Stochastic K-TSS bi-languages for Machine Translation

M. Inés Torres
Depto. de Electricidad y Electrónica
Universidad del País Vasco
Bilbao, Spain
manes.torres@ehu.es

Francisco Casacuberta
Instituto Tecnológico de Informática
Universidad Politécnica de Valencia
Valencia, Spain
fcn@iti.upv.es

Abstract

One of the approaches to statistical machine translation is based on joint probability distributions over some source and target languages. In this work we propose to model the joint probability distribution by stochastic regular *bi-languages*. Specifically we introduce the stochastic k-testable in the strict sense *bi-languages* to represent the joint probability distribution of source and target languages. With this basis we present a reformulation of the GIATI methodology to infer stochastic regular *bi-languages* for machine translation purposes.

1 Introduction

The goal of *statistical machine translation* (SMT) is to search for the sentence $\hat{\mathbf{t}}$ that maximizes the a-posteriori probability $P(\mathbf{t}|\mathbf{s})$ of the target sentence $\mathbf{t}$ being the translation of a given sentence $\mathbf{s}$ from the source language. The translation models in SMT are automatically learned from bilingual samples. In the early nineties machine translation was tackled as a pure probabilistic process by the IBM research group (Brown et al., 1993). Within the SMT framework, *stochastic-finite-state transducers* (SFSTs) have also been proposed for machine translation purposes (Bangalore and Riccardi, 2002) (Shankar et al., 2005) (Casacuberta and Vidal, 2004) (Casacuberta and Vidal, 2007) (Blackwood et al., 2009). In such a context, SMT can be viewed as the problem of computing the joint probability distribution of some source and target languages. i.e. $P(\mathbf{t}, \mathbf{s})$, inferred from a bi-lingual corpus. The joint probability distributions of pairs of strings may be modeled by a probability distribution on a set of strings based on bi-lingual units as proposed in (Bangalore and Riccardi, 2002) for SFSTs. Alternatively (Casacuberta and Vidal, 2004) (Mariño et al., 2006) proposed n-grams models of bi-lingual units. However, only a few techniques to learn finite-state transducers for machine translation purposes can be found (Bangalore and Riccardi, 2002) (Oncina et al., 1993) (Knight and Al-Onaizan, 1998) (Casacuberta and Vidal, 2007). On the other hand, a method of inference of SFST based on the inference of stochastic finite-state automata (Casacuberta and Vidal, 2004) was proposed and then used in machine translation applications (Casacuberta and Vidal, 2007) (Pérez et al., 2008) (González and Casacuberta, 2009). This method was called *grammatical inference and alignments for transducer inference* (GIATI) and is based on some important properties relating regular translations generated by finite-state-transducers and regular languages over some bi-lingual alphabet (Berstel, 1979).

On the other hand, different stochastic regular *bi-languages* can be introduced to model $P(\mathbf{s}, \mathbf{t})$ distribution. Turning to stochastic regular languages, let us note that the class of stochastic *k-testable in the strict sense* (k-TSS) languages is a subclass of stochastic regular languages that can be inferred from a set of positive training data (Torres and Varona, 2001) (Vidal et al., 2005a) (Torres and Casacuberta, 2011) by some stochastic extension of the inference algorithm in (García and Vidal, 1990). Thus, they belong to the subset of regular languages that can be used to characterize some pattern recog-

Proceedings of the 9th International Workshop on Finite State Methods and Natural Language Processing, pages 98–106,
Blois (France), July 12-15, 2011.

nition tasks. In particular, stochastic k-TSS has been used in many natural language processing tasks such as phone recognition (Galiano and Segarra, 1993), speech recognition (Torres and Varona, 2001), language identification (Guijarrubia and Torres, 2010), language modeling (Justo and Torres, 2009) or machine translation (Pérez et al., 2008).

In this work we propose to model the joint probability distribution $P(\mathbf{t}, \mathbf{s})$ by stochastic regular *bi-languages*. A first contribution of our work is the reformulation of the GIATI methodology to infer stochastic regular *bi-languages* for machine translation purposes. This proposal allows the use of some stochastic *bi-automaton* to get the sentence $\hat{\mathbf{t}}$ that corresponds to the source sentence $\hat{\mathbf{s}}$. This stochastic *bi-automaton* need to be inferred from a sample set of *bi-strings*. As a consequence, this methodology does not required any SFST as original GIATI did. Thus, there is no need to any property relating stochastic regular translations and stochastic regular languages to support the proposed method. On the other hand, different stochastic regular *bi-languages* can be introduced to model the joint probability distribution. As a second contribution we propose in this work the use of stochastic k-TSS *bi-languages* to model $Pr(\mathbf{s}, \mathbf{t})$. For this purpose we extend definitions and theorems of stochastic k-TSS languages (Vidal et al., 2005a) (Torres and Casacuberta, 2011) to stochastic k-TSS *bi-languages* and then write a corollary to the stochastic extension of the morphism theorem.

