Machine Translation with Inferred Stochastic Finite-State Transducers

Francisco Casacuberta*
Universidad Politécnica de Valencia

Enrique Vidal*
Universidad Politécnica de Valencia

Finite-state transducers are models that are being used in different areas of pattern recognition and computational linguistics. One of these areas is machine translation, in which the approaches that are based on building models automatically from training examples are becoming more and more attractive. Finite-state transducers are very adequate for use in constrained tasks in which training samples of pairs of sentences are available. A technique for inferring finite-state transducers is proposed in this article. This technique is based on formal relations between finite-state transducers and rational grammars. Given a training corpus of source-target pairs of sentences, the proposed approach uses statistical alignment methods to produce a set of conventional strings from which a stochastic rational grammar (e.g., an n*-gram) is inferred. This grammar is finally converted into a finite-state transducer. The proposed methods are assessed through a series of machine translation experiments within the framework of the* EuTrans *project.*

1. Introduction

Formal transducers give rise to an important framework in **syntactic-pattern recognition** (Fu 1982; Vidal, Casacuberta, and García 1995) and in **language processing** (Mohri 1997). Many tasks in automatic speech recognition can be viewed as simple translations from acoustic sequences to sublexical or lexical sequences (**acoustic-phonetic decoding**) or from acoustic or lexical sequences to query strings (for database access) or (robot control) commands (**semantic decoding**) (Vidal, Casacuberta, and García 1995; Vidal 1997; Bangalore and Ricardi 2000a, 2000b; Hazen, Hetherington, and Park 2001; Mou, Seneff, and Zue 2001; Segarra et al. 2001; Seward 2001).

Another similar application is the recognition of continuous hand-written characters (González et al. 2000). Yet a more complex application of formal transducers is **language translation**, in which input and output can be text, speech, (continuous) handwritten text, etc. (Mohri 1997; Vidal 1997; Bangalore and Ricardi 2000b, 2001; Amengual et al. 2000).

Rational transductions (Berstel 1979) constitute an important class within the field of formal translation. These transductions are realized by the so-called **finite-state transducers**. Even though other, more powerful transduction models exist, finite-state transducers generally entail much more affordable computational costs, thereby making these simpler models more interesting 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, Casacuberta, and García 1995). Nowadays, only a few techniques exist for inferring finite-state transducers (Vidal, García, and Segarra 1989; Oncina,

* Departamento de Sistemas Informáticos y Computación, Instituto Tecnológico de Informática, 46071 Valencia, Spain. E-mail:{fcn, evidal}@iti.upv.es.

García, and Vidal 1993; Mäkinen 1999; Knight and Al-Onaizan 1998; Bangalore and Ricardi 2000b; Casacuberta 2000; Vilar 2000). Nevertheless, there are many techniques for inferring regular grammars from finite sets of learning strings which have been used successfully in a number of fields, including automatic speech recognition (Vidal, Casacuberta, and García 1995). Some of these techniques are based on results from formal language theory. In particular, complex regular grammars can be built by inferring simple grammars that recognize local languages (García, Vidal, and Casacuberta 1987).

Here we explore this idea further and propose methods that use (simple) finite-state grammar learning techniques, such as n-gram modeling, to infer rational transducers which prove adequate for language translation.

The organization of the article is as follows. Sections 2 and 3 give the basic definitions of a finite-state transducer and the corresponding stochastic extension, presented within the statistical framework of language translation. In Section 4, the proposed method for inferring stochastic finite-state transducers is presented. The experiments are described in Section 5. Finally, Section 6 is devoted to general discussion and conclusions.

2. Finite-State Transducers

A finite-state transducer, $\mathcal{T}$, is a tuple $\langle \Sigma, \Delta, Q, q_0, F, \delta \rangle$, in which Σ is a finite set of **source symbols**, Δ is a finite set of **target symbols** ($\Sigma \cap \Delta = \emptyset$), Q is a finite set of **states**, q_0 is the initial state, $F \subseteq Q$ is a set of final states, and $\delta \subseteq Q \times \Sigma \times \Delta^\star \times Q$ is a set of transitions.[1] A **translation form** ϕ of length I in $\mathcal{T}$ is defined as a sequence of transitions:

$$\phi = (q_0^\phi, \mathrm{s}_1^\phi, \bar{\mathbf{t}}_1^\phi, q_1^\phi)(q_1^\phi, \mathrm{s}_2^\phi, \bar{\mathbf{t}}_2^\phi, q_2^\phi)(q_2^\phi, \mathrm{s}_3^\phi, \bar{\mathbf{t}}_3^\phi, q_3^\phi) \ldots (q_{I-1}^\phi, \mathrm{s}_I^\phi, \bar{\mathbf{t}}_I^\phi, q_I^\phi) \tag{1}$$

where $(q_{i-1}^\phi, \mathrm{s}_i^\phi, \bar{\mathbf{t}}_i^\phi, q_i^\phi) \in \delta$, $q_0^\phi = q_0$, and $q_I^\phi \in F$. A pair $(\mathbf{s}, \mathbf{t}) \in \Sigma^\star \times \Delta^\star$ is a **translation pair** if there is a translation form ϕ of length I in $\mathcal{T}$ such that $I = |\mathbf{s}|$ and $\mathbf{t} = \bar{\mathbf{t}}_1^\phi \bar{\mathbf{t}}_2^\phi \ldots \bar{\mathbf{t}}_I^\phi$. By $d(\mathbf{s}, \mathbf{t})$ we will denote the set of translation forms[2] in $\mathcal{T}$ associated with the pair $(\mathbf{s}, \mathbf{t})$. A **rational translation** is the set of all translation pairs of some finite-state transducer $\mathcal{T}$.

This definition of a finite-state transducer is similar to the definition of a regular or finite-state grammar $\mathcal{G}$. The main difference is that in a finite-state grammar, the set of target symbols Δ does not exist, and the transitions are defined on $Q \times \Sigma \times Q$. A translation form is the transducer counterpart of a **derivation** in a finite-state grammar, and the concept of rational translation is reminiscent of the concept of (**regular**) **language**, defined as the set of strings associated with the derivations in the grammar $\mathcal{G}$.

Rational translations exhibit many properties similar to those shown for regular languages (Berstel 1979). One of these properties can be stated as follows (Berstel 1979):

Theorem 1
$T \subseteq \Sigma^\star \times \Delta^\star$ is a rational translation if and only if there exist an alphabet Γ, a regular language $L \subset \Gamma^\star$, and two morphisms $h_\Sigma : \Gamma^\star \to \Sigma^\star$ and $h_\Delta : \Gamma^\star \to \Delta^\star$, such that $T = \{(h_\Sigma(w), h_\Delta(w)) \mid w \in L\}$.

1 By Δ^* and Σ^*, we denote the set of finite-length strings on Δ and Σ, respectively.
2 To simplify the notation, we will remove the superscript ϕ from the components of a translation form if no confusion is induced.

As will be discussed later, this theorem directly suggests the transducer inference methods proposed in this article.

3. Statistical Translation Using Finite-State Transducers

In the **statistical translation** framework, the translation of a given source string $\mathbf{s}$ in $\Sigma^\star$ is a string $\hat{\mathbf{t}} \in \Delta^\star$, such that[3]

$$\hat{\mathbf{t}} = \underset{\mathbf{t}\in\Delta^\star}{\operatorname{argmax}} \Pr(\mathbf{t} \mid \mathbf{s}) = \underset{\mathbf{t}\in\Delta^\star}{\operatorname{argmax}} \Pr(\mathbf{s}, \mathbf{t}) \tag{2}$$

$\Pr(\mathbf{s}, \mathbf{t})$ can be modeled by the stochastic extension of a finite-state transducer. A **stochastic finite-state transducer**, $\mathcal{T}_P$, is defined as a tuple $\langle \Sigma, \Delta, Q, q_0, p, f \rangle$, in which Q, q_0, Q, Σ, and Δ are as in the definition of a finite-state transducer and p and f are two functions $p : Q \times \Sigma \times \Delta^\star \times Q \to [0,1]$ and $f : Q \to [0,1]$ that satisfy, $\forall q \in Q$,

$$f(q) + \sum_{(\mathrm{a},\omega,\mathrm{q}')\in\Sigma\times\Delta^\star\times\mathrm{Q}} p(q, a, \omega, q') = 1$$

In this context, $\mathcal{T}$ will denote the natural finite-state transducer associated with a stochastic finite-state transducer $\mathcal{T}_P$ (**characteristic finite-state transducer**). The set of transitions of $\mathcal{T}$ is the set of tuples $(q, s, \mathbf{t}, q')$ in $\mathcal{T}_P$ with probabilities greater than zero, and the set of final states is the set of states with nonzero final-state probabilities.

