Fast production of *ad hoc* translation tables using the sampling-based method

Junjun LEE　　Yves LEPAGE
Graduate School of Information, Production and Systems, Waseda University
{junjun_lee99@akane, yves.lepage@aoni}.waseda.jp

Abstract

This paper deals with the production of *ad hoc* translation tables from the sentences to be translated in addition to the training data. This way of doing is different from the standard phrase-based statistical machine translation approach (PB-SMT), but similar to some approaches in example-based machine translation (EBMT).

We compare the time necessary to obtain translation tables using the sampling-based technique in the PB-SMT framework and the EBMT framework so as to obtain the same translation quality. In addition, we conduct a comparison of the sizes of the translation tables obtained in both ways as well as a comparison of their contents.

1 Introduction

In phrase-based statistical-based machine translation (PB-SMT), one compiles a translation table (TT) from a bilingual corpus, the training data, beforehand. This may be seen as a type of *eager learning* where knowledge is built as soon as data is fed into the system. Translation tables are usually large (millions of entries) and long to compile (usually hours).

By opposition, in some approaches of example-based machine translation (EBMT), one extracts *ad hoc* translation information from a bilingual corpus during the translation of the test set. This may be seen as a type of *lazy learning* where specific knowledge is built only when the system needs to perform some specific task. This way of doing is usually faster and leads to a smaller translation table, but it usually results in a lesser translation quality.

Recently the sampling-based technique for translation table (TT) production has been proposed. It can produce translation tables ready-to-use in the PB-SMT framework. Due to its mechanism, it is easy to modify so as to be able to produce translation tables on-demand as in the EBMT approach.

The paper is divided as follows. Section 2 gives an overview of the sampling-based method to produce translation tables and introduces the simple modification that allows for the on-demand production of translation tables. Section 3 describes the experimental data and the various results obtained. In particular, we compare the time necessary to obtain translation tables using the sampling-based technique in the PB-SMT frame and the EBMT frame so as to obtain the same translation quality. In addition, we conduct a comparison of the sizes of the translation tables obtained in both frames as well as a comparison of their contents.

2 Sampling-based TT production method and *ad hoc* TTs

2.1 Anymalign

The standard PB-SMT approach requires translation tables. Each line in a translation table is made out of four pieces of information: source language word sequence, target language word sequence, lexical weights [5] in both directions (source knowing target and target knowing the source), translation probabilities in both directions too, and a phrase penalty.

The quality of translation tables is crucial for the quality of translation. The most widely used way of producing phrase translation tables is to use MGIZA++ (IBM models and HMM-model in combination with the Moses toolkit These tools belong to the estimative trend [1].

We use a different approach for the production of phrase translation tables: the sampling-based approach [6]. It is available as a free open-source tool called Anymalign.[1] Being in line with the associative alignment trend [2], it is much simpler than the models implemented in MGIZA++, and better suited to our purpose of producing *ad hoc* TTs.

In sampling-based TT production, only those sequences of words that appear exactly in the same sentences of the corpus are considered for alignment. The key idea is to produce more candidates by artificially reducing the size of the input corpus, *i.e.,*

[1] http:www.limsi.fr/Individu/alardill/anymalign/

many subcorpora of small sizes are obtained by sampling and processed one after another. Indeed, the smaller a subcorpus, the less frequent its words, and the more likely they are to share the same distribution. The subcorpus selection process is guided by a probability distribution that ensures a proper coverage of the input parallel corpus. Eventually, the list of alignments is turned into a full-fledged translation table by calculating the necessary features for each alignment from the number of times each sequence of words was obtained from subcorpora.

One important feature of the sampling-based TT production method is that it is *anytime* in essence: the number of random subcorpora to be processed is not set in advance, so the alignment process can be interrupted at any moment. Contrary to many approaches, *quality* is not a matter of time, however *quantity* is: the longer the aligner runs (i.e. the more subcorpora processed), the more alignments produced, and the more reliable their associated translation probabilities. The experiments reported in this paper make use of the anytime feature of Anymalign and of the possibility of allotting time freely.

