Multi-Target Machine Translation using Finite-State Transducers

María-Teresa González*, **Francisco Casacuberta***

*Dept. de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
46071 Valencia, Spain
{mgonzalez,fcn}@dsic.upv.es

Abstract

Finite-state transducers are very useful models in different areas of pattern recognition, natural language processing, and machine translation. Two methods for inferring a finite-state transducer that accept a source sentence and produce several targets are proposed in this article. Transducers of this kind are called multiple finite-state transducers. In the first method, a multiple finite-state transducer will be inferred by a multilingual corpus. In the second method, the models will be inferred using different bilingual corpora, since in the real world there are more bilingual corpora than multilingual corpora. The proposed methods are assessed through a series of machine translation experiments.

1. Introduction

Finite-state transducers (Vidal and Casacuberta, 2004) through the years have given rise to an important framework in syntactic-pattern recognition (Fu, 1982; Vidal et al., 1995) and in language processing (Mohri, 1997). Another more complex application of formal transducers is *language translation*. By translation, we mean the process where a source sentence (input) in one language is translated into a target sentence (output) in a different language, preserving the original meaning.

We know that there are more powerful transduction models than finite-state transducers (Casacuberta et al., 2004), but we have chosen them because they generally entail more affordable computational costs and obtain rather good results in practice.

One of the main reasons for the interest in finite-state machines for language translation comes from the fact that these machines can be learned automatically from examples (Vidal et al., 1995).

Nowadays, to simultaneously translate a sentence from one language to sentences in several languages, a finite-state transducer is inferred for each language pair. In this work, we propose two methods for inferring only one finite-state transducer that will be multiple, that is, permit translation from one language to several languages simultaneously, which has a lower computational cost. Our main goal in both models is to achieve this computational improvement without degrading the translation quality. In the first model, we use a multilingual corpus to infer the transducers, where the samples are (source,target1,…,targetN). We consider that a collection of tuples for this kind of task can be adequate for a modified version of the GIATI method (Casacuberta and Vidal, 2004). We also propose a second method for inferring multiple finite-state transducers that are from different but related bilingual corpora since in the real world there are more bilingual corpora than multiple corpora. Our goal for proposing this second method was to maintain the advantages that are obtained by inferring a transducer with a multilingual corpus, but without having this corpus *a priori*. By a multiple finite-state transducer, we mean a finite-state transducer where the output is formed by a string of several languages.

The results show that both of these methods have the advantage that a transducer is not required for each language pair. Another advantage is that a language can benefit from the influence of another language by inferring the multiple finite-state transducers.

The organization of the article is as follows. Section 2. presents, the two proposed methods for inferring multiple finite-state transducers from multiple and bilingual corpora. Section 3. describes, the experiments using the two methods and presents the corresponding results. The corpora used in this section were derived from the *EuTrans* project (Instituto Tecnológico de Informática et al., 2000) corpora. Finally, Section 4. is devoted to conclusions and further work.

2. Methods for Inferring Multiple Finite-State Transducers

In this section, we introduce some basic definitions of statistical alignments and search in finite-state transducers. We also describe the two methods proposed for inferring multiple finite-state transducers.

2.1. Statistical Alignments

The statistical translation models introduced by (Brown et al., 1990; Brown et al., 1993) are based on the concept of alignment between source and target words. Formally, an *alignment* of a translation pair $(\mathbf{s}, \mathbf{t}) \in \Sigma^* \times \Delta^*$ is a function $\mathbf{a} : \{1, \ldots, |\mathbf{t}|\} \rightarrow \{0, \ldots, |\mathbf{s}|\}$. The particular case $\mathbf{a}(j) = 0$ means that the position j in $\mathbf{t}$ is not aligned with any position in $\mathbf{s}$. The probability of translating a given $\mathbf{s}$ into $\mathbf{t}$ by an alignment $\mathbf{a}$ is denoted by $\Pr(\mathbf{t}, \mathbf{a}|\mathbf{s})$.

Different approaches for estimating the optimal $\mathbf{a}$ were proposed in (Brown et al., 1993). Nowadays, there are adequate software packages that are publicly available for obtaining good alignments between pairs of sentences such as is GIZA++ (Och and Ney, 2000). An example of Spanish-English sentence alignment is given below:

¿ Cuánto cuesta una habitación individual por semana ?