The **probability of a translation pair** $(\mathbf{s}, \mathbf{t}) \in \Sigma^\star \times \Delta^\star$ according to $\mathcal{T}_P$ is the sum of the probabilities of all the translation forms of $(\mathbf{s}, \mathbf{t})$ in $\mathcal{T}$:

$$P_{\mathcal{T}_P}(\mathbf{s}, \mathbf{t}) = \sum_{\phi\in d(\mathbf{s},\mathbf{t})} P_{\mathcal{T}_P}(\phi)$$

where the probability of a translation form ϕ (as defined in equation (1)) is

$$P_{\mathcal{T}_P}(\phi) = \prod_{i=0}^{I} p(q_{i-1}, \mathrm{s}_i, \bar{\mathbf{t}}_i, q_i) \cdot f(q_I) \tag{3}$$

that is, the product of the probabilities of all the transitions involved in ϕ.

We are interested only in transducers without **useless states**, that is, those in which for every state in $\mathcal{T}$, there is a path leading to a final state. If we further assume that $P_{\mathcal{T}_P}(\mathbf{s}, \mathbf{t})$ is zero when no translation form exists for $(\mathbf{s}, \mathbf{t})$ in $\mathcal{T}$, it can be easily verified that

$$\sum_{(\mathbf{s},\mathbf{t})\in\Sigma^\star\times\Delta^\star} P_{\mathcal{T}_P}(\mathbf{s}, \mathbf{t}) = 1$$

That is, $P_{\mathcal{T}_P}$ is a joint distribution on $\Sigma^\star \times \Delta^\star$ which will be called the **stochastic translation** defined by $\mathcal{T}_P$. [4]

Finally, the translation of a source string $\mathbf{s} \in \Sigma^\star$ by a stochastic finite-state transducer $\mathcal{T}_P$ is

$$\hat{\mathbf{t}} = \underset{\mathbf{t}\in\Delta^\star}{\operatorname{argmax}} P_{\mathcal{T}_P}(\mathbf{s}, \mathbf{t}) \tag{4}$$

3 For the sake of simplicity, we will denote $\Pr(X = x)$ as $\Pr(x)$ and $\Pr(X = x \mid Y = y)$ as $\Pr(x \mid y)$.

4 This concept is similar to the **stochastic regular language** for a stochastic regular grammar. In that case, the probability distribution is defined on the set of finite-length strings rather than on the set of pairs of strings.

A stochastic finite-state transducer has stochastic source and target regular languages embedded (P_i and P_o, respectively.):

$$P_i(\mathbf{s}) = \sum_{\mathbf{t}\in\Delta^\star} P_{\mathcal{T}_p}(\mathbf{s},\mathbf{t}), \quad P_o(\mathbf{t}) = \sum_{\mathbf{s}\in\Sigma^\star} P_{\mathcal{T}_p}(\mathbf{s},\mathbf{t})$$

In practice, these source or target regular languages are obtained, by dropping the target or the source symbols, respectively, from each transition of the finite-state transducer.

The following theorem naturally extends Theorem 1 to the stochastic framework (Casacuberta, Vidal, and Picó 2004):

Theorem 2
A distribution $P_T : \Sigma^\star \times \Delta^\star \to [0,1]$ is a stochastic rational translation if and only if there exist an alphabet Γ, two morphisms $h_\Sigma : \Gamma^\star \to \Sigma^\star$ and $h_\Delta : \Gamma^\star \to \Delta^\star$, and a stochastic regular language P_L such that, $\forall(\mathbf{s},\mathbf{t}) \in \Sigma^\star \times \Delta^\star$,

$$P_T(\mathbf{s},\mathbf{t}) \;=\; \sum_{\substack{\omega\in\Gamma^\star:\\ (h_\Sigma(\omega),h_\Delta(\omega))=(\mathbf{s},\mathbf{t})}} P_L(\omega) \tag{5}$$

3.1 Search with Stochastic Finite-State Transducers

The search for an optimal $\hat{\mathbf{t}}$ in Equation (4) has proved to be a difficult computational problem (Casacuberta and de la Higuera 2000). In practice, an approximate solution can be obtained (Casacuberta 2000) on the basis of the following approximation to the probability of a translation pair (**Viterbi score of a translation**):

$$P_{\mathcal{T}_p}(\mathbf{s},\mathbf{t}) \;\approx\; V_{\mathcal{T}_p}(\mathbf{s},\mathbf{t}) \;=\; \max_{\phi\in d(\mathbf{s},\mathbf{t})} P_{\mathcal{T}_p}(\phi) \tag{6}$$

An **approximate translation** can now be computed as

$$\tilde{\mathbf{t}} = \operatorname*{argmax}_{\mathbf{t}\in\Delta^\star} V_{\mathcal{T}_p}(\mathbf{s},\mathbf{t}) = \operatorname*{argmax}_{\mathbf{t}\in\Delta^\star} \max_{\phi\in d(\mathbf{s},\mathbf{t})} P_{\mathcal{T}_p}(\phi) \tag{7}$$

This computation can be carried out efficiently (Casacuberta 1996) by solving the following recurrence by means of **dynamic programming**:

$$\max_{\mathbf{t}\in\Delta^*} V_{\mathcal{T}_p}(\mathbf{s},\mathbf{t}) \;=\; \max_{q\in Q}\big(V(|\mathbf{s}|,q)\cdot f(q)\big) \tag{8}$$

$$V(i,q) \;=\; \max_{q'\in Q, w\in\Delta^\star}\big(V(i-1,q')\cdot p(q',s_i,w,q)\big) \quad \text{if } i\neq 0, q\neq q_0 \tag{9}$$

$$V(0,q_0) \;=\; 1 \tag{10}$$

Finally, the approximate translation $\tilde{\mathbf{t}}$ is obtained as the concatenation of the target strings associated with the translation form

$$\tilde{\phi} = (q_0,\mathbf{s}_1,\bar{\mathbf{t}}_1,q_1)(q_1,\mathbf{s}_2,\bar{\mathbf{t}}_2,q_2)\ldots(q_{I-1},\mathbf{s}_{I-1},\bar{\mathbf{t}}_I,q_I),$$

corresponding to the optimal sequence of states involved in the solution to Equation (8); that is,

$$\tilde{\mathbf{t}} = \bar{\mathbf{t}}_1\,\bar{\mathbf{t}}_2\ldots\bar{\mathbf{t}}_I$$

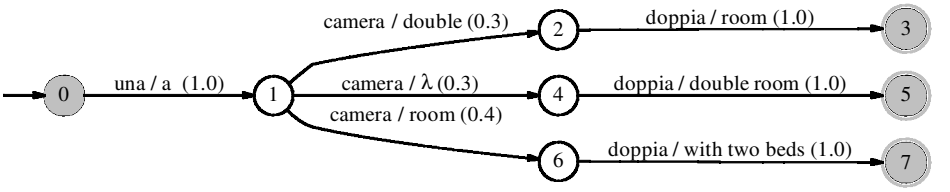

Figure 1
Example of Viterbi score-based suboptimal result. The probability $P_{\mathcal{T}_p}$ of the pair *una camera doppia/a double room* is $(1.0 \cdot 0.3 \cdot 1.0) + (1.0 \cdot 0.3 \cdot 1.0) = 0.6$. This is greater than the probability $P_{\mathcal{T}_p}$ of the pair *una camera doppia/a room with two beds*, $1.0 \cdot 0.4 \cdot 1.0 = 0.4$. However, the Viterbi score $V_{\mathcal{T}_p}$ for the first pair is $1.0 \cdot 0.3 \cdot 1.0 = 0.3$, which is lower than the Viterbi score $V_{\mathcal{T}_p}$ for the second pair, $1.0 \cdot 0.4 \cdot 1.0 = 0.4$. Therefore this second pair will be the approximate result given by equation (7).

