Thot: a Toolkit To Train Phrase-based Statistical Translation Models*

Daniel Ortiz-Martínez
Dpto. de Sist Inf. y Comp.
Univ. Politéc. de Valencia
46071 Valencia, Spain
dortiz@dsic.upv.es

Ismael García-Varea
Dpto. de Informática
Univ. de Castilla-La Mancha
02071 Albacete, Spain
ivarea@info-ab.uclm.es

Francisco Casacuberta
Dpto. de Sist Inf. y Comp.
Univ. Politéc. de Valencia
46071 Valencia, Spain
fcn@dsic.upv.es

Abstract

In this paper, we present the Thot toolkit, a set of tools to train phrase-based models for statistical machine translation, which is publicly available as open source software. The toolkit obtains phrase-based models from word-based alignment models; to our knowledge, this functionality has not been offered by any publicly available toolkit. The Thot toolkit also implements a new way for estimating phrase models, this allows to obtain more complete phrase models than the methods described in the literature, including a segmentation length submodel. The toolkit output can be given in different formats in order to be used by other statistical machine translation tools like Pharaoh, which is a beam search decoder for phrase-based alignment models which was used in order to perform translation experiments with the generated models. Additionally, the Thot toolkit can be used to obtain the best alignment between a sentence pair at phrase level.

1 Introduction

Since the beginning of the 90's interest in the statistical approach to machine translation (SMT) has greatly increased due to the successful results obtained for typical restricted-domain translation tasks.

The translation process can be formulated from a statistical point of view as follows: A source language string $f_1^J = f_1 \dots f_J$ is to be translated into a target language string $e_1^I = e_1 \dots e_I$. Every target string is regarded as a possible translation for the source language string with maximum a posteriori probability $Pr(e_1^I|f_1^J)$. According to Bayes' decision rule, the target string $\hat{e}_1^I$ that maximizes[1] the product of both the target language model $Pr(e_1^I)$ and the string translation model $Pr(f_1^J|e_1^I)$ must be chosen. The equation that models this process is:

$$\hat{e}_1^I = \arg\max_{e_1^I} \{Pr(e_1^I) \cdot Pr(f_1^J|e_1^I)\} \qquad (1)$$

Different *translation models* (TMs) have been proposed depending on how the relation between the source and the target languages is structured; that is, the way a target sentence is generated from a source sentence. This relation is summarized using the concept of *alignment*; that is, how the words of a pair of sentences are aligned to each other. Different *statistical alignment models* (SAMs) have been proposed. The well-known IBM and HMM alignment models were proposed in (Brown et al., 1993) and in (Ney et al., 2000) respectively. All these models fall into the category of single-word-based (SWB) SAM. Recent research in the field has demonstrated that phrase-based or context-based translation models outperform the first propose word-based statistical translation models (Brown et al., 1993). Since then, some useful tools have been made to help researchers in the field improve their own machine translation systems. These tools range from software for training single word-based translation models (as the Giza++ software (Och, 2000)) and some specific word-based decoders, to a recently available phrase-based decoder, like Pharaoh (Koehn, 2003). For SMT software, a tool to train phrase-based is essential in order to continue the research. In this paper we presented a publicly available toolkit to train phrase-based SMT models. Different models that deal with structures or phrases instead of single words have also been proposed: the

* This work has been partially supported by the Spanish project TIC2003-08681-C02-02, the *Agencia Valenciana de Ciencia y Tecnología* under contract GRUPOS03/031, the *Generalitat Valenciana*, and the project HERMES (Vicerrectorado de Investigación - UCLM-05)

[1] Note that the expression should also be maximized by I; however, for the sake of simplicity we suppose that it is known.

syntax translation models are described in (Yamada and Knight, 2001) , alignment templates are used in (Och, 2002), and the alignment template approach is re-framed into the so-called *phrase based translation* (PBT) in (Tomás and Casacuberta, 2001; Marcu and Wong, 2002; Zens et al., 2002; Koehn et al., 2003). In (Venugopal et al., 2003), two methods of phrase extractions are proposed (based on source n-grams and HMM alignments respectively). They improve a translation lexicon, instead of defining a phrase-based model, which is also used within a word-based decoder. In the same line, a method to produce phrase-based alignments from word-based alignments is proposed in (Lambert and Castell., 2004).