How (2) much (2) does (3) a (4) single (6) room (5) cost (3) per (7) week (8) ? (9)

Each number within parentheses in the example represents the position in the source sentence (Spanish) that is aligned with the (position of the) preceding target word (English). For example, position 1 of the target sentence (*How*) is aligned with position 2 of the source sentence (*Cuánto*). Formally, it will be represented as $\mathbf{a}(1) = 2$.

2.2. Method using a Multilingual Corpus

In (Casacuberta and Vidal, 2004), a method for inferring a finite-state transducer is presented. This is called *grammatical inference and alignments for transducer inference* (GIATI). The method that is proposed in this document is an adaptation of GIATI for inferring a multiple finite-state transducer from a multilingual corpus. In other words, training samples are of the form (source,target1,...,targetN); where the target sentence *i* is the translation of the source sentence in language *i*.

Formally, given a finite sample A of strings $(\mathbf{s}, \mathbf{t}_1, \ldots, \mathbf{t}_N) \in \Sigma^* \times \Delta_1^* \times \ldots \times \Delta_N^*$ with $N \geq 1$ (a multilingual corpus):

1. Each training sample $(\mathbf{s}, \mathbf{t}_1, \ldots, \mathbf{t}_N)$ from A is transformed into a string z from an extended alphabet Γ (strings of Γ-symbols) yielding a sample S of strings $S \subset \Gamma^*$.
2. A (stochastic) regular grammar G is inferred from S.
3. The Γ-symbols of the grammar rules are transformed back into samples of *source/target1/.../targetN* (from $\Sigma^* \times \Delta_1^* \times \ldots \times \Delta_N^*$).

The method presented here is called *grammatical inference and alignments for multiple transducer inference* (GIAMTI).

2.2.1. First Step of the GIAMTI Methodology: Transformation of Training Samples into Strings

The first step of the proposed method is very similar to the GIATI method (Casacuberta and Vidal, 2004), but it is extended to several targets. Thus, the GIAMTI methodology consists in a labeling process (L) that builds a string of certain extended symbols from each training string pair source/target and its corresponding statistical alignment. An alignment $\mathbf{a}_n$ is computed for each $(\mathbf{s}, \mathbf{t}_n)$ $1 \leq n \leq N$. The main idea is to assign each word from $\mathbf{t}_n$ to the corresponding word from $\mathbf{s}$ given by the alignment $\mathbf{a}_n$. When this assignment produces a violation of the sequential order of the words in $\mathbf{t}_n$, we assign the word in $\mathbf{t}_n$ to the word in $\mathbf{s}$ that is closest to the aligned word but without reordering the target words. An example of the GIAMTI methodology in this step can be seen below.

Example 1 *Let A be a training sample composed by the following strings (Spanish,English,Italian,German):*

una habitación doble
 # a double room
 # una camera doppia
 # ein Doppelzimmer
una habitación
 # a room
 # una camera
 # ein zimmer
la habitación individual
 # the single room
 # la camera singola
 # das Einzelzimmer
la habitación
 # the room
 # la camera
 # das zimmer

Suitable alignments for this sample are

una habitación doble
 # a (1) double (3) room (2)
 # una (1) camera (2) doppia (3)
 # ein (1) Doppelzimmer (3)
una habitación
 # a (1) room (2)
 # una (1) camera (2)
 # ein (1) zimmer (2)
la habitación individual
 # the (1) single (3) room (2)
 # la (1) camera (2) singola (3)
 # das (1) Einzelzimmer (3)
la habitación
 # the (1) room (2)
 # la (1) camera (2)
 # das (1) zimmer (2)

In the pair *Spanish-English* pair, in the first aligned sample in Example 1, the English word (*double*) is assigned to the third Spanish word (*doble*), and the English word (*room*) is assigned to the second Spanish word (*habitación*). This would imply a reordering of the words *double* and *room*, which is not appropriate in our finite-state framework.