The computational cost of the iterative version of this algorithm is $O(|\,\mathbf{s}\,| \cdot\ |Q| \cdot\ B)$, where B is the (average) branching factor of the finite-state transducer.

Figure 1 shows a simple example in which Viterbi score maximization (7) leads to a suboptimal result.

4. A Method for Inferring Finite-State Transducers

Theorems 1 and 2 establish that any (stochastic) rational translation T can be obtained as a homomorphic image of certain (stochastic) regular language L over an adequate alphabet Γ. The proofs of these theorems are constructive (Berstel 1979; Casacuberta, Vidal, and Picó 2004) and are based on building a (stochastic) finite-state transducer $\mathcal{T}$ for T by applying certain morphisms h_Σ and h_Δ to the symbols of Γ that are associated with the rules of a (stochastic) regular grammar that generates L.

This suggests the following general technique for learning a stochastic finite-state transducer, given a finite sample A of string pairs $(\mathbf{s}, \mathbf{t}) \in \Sigma^\star \times \Delta^\star$ (a **parallel corpus**):

1. Each training pair $(\mathbf{s}, \mathbf{t})$ from A is transformed into a string $\mathbf{z}$ from an **extended alphabet** Γ (strings of Γ-symbols) yielding a sample S of strings $S \subset \Gamma^\star$.
2. A (stochastic) regular grammar $\mathcal{G}$ is inferred from S.
3. The Γ-symbols of the grammar rules are transformed back into pairs of source/target symbols/strings (from $\Sigma^\star \times \Delta^\star$).

This technique, which is very similar to that proposed in García, Vidal, and Casacuberta (1987) for the inference of regular grammars, is illustrated in Figure 2.

The first transformation is modeled by the labeling function $\mathcal{L} : \Sigma^\star \times \Delta^\star \rightarrow \Gamma^\star$, while the last transformation is carried out by an "inverse labeling function" $\Lambda(\cdot)$, that is, one such that $\Lambda(\mathcal{L}(A)) = A$. Following Theorems 1 and 2, $\Lambda(\cdot)$ consists of a couple of morphisms, h_Σ, h_Δ, such that for a string $\mathbf{z} \in \Gamma^\star$, $\Lambda(\mathbf{z}) = (h_\Sigma(\mathbf{z}), h_\Delta(\mathbf{z}))$.

Without loss of generality, we assume that the method used in the second step of the proposed method consists of the inference of ***n*-grams** (Ney, Martin, and Wessel 1997) **with final states**, which are particular cases of stochastic regular grammars. This simple method automatically derives, from the strings in S, both the structure of $\mathcal{G}$ (i.e., the rules—states and transitions) and the associated probabilities.

Since Λ is typically the inverse of $\mathcal{L}$, the morphisms h_Σ and h_Δ needed in the third step of the proposed approach are determined by the definition of $\mathcal{L}$. So a key

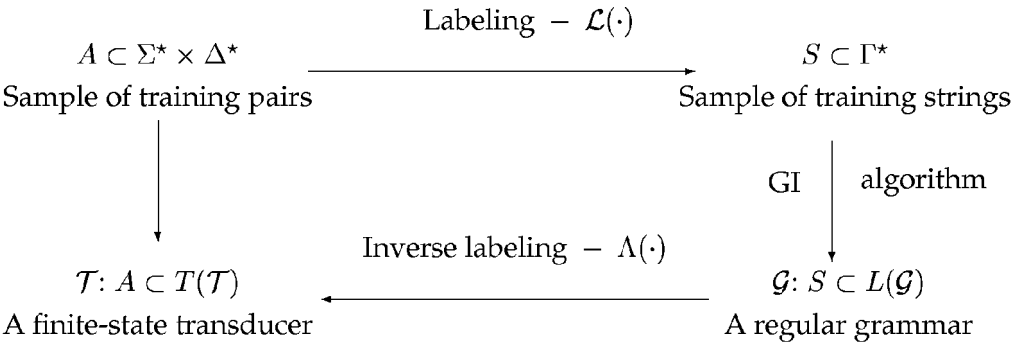

Figure 2
Basic scheme for the inference of finite-state transducers. A is a finite sample of training pairs. S is the finite sample of strings obtained from A using $\mathcal{L}$. $\mathcal{G}$ is a grammar inferred from S such that S is a subset of the language, $L(\mathcal{G})$, generated by the grammar $\mathcal{G}$. $\mathcal{T}$ is a finite-state transducer whose translation ($T(\mathcal{T})$) includes the training sample A.

point in this approach is its first step, that is, how to conveniently transform a *parallel* corpus into a *string* corpus. In general, there are many possible transformations, but if the source–target correspondences are complex, the design of an adequate transformation can become difficult. As a general rule, the labeling process must capture these source–target word correspondences and must allow for a simple implementation of the inverse labeling needed in the third step.

A very preliminary, nonstochastic version of this finite-state transducer inference technique was presented in Vidal, García, and Segarra (1989) An important drawback of that early proposal was that the methods proposed for building the $\Gamma^\star$ sentences from the training pairs did not adequately cope with the dependencies between the words of the source sentences and the words of the corresponding target sentences. In the following section we show how this drawback can be overcome using **statistical alignments** (Brown et al. 1993).

The resulting methodology is called **grammatical inference and alignments for transducer inference** (GIATI).[5] A related approach was proposed in Bangalore and Ricardi (2000b). In that case, the extended symbols were also built according to previously computed alignments, but the order of target words was not preserved. As a consequence, that approach requires a postprocess to try to restore the target words to a proper order.

4.1 Statistical Alignments

The statistical translation models introduced by Brown et al. (1993) are based on the concept of alignment between source and target words (**statistical alignment models**). Formally, an **alignment** of a translation pair $(\mathbf{s}, \mathbf{t}) \in \Sigma^\star \times \Delta^\star$ 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 $\mathbf{s}$. All the possible alignments between $\mathbf{t}$ and $\mathbf{s}$ are denoted by $\mathcal{A}(\mathbf{s}, \mathbf{t})$, and the probability of translating a given $\mathbf{s}$ into $\mathbf{t}$ by an alignment $\mathbf{a}$ is $\Pr(\mathbf{t}, \mathbf{a} \mid \mathbf{s})$.

Thus, an optimal alignment between $\mathbf{s}$ and $\mathbf{t}$ can be computed as

$$\hat{\mathbf{a}} = \underset{\mathbf{a} \in \mathcal{A}(\mathbf{s},\mathbf{t})}{\operatorname{argmax}} \Pr(\mathbf{t}, \mathbf{a} \mid \mathbf{s}) \tag{11}$$

5 In previous work, this idea was often called **morphic generator transducer inference.**

Different approaches for estimating Pr(**t**, **a** | **s**) were proposed in Brown et al. (1993). These approaches are known as models 1 through 5. Adequate software packages are publicly available for training these statistical models and for obtaining good alignments between pairs of sentences (Al-Onaizan et al. 1999; Och and Ney 2000). An example of Spanish-English sentence alignment is given below:

Example 1

¿ 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 that is aligned with the (position of the) preceding target word. A graphical representation of this alignment is shown in Figure 3.

