar, en-zh) and all the others on the second one. The first server can be used to train new models while still being used in production as a translation server.

We run a web application server (Apache Tomcat) that connects to a translation server (in-house developed Java-RMI) that queues requests and selects the next available Moses engine to handle translation.

2.5 Translating interface

In addition to a raw translation service (basic "Moses" Unix command), for final users we developed a web graphical interface, together with a web service, to help users in their daily work. The graphical user interface is developed as a Java-Jsf 2.0 web application running on Tomcat.

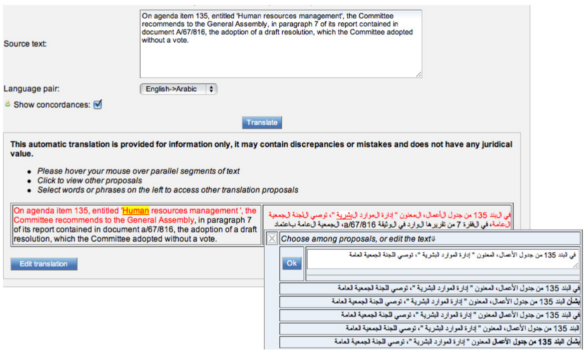

Figure 1: Graphical user interface example from English into Arabic. Note that parallel sentences and translated words are highlighted; users can select alternative translations for each segment

This interface allows users to cut and paste any text and get the corresponding translation. Any click on a segment will display alternative translations (up to 24) and users can select any alternative or write their own version. Each word gets highlighted together with its translation (same for segments).

[6] Memory: 64Gb, 8 cores, 400 Gb disk,

[7] For every language pair the current implementation uses several Moses instances that do not share resources

In parallel, a more general web service allows for an easy integration with any other application, the client provides the source text, the language pair and receives an XML response containing various pieces of information, including alternative translations.

Example: http://my.server/translateXml.jsf?langpair=en-es&q=green%20economy

<xml>economía ecológica
<raw_mt>economía ecológica</raw_mt>
<alternatives>
<alt n="0">economía ecológica</alt>
<alt n="1">economía verde</alt>
<alt n="2">en favor de una economía verde</alt>
...
</alternatives>
</xml>

This web service has been used to integrate the system with two computer-assisted translation tools, one commercial (SDL Trados Studio 2011) and one internally developed (eLuna) that will be used as the main translation interface for all UN translators in all duty stations after 2014.

2.6 Usage

Immediately after training, the systems were made available to all translators, before a human evaluation was conducted.

We monitored the use of the web interface during four months, from 1st of December 2012 up to 31st of March 2013 (121 days). Table 3 gives a good indication of the server usage for given language pairs.

Table 3: Usage statistics over 121 days, average segments submitted for translation (together with the number of words of the source language). Numbers are given per day (including non working days).

	# segments/day	# words/day
ar-en	2.32	55.19
en-ar	3'124.51	55'659.16
en-es	701.98	13'347.09
en-fr	82.60	2'024.41
en-ru	468.73	7'459.43
en-zh	491.85	11'270.80
es-en	29.69	701.61
fr-en	1.66	30.11
ru-en	2.18	42.54
zh-en	0.45	(estimated)11.76

The average daily amount of translated words for the English-Spanish language pair at the UN is estimated at 42,000 (also including non-working days). The amount of words submitted to TAPTA4UN by users during this period accounted for more than one quarter of that total.

3 Evaluation

In order to have comparable metrics across languages we have chosen to work on a test set where every single sentence is available in the 6 languages.

We took new documents published between November and December 2012 (after our training). Five bitexts were exported containing between 163 and 185 documents.

We applied our sentence-alignment tool to extract parallel segments from each bitext, and ended up with 1464 segments available in 6 languages. Then the following empirically set filtering criteria were applied:

- English segment length is between 40 and 600 characters
- Translation has between 33% and 133% of the English length (except Chinese: between 12% and 53% of English length)
- We filtered out any segment starting with a number as we may have paragraph numbers missing in one of the 6 segments (ie. "6. introduction" and "introduction").