Given $\mathbf{s}, \mathbf{t}_1, \ldots, \mathbf{t}_N$ and $\mathbf{a_1^N}$ (source and target strings and associated alignment, respectively), the proposed transformation $L(\mathbf{s}, \mathbf{t}_1, \ldots, \mathbf{t}_N)$ avoids this problem as follows:

First, we apply a transformation $z_n = L_n(\mathbf{s}, \mathbf{t}_n, \mathbf{a_n})$ for each N, that is, for each pair source-targetN.

$$z_{ni} = \begin{cases} & |\mathbf{z}_n| = |\mathbf{s}| \\ & 1 \leq i \leq |\mathbf{z}_n| \\ (s_i, t_j^n t_{j+1}^n \ldots t_{j+l}^n) & \text{if } \exists j : \mathbf{a}(j) = i \text{ and} \\ & \nexists j\prime < j : \mathbf{a}(j\prime) > \mathbf{a}(j) \\ & \text{and for } j\prime\prime : \\ & j \leq j\prime\prime \leq j + l, \\ & \mathbf{a}(j\prime\prime) \leq \mathbf{a}(j) \\ (s_i, \lambda) & \text{otherwise} \end{cases}$$

Then, the above transformations L_n are merged by the Spanish word, and the final transformation L is obtained.

The application of L to Example 1 generates the following strings of extended symbols:

(una,a,una,ein)
(habitación,λ,camera,λ)
(doble,double room,doppia,Doppelzimmer)
(una,a,una,ein)
(habitación,room,camera,zimmer)

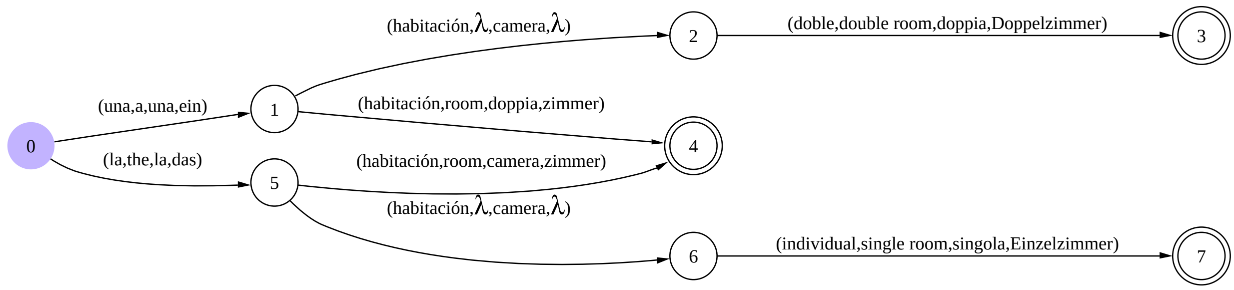

Figure 1: Trigram model inferred from strings obtained by the transformation L in Example 1.

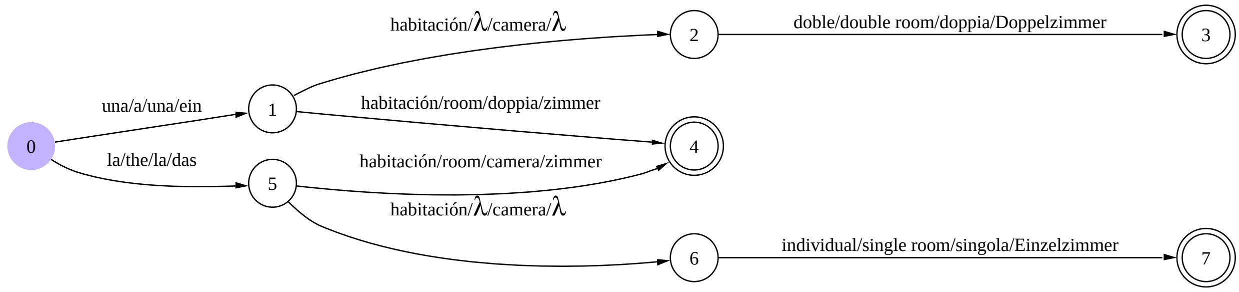

Figure 2: A multiple finite-state transducer built from the n-gram of Figure 1.