4.2 First Step of the GIATI Methodology: Transformation of Training Pairs into Strings

The first step of the proposed method consists in a labeling process ($\mathcal{L}$) that builds a string of certain **extended symbols** from each training string pair and its corresponding statistical alignment. The main idea is to assign each word from **t** to the corresponding word from **s** given by the alignment **a**. But sometimes this assignment produces a violation of the sequential order of the words in **t**. To illustrate the GIATI methodology we will use example 2:

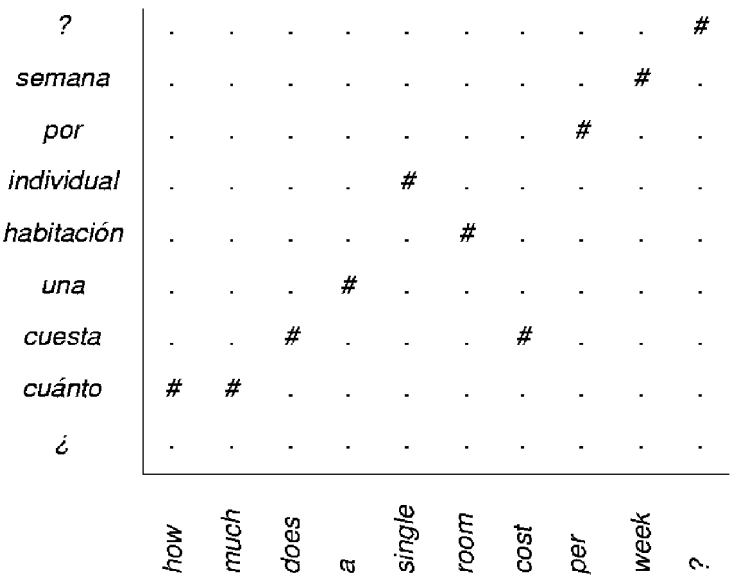

Figure 3
Graphical representation of the alignment between a source (Spanish) sentence (*¿ Cuánto cuesta una habitación individual por semana ?*) and a target (English) sentence (*How much does a single room cost per week ?*). Note the correspondence between the Spanish *cuesta* and the English *does* and *cost*. Note also that the model does not allow for alignments between sets of two or more source words and one target word.

Example 2
Let A be a training sample composed by the following pairs (Italian/English):

una camera doppia	#	*a double room*
una camera	#	*a room*
la camera singola	#	*the single room*
la camera	#	*the room*

Suitable alignments for these pairs are

una camera doppia	#	*a (1) double (3) room (2)*
una camera	#	a (1) *room (2)*
la camera singola	#	*the (1) single (3) room (2)*
la camera	#	*the (1) room (2)*

In the first pair of this example, the English word *double* could be assigned to the third Italian word (*doppia*) and the English word *room* to the second Italian word (*camera*). This would imply a "reordering" of the words *double* and *room*, which is not appropriate in our finite-state framework.

Given **s**, **t**, and **a** (source and target strings and associated alignment, respectively), the proposed transformation $\mathbf{z} = \mathcal{L}_1(\mathbf{s}, \mathbf{t})$ avoids this problem as follows:

$$|\,\mathbf{z}\,| = |\,\mathbf{s}\,|$$

$$1 \leq i \leq |\,\mathbf{z}\,|$$

$$z_i = \begin{cases} (s_i\ ,\ t_j\, t_{j+1} \ldots t_{j+l}) & \text{if } \exists j : a(j) = i \text{ and } \nexists\, j' < j : a(j') > a(j) \\ & \text{and for } j'' : j \leq j'' \leq j + l, a(j'') \leq a(j) \\ (s_i\ ,\ \lambda) & \text{otherwise} \end{cases}$$

Each word from **t** is joined with the corresponding word from **s** given by the alignment **a** if the target word order is not violated. Otherwise, the target word is joined with the first source word that does not violate the target word order.

The application of $\mathcal{L}_1$ to example 2 generates the following strings of extended symbols:

(una , a) (camera , λ) (doppia , double room)
(una , a) (camera , room)
(la , the) (camera , λ) (singola , single room)
(la , the) (camera , room)

As a more complicated example, the application of this transformation to example 1 generates the following string:

(¿ , λ) (Cuánto , how much) (cuesta , does) (una , a) (habitación , λ)
(individual , single room cost) (por , per) (semana , week) (? , ?)

In this case the unaligned token ? has an associated empty target string, and the target word *cost*, which is aligned with the source word *cuesta*, is associated with the nearby source word *individual*. This avoids a "reordering" of the target string and entails an (apparently) lower degree of nonmonotonicity. This is achieved, however, at the expense of letting the method generalize from word associations which can be considered improper from a linguistic point of view (e.g., (*cuesta*, *does*), (*individual*, *single*

room cost)). While this would certainly be problematic for general language translation, it proves not to be so harmful when the sentences to be translated come from limited-domain languages.

Obviously, other transformations are possible. For example, after the application of the above procedure, successive isolated source words (without any target word) can be joined to the first extended word which has target word(s) assigned. Let $\mathbf{z} = \mathcal{L}_1(\mathbf{s}, \mathbf{t})$ be a transformed string obtained from the above procedure and let

$$(s_{k-1}\ ,\ t_j\, t_{j+1} \ldots t_{j+m})(s_k\ ,\ \lambda) \ldots (s_{k+l-1}\ ,\ \lambda)(s_{k+l}\ ,\ t_{j+m+1} \ldots t_{j+n})$$

be a subsequence within $\mathbf{z}$. Then the subsequence

$$(s_k\ ,\ \lambda) \ldots (s_{k+l-1}\ ,\ \lambda)(s_{k+l}\ ,\ t_{j+m+1} \ldots t_{j+n})$$

is transformed by $\mathcal{L}_2$ into

$$(s_k \ldots s_{k+l-1}\, s_{k+l}\ ,\ t_{j+m+1} \ldots t_{j+n})$$

The application of $\mathcal{L}_2$ to example 2 leads to

(una , a) (camera doppia , double room)
(una , a) (camera , room)
(la , the) (camera singola , single room)
(la , the) (camera , room)

Although many other sophisticated transformations can be defined following the above ideas, only the simple $\mathcal{L}_1$ will be used in the experiments reported in this article.

4.3 Second Step of the GIATI Methodology: Inferring a Stochastic Regular Grammar from a Set of Strings

Many grammatical inference techniques are available to implement the second step of the proposed procedure. In this work, (smoothed) n-grams are used. These models have proven quite successful in many areas such as language modeling (Clarkson and Rosenfeld 1997; Ney, Martin, and Wessel 1997).

Figures 4 and 5 show the (nonsmoothed) bigram models inferred from the sample obtained using $\mathcal{L}_1$ and $\mathcal{L}_2$, respectively, in example 2. Note that the generalization achieved by the first model is greater than that of the second.

The probabilities of the n-grams are computed from the corresponding counts in the training set of extended strings. The probability of an extended word $\mathbf{z}_j = (s_i, \bar{\mathbf{t}}_i)$ given the sequence of extended words $\mathbf{z}_{i-n+1}, \ldots, \mathbf{z}_{i-1} = (s_{i-n+1}, \bar{\mathbf{t}}_{i-n+1}) \ldots (s_{i-1}, \bar{\mathbf{t}}_{i-1})$

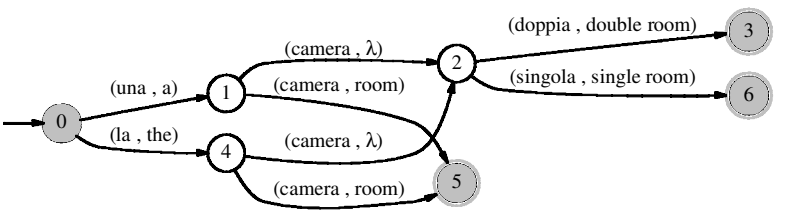

Figure 4
Bigram model inferred from strings obtained by the transformation $\mathcal{L}_1$ in example 2.

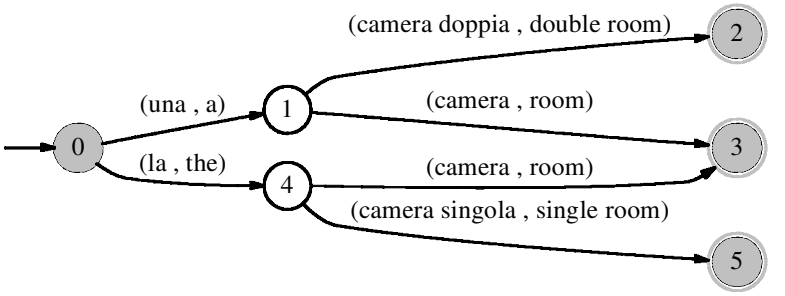

Figure 5
Bigram model inferred from strings obtained by the transformation $\mathcal{L}_2$ in example 2.

is estimated as

$$p_n(\mathbf{z}_i \mid \mathbf{z}_{i-n+1} \dots \mathbf{z}_{i-1}) = \frac{c(\mathbf{z}_{i-n+1}, \dots, \mathbf{z}_{i-1}, \mathbf{z}_i)}{c(\mathbf{z}_{i-n+1}, \dots, \mathbf{z}_{i-1})} \tag{12}$$

where $c(\cdot)$ is the number of times that an event occurs in the training set. To deal with unseen n-grams, the **back-off** smoothing technique from the CMU Statistical Language Modeling (SLM) Toolkit (Rosenfeld 1995) has been used.