2 Phrase Based Translation

One important disadvantage of the SWB SAMs is that contextual information is not taken into account. Another important disadvantage of the SWB models (and specifically, of the widely-used IBM models), consists of the definition of alignment as a function. This implies that a source word can only be aligned to zero or one target word (see (Brown et al., 1993)).

One way to solve these disadvantages consists of learning translations for whole phrases instead of single words, where a phrase is defined as a consecutive sequence of words.

PBT can be explained from a generative point of view as follows (Zens et al., 2002):

1. The source sentence f_1^J is segmented into K phrases ($\tilde{f}_1^K$).
2. Each source phrase $\tilde{f}_k$ is translated into a target phrase $\tilde{e}$.
3. Finally, the target phrases are reordered in order to compose the target sentence $\tilde{e}_1^K = e_1^I$.

2.1 Phrase-based models

In PBT, it is assumed that the relations between the words of the source and target sentences can be explained by means of the hidden variable $\tilde{\mathbf{a}} = \tilde{a}_1^K$, which contains all the decisions made during the generative story.

$$\begin{aligned} Pr(f_1^J|e_1^I) &= \sum_{\tilde{\mathbf{a}}} Pr(\tilde{\mathbf{a}}, \tilde{f}_1^J|\tilde{e}_1^I) \\ &= \sum_{\tilde{\mathbf{a}}} Pr(\tilde{\mathbf{a}}|\tilde{e}_1^I) Pr(\tilde{f}_1^J|\tilde{\mathbf{a}}, \tilde{e}_1^I) \quad (2) \end{aligned}$$

Different assumptions can be made from the previous equation. For example, in (Tomás and Casacuberta, 2001) the following model is proposed:

$$p_\theta(f_1^J, e_1^I) = \alpha(e_1^I) \sum_{\tilde{\mathbf{a}}} \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_{\tilde{a}_k}) \quad (3)$$

where $\tilde{a}_k$ notes the index of the source phrase $\tilde{e}$ that is aligned with the k-th target phrase $\tilde{f}_k$ and that all possible segmentations have the same probability. In (Zens et al., 2002), it also is assumed that the alignments must be monotonic. This led us to the following equation:

$$p_\theta(f_1^J|e_1^I) = \alpha(e_1^I) \sum_{\tilde{\mathbf{a}}} \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k) \quad (4)$$

In both cases the model parameters that have to be estimated are the translation probabilities between phrase pairs ($\theta = \{p(\tilde{f}|\tilde{e})\}$).

2.2 Model estimation

As mentioned above, PBTs are based on a set of bilingual phrases that must be previously obtained in order to perform the translation.

Three ways of obtaining the bilingual phrases from a parallel training corpus are described in (Koehn et al., 2003):

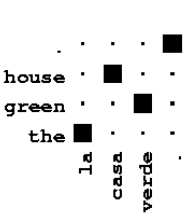
1. From word-based alignments.
2. From syntactic phrases (see (Yamada and Knight, 2001) for more details).
3. From sentence alignments, by means of a joint probability model (see (Marcu and Wong, 2002)).

In this paper, we focus on the first method, in which the bilingual phrases are extracted from a bilingual, word-aligned training corpus. The extraction process is driven by an additional constraint: the bilingual phrase must be consistent with its corresponding word alignment matrix A as shown in equation (5) (which is the same given in (Och, 2002) for the alignment template approach).

$$\begin{aligned} \mathcal{BP}(f_1^J, e_1^I, A) = \{(f_j^{j+m}, e_i^{i+n} : \forall (i', j') \in A : \\ j \leq j' \leq j+m \iff i \leq i' \leq i+n\} \quad (5) \end{aligned}$$

See Figure 1 for a word alignment matrix example and its corresponding set of consistent, bilingual phrases. The word alignment matrices are supposed to be manually generated by linguistic experts; however, due to the cost of such generation, in practice they are obtained using SWB

alignment models. This can be done by means of the Giza++ toolkit (Och, 2000), which generates word alignments for the training data as a by-product of the estimation of IBM models.



source phrase	target phrase
La	the
casa	house
verde	green
casa verde	green house
La casa verde	the green house
.	.
casa verde .	green house .
La casa verde .	the green house .

Figure 1: Set of consistent bilingual phrases (right) given a word alignment matrix (left).