We contribute in Section 2 with some definitions of *bi-strings*, stochastic *bi-languages* and stochastic *bi-automata*. In Section 3 we propose to model the joint probability distribution through stochastic *bi-language* and then use stochastic *bi-automaton* for translation purposes. In Section 4 we deal with stochastic k-TSS *bi-languages* and *bi-automaton*, introducing some definitions and theorem applications. Then we present in Section 5 the inference of stochastic k-TSS *bi-automata* for machine translation as a reformulation of the GIATI methodology. Finally Section 6 deals with some concluding remarks and future work.

2 Stochastic regular bi-languages

In this Section we first provide the basic definitions of bi-string, stochastic regular bi-language and stochastic and deterministic finite state bi-automata proposed in this work.

Let Σ and Δ be two finite alphabets and $\Sigma^{\leq m}$ and $\Delta^{\leq n}$, the finite sets of sequences of symbols in Σ and Δ of length up to m and n respectively. Let $\Gamma \subseteq (\Sigma^{\leq m} \times \Delta^{\leq n})$ be a finite alphabet (*extended alphabet*) consisting of pairs of strings, that we call *extended symbols*, $(s_1 \dots s_i : t_1 \dots t_j) \in \Gamma$ such that $s_1 \dots s_i \in \Sigma^{\leq m}$ and $t_1 \dots t_j \in \Delta^{\leq n}$ with $0 \leq i \leq m$ and $0 \leq j \leq n$.

Definition 2.1. *A bi-language is a set of strings over an extended alphabet Γ, i.e., a set of strings of the form $\mathbf{b} = b_1 \dots b_k$ such that $b_i \in \Gamma$ for $0 \leq i \leq k$. A string over an extended alphabet Γ will be called bi-string.*

Alternatively (Kornai, 2008) defines a *bi-string* as composed by two strings and an association relation. In the same way, *bi-languages* are defined as sets of well-formed *bi-strings* that undergo the usual set-theoretic operations of intersection, union and complementation. Concatenation of such *bi-strings* is also defined in (Kornai, 2008). In this context, regular *bi-languages* were previously defined in (Kornai, 1995). In the context of machine translation, (Mariño et al., 2006) defines a *bi-language* as composed of bi-lingual units which were referred to as *tuples* extracted from alignments of a bilingual corpus. This definition could be consistent with the one provided in definition 2.1. Also in machine translation, (Bangalore and Riccardi, 2002) defines a *bi-language* corpus as consisting of source-target symbol pair sequences $(s_1 : t_1) \dots (s_i : t_i) \dots (s_n : t_n)$ such that $s_i \in L_s \cup \{\lambda\}$ and its aligned symbol $t_i \in L_t \cup \{\lambda\}$ where L_s and L_t are a couple of related languages. This definition allows for pairs of symbols by contrast with definition 2.1 where pairs of finite-length strings are considered. Finally, let us note that regular tree languages were also been referred as bilanguages (Pair and Quere, 1968) (Berger and Pair, 1978).

We are now referring to the work by (Vidal et al., 2005a). This work is a survey of probabilistic finite-state machines and related definitions and properties. In this survey, the authors provide a def-

inition of probabilistic automata that corresponds to *generative* models. Note that in classical (and non probabilistic) formal theory strings are generated by *grammars*. In this paper we are using the formalism developed in (Vidal et al., 2005a).

Given a finite alphabet Σ, a *stochastic language* is defined in (Vidal et al., 2005a) as a probability distribution over Σ^*. Let us extend this definition to consider *bi-strings* and then get stochastic *bi-languages*.