The (smoothed) n-gram model obtained from the set of extended symbols is represented as a stochastic finite-state automaton (Llorens, Vilar, and Casacuberta 2002). The states of the automaton are the observed $(n-1)$-grams. For the n-gram $(\mathbf{z}_{i-n+1} \dots \mathbf{z}_i)$, there is a transition from state $(\mathbf{z}_{i-n+1} \dots \mathbf{z}_{i-1})$ to state $(\mathbf{z}_{i-n+2} \dots \mathbf{z}_i)$ with the associated extended word $\mathbf{z}_i$ and a probability $p_n(\mathbf{z}_i \mid \mathbf{z}_{i-n+1} \dots \mathbf{z}_{i-1})$. The back-off smoothing method supplied by the SLM Toolkit is represented by the states corresponding to k-grams ($k < n$) and by special transitions between k-gram states and $(k-1)$-gram states (Llorens, Vilar, and Casacuberta 2002). The final-state probability is computed as the probability of a transition with an end-of-sentence mark.

4.4 Third Step of the GIATI Methodology: Transforming a Stochastic Regular Grammar into a Stochastic Finite-State Transducer

In order to obtain a finite-state transducer from a grammar of $\mathcal{L}_1$–transformed symbols, an "inverse transformation" $\Lambda(\cdot)$ is used which is based on two simple morphisms:

$$\begin{aligned}
&\text{if } (a, b_1 b_2 \dots b_k) \in \Gamma \text{ with } a \in \Sigma \text{ and } b_1, b_2, \dots, b_k \in \Delta, \\
&h_\Sigma((a, b_1 b_2 \dots b_k)) = a \\
&h_\Delta((a, b_1 b_2 \dots b_k)) = b_1\ b_2 \dots b_k
\end{aligned}$$

It can be verified that this constitutes a true inverse transformation; that is, for every training pair $\forall(\mathbf{s}, \mathbf{t}) \in A$

$$\mathbf{s} = h_\Sigma(\mathcal{L}_1(\mathbf{s}, \mathbf{t})), \quad \mathbf{t} = h_\Delta(\mathcal{L}_1(\mathbf{s}, \mathbf{t}))$$

If $\mathbf{z}_i$ is a transition of the inferred regular grammar, where $\mathbf{z}_i = (a, b_1 b_2 \dots b_k) \in \Gamma$, the corresponding transition of the resulting finite-state transducer is $(q, a, b_1 b_2 \cdots b_k, q')$.

This construction is illustrated in Figures 6 and 7 for the bigrams of Figures 4 and 5, respectively. Note that in the second case, this construction entails the trivial addition of a few states which did not exist in the corresponding bigram. As previously discussed, the first transformation ($\mathcal{L}_1$) definitely leads to a greater translation generalization than the second ($\mathcal{L}_2$) (Casacuberta, Vidal, and Picó 2004). The probabilities associated with

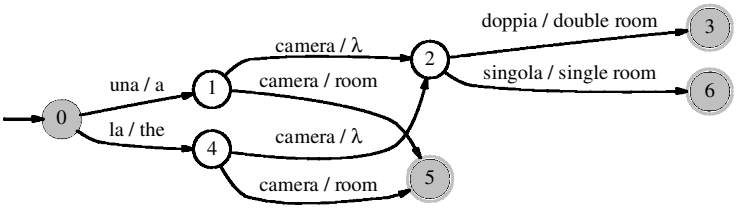

Figure 6
A finite-state transducer built from the n-gram of Figure 4.

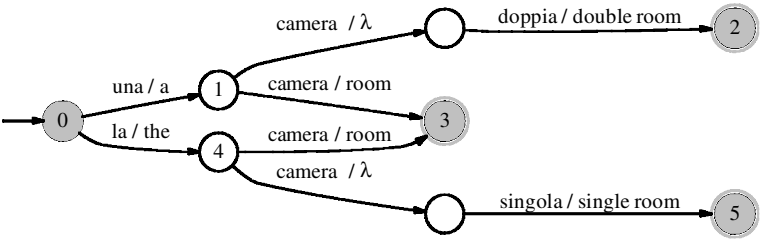

Figure 7
A finite-state transducer built from the n-gram of Figure 5.

the transitions and the final states of the finite-state transducer are the same as those of the original stochastic regular grammar.

Since we are using n-grams in the second step, a transition $(q, a, b_1 b_2 \cdots b_k, q')$ is in the finite-state transducer if the states q and q' are $(\mathbf{z}_{i-n+1} \ldots \mathbf{z}_{i-1})$, $(\mathbf{z}_{i-n+2} \ldots \mathbf{z}_i)$, respectively, and $(a, b_1 b_2 \cdots b_k)$ is $\mathbf{z}_i$. The probability of the transition is $p_n(\mathbf{z}_i \mid \mathbf{z}_{i-n+1} \ldots \mathbf{z}_{i-1})$. The transitions associated with back-off are labeled with a special source symbol (not in the source vocabulary) and with an empty target string. The number of states is the overall number of k-grams ($k < n$) that appear in the training set of extended strings plus one (the unigram state). The number of transitions is the overall number of k-grams ($k \leq n$) plus the number of states (back-off transitions). The actual number of these k-grams depends on the degree of nonmonotonicity of the original bilingual training corpus. If the corpus if completely monotone, this number would be approximately the same as the number of k-grams in the source or target parts of the training corpus. If the corpus in not monotone, the vocabulary of expanded strings becomes large, and the number of k-grams can be much larger than the number of training source or target k-grams. As a consequence, an interesting property of this type of transformations is that the source and target languages embedded in the final finite-state transducer are more constrained than the corresponding n-gram models obtained from either the source or the target strings, respectively, of the same training pairs (Casacuberta, Vidal, and Picó 2004).

While n-grams are deterministic (hence nonambiguous) models, the finite-state transducers obtained after the third-step inverse transformations (h_Σ, h_Δ) are often nondeterministic and generally ambiguous; that is, there are source strings which can be parsed through more than one path. This is in fact a fundamental property, directly coming from expression (5) of Theorem 2, on which the whole GIATI approach is essentially based. As a consequence, all the search issues discussed in Section 3.1 do apply to GIATI-learned transducers.

5. Experimental Results

Different translation tasks of different levels of difficulty were selected to assess the capabilities of the proposed inference method in the framework of the EuTrans project (ITI et al. 2000): two Spanish-English tasks (EuTrans-0 and EuTrans-I), an Italian-English task (EuTrans-II) and a Spanish-German task (EuTrans-Ia). The EuTrans-0 task, with a large semi-automatically generated training corpus, was used for studying the convergence of transducer learning algorithms for increasingly large training sets (Amengual et al. 2000). In this article it is used to get an estimation of performance limits of the GIATI technique by assuming an unbounded amount of training data. The EuTrans-I task was similar to EuTrans-0 but with a more realistically sized training corpus. This corpus was defined as a first benchmark in the EuTrans project, and therefore results with other techniques are available. The EuTrans-II task, with a quite small and highly spontaneous natural training set, was a second benchmark of the project. Finally, EuTrans-Ia was similar to EuTrans-I, but with a higher degree of nonmonotonicity between corresponding words in input/output sentence pairs.

Tables 1, 4, and 7 show some important features of these corpora. As can be seen in these tables, the training sets of EuTrans-0, EuTrans-I and EuTrans-Ia contain non-negligible amounts of repeated sentence pairs. Most of these repetitions correspond to simple and/or usual sentences such as *good morning*, *thank you*, and *do you have a single room for tonight*. The repetition rate is quite significant for EuTrans-0, but it was explicitly reduced in the more realistic benchmark tasks EuTrans-I and EuTrans-Ia. It is worth noting, however, that no repetitions appear in any of the test sets of these tasks. While repetitions can be helpful for probability estimation, they are completely useless for inducing the transducer structure. Moreover, since no repetitions appear in the test sets, the estimated probabilities will not be as useful as they could be if test data repetitions exhibited the same patterns as those in the corresponding training materials.