Since word alignment matrices obtained via the estimation of IBM models are restricted to being functions (as we mentioned at the beginning of this section), some authors (Och, 2002) have proposed performing operations between matrices in order to obtain better alignments. The common procedure consists of estimating IBM models in both directions and performing different operations with the resulting alignment matrices such as union or intersection.

Another negative consequence of the word-alignment matrix generation using IBM model information is the appearance of words that are not aligned into the matrices (the so-called *spurious* and *zero fertility* words, see (Brown et al., 1993)). These special words are not taken into account by equation (5) and must be considered separately. A simple way to solve this problem consists of putting the words that are not aligned at the right or at the left of phrases composed with aligned words. This solution generates a greater number of bilingual phrases.

Once the phrase pairs are collected, the phrase translation probability distribution is calculated by relative frequency (RF) estimation as follows:

$$p(\tilde{f}|\tilde{e}) = \frac{count(\tilde{f}, \tilde{e})}{\sum_{\tilde{f}} count(\tilde{f}, \tilde{e})} \qquad (6)$$

3 Toolkit Description

Thot toolkit has been developed using the C++ programming language. The design principles that have led the development process were: efficiency, extensibility, flexibility (it works with different and well-known data formats) and usability (the toolkit functionality is easy to use, the code is easy to incorporate to new code).

In the following subsections, we describe the basic functionality of the toolkit.

3.1 Operations between alignments

As stated in section 2.2 it is common to apply operations between alignments in order to make them better. The toolkit provides the following operations:

Union : Obtains the union of two matrices.

Intersection : Obtains the intersection of two matrices.

Sum : Obtains the sum of two or more matrices.

Symmetrization : Obtains "something" between the union and the intersection of two matrices. It was defined in (Och, 2002) for the first time, and there exist different versions.

The expected input format for the alignments is the one generated by Giza++. The output can be given in the Giza++ or in two other formats: as a bidimensional matrix (which is easily readable by a human), or a format which can be easily converted to different formats by using, for example, the `Lingua-Alignment` visualization tool (Lambert and Castell., 2004). Two or more alignment files can be supplied simultaneously, which increases the flexibility of the toolkit (the alignment information within them can appear in any order).

3.2 RF and pseudo-ML estimation

Thot toolkit provides model estimation based on single-word alignments (see section 2.2) given in Giza++ format. This estimation method is heuristic for two reasons. First, the bilingual phrases are obtained from a given single-word alignment matrix, which forces us to impose a heuristic consistence restriction in order to extract them. Second, the extracted bilingual phrases are not considered as part of complete bisegmentations when doing the model estimation. The first problem cannot be solved without changing the whole extraction method (for example, using EM algorithm as in (Marcu and Wong, 2002)). In contrast, a possible solution for the second problem can be proposed.

For this purpose, the toolkit implements a new proposal for model estimation that we have called *pseudo_ML*[2] (pML) estimation which is

[2]We use this name because actually this estimation

different from the classical approach. The estimation procedure has three steps that are repeated for each sentence pair and its corresponding alignment matrix (f_1^J, e_1^I, A):

1. Obtain the set $\mathcal{BP}(f_1^J, e_1^I, A)$ of all consistent bilingual phrases.

2. Obtain the set $\mathcal{S}_{\mathcal{BP}(f_1^J, e_1^I, A)}$ of all possible bilingual segmentations[3] of the pair (f_1^J, e_1^I) that can be composed using the extracted bilingual phrases.

3. Update the counts (actually fractional counts) for every different phrase pair $(\tilde{f}, \tilde{e})$ in the set $\mathcal{S}_{\mathcal{BP}(f_1^J, e_1^I, A)}$, as:

$$fracCount(\tilde{f}, \tilde{e}) += \frac{N(\tilde{f}, \tilde{e})}{|\mathcal{S}_{\mathcal{BP}(f_1^J, e_1^I, A)}|}$$

   where $N(\tilde{f}, \tilde{e})$ is the number of times that the pair $(\tilde{f}, \tilde{e})$ occurs in $\mathcal{S}_{\mathcal{BP}(f_1^J, e_1^I, A)}$, and $|\cdot|$ denotes the sizeof operation.