2.2 *Ad hoc* translation tables

The main reason to produce *ad hoc* translation tables is to reduce the size of the TTs used during translation. Basically, it suffices to filter out those entries in the TT that would never be used during translation in the general case, or that will not be used in the particular case of a particular text. For instance, Johnson et al. [3] showed how to filter out unnecessary entries in the TT before any translation by relying on some confidence measure. When the data from which to extract the TTs reach the terabytes, compiling only a partial but sufficient TT becomes unavoidable [7]. This way of doing is indeed characteristic of the EBMT approach, as instantiated in the Cunei system [8] in which only the necessary entries for the sentences to translate are produced.

We compile *ad hoc* TTs according to a naïve approach: filter out any entry in the TT that contains a source word not present in the text to translate. This is made possible by the specific feature of the sampling-based method mentioned in Section 2.1: its *anytime* nature ensures the quality of TT entries over production time. Time does not influence the final quality of the entries, only their quantity. Consequently, after the necessary quantity of entries has been produced for the text to translate (which is achieved in seconds), producing new entries that will merely be filtered out is simply without object.

The necessary modifications to Anymalign thus reduced to:

- form the set of words present in the text to translate. This is performed before subcorpora sampling starts;
- check for the absence of a source word from the above set in any source sequence as soon as it is produced. This is done while processing each subcorpus.

As explained in Section 2.1, the computation of the numerical features is performed after subcorpora sampling and the production of the entries of the TT.

As a result, the data stored during the TT production process is by far much smaller than in the general case and results in a gain in time. However, there is an obvious danger that the entries produced would not permit the accurate computation of the numerical features afterwards. The following sections that report our experiments will inspect these points.

Table 1: Translation quality in BLEU scores for various amounts of time using the standard and the modified versions of Anymalign. The last columns give the sizes of the translation tables.

Standard		
Time	BLEU	Size
10s	0.065	1,562
20s	0.087	6,779
30s	0.101	7,598
10min	0.161	107,791
1h	0.182	398,075
2h	0.190	625,357
3h	0.196	776,086
5h	0.194	1,042,389

Modified		
Time	BLEU	Size
10s	0.081	2,113
20s	0.098	3,728
30s	0.105	5,474
40s	0.108	5,523
50s	0.108	6,222
1min	0.113	7,944
2min	0.134	14,836
3min	0.143	20,477
4min	0.148	26,717
5min	0.153	31,384
10min	0.162	52,122
2h	0.185	201,527

3 Experiments

We now turn to the assessment of our proposed technique using relevant parameters:

- time of TT production (in seconds);
- sizes of TTs (in number of entries);

Table 2: Times and sizes required for both ways of production translation tables, standard (std.) and modified (mod.) obtained by linear regression for the same BLEU scores. The last figures on each line give the reduction in relative size of the translation table that can be obtained using the modified version of Anymalign.

BLEU	times (sec.)		sizes (# of lines)	
	std.	mod.	std.	mod.
0.08	17	10	15,060	1,985 (−87%)
0.09	22	16	19,112	2,832 (−85%)
0.10	29	23	25,211	3,761 (−85%)
0.11	113	55	76,481	7,952 (−90%)
0.12	209	81	121,183	10,059 (−92%)
0.13	306	110	163,630	13,448 (−92%)
0.14	402	161	189,666	17,047 (−91%)
0.15	499	269	230,468	26,167 (−89%)
0.16	595	537	264,959	40,111 (−85%)
0.17	3951	2947	862,245	159,869 (−81%)
0.18	6373	5880	1,126,768	250,169 (−78%)
0.19	8490	7724	1,315,786	295,643 (−78%)
0.20	17947	8269	1,943,894	311,226 (−84%)

- quality of translation (in BLEU scores as is standard practice).

Indeed, quality of translation is a non-adjustable parameter. We will thus compare time and sizes for fixed values of translation quality.

3.1 Data

The data used in the experiments come from the German-Spanish part of the Europarl corpus [2] [4]. We used 300,000 pairs of aligned sentences for the training data (8,211,384 words in German and 9,039,118 in Spanish). The tuning data consists of 500 pairs of aligned sentences. The test set comprises of 1,000 German sentences (27,264 words). The references are the corresponding translations of the test set sentences in the Spanish data (1 reference per test sentence).

3.2 Experimental setting

We use the Moses toolkit to produce machine translation systems. The procedure is as follows. We first produce a translation table using either Anymalign in its standard distribution or our proposed modified version. We then apply tuning with the Moses toolkit, translate the test set with the Moses decoder, and then compute BLEU scores using the mteval toolkit.

The experiments were run on a computer with 2.5 GB Memory and a 6 core processor at 2.80 GHz. In average, tuning takes about 3 to 4 hours and decoding about 1 hour.

[2] `http://www.statmt.org/europarl/`

3.3 Experimental results

We run the standard and modified versions of Anymalign for different amounts of time. Table 1 gives the translation qualities obtained for various amounts of time.

From the previous results given in Table 1 we deduced the results given in Table 2 by linear regression, *i.e.*, we assumed that the increase in quality was linear between two consecutive times.

In a further step, we inspected the content of the translation tables to understand why the ad hoc translation tables can properly translate the data set up to the level of the standard configuration. The explanation lies in the content. Table 3 gives the size of the entries in common between both configurations as well as a comparison in terms of average and standard deviation of the numerical features associated with each entry (lexical weights and translation probabilities) for the common part.

4 Conclusion

This paper has dealt with the production of *ad hoc* translation tables from the training data and from the sentences to be translated. The technique used was the sampling-based TT production method as implemented in the freely downloadable software tool Anymalign.

It has been shown that a simple modification of the tool can lead to the production of *ad hoc* translation tables in only half of the time usually needed during training (but not in tuning in our present experimental setting) and with no loss in translation quality as desired.

By conducting a comparison of the sizes of the

Table 3: Comparison of the contents of the standard (std.) and modified (mod.) version of Anymalign for the same BLEU scores: number of entries in common and comparison of the feature differences.

BLEU score	Common part in % of		Average ± standard deviation of difference between std. and mod.			
	std. TT	mod. TT	lw($s\|t$)	lw($t\|s$)	p($s\|t$)	p($t\|s$)
0.08	6 %	46 %	$2 \times 10^{-7} \pm 1 \times 10^{-5}$	$2 \times 10^{-7} \pm 1 \times 10^{-5}$	0.07 ± 0.19	0.08 ± 0.20
0.09	7 %	47 %	$3 \times 10^{-7} \pm 1 \times 10^{-5}$	$8 \times 10^{-9} \pm 8 \times 10^{-6}$	0.07 ± 0.18	0.08 ± 0.20
0.10	6 %	43 %	$2 \times 10^{-7} \pm 1 \times 10^{-5}$	$2 \times 10^{-7} \pm 1 \times 10^{-5}$	0.05 ± 0.17	0.06 ± 0.19
0.11	5 %	48 %	$1 \times 10^{-6} \pm 1 \times 10^{-4}$	$5 \times 10^{-7} \pm 2 \times 10^{-5}$	0.06 ± 0.18	0.08 ± 0.19
0.12	4 %	54 %	$1 \times 10^{-6} \pm 1 \times 10^{-4}$	$9 \times 10^{-7} \pm 8 \times 10^{-5}$	0.07 ± 0.18	0.08 ± 0.20
0.13	4 %	54 %	$1 \times 10^{-6} \pm 9 \times 10^{-5}$	$1 \times 10^{-7} \pm 1 \times 10^{-5}$	0.06 ± 0.18	0.09 ± 0.21
0.14	5 %	56 %	$7 \times 10^{-7} \pm 8 \times 10^{-5}$	$3 \times 10^{-7} \pm 2 \times 10^{-5}$	0.06 ± 0.18	0.09 ± 0.21
0.15	6 %	54 %	$2 \times 10^{-7} \pm 2 \times 10^{-5}$	$9 \times 10^{-7} \pm 1 \times 10^{-4}$	0.05 ± 0.17	0.08 ± 0.21
0.16	8 %	51 %	$3 \times 10^{-7} \pm 6 \times 10^{-5}$	$1 \times 10^{-7} \pm 4 \times 10^{-5}$	0.04 ± 0.15	0.08 ± 0.21
0.17	11 %	58 %	$1 \times 10^{-7} \pm 3 \times 10^{-5}$	$8 \times 10^{-8} \pm 2 \times 10^{-5}$	0.02 ± 0.10	0.07 ± 0.19
0.18	13 %	58 %	$9 \times 10^{-8} \pm 2 \times 10^{-5}$	$5 \times 10^{-8} \pm 3 \times 10^{-5}$	0.01 ± 0.09	0.06 ± 0.18
0.19	13 %	59 %	$8 \times 10^{-8} \pm 2 \times 10^{-5}$	$1 \times 10^{-7} \pm 4 \times 10^{-5}$	0.01 ± 0.08	0.06 ± 0.18
0.20	11 %	68 %	$4 \times 10^{-8} \pm 2 \times 10^{-5}$	$9 \times 10^{-8} \pm 2 \times 10^{-5}$	0.02 ± 0.09	0.06 ± 0.18

translation tables obtained in both ways, we have shown that a reduction of around 85% of the translation table size may be obtained for the same translation quality.

A comparison on the contents of the translation tables obtained in both ways showed that more than half of the common part of the translation tables obtained exactly the same numerical features (lexical weights and translation probabilities) while the rest was different by only a small amount.

References

[1] P. Brown, J. Lai, and R. Mercer. Aligning sentences in parallel corpora. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL'91)*, pages 169–176, Berkeley (California, USA), June 1991. URL `http://www.aclweb.org/anthology/P91-1022`.

[2] William Gale and Kenneth Church. Identifying word correspondences in parallel texts. In *Proceedings of the fourth DARPA workshop on Speech and Natural Language*, pages 152–157, Pacific Grove, feb 1991. URL `http://www.aclweb.org/anthology/H/H91/H91-1026.pdf`.

[3] Howard Johnson, Joel D. Martin, George F. Foster, and Roland Kuhn. Improving translation quality by discarding most of the phrasetable. In *EMNLP-CoNLL'07*, pages 967–975, 2007. URL `http://www.aclweb.org/anthology-new/D/D07/D07-1103.pdf`.

[4] Philipp Koehn. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of the tenth Machine Translation Summit (MT Summit X)*, pages 79–86, Phuket, September 2005. URL `http://www.mt-archive.info/MTS-2005-Koehn.pdf`.

[5] Philipp Koehn, Franz J. Och, and Daniel Marcu. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 48–54, Edmonton, may 2003. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology-new/N/N03/N03-1017.pdf`.

[6] Adrien Lardilleux and Yves Lepage. Sampling-based multilingual alignment. In *International Conference on Recent Advances in Natural Language Processing (RANLP 2009)*, pages 214–218, Borovets, Bulgaria, sept 2009.

[7] Adam Lopez. Tera-scale translation models via pattern matching. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 505–512, Manchester, UK, August 2008. Coling 2008 Organizing Committee. URL `http://www.aclweb.org/anthology/C08-1064`.

[8] Aaron B. Phillips and Ralf D. Brown. Cunei machine translation platform: System description. In Mike L. Forcada and Andy Way, editors, *Proceedings of the 3rd Workshop on Example-Based Machine Translation (EBMT 3)*, novembre 2009.