(la,the,la,ins)
(habitación,λ,camera,λ)
(individual,single room,singola,Einzelzimmer)
(la,the,la,ins)
(habitación,room,camera,zimmer)

2.2.2. The Second Step of the GIAMTI Methodology: Iinferring a Stochastic Regular Grammar from a Set of Strings

The second step of the proposed method is the same as the one in the GIATI methodology (Casacuberta and Vidal, 2004). The n-grams are computed with the SRILM toolkit (Stolcke, 2002) using the *back-off* smoothing technique with the training set of extended strings obtained from the L transformation.

Figure 1 shows the trigram model inferred from the sample obtained using the L transformation in Example 1.

2.2.3. The Third Step of the GIAMTI Methodology: Transforming a Stochastic Regular Grammar into a Multiple Stochastic Finite-State Transducer

The third step of the proposed method is similar to the one in the GIATI methodology (Casacuberta and Vidal, 2004). An "inverse transformation" $\Lambda(\cdot)$ from a grammar of L-transformed symbols is used in order to obtain a finite-state transducer. $\Lambda(\cdot)$ is based on the following simple morphim where n is the number of translations (targets):

$$\text{if } (a,b) \in \Gamma: \quad \begin{array}{l} h_\Sigma((a,b)) = a, \\ h_{\Delta_1}((a,b)) = b_1, \\ \ldots, \\ h_{\Delta_n}((a,b)) = b_n \end{array}$$

$$\text{where} \quad \begin{array}{l} a \in \Sigma, \\ b_1 = b_{1,1}, b_{1,2}, \ldots, b_{1,k} \in \Delta_1, \\ \ldots, \\ b_n = b_{n,1}, b_{n,2}, \ldots, b_{n,j} \in \Delta_n, \\ b = b_1, \ldots, b_n \in \Delta \end{array}$$

Figure 2 shows the finite-state transducer inferred from the n-gram in Figure 1.

2.3. Search in Stochastic Finite-State Transducers with the Viterbi Algorithm

The search for an optimal $\hat{\mathbf{t}}$ in a stochastic finite-state transducer $\mathcal{T}_P$ for a source sentence s

$$\hat{\mathbf{t}} = \operatorname*{argmax}_{\mathbf{t} \in \Delta^*} P_{\mathcal{T}_P}(\mathbf{s}, \mathbf{t}) \tag{1}$$

has proved to be a difficult computational problem (Casacuberta and de la Higuera, 2000). In practice, an approximate solution can be obtained (F.Casacuberta, 2000) on the basis of the following Viterbi approximation to the probability of a translation pair (source, target):

$$P_{\mathcal{T}_P}(\mathbf{s}, \mathbf{t}) \approx V_{\mathcal{T}_P}(\mathbf{s}, \mathbf{t}) = \max_{\phi \in d(\mathbf{s}, \mathbf{t})} P_{\mathcal{T}_P}(\phi) \tag{2}$$

When ϕ is a translation form, an approximate translation can now be computed as

$$\tilde{\mathbf{t}} = \operatorname*{argmax}_{\mathbf{t} \in \Delta^*} V_{\mathcal{T}_P}(\mathbf{s}, \mathbf{t}) = \operatorname*{argmax}_{\mathbf{t} \in \Delta^*} \max_{\phi \in d(\mathbf{s}, \mathbf{t})} P_{\mathcal{T}_P}(\phi) \tag{3}$$

This computation can be carried out efficiently (Casacuberta, 1996) by using dynamic programming techniques.
Finally, the approximate translation $\tilde{\mathbf{t}}$ is obtained as the concatenation of the target strings associated with the translation form

		Spanish	English	Italian	German
Train	Sentence pairs	10000			
	Distinct pairs	6812			
	Running words	119725	123559	112933	109911
	Vocabulary	586	430	493	470
Test	Sentence pairs	2985			
	Running words	30554	31701	28671	27899
	Vocabulary	357	237	288	261
	Bigram perplexity	3.6	2.7	3.2	3.1

Table 1: The Spanish-English-Italian-German multiple corpus. There was no overlap between training and test sentences, and the test set did not contain out-of-vocabulary words with respect to any of the training sets.