Afterwards the probability of every phrase pair $(\tilde{f}, \tilde{e})$ is computed as:

$$p(\tilde{f}|\tilde{e}) = \frac{fracCount(\tilde{f}, \tilde{e})}{\sum_{\tilde{f}} fracCount(\tilde{f}, \tilde{e})}$$

Step 2 implies that if a bilingual phrase cannot be part of any bisegmentation for a given sentence pair, this bilingual phrase will not be extracted. For this reason, pML estimation extracts fewer bilingual phrases than the RF estimation.

Figure 2 shows all possible segmentations for the word alignment matrix given in Figure 1. The *counts* and *fractional counts* for each extracted bilingual phrase will differ for each estimation method, as shown in Table 1 for the RF and pML estimation methods respectively.

In addition, pML estimation allows us to obtain more complete models including, for example, a sub-model for the segmentation length K. This functionality has been included in the toolkit.

method is equivalent to the first iteration of the EM algorithm which finally might be used to perform a correct estimation of the model

[3]A bilingual segmentation or *bisegmentation* of length K of a sentence pair (f_1^J, e_1^I) is defined as a triple $(\tilde{f}_1^K, \tilde{e}_1^K, \tilde{a}_1^K)$, where $\tilde{a}_1^K$ is a specific one-to-one mapping between the K segments/phrases of both sentences.

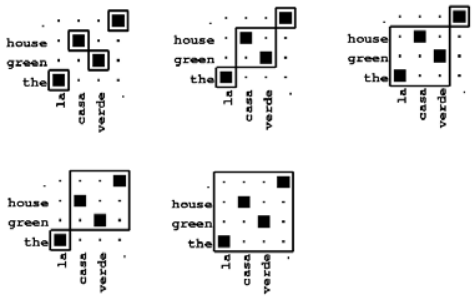


Figure 2: Possible segmentations for a given word-alignment matrix.

$\tilde{f}$ — $\tilde{e}$	**RF**	**pML**
La — the	1	3/5
casa — house	1	1/5
verde — green	1	1/5
casa verde — green house	1	1/5
La casa verde — the green house	1	1/5
. — .	1	3/5
casa verde . — green house .	1	1/5
La casa verde . — the green house .	1	1/5

Table 1: Bilingual phrase counts and fractional counts for RF and pML estimation, respectively, for the sentence pair shown in Figure 1.

pML estimation has a high computational cost due to the need to obtain the bisegmentation of each phrase pair. In order to keep these costs under control, the toolkit limits the maximum number of bisegments that can be obtained. When the maximum is reached, the bisegmentation is pruned.

One major disadvantage of the phrase-based translation models is their high memory allocation size. These sizes can be reduced if we impose a restriction over the length of the bilingual phrases, at the risk of obtaining poorer models. However, as stated in (Koehn et al., 2003), the length of the extracted phrases can be limited without decreasing the performance of a PBT system. For this reason, the model estimation with the Thot toolkit incorporates a maximum phrase length parameter.

Finally, RF and pML estimation can be restricted to be monotonic. All these variants of the estimation methods are also implemented by the toolkit, whose output can be given in the toolkit native format, or in the input format expected by the publicly available translator software Pharaoh (Koehn, 2003).

3.3 Segmentation of bilingual corpora

Given a pair of sentences (f_1^J, e_1^I) and a word alignment between them, the toolkit provides

an additional functionality that allows to obtain the best bisegmentation in K bisegments, and implicitly the best phrase-alignment $\tilde{a}_1^K$ (or Viterbi phrase-alignment) between them, according to the following algorithm:

1. For every possible $K \in \{1 \cdots \min(J, I)\}$
   (a) Extract all possible bilingual segmentations of size K according to the restrictions of $A(f_1^J, e_1^I)$.
   (b) Compute and store the probability $p(\tilde{f}_1^K, \tilde{a}_1^K | \tilde{e}_1^K)$ of these bisegmentations.
2. Return the bilingual segmentation $(\tilde{f}_1^K, \tilde{e}_1^K, \tilde{a}_1^K)$ of highest probability.

where $p(\tilde{f}_1^K, \tilde{a}_1^K | \tilde{e}_1^K) = \prod_{k=1}^K p(\tilde{f}_{\tilde{a}_k} | \tilde{e}_k)$

3.4 Applications

As a forward to the next section, we present different applications on where the Thot toolkit can be used.