Definition 2.2. *Given two finite alphabets Σ and Δ, a stochastic bi-language $\mathcal{B}$ is a probability distribution over Γ^* where $\Gamma \subseteq (\Sigma^{\leq m} \times \Delta^{\leq n})$, $m, n \geq 0$. Let $\mathbf{z} = z_1 \dots z_{|\mathbf{z}|}$ be a bi-string such that $z_i \in \Gamma$ for $1 \leq i \leq |\mathbf{z}|$. If $Pr_{\mathcal{B}}(\mathbf{z})$ denotes the probability of the bi-string $\mathbf{z}$ under the distribution $\mathcal{B}$ then $\sum_{\mathbf{z} \in \Gamma^*} Pr_{\mathcal{B}}(\mathbf{z}) = 1$.*

Let now define a deterministic and probabilistic finite-state *bi-automaton* (DPFBA) by extending the standard definition of a deterministic and probabilistic finite-state automaton (DPFA) as follows:

Definition 2.3. *A DPFBA is a probabilistic finite-state bi-automaton $\mathcal{BA} = (Q, \Sigma,$ $\Delta, \Gamma, \delta, q_0, P_f, P)$ if Q is a finite set of states, Σ and Δ are two finite alphabets, Γ is an extended alphabet such that $\Gamma \subseteq (\Sigma^{\leq m} \times \Delta^{\leq n})$, $m, n \geq 0$, $\delta \subseteq Q \times \Gamma \times Q$ is a set of transitions of the form $(q, (\tilde{s}_i : \tilde{t}_i), q')$ where $q, q' \in Q$ and , $(\tilde{s}_i : \tilde{t}_i) \in \Gamma$, $q_0 \in Q$ is the unique initial state, $P_f : Q \rightarrow [0, 1]$ is the final-state probabilistic distribution and $P : \delta \rightarrow [0, 1]$ defines transition probabilistic distributions ($P(q, b, q') \equiv Pr(q', b|q)$ for $b \in \Gamma$ and $q, q' \in Q$) such that:*

$$P_f(q) + \sum_{b \in \Gamma, q' \in Q} P(q, b, q') = 1 \quad \forall q \in Q \qquad (1)$$

where a transition (q, b, q') is completely defined by q and b. Thus, $\forall q \in Q$, $\forall b \in \Gamma$ $|\{q' : (q, b, q')\}| \leq 1$

Finally let $\mathbf{z} \in \Gamma^*$ and let $\theta = (q_0, z_1, q_1, z_2, q_2, ..., q_{|\mathbf{z}|-1}, z_{|\mathbf{z}|}, q_{|\mathbf{z}|})$ be a path for $\mathbf{z}$ in $\mathcal{BA}$. The probability of generating θ is:

$$Pr_{\mathcal{BA}}(\theta) = \left(\prod_{j=1}^{|\mathbf{z}|} P(q_{j-1}, z_j, q_j) \right) \cdot P_f(q_{|\mathbf{z}|}) \quad (2)$$

$\mathcal{BA}$ is a DPFBA and thus unambiguous. Then, a given *bi-string* $\mathbf{z}$ can only be generated by $\mathcal{BA}$ through a unique valid path $\theta(\mathbf{z})$. Thus, the probability of generating $\mathbf{z}$ with $\mathcal{BA}$ is $Pr_{\mathcal{BA}}(\mathbf{z}) = Pr_{\mathcal{BA}}(\theta(\mathbf{z}))$.

3 Statistical translation with bi-automata

Let us consider a source and a target languages from a source vocabulary Σ and a target vocabulary Δ, respectively. The goal of machine translation is to map a sentence in the source language, i.e. a string of symbols $\mathbf{s} = s_1 \dots s_{|\mathbf{s}|}$, $s_i \in \Sigma$ into a sentence in the target language $\mathbf{t} = t_1 \dots t_{|\mathbf{t}|}$, $t_i \in \Delta$. Statistical machine translation (SMT) is based on the *noisy channel* approach (Shannon, 1948) where $\mathbf{t}$ is considered to be a noisy version of $\mathbf{s}$ (Brown et al., 1993). Thus, the translation of a given string $\mathbf{s} \in \Delta^*$ in the source language is a string $\hat{\mathbf{t}} \in \Delta^*$ in the target language such that:

$$\hat{\mathbf{t}} = \arg\max_{\mathbf{t} \in \Delta^*} Pr(\mathbf{t}|\mathbf{s})$$

Alternatively, a joint probability distribution can be used by developing $Pr(\mathbf{t}|\mathbf{s})$ in previous Equation as follows:

$$\hat{\mathbf{t}} = \arg\max_{\mathbf{t} \in \Delta^*} \frac{Pr(\mathbf{s}, \mathbf{t})}{Pr(\mathbf{s})} = \arg\max_{\mathbf{t} \in \Delta^*} Pr(\mathbf{s}, \mathbf{t}) \quad (3)$$

since, $Pr(\mathbf{s})$ does not depend on $\mathbf{t}$. Distribution $Pr(\mathbf{s}, \mathbf{t})$ can be modeled by a stochastic finite state transducer (Bangalore and Riccardi, 2002) (Casacuberta and Vidal, 2004). Alternatively in this