In all the experiments reported in this article, the approximate optimal translations (equation (7)) of the source test strings were computed and the word error rate (WER), the sentence error rate (SER), and the bilingual evaluation understudy (BLEU) metric for the translations were used as assessment criteria. The WER is the minimum number of substitution, insertion, and deletion operations needed to convert the word string hypothesized by the translation system into a given single reference word string (ITI et al. 2000). The SER is the result of a direct comparison between the hypothesized and reference word strings as a whole. The BLEU metric is based on the n-grams of the hypothesized translation that occur in the reference translations (Papineni et al 2001). The BLEU metric ranges from 0.0 (worst score) to 1.0 (best score).

5.1 The Spanish-English Translation Tasks

A Spanish-English corpus was semi-automatically generated in the first phase of the EuTrans project (Vidal 1997). The domain of the corpus involved typical human-to-human communication situations at a reception desk of a hotel.

A summary of this corpus (EuTrans-0) is given in Table 1 (Amengual et al 2000; Casacuberta et al. 2001). From this (large) corpus, a small subset of ten thousand training sentence pairs (EuTrans-I) was randomly selected in order to approach more realistic training conditions (see also Table 1). From these data, completely disjoint training and test sets were defined. It was guaranteed, however, that all the words in the source test sentences were contained in both training sets (closed vocabulary).

Results for the EuTrans-0 and EuTrans-I corpora are presented in Tables 2 and 3, respectively. The best results obtained using the proposed technique were 3.1% WER

Table 1
The Spanish-English 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.

		Spanish	English
EuTrans-0 Train:	Sentence pairs	490,000	
	Distinct pairs	168,629	
	Running words	4,655,000	4,802,000
	Vocabulary	686	513
EuTrans-I Train:	Sentence pairs	10,000	
	Distinct pairs	6,813	
	Running words	97,131	99,292
	Vocabulary	683	513
Test:	Sentences	2,996	
	Running words	35,023	35,590
EuTrans-0	Bigram test perplexity	6.8	5.8
EuTrans-I	Bigram test perplexity	8.6	6.3

Table 2
Results with the standard corpus EuTrans-0. The underlying regular models were smoothed n-grams for different values of n.

n-grams	States	Transitions	WER %	SER %	BLEU
2	4,056	67,235	8.8	50.0	0.86
3	33,619	173,500	4.7	27.2	0.94
4	110,321	364,373	4.2	23.2	0.94
5	147,790	492,840	3.8	20.5	0.95
6	201,319	663,447	3.6	19.0	0.96
7	264,868	857,275	3.4	18.0	0.96
8	331,598	1,050,949	3.3	17.4	0.96
9	391,812	1,218,367	3.3	17.2	0.96
10	438,802	1,345,278	3.2	16.8	0.96
11	471,733	1,432,027	3.1	16.4	0.96
12	492,620	1,485,370	3.1	16.4	0.96

Table 3
Results with the standard corpus EuTrans-I. The underlying regular models were smoothed n-grams for different values of n.

n-grams	States	Transitions	WER %	SER %	BLEU
2	1,696	17,121	9.0	53.7	0.86
3	8,562	36,763	6.7	38.9	0.90
4	21,338	64,856	6.7	37.9	0.91
5	23,879	72,006	6.6	37.1	0.91
6	25,947	77,531	6.6	37.0	0.91
7	27,336	81,076	6.6	37.0	0.91

for EuTrans-0 and 6.6% WER for EuTrans-I. These results were achieved using the statistical alignments provided by model 5 (Brown et al. 1993; Och and Ney 2000) and smoothed 11-grams and 6-grams, respectively.

These results were obtained using the first type of transformation described in Section 4.2 ($\mathcal{L}_1$). Similar experiments with the second type of transformation ($\mathcal{L}_2$) produced slightly worse results. However, $\mathcal{L}_2$ is interesting because many of the extended symbols obtained in the experiments involve very good relations between some source word groups and target word groups which could be useful by themselves. Consequently, more research work has to be done with this second type of transformation.

The results on the (benchmark) EuTrans-I corpus can be compared with those obtained using other approaches. GIATI outperforms other finite-state techniques in similar experimental conditions (with a best result of 8.3% WER, using another transducer inference technique called OMEGA [ITI et al. 2000]). On the other hand, the best result achieved by the **statistical templates** technique (Och and Ney 2000) was 4.4% WER (ITI et al. 2000). However, this result cannot be exactly compared with that achieved by GIATI, because the statistical templates approach used an explicit (automatic) categorization of the source and the target words, while only the raw word forms were used in GIATI. Although GIATI is compatible with different forms of word categorization, the required finite-state expansion is not straightforward, and some work is still needed in order to actually allow this technique to be taken advantage of.

5.2 The Italian-English Task

The Italian-English translation task of the EuTrans project (ITI et al. 2000) consisted of spoken person-to-person telephone communications in the framework of a hotel reception desk. A text corpus was collected with the transcriptions of dialogues of this type, along with the corresponding (human-produced) translations. A summary of the corpus used in the experiments (EuTrans-II) is given in Table 4. There was a small overlap of seven pairs between the training set and the test set, but in this case, the vocabulary was not closed (there were 107 words in the test set that did not exist in the training-set vocabulary). The processing of words out of the vocabulary was very simple in this experiment: If the word started with a capital letter, the translation was the source word; otherwise it was the empty string.

The same translation procedure and evaluation criteria used for EuTrans-0 and EuTrans-I were used for EuTrans-II. The results are reported in Table 5.

Table 4
The EuTrans-II corpus. There was a small overlap of seven pairs between the training and test sets, but 107 source words in the test set were not in the (training-set-derived) vocabulary.

		Italian	English
Train:	Sentence pairs	3,038	
	Running words	55,302	64,176
	Vocabulary	2,459	1,712
Test:	Sentences	300	
	Running words	6,121	7,243
	Bigram test perplexity	31	25

Table 5
Results with the standard EuTrans-II corpus. The underlying regular models were smoothed n-grams (Rosenfeld 1995) for different values of n.

n-grams	States	Transitions	WER %	SER %	BLEU
2	5,909	49,701	27.2	96.7	0.56
3	24,852	97,893	27.3	96.0	0.56
4	54,102	157,073	27.4	96.0	0.56

Table 6
Results with the standard EuTrans-II corpus. The underlying regular models were smoothed n-grams (Rosenfeld 1994) for different values of n. The training set was (automatically) segmented using a priori knowledge. The statistical alignments were constrained to be within each parallel segment.

n-grams	States	Transitions	WER %	SER %	BLEU
2	6,300	52,385	24.9	93.0	0.62
3	26,194	102,941	25.5	93.3	0.61
4	56,856	164,972	25.5	93.3	0.61

This corpus contained many long sentences, most of which were composed of rather short segments connected by punctuation marks. Typically, these segments can be monotonically aligned with corresponding target segments using a simple dynamic programming procedure (prior segmentation) (ITI et al. 2000). We explored computing the statistical alignments within each pair of segments rather than in the entire sentences. Since the segments were shorter than the whole sentences, the alignment probability distributions were better estimated. In the training phase, extended symbols were built from these alignments, and the strings of extended symbols corresponding to the segments of the same original string pair were concatenated. Test sentences were directly used, without any kind of segmentation.

The translation results using prior segmentation are reported in Table 6. These results were clearly better than those of the corresponding experiments with nonsegmented training data.

The accuracy of GIATI in the EuTrans-II experiments was significantly worse than that achieved in EuTrans-I, and best performance is obtained with a lower-order n-gram. One obvious reason for this behavior is that this corpus is far more spontaneous than the first one, and consequently, it has a much higher degree of variability. Moreover, the training data set is about three times smaller than the corresponding data of EuTrans-I, while the vocabularies are three to four times larger.

The best result achieved with the proposed technique on EuTrans-II was 24.9% WER, using prior segmentation of the training pairs and a smoothed bigram model. This result was comparable to the best among all those reported in RWTH Aachen and ITI (1999). The previously mentioned statistical templates technique achieved 25.1% WER in this case. In this application, in which categories are not as important as in EuTrans-I, statistical templates and GIATI achieved similar results.