The most immediate application of the phrase-based models is in the field of machine translation. For this purpose an appropriate search engine is required, such us Pharaoh.

A second application is to obtain a *bisegmentation* for a given corpus. The usefulness of this application is two fold:

- With this *bisegmentation*, can be evaluated the quality of the phrase model when it is compared with a test corpus that is manually aligned by experts.
- The *bisegmentation* of a given test corpus can be used as a preprocessing step to other machine translation systems, such as the one presented in (Casacuberta and Vidal, 2004), which is based on finite-state technology.

In addition other NLP applications can take advantage of phrase-based translation models. Some of them are: document classification, information retrieval, word-sense disambiguation, question-answering systems, etc.

4 Experiments and results

In this section, we present some experimental results using the most important features of the Thot toolkit. The corpora we have used in the experiments are outlined in Table 2 for the two well-known EuTrans-I and Hansards tasks, respectively.

4.1 Bilingual segmentation experiments

For the bilingual segmentation experiments, we selected a subset of the EuTrans-I test corpus consisting of 40 randomly selected pairs of sentences. This corpus was bilingually segmented by human experts (Nevado et al., 2004).

Table 3 shows the well-known *Recall*, *Precision*, and *F-measure* bisegmentation-quality measures for three different bisegmentation techniques including the one provided by the Thot toolkit. The other two techniques are the recursive alignments (RECalign) and the GIATI alignments (GIATIalign) that are described and tested in (Nevado et al., 2004).

As table 3 shows, the bisegmentation quality for the Thot toolkit outperforms the other two.

Technique	Recall	Precision	F-measure
RECalign	52.96	79.01	**63.41**
GIATIalign	39.99	85.52	**54.50**
Thot	72.58	65.49	**68.85**

Table 3: Bisegmentation results for 40 randomly selected test sentences for EuTrans-I task.

4.2 Machine translation experiments

We carried out a set of machine translation experiments using the functionality of the Thot toolkit and the Pharaoh translation tool; namely operations between alignments, RF and pML estimation and its application in translation quality experiments. For the experiments, we used the common definitions for Word Error Rate (WER), Position independent Error Rate (PER) and Bleu.

4.2.1 Alignment operations

Using the toolkit functionality, we estimated an RF phrase-based model in order to translate the EuTrans-I test corpus with the Pharaoh translation tool. The model estimation was performed from a set of word-alignment matrices that had been obtained by means of different alignment operations. The maximum phrase length parameter was set to 6.

Table 4 shows WER, PER, Bleu and the number of extracted phrases for each alignment operation described in section 3.1 (*none* means that no alignment operations were applied). As 4 shows, alignment symmetrization obtains the best results. As expected, the worst results are obtained when any operation is made. The intersection operation extracts the greatest number of bilingual phrases due to the

		EuTrans-I		Hansards	
		Spanish	**English**	**French**	**English**
Training	Sentences	10,000		128,000	
	Words	97,131	99,292	2,062,403	1,929,186
	Vocabulary size	686	513	37,542	29,414
Test	Sentence	2,996		500	
	Words	35,023	35,590	3,890	3,929
	Perplexity (Trigrams)	–	3.62	–	30.0

Table 2: EuTrans-I and Hansards corpus statistics

greater frequency of words that are not aligned in the word alignment matrix (as stated in section 2.2).

Op	**WER**	**PER**	**Bleu**	**#Phrases**
none	7.93	6.76	0.887	63528
and	7.56	6.95	0.883	133322
or	7.93	6.05	0.892	33350
sum	8.01	6.14	0.891	33350
Symmetr.	7.17	5.80	0.902	42414

Table 4: Alignment operation influence, maximum phrase length=6, non-monotone RF estimation, for EuTrans-I task.

4.2.2 RF vs. pML estimation

We carried out an exhaustive experimentation applying the different estimation variants described in section 3.2 over the EuTrans-I training corpus.

Table 5 shows the number of extracted bilingual phrases (no alignment operations were used, the maximum phrase length was equal to 6), the training time[4] and the amount of sentence pairs that were not completely bisegmented. As expected, monotone extraction decreases the amount of phrase pairs. pML estimation took a lot of more time and extracted fewer phrase pairs than the RF estimation, which is due to the fact mentioned in section 3.2.