$$\phi = (q_0, s_1, \bar{\mathbf{t}}_1, q_1)(q_1, s_2, \bar{\mathbf{t}}_2, q_2)\dots(q_{I-1}, s_I, \bar{\mathbf{t}}_I, q_I)$$

corresponding to the optimal sequence of states involved in the solution, that is,

$$\tilde{\mathbf{t}} = \bar{\mathbf{t}}_1\bar{\mathbf{t}}_2\dots\bar{\mathbf{t}}_I \tag{4}$$

In our particular case, which is multiple the samples are *(source,target1,...,targetN)* and, the search is carried out in the same way as for two languages. However, when a sentence is translated, the targetN we want to obtain is chosen. For example, in the transducer shown in Figure 2, if the optimal translation form for the source "*una habitación doble*" is "*una/a/una/ein habitación/λ/camera/λ doble/double room/doppia/Doppelzimmer*", the English translation will be "*a λ double room*", the Italian translation will be "*a camera doppia*", and the German translation will be "*ein λ Doppelzimmer*".

2.4. The Method using Bilingual Corpora

In the real world, there are more bilingual corpora than multilingual corpora. Our goal is to try to take advantage of the benefits of the GIAMTI method (described in Section 3.) but without requiring a multilingual corpus *a priori*:
Formally, the method below follows a technique for learning a multiple (stochastic) finite-state transducer. Given several bilingual corpora where the source in each corpus is in the same language, a finite sample A_1 of strings $(\mathbf{s_1}, \mathbf{t}_1) \in \Sigma^* \times \Delta_1^*$, A_2 of strings $(\mathbf{s_2}, \mathbf{t}_2) \in \Sigma^* \times \Delta_2^*, \dots,$ and A_N of strings $(\mathbf{s_N}, \mathbf{t}_N) \in \Sigma^* \times \Delta_N^*$ with $N \geq 1$ then:

1. A (stochastic) finite-state transducer is inferred with GIATI for each bilingual corpus.

2. A multilingual corpus is created: the source sentences of each A_n are translated using the finite-state transducers of step 1. Therefore, for each A_n, there will be a set of source sentences and the corresponding translations. The union of all these bilingual corpora is the final multilingual corpus that is used in the next step.

3. A multiple finite-state transducer is inferred with GIAMTI.

The method presented is called GIAMTI2.

3. Experimental Results

3.1. GIAMTI Experiments

For the experiments, a multilingual corpus for more than two languages was needed; that is, each sample was composed of a source sentence and its target sentences in different languages. This corpus was created from the corpora generated in the *EuTrans* project (Instituto Tecnológico de Informática et al., 2000), where several corpora of $500,000$ sentence pairs were generated: *Spanish-English* ($C1$), *Spanish-Italian* ($C2$), and *Spanish-German* ($C3$).
A small subset ($B1$) was created for $C1$. It had 10000 sentence pairs for training and 3000 sentence pairs for test. It was used as a benchmark for the different methods in machine translation. A new multiple corpus ($B2$) was created from this and using the whole corpora $C1$, $C2$, and $C3$. The source sentences of $B2$ were the same or the most similar to those in $B1$[1]. Then, the translations of the sentences to *English*, *Italian* and *German* were obtained with $C1$, $C2$, and $C3$, respectively. The new multilingual corpus was not composed by sentence pairs but by samples with a source sentence and three target sentences. An example is: *una habitación doble # a double room # una camera doppia # ein Doppelzimmer*.
A summary of this multilingual corpus is given in Table 1.
The following experiments were carried out with this multilingual corpus:

1. Obtain results using GIATI (Casacuberta and Vidal, 2004) for the sentence pairs *Spanish-English*, *Spanish-Italian*, and *Spanish-German* from the new multilingual corpus. These results were used as a baseline.

2. Obtain results using GIAMTI (Subsection 2.2.) for samples *Spanish-English-Italian-German*, *Spanish-English-Italian*, *Spanish-English-German*, and *Spanish-Italian-German* from the new multilingual corpus. These results were compared with the GIATI results in experiment 1 and demostrate the advantages of the GIAMTI method.

The results of these experiments are shown in Table 2. All the results were evaluated with the error measure WER

[1] *Similar* means the sentences with the lowest WER that were in all three corpora ($C1$, $C2$, $C3$).