Table 7
The Spanish-German corpus. There was no overlap between training and test sets and no out-of-vocabulary words in the test set.

		Spanish	German
Train:	Sentence pairs	10,000	
	Distinct pairs	6,636	
	Running words	96,043	90,099
	Vocabulary	6,622	4,890
Test:	Sentences	2,862	
	Running words	33,542	31,103
	Bigram test perplexity	8.3	6.6

Table 8
Results with the standard corpus EuTrans-Ia. The underlying regular models were smoothed n-grams for different values of n.

n-grams	States	Transitions	WER %	SER %	BLEU
2	2,441	21,181	16.0	78.1	0.74
3	10,592	43,294	11.3	65.3	0.82
4	24,554	74,412	10.6	62.3	0.83
5	27,748	83,553	10.6	62.5	0.83
6	30,501	91,055	10.6	62.4	0.83
7	32,497	96,303	10.7	62.7	0.83

5.3 The Spanish-German Task

The Spanish-German translation task is similar to EuTrans-I, but here the target language is German instead of English. It should be noted that Spanish syntax is significantly more different from that of German than it is from that of English, and therefore, the corresponding corpus exhibited a higher degree of nonmonotonicity. The features of this corpus (EuTrans-Ia) are summarized in Table 7. There was no overlap between training and test sets, and the vocabulary was closed.

The translation results are reported in Table 8. As expected from the higher degree of nonmonotonicity of the present task, these results were somewhat worse than those achieved with EuTrans-I. This is consistent with the larger number of states and transitions of the EuTrans-Ia models: The higher degree of word reordering of these models is achieved at the expense of a larger number of extended words.

The way GIATI transducers cope with these monotonicity differences can be more explicitly illustrated by estimating how many target words are produced after some delay with respect to the source. While directly determining (or even properly defining) the actual production delay for each individual (test) word is not trivial, an approximation can be indirectly derived from the number of target words that are preceded by sequences of λ symbols (from target-empty transitions) in the parsing of a source test text with a given transducer. This has been done for the EuTrans-I and EuTrans-Ia test sets with GIATI transducers learned with $n = 6$. On the average, the EuTrans-I transducer needed to introduce delays ranging from one to five positions for approximately 15% of the English target words produced, while the transducer for EuTrans-Ia had to introduce similar delays for about 20% of the German target words produced.

5.4 Error Analysis

The errors reported in the previous sections can be attributed to four main factors:

1. Correct translations which differ from the given (single) reference
2. Wrong alignments of training pairs
3. Insufficient or improper generalization of n-gram-based GIATI learning
4. Wrong approximate Viterbi score–based search results

An informal inspection of the target sentences produced by GIATI in all the experiments reveals that the first three factors are responsible for the vast majority of errors. Table 9 shows typical examples for the results of the EuTrans-I experiment with 6-gram-based GIATI transducers.

The first three examples correspond to correct translations which have been wrongly counted as errors (factor 1). Examples 4 and 5 are probably due to alignment problems (factor 2). In fact, more than half of the errors reported in the EuTrans-I experiments are due to misuse or misplacement of the English word *please*. Examples 6–8 can also be considered minor errors, probably resulting from factors 2 and 3. Examples 9 and 10 are clear undergeneralization errors (factor 3). These errors could have been easily overcome through an adequate use of bilingual lexical categorization. Examples 11 and 12, finally, are more complex errors that can be attributed to (a combination of) factors 2, 3, and 4.

6. Conclusions

A method has been proposed in this article for inferring stochastic finite-state transducers from stochastic regular grammars. This method, GIATI, allowed us to achieve good results in several language translation tasks with different levels of difficulty. It works better than other finite-state techniques when the training data are scarce and achieves similar results with sufficient training data.

The GIATI approach produces transducers which *generalize* the information provided by the (aligned) training pairs. Thanks to the use of n-grams as a core learning procedure, a wide range of generalization degrees can be achieved. It is well-known that a 1-gram entails a maximum generalization, allowing (extended) words to follow one another. On the other hand, for sufficiently large m, a (nonsmoothed) m-gram is just an exact representation of the training strings (of extended words, in our case). Such a representation can thus be considered a simple "translation memory" that just contains the (aligned) training pairs. For any new source sentence, this "memory" can be easily and quite efficiently searched through finite-state parsing. For other intermediate values of n, $1<n<m$, GIATI obtains increasing degrees of generalization. As in the case of language modeling, the generalization degree (n) has to be tuned so as to take maximum advantage of the available training data. As training pairs become scarce, more generalization is needed to allow GIATI to adequately accept new test sentences. This behavior can be clearly observed throughout the results presented in this article.

Another feature of the GIATI approach is the use of smoothed n-grams of extended words as the basic mechanism for producing smoothed transducers. The combination of this feature with the intrinsic generalization provided by the n-gram modeling itself has proved very adequate to deal with the problem of unseen source (sub)strings.

Obviously, the overall quality of the generalizations achieved by GIATI strongly relies on the quality of the statistical alignments used and on the way word order is preserved in the source-target strings of each training pair. Taking into account the

Table 9
Examples of typical errors produced by a 6-gram-based GIATI transducer in the the EuTrans-I task. For each Spanish source sentence, the corresponding target reference and GIATI translations are shown in successive lines.

1	*¿ les importaría bajarnos nuestras bolsas a recepción ?* *would you mind sending our bags down to reception ?* *would you mind sending down our bags to reception ?*
2	*explique la cuenta de la habitación cuatro dieciséis .* *explain the bill for room number four one six .* *explain the bill for room number four sixteenth .*
3	*¿ cuánto vale una habitación doble para cinco días incluyendo desayuno ?* *how much is a double room including breakfast for five days ?* *how much is a double room for five days including breakfast ?*
4	*por favor , deseo una habitación individual para esta semana .* *I want a single room for this week , please .* *I want a single room for this week .*
5	*¿ le importaría despertarnos a las cinco ?* *would you mind waking us up at five ?* *would you mind waking us up at five , please ?*
6	*¿ hay televisión , aire acondicionado y caja fuerte en las habitaciones ?* *are there a tv , air conditioning and a safe in the rooms ?* *is there a tv , air conditioning and a safe in the rooms ?*
7	*¿ tiene habitaciones libres con teléfono ?* *do you have any rooms with a telephone available ?* *do you have any rooms with a telephone ?*
8	*¿ querría llamar a mi taxi ?* *would you call my taxi , please ?* *would you call my taxi for me , please ?*
9	*hemos de marcharnos el veintiséis de marzo por la tarde .* *we should leave on March the twenty-sixth in the afternoon .* *we should leave on March the twenty-seventh in the afternoon*
10	*por favor , ¿ nos podría dar usted la llave de la ochocientos ochenta y uno ?* *could you give us the key to room number eight eight one , please ?* *could you give us the key to room number eight oh eight one , please ?*
11	*quiero cambiarme de habitación .* *I want to change rooms .* *I want to move .*
12	*¿ tiene televisión nuestra habitación ?* *does our room have a tv ?* *does our room ?*

finite-state nature of GIATI transducers, certain heuristics have been needed in order to avoid a direct use of too-long-distance alignments ($\mathcal{L}_1$ in Section 4.2). This has proved adequate for language pairs with not too different (syntactic) structure and more so if the domains are limited. As we relax these restrictions, we might have to relax the not-too-long-distance assumption correspondingly. In this respect, the bilingual word reordering ideas of Vilar, Vidal, and Amengual (1996), Vidal (1997), and Bangalore and Ricardi (2000a) may certainly prove useful in future developments.

Acknowledgments

This work has been partially supported by the European Union under grants IT-LTR-OS-30268, IST-2001-32091 and Spanish project TIC 2000-1599-C02-01. The authors wish to thank the anonymous reviewers for their criticisms and suggestions.

References

Al-Onaizan, Yaser, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, Dan Melamed, Franz-Josef Och, David Purdy, Noah A. Smith, and David Yarowsky. 1999. Statistical machine translation. Final Report, JHU Workshop, Johns Hopkins University, Baltimore.

Amengual, Juan-Carlos, Jose-Miguel Benedí, Francisco Casacuberta, Asunción Casta no, Antonio Castellanos, Víctor Jiménez, David Llorens, Andrés Marzal, Moisés Pastor, Federico Prat, Enrique Vidal, and Juan-Miguel Vilar. 2000. The EUTRANS-I speech translation system. *Machine Translation Journal*, 15(1–2):75–103.