We also carried out translation experiments with the above-mentioned estimation methods (again without using alignment operations and maximum phrase length equal to 6). Table 6 shows the WER, PER and Bleu error measures. As table 6 shows, pseudo-ML estimation obtains similar results than RF estimation, but a little bit worse than RF models. Despite the fact that the differences are not significant, we have two

[4]The results were obtained on a PC with a 1.6Ghz AMD Athlon processor and 512 MB of memory using Linux as the operating system. All times are given in seconds.

Estimation	**#Pairs**	**Time**	**#prunings**
Mon. RF	58,099	13.5	-
RF	63,528	14.8	-
Mon. pML	53,249	1245.6	63
pML	58,980	1637.7	83

Table 5: Number of extracted bilingual phrases for each estimation method, for EuTrans-I task.

hypotheses about this unexpected result. The first hypothesis is that it could be due to the small size of training samples used in the experiments, which finally causes an overfitting of pML model parameters to the training sample. The second hypothesis is that the RF estimation method performs a kind of smoothing because of the way of phrase-extraction technique, actually this fact can be observed in the number of bilingual phrases obtained by this technique (see Table 5), which can help to obtain better translations for a given test set.

Estimation	**WER**	**PER**	**Bleu**
Mon. RF	9.03	7.64	0.874
RF	7.93	6.76	0.887
Mon. pML	9.34	7.89	0.870
pML	8.36	7.09	0.884

Table 6: Translation experiments for the different estimation methods, for EuTrans-I task.

In contrast to these results we computed the log-likelihood, for equation 3, of the training and the test sets for both estimation methods. As we expected the pML estimation obtained better log-likelihood than the RF estimation in training and test (also for the maximum approximation which is the most commonly used search criterion). Despite the translation results showed above, this result proves that the proposed estimation pML obtains a better parameter estimation for the phrase-based translation

model.

Additional experiments were performed in order to determine the effect of the maximum phrase length parameter. See Table 7 for the influence of this parameter in RF estimation. In this table, the training time, the WER and PER error measures and the number of extracted pairs are given. As table 7 shows, parameter values greater than 4 do not improve the results and increase the estimation time. We have observed the same situation for pML estimation.

	Time	WER	PER	Bleu	#pairs
1	6.030	31.36	26.48	0.582	1736
3	8.500	9.64	7.95	0.867	14953
5	12.300	8.19	6.89	0.884	44056
7+	16.750	7.88	6.75	0.888	84145

Table 7: Phrase length parameter influence, RF estimation, for EuTrans-I task.

4.2.3 Translation quality experiments

Finally, we carried out a translation quality experiment adjusting both the Thot toolkit parameters and the Pharaoh parameters appropiately. Specifically, a RF model was estimated from symmetrized word alignment matrices. The maximum phrase length parameter was set to 6.

Table 8 shows the WER and PER error measures for the EuTrans-I corpus. We compared the Pharaoh translation quality with the quality obtained by two other translation tools: the *ISI ReWrite Decoder*, a publicly available translation tool that implements a greedy decoder (see (Germann et al., 2001)), and GIATI, a stochastic finite state transductor (see (Casacuberta and Vidal, 2004)). The results obtained by Pharaoh and GIATI were very similar and clearly outperformed the results of the greedy decoder.

Decoder	WER	PER	Bleu
Greedy	25.2	22.3	0.55
Pharaoh	6.7	5.3	0.90
GIATI	6.6	-	0.91

Table 8: Translation quality results for the EuTrans-I task.

A similar experimentation is shown in Table 9 for the Hansards task. In this case, the results obtained by the greedy decoder are closer to the results obtained by Pharaoh. (In Table 9 results with the GIATI technique are not available since they have not been obtained so far.)

Decoder	WER	PER	BLEU
Greedy	57.0	52.0	0.22
Pharaoh	52.8	48.1	0.31

Table 9: Translation quality results for the Hansards task.

5 Concluding Remarks

In this paper, we have given a description of the Thot toolkit, which is publicly available as open source software at `http://www.info-ab.uclm.es/simd/software/thot`.

The main purpose of the toolkit is to provide an easy, effective, and useful way to train phrase-based statistical translation models to be used as part of a statistical machine translation system, or for other different NLP related tasks.