	GIATI	GIATIM			
		eng-ita-ger	eng-ita	eng-ger	ita-ger
English	7.14	6.96	7.23	**6.82**	XXX
Italian	2.20	2.17	2.32	XXX	**2.08**
German	12.40	12.00	XXX	**11.80**	12.52

Table 2: Results with GIATI and with GIAMTI using the multiple corpus and translating from Spanish to English, Italian, or German. The results are shown with WER (Word Error Rate).

		Spanish	English	Spanish	Italian	Spanish	German
Train	Sentence pairs	10000					
	Distinct pairs	9704		9707		9671	
	Running words	127701	128225	126023	117634	128634	120572
	Vocabulary	680	505	680	569	680	553
Test	Sentence pairs	2870		2865		2969	
	Running words	37147	37462	36094	33579	38108	35657
	Vocabulary	579	442	587	503	587	495
	Bigram perplexity	7.9	5.7	8.3	6.9	7.7	6.0

Table 3: The bilingual corpora: Spanish-English, Spanish-Italian, and Spanish-German. There was no overlap between training and test sentences, and the test set did not contain out-of-vocabulary words with respect to any of the training sets. All corpora had the same source training vocabulary.

(Word Error Rate)[2].
As the results show, we can state the following:

- The translation quality was not degraded with this method. Moreover, they were improved.
- *Italian* does not help to improve the translation from *Spanish* to *English* or to *German*. And *English* and *German* do not help to improve the translation from *Spanish* to *Italian*.
- Two Germanic languages can help each other; that is, *English* helps in the translation from *Spanish* to *German*, and *German* helps in the translation from *Spanish* to *English*.

3.2. GIAMTI2 Experiments

To carrying out the experiments for GIAMTI2 method, several bilingual corpora were created from the corpora of the *EuTrans* project (Instituto Tecnológico de Informática et al., 2000)(See Subsection 3.1.).
The three subsets of 10000 training sentence pairs were randomly selected from $C1$, $C2$, and $C3$ corpora of the *EuTrans* project (see Subsection 3.1.). The only constraint was that each new corpus had the same source vocabulary. Since the source sentences were not the same, and did not make up a multilingual corpus the GIAMTI2 method was used.
A summary of these bilingual corpora is given in Table 3.
The following experiments were carried out with these bilingual corpora:

1. Obtain results with GIATI (Casacuberta and Vidal, 2004) for each new bilingual corpus. These results were used as a baseline.
2. Obtain results with GIAMTI2 (see Subsection 2.2.) for samples *Spanish-English-Italian-German*, *Spanish-English-Italian*, *Spanish-English-German*, and *Spanish-Italian-German* from the new bilingual corpora. These results were compared with the GIATI results and demonstrated the advantages of the GIAMTI2 method.

The results of these experiments are shown in Table 4. All the results were evaluated with the error measure WER (Word Error Rate).
As results show, we can state the following:

- Since error are introduced in step 2 of the GIAMTI2 methodology, the improvements achieved with GIAMTI were not achieved with GIAMTI2. However, the translation quality was not degraded, even improving in some cases.
- Two Germanic languages can help to improve each other. This is shown in the case *Spanish-English-German* where the *German* helps to improve the final result of the *English* translations.
- Since Italian and Spanish are Romanic languages, the influence of English and German did not improve the Italian translations.

4. Conclusions and Further Work

This work has shown that it is possible to infer multiple finite-state transducers such as GIAMTI (using a multilingual corpus) and GIAMTI2 (using bilingual corpora). It

[2]The minimum number of substitution, insertion, and deletion operations that is needed to convert the word string hypothesized by the translation system into a given single reference word string.