Bangalore, Srinivas and Giuseppe Ricardi. 2000a. Finite-state models for lexical reordering in spoken language translation. In *Proceedings of the International Conference on Speech and Language Processing*, Beijing, China, October.

Bangalore, Srinivas and Giuseppe Ricardi. 2000b. Stochastic finite-state models for spoken language machine translation. In *Proceedings of the Workshop on Embedded Machine Translation Systems, North American Association for Computational Linguistics*, pages 52–59, Seattle, May.

Bangalore, Srinivas and Giuseppe Ricardi. 2001. A finite-state approach to machine translation. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics 2001*, Pittsburgh, May.

Berstel, Jean. 1979. *Transductions and context-free languages*. B. G. Teubner, Stuttgart.

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–310.

Casacuberta, Francisco. 1996. Maximum mutual information and conditional maximum likelihood estimation of stochastic regular syntax-directed translation schemes. In *Grammatical Inference: Learning Syntax from Sentences* (volume 1147 of Lecture Notes on Computer Science). Springer-Verlag, Berlin and Heidelberg, pages 282–291.

Casacuberta, Francisco. 2000. Inference of finite-state transducers by using regular grammars and morphisms. In *Grammatical Inference: Algorithms and Applications* (volume 1891 of Lecture Notes in Artificial Intelligence). Springer-Verlag, Berlin and Heidelberg, pages 1–14.

Casacuberta, Francisco and Colin de la Higuera. 2000. Computational complexity of problems on probabilistic grammars and transducers. In *Grammatical Inference: Algorithms and Applications* (volume 1891 of Lecture Notes in Artificial Intelligence). Springer-Verlag, Berlin and Heidelberg, pages 15–24.

Casacuberta, Francisco, David Llorens, Carlos Martínez, Sirko Molau, Francisco Nevado, Hermann Ney, Moiseés Pastor, David Picó, Alberto Sanchis, Enrique Vidal, and Juan-Miguel Vilar. 2001. Speech-to-speech translation based on finite-state transducers. In *Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing*, volume 1. IEEE Press, Piscataway, NJ, pages 613–616.

Casacuberta, Francisco, Enrique Vidal, and David Picó. 2004. Inference of finite-state transducers from regular languages. *Pattern Recognition*, forthcoming.

Clarkson, Philip and Ronald Rosenfeld. 1997. Statistical language modeling using the CMU-Cambridge toolkit. In *Proceedings of EUROSPEECH*, volume 5, pages 2707–2710, Rhodes, September.

Fu, King-Sun. 1982. *Syntactic pattern recognition and applications*. Prentice-Hall, Englewood Cliffs, NJ.

García, Pedro, Enrique Vidal, and Francisco Casacuberta. 1987. Local languages, the successor method, and a step towards a general methodology for the inference of regular grammars. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(6):841–845.

González, Jorge, Ismael Salvador, Alejandro Toselli, Alfons Juan, Enrique Vidal, and Francisco Casacuberta. 2000. Offline recognition of syntax-constrained cursive handwritten text. In *Advances in Pattern Recognition* (volume 1876 of Lecture Notes in Computer Science). Springer-Verlag, Berlin and Heidelberg, pages 143–153.

Hazen, Timothy, I. Lee Hetherington, and Alex Park. 2001. FST-based recognition techniques for multi-lingual and multi-domain spontaneous speech. In *Proceedings of EUROSPEECH2001*, pages 1591–1594, Aalborg, Denmark, September.

ITI, FUB, RWTH Aachen, and ZERES. 2000. Example-based language translation systems: Final report. Technical Report D0.1c, Instituto Tecnológico de Informática, Fondazione Ugo Bordoni, Rheinisch Westfälische Technische Hochschule Aachen Lehrstuhl für Informatik V and Zeres GmbH Bochum. Information Technology. Long Term Research Domain. Open scheme.

Knight, Kevin and Yaser Al-Onaizan. 1998. Translation with finite-state devices. In *Proceedings of the Fourth. AMTA Conference* (volume 1529 of Lecture Notes in Artificial Intelligence). Springer-Verlag, Berlin and Heidelberg, pages 421–437.

Llorens, David, Juna-Miguel Vilar, and Francisco Casacuberta. 2002. Finite state language models smoothed using n-grams. *International Journal of Pattern Recognition and Artificial Intelligence*, 16(3):275–289.

Mäkinen, Erkki. 1999. Inferring finite transducers. Technical Report A-1999-3, University of Tampere, Tampere, Finland.

Mohri, Mehryar. 1997. Finite-state transducers in language and speech processing. *Computational Linguistics*, 23(2):1–20.

Mou, Xiaolong, Stephanie Seneff, and Victor Zue. 2001. Context-dependent probabilistic hierarchical sub-lexical modelling using finite-state transducers. In *Proceedings of EUROSPEECH2001*, pages 451–454, Aalborg, Denmark, September.

Ney, Hermann, Sven Martin, and Frank Wessel. 1997. Statistical language modeling using leaving-one-out. In S. Young and G. Bloothooft, editors, *Corpus-Based Statiscal Methods in Speech and Language Processing*. Kluwer Academic, Dordrecht, the Netherlands, pages 174–207.

Och, Franz-Josef and Hermann Ney. 2000. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 440–447, Hong Kong, October.

Oncina, José, Pedro García, and Enrique Vidal. 1993. Learning subsequential transducers for pattern recognition interpretation tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(5):448–458.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: A method for automatic evaluation of machine translation. Technical Report RC22176(W0109-022), IBM Research Division, Yorktown Heights, NY, September.

Rosenfeld, Ronald. 1995. The CMU statistical language modeling toolkit and its use in the 1994 ARPA CSR evaluation. In *Proceedings of the ARPA Spoken Language Technology Workshop*, Princeton, NJ. Morgan Kaufmann, San Mateo, CA.

RWTH Aachen and ITI. 1999. Statistical modeling techniques and results and search techniques and results. Technical Report D3.1a and D3.2a, Rheinisch Westfälische Technische Hochschule Aachen Lehrstuhl für Informatik VI and Instituto Tecnológico de Informática. Information Technology. Long Term Research Domain. Open scheme.

Segarra, Encarna, María-Isabel Galiano Emilio Sanchis, Fernando García, and Luis Hurtado. 2001. Extracting semantic information through automatic learning techniques. In *Proceedings of the Spanish Symposium on Pattern Recognition and Image Analysis*, pages 177–182, Benicasim, Spain, May.

Seward, Alexander. 2001. Transducer optimizations for tight-coupled decoding. In *Proceedings of EUROSPEECH2001*, pages 1607–1610, Aalborg, Denmark, September.

Vidal, Enrique. 1997. Finite-state speech-to-speech translation. In *Proceedings of the International Conference on Acoustic Speech and Signal Processing*, Munich. IEEE Press, Piscataway, NJ, pages 111–114.

Vidal, Enrique, Francisco Casacuberta, and Pedro García. 1995. Grammatical inference and automatic speech recognition. In A. Rubio, editor, *New Advances and Trends in Speech Recognition and Coding* (volume 147 of NATO-ASI Series F: Computer and

Systems Sciences). Springer-Verlag, Berlin and Heidelberg, pages 174–191.
Vidal, Enrique, Pedro García, and Encarna Segarra. 1989. Inductive learning of finite-state transducers for the interpretation of unidimensional objects. In R. Mohr, T. Pavlidis, and A. Sanfeliu, editors, *Structural Pattern Analysis*. World Scientific, Singapore, pages 17–35.
Vilar, Juan-Miguel. 2000. Improve the learning of subsequential transducers by using alignments and dictionaries. In *Grammatical Inference: Algorithms and Applications* (volume 1891 of *Lecture Notes in Artificial Intelligence*). Springer-Verlag, Berlin and Heidelberg, pages 298–312.
Vilar, Juan-Miguel, Enrique Vidal, and Juan-Carlos Amengual. 1996. Learning extended finite state models for language translation. In András Kornai, editor, *Proceedings of the Extended Finite State Models of Language Workshop*, pages 92–96, Budapest, August.