The main features (among others) that this toolkit offers are:

- Different combinations of single, word-based alignments to obtain better alignment matrices or to directly obtain phrase-based statistical lexicons.
- Training of phrase-based translation in accordance with some of the different approaches mentioned above, and a new approach that we call *pseudo_ML* estimation.

According to the results presented in section 4.2.2, it is important to note that the pML estimation proposed in this paper obtains similar results than those obtained with the RF estimation. Despite the fact that the differences are not significant and that the log-likelihood for pseudo-ML estimation is better than the RF estimation, much more detailed experimentation must be carried out in order to give a reasonable explanation for the very similar translation results obtained with both techniques.

We believe that this toolkit (in conjunction with other freely available statistical machine translation tools) can provide the MT community with a valuable resource, which can be used to build their own in-house statistical machine translation systems with a very low development cost. The toolkit has been developed and implemented following standard principles of design such as usability and versatility in formats. These features make it attractive not only

for experts in the field of SMT but to a general audience whose knowledge of the mathematical details of this approach is limited.

6 Future Works

There are still features of Thot toolkit that should be improved. One of these is the estimation of an alignment/distortion model to improve the phrase-based models.

We also have in mind for a near future:

- To make a formal derivation of the phrase-based translation models, in order to obtain explicitly mathematical formulation to implement an EM estimation of the phrase-based model parameters.
- To implement our own phrase-based decoder, specially designed to be used with this toolkit, which also will be publicly available as open source software. The new decoder should have lower memory requirements than the Pharaoh decoder, in order to be used with complex corpora like HANSARDS.
- To include more complex ways to combine word-based alignment matrices as the ones described in (Venugopal et al., 2003) and in (Lambert and Castell., 2004).

References

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and R. L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

F. Casacuberta and E. Vidal. 2004. Machine translation with inferred stochastic finite-state transducers. *Computational Linguistics*, 30(2):205–225.

Ulrich Germann, Michael Jahr, Kevin Knight, Daniel Marcu, and Kenji Yamada. 2001. Fast decoding and optimal decoding for machine translation. In *Proc. of the 39th Annual Meeting of ACL*, pages 228–235, Toulouse, France, July.

P. Koehn, F. J. Och, and D. Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the HLT/NAACL*, Edmonton, Canada, May.

Phillip Koehn. 2003. Pharaoh: a beam search decoder for phrase-based statistical machine translation models. User manual and description. Technical report, USC Information Science Institute, December.

Patrik Lambert and Núria Castell. 2004. Alignment of parallel corpora exploiting asymmetrically aligned phrases. In *Proc. of the Fourth Int. Conf. on LREC*, Lisbon, Portugal.

Daniel Marcu and William Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proceedings of the EMNLP Conference*, pages 1408–1414, Philadelphia, USA, July.

F. Nevado, F. Casacuberta, and J. Landa. 2004. Translation memories enrichment by statistical bilingual segmentation. In *Proc. of the Fourth Int. Conf. on LREC*, Lisbon.

Hermann Ney, Sonja Nießen, Franz J. Och, Hassan Sawaf, Christoph Tillmann, and Stephan Vogel. 2000. Algorithms for statistical translation of spoken language. *IEEE Trans. on Speech and Audio Processing*, 8(1):24–36, January.

Franz J. Och. 2000. GIZA++: Training of statistical translation models. `http://www-i6.informatik.rwth-aachen.de/\~och/software/GIZA++.html`.

Franz Joseph Och. 2002. *Statistical Machine Translation: From Single-Word Models to Alignment Templates*. Ph.D. thesis, Computer Science Department, RWTH Aachen, Germany, October.

J. Tomás and F. Casacuberta. 2001. Monotone statistical translation using word groups. In *Procs. of the Machine Translation Summit VIII*, pages 357–361, Santiago de Compostela, Spain.

Ashish Venugopal, Stephan Vogel, and Alex Waibel. 2003. Effective phrase translation extraction from alignment models. In *Proc. of the 41th Annual Meeting of ACL*, pages 319–326, Sapporo, Japan, July.

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In *Proc. of the 39th Annual Meeting of ACL*, pages 523–530, Toulouse, France, July.

R. Zens, F.J. Och, and H. Ney. 2002. Phrase-based statistical machine translation. In *Advances in artificial intelligence. 25. Annual German Conference on AI*, volume 2479 of *Lecture Notes in Computer Science*, pages 18–32. Springer Verlag, September.