As a result, 1175 segments were kept as part of our test set. This test set will be used for our automatic evaluation (computing BLEU/ METEOR for any language pair) as well as for a human evaluation.

3.1 Automatic evaluation

We use the common measure BLEU and METEOR (Denkowski and Lavie, 2011) to compare the MT output with the reference translation. Table 4 sums up the BLEU/METEOR scores per language pair. For comparison, the same input text has been submitted to two freely available translation tools: Google Translate and Bing Translator.

Table 4: Automatic evaluation, computed BLEU/METEOR scores for 1175 segments, by language pair

Lang-pair	Our SMT		Google		Bing	
	Bleu	Meteor	Bleu	Meteor	Bleu	Meteor
ar-en	55.25	43.79	40.71[8]	40.10[8]	51.17	43.22
en-ar	44.10	18.72	33.74	17.13	28.94	15.91
en-es	61.81	76.69	53.39	71.24	46.86	67.54
en-fr	51.23	67.11	45.58	62.39	42.19	60.07
en-ru	50.85	66.53	39.67	58.22	38.96	58.22
en-zh	43.17	n/a	34.16	n/a	32.77	n/a
es-en	60.32	47.87	52.54	43.95	49.18	43.79
fr-en	53.36	44.11	46.46	40.18	43.39	39.85
ru-en	58.56	45.11	47.71	42.10	47.09	41.70
zh-en	42.31	39.71	36.55	37.99	30.60	34.64

Our SMT system clearly outperforms the two other engines in all the language pairs. This does not come as a surprise as Google and Bing are general purpose MT tools while our own system is trained exclusively on in-domain documents.

The first users of our prototypes also came to the conclusion that the system was good enough to accelerate translation through post edition. We decided to proceed with a more systematic evaluation of translators' impressions of the tool.

3.2 Human evaluation

The systems were made available to all UN translators in New York and other duty stations for several months before the human evaluation. Translators were free to use them as they wished. Some patterns of usage emerged, where en-ar, en-es and en-ru, in decreasing order, were the most used.

During the human evaluation, every test segment was translated with the corresponding system. For each language pair from English into one other official language, three translators were assigned by the Chief of each Service (one P5, Senior Reviser, one P4, Reviser, and one P3, Translator). Due to the smaller size of the English Translation Service and the high number of systems to be evaluated by its staff (from each official language into English), only two English translators were assigned per language combina-

[8] Despite many attempts, we were not able to translate some Arabic segments into English with Google (for these sentences only empty results were returned). This evaluation was done on the remaining 1022 segments for ar-en.

tion. Two verbatim reporters (a separate category of translators tasked to prepare and translate *procès-verbaux* of main UN bodies) requested to participate in the evaluation for en-ar and en-ru. The participation of these additional translators explains the varying number of evaluators across the language combinations. All evaluators were asked to judge the output of the system by their degree of fluency and adequacy (defined as metrics in Denkowski & Lavie, 2010).

Table 5 summarizes the result of the evaluation. The average adequacy was measured on a scale from 5 (all the input information has been transferred to the translation) to 1 (none of the original information has been transferred), and the average fluency from 5 (flawless) to 1 (incomprehensible). We also wanted to assess the inter-evaluator-agreement; for this purpose, the "average maximum divergence" score was computed, which measures the maximum difference of one metric – fluency/adequacy – between all evaluators). We decided to set a threshold of 1.3 in order to discard segments where the evaluators disagreed too much on their evaluation. Then we computed "filtered adequacy" and "filtered fluency" together with the average maximum divergence score (which was used as a metric of how well the evaluators agreed during this exercise).

3.3 Discussion

Given the huge structural differences among languages, it is difficult to conduct a meaningful comparative analysis. Also, cultural attitudes towards MT might explain the gap between some human evaluations and the corresponding BLEU scores.

As expected, the results for en-es were the highest among all language combinations. The evaluators were the same translators as in our previous experiment and they were familiar with the system as they use it regularly in their daily work. This language combination was the first to be implemented and MT is perceived by Spanish translators as yet another translation tool.