	GIATI	GIATIM2			
		eng-ita-ger	eng-ita	eng-ger	ita-ger
English	7.17	7.41	7.35	**6.97**	XXX
Italian	**2.37**	2.58	2.47	XXX	2.43
German	10.50	10.81	XXX	10.71	**10.46**

Table 4: Results with GIATI and with GIAMTI2 using bilingual corpora and translating from Spanish to English, Italian, or German. The results are shown with WER (Word Error Rate).

was also shown that these models can produce improvements in the translations where more than one language is available. The results obtained using a multilingual corpus were expected to be better than the results using bilingual corpora. This is because with GIAMTI2 we know *a priori* that some errors are introduced in the second step of the method. The improvement with GIAMTI2 are not as good as the improvements that are obtained with GIAMTI. However, in some cases, the GIAMTI2 method maintain the improvements that were obtained with the GIAMTI method. It is importat to note that by inferring multiple finite-state transducers when translating a source sentence to several languages, it is only necessary to compute a single search and not one for each language. These results leads us to look for new ways for inferring multiple finite-state transducers without requiring a multilingual corpus *a priori*. Inferring multiple finite-state transducers from bilingual corpora is a more realistic way than inferring them from a multiple corpus. Our results are all assessed through the Eutrans corpus. In future works, we expect to obtain results with other corpora such as the Europarl corpus which is a publicly available corpus and is a multiplingual resource that would be very well-suited for this task.

5. Acknowledgements

This work has been partially supported by Itefte project (CICYT TIC2003-08681-C02-02).

6. References

Peter F. Brown, John Cocke, Stephen Della Pietra, Vincent J. Della Pietra, Frederick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85.

Peter F. Brown, S. Della Pietra, V. Della Pietra, and R. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–312.

F. Casacuberta and C. de la Higuera. 2000. Computational complexity of problems on probabilistic grammars and transducers. In A.L. Oliveira, editor, *Grammatical Inference: Algorithms and Applications*, volume 1891 of *Lecture Notes in Computer Science*, pages 15–24. Springer-Verlag. 5th International Colloquium Grammatical Inference -ICGI2000-. Lisboa. Portugal. Septiembre.

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

F. Casacuberta, H. Ney, F. J. Och, E. Vidal, J. M. Vilar, S. Barrachina, I. Garcia-Varea, D. Llorens, C. Martinez, S. Molau, F. Nevado, M. Pastor, D. Pico, and A. Sanchis. 2004. Some approaches to statistical and finite-state speech-to-speech translation. *Computer Speech and Language*, 18:25–47.

Francisco Casacuberta. 1996. Maximum mutual information and conditional maximum likelihood estimations of stochastic syntax-directed translation schemes. In L. Miclet and C. de la Higuera, editors, *Grammatical Inference: Learning Syntax from Sentences*, volume 1147 of *Lecture Notes in Artificial Intelligence*, pages 282–291. Springer-Verlag.

F.Casacuberta. 2000. Inference of finite-state transducers by using regular grammars and morphisms. In A.L. Oliveira, editor, *Grammatical Inference: Algorithms and Applications*, volume 1891 of *Lecture Notes in Computer Science*, pages 1–14. Springer-Verlag. 5th International Colloquium Grammatical Inference -ICGI2000-. Lisboa. Portugal. Septiembre.

K. Fu. 1982. *Syntactic pattern recognition and applications*. Pretice-Hall, Englewood Cliffs, MJ.

Instituto Tecnológico de Informática, Fondazione Ugo Bordoni, Rheinisch Westfälische Technische Hoschschule Aachen Lehrstuhl für Informatik VI, and Zeres GmbH Bochum. 2000. Example-based language transalation systems: Final report. Technical report.

Mehryar Mohri. 1997. Finite-state transducers in language and speech processing. *Computational Linguistics*, 23(2):269–311.

Franz J. Och and Hermann Ney. 2000. Improved statistical alignment models. In *Proc. of ACL'00*, pages 440–447, Hong Kong, China, October.

A. Stolcke. 2002. Srilm - an extensible language modeling toolkit. In *Proc. Intl. Conf. Spoken Language Processing*.

E. Vidal and F. Casacuberta. 2004. Learning finite-state models for machine translation. In *Grammatical Inference: Algorithms and Applications. Proceedings of the 7th International Coloquium ICGI 2004*, volume 3264 of *Lecture Notes in Artificial Intelligence*, pages 16–27. Springer, Athens, Greece, October. Invited conference.

Enrique Vidal, Francisco Casacuberta, and Pedro García. 1995. Grammatical inference and automatic speech recognition. In A. Rubio, editor, *New Advantages and Trends in Speech Recognition and Coding*, volume 147 of *NATO-ASI Series F:Computer and Systems Sciences*, pages 174–191. Springer-Verlag.