Scores for en-ar and en-ru were surprisingly high and are correlated with usage statistics that show that the en-ar system is the most used, while en-ru has experienced a consistent increase in the number of users. Being inflected languages, the scores for the human evaluation of Arabic and Russian show a considerable gap between adequacy and fluency. The need to make many small corrections across each segment significantly increases the necessary post-editing time for such languages. This, in turn, may have generated a worse impression in reviewers, thus lowering the overall fluency score.

Table 5: Human evaluation results, indicating, for each language pair, the number of evaluators, the average adequacy, the average fluency, the filtered adequacy, the filtered fluency, the number of filtered segments, the average maximum disagreement score and the MT BLEU score

		Scores		*Filtered scores*				
Language pair	*# evaluators*	*Adequacy*	*Fluency*	*Filtered adequacy*	*Filtered fluency*	*# Lines after filtering*	*Average Max divergence*	*Machine translation BLEU*
ar-en	2	3.47	3.00	3.50	3.01	1094	0.30	55.25
en-ar	4	3.30	3.20	3.74	3.61	662	1.37	44.10
en-es	3	4.01	3.52	4.17	3.70	894	1.04	61.81
en-fr	3	3.34	2.95	3.37	3.04	711	1.39	51.23
en-ru	4	3.84	3.35	3.97	3.57	813	1.23	50.85
en-zh	3	2.00	1.44	1.88	1.31	896	1.23	43.17
es-en	3	3.97	3.35	4.20	3.59	576	1.56	60.32
fr-en	2	3.97	3.26	3.99	3.24	933	1.00	53.36
ru-en	2	4.08	3.48	4.13	3.54	1016	0.84	58.56
zh-en	n/a	n/a	n/a	n/a	n/a	n/a	n/a	42.31

It should be stressed than most users for these language combinations use our system through SDL Trados Studio 2011[9]. We have found that UN translators tend to use MT more and more readily when it is integrated in a computer-assisted translation environment. Also, one factor for the popularity of our system in these two language combinations is the fact that MT facilitates the switch between alphabets.

The evaluation scores for en-fr and en-zh were quite low, which was unexpected when considering that both systems got solid BLEU scores that were better than Google and Bing MT systems (51.23 and 43.17, respectively, for our system). At the same time, the server statistics for en-fr show that the system is seldom used by translators.

For the evaluation of the systems with English as target language, it is possible that some of the reviewers may have applied standard editorial criteria (i.e. those criteria applied to human translators, which are fairly strict) to the test, which would explain the extremely high amount of sentences filtered out from the ar-en, ru-en and fr-en tests.

For the future, it would be interesting to come up with a way to factor in cultural attitudes towards MT across the different UN translation services and across duty stations.

Conclusion and future work

The tool is currently used in production and the server receives on average 7000 segment translation requests per day. We receive positive feedback from professional translators.

We are considering putting in place an automatic procedure to retrain regularly with up-to-date exports of parallel text. We would also like to retrain the system using other UN duty stations as domains.

When translating from/to Russian, Chinese or Arabic, the tool does not translate unknown proper names. An interesting development would be to investigate automatic transliteration.

We would like to implement some post-processing procedure to verify terminology compliance against the UN mandatory terminology database.

[9] http://www.sdl.com/products/sdl-trados-studio/

Acknowledgement

Many thanks to our evaluators: Manar Badran, Maria Barros, Kieran Burns, Imane Cherif, Isabelle Delatour, Kathy Durnin, Rosario Fernandez, Valerie Fonseca, Ann Getzinger, Benoît Guyard, Adnan Jarrar, Megin Jimenez, Janine Lissandre, Nadia Lotfi, Denis McAuley, Halina Palyvian, Tracey Parsons, Ekaterina Pugacheva, Carla Raffo, Dmitry Sapsay, Sally Shaw, Chun Shi, Valérie Troude, Vladimir Tsivilka, Susan Walters, Laura Wolfson, Gaojing Xu and Jiansheng Yang, and the Chiefs of the Translation Services who authorized their participation in this evaluation: Latifa Amine Saint-Roch, Alassane Diatta, Xuesong Ma, Maria Nóbrega, Michael Ten-Pow and Nikolai Pakhomov, as well as Ricardo Pérez. Special thanks to Philip Dale, who edited the first version of this paper.

Thanks to Paul Halfpenny for the proofreading, special thanks to Michal Ziemski, Hussein Ghaly, Yaroslav Kurennoy, Laurent Gottardo and Abderrezak Smahi who all contributed to the technical improvement of the tool.

References

Denkowski, Michael and Alon Lavie. 2011. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. Proc of WMT 2011, Edinburgh, UK. pp. 85–91

Denkowski, Michael and Alon Lavie. 2012. Challenges in Predicting Machine Translation Utility for Human Post-Editors. Proc of AMTA 2012. San Diego, CA. 10pp.

Federico, Marcello, Nicola Bertoldi, Mauro Cettolo 2008. IRSTLM: an Open Source Toolkit for Handling Large Scale Language Models, Proc. of Interspeech 2008, Brisbane, Australia, pp. 1618-1621

Franz, Alex, Shankar Kumar, Thorsten Brants, 2013, 1993-2007 United Nations Parallel Text, Linguistic Data Consortium, Philadelphia.

Gao, Qin & Stephan Vogel. 2008. Parallel Implementations of Word Alignment Tool. Proc of ACL 2008 HLT: Software Engineering, Testing, and Quality Assurance Workshop, Columbus, Ohio, USA. pp. 49-57

Johnson, Howard, Joel Martin, George Foster, Roland Kuhn. 2007. Improving Translation Quality by Discarding Most of the Phrasetable, proc of EMNLP-CoNLL 2007: 967-975

Junczys-Dowmunt, Marcin 2012. Phrasal Rank-Encoding: Exploiting Phrase Redundancy and Translational Relations for Phrase Table Compression. The Prague Bulletin of Mathematical Linguistics, vol. 98. pp. 63-74

Khalilov, Maxim & Rahzeb Choudhury, 2011, Building English-Chinese and Chinese-English MT Engines for the Computer Software Domain, proc. of EAMT 2012, Trento, Italy

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris C. Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation, proc of ACL 2007. Morristown, NJ, USA, pp. 177-180.

Koehn, Phillip, Alexandra Birch & Ralf Steinberger, 2009, 462 Machine Translation Systems for Europe, proc of MT Summit XII, Ottawa, Ontario, Canada; pp. 65-72.

Huang, Liang, David Chiang, 2007, Forest Rescoring: Faster Decoding with Integrated Language Models, proc of ACL 2007, p. 144.

Ma, Xiaoyi. 2006. Champollion: A Robust Parallel Text Sentence Aligner, proc of LREC2006, pp. 489-492

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu, 2002, BLEU: a Method for Automatic Evaluation of Machine Translation, proc. of ACL 2002, pp. 311-318

Pouliquen, Bruno, Christophe Mazenc, Cecilia Elizalde, & José García-Verdugo, 2012, Statistical Machine Translation Prototype using UN Parallel Documents, proc of EAMT 2012, Trento, Italy, May 28-30 2012, pp.12-19

Pouliquen, Bruno, Christophe Mazenc & Aldo Iorio, 2011, Tapta: a User-Driven Translation System for Patent Documents based on Domain-Aware Statistical Machine Translation, proc of EAMT 2011, , Leuven, Belgium; 30-31 May 2011, pp. 5-12

Soudi, Abdelhadi, Ali Farghaly, Günter Neumann & Rabih Zbib (eds.) 2012, Challenges for Arabic Machine Translation. John Benjamins publishing

Zhechev, Ventsislav, 2012, Machine Translation Infrastructure and Post-editing Performance at Autodesk, proc of AMTA 2012 Workshop on Post-editing Technology and Practice (WPTP ’12), San Diego, CA, USA. pp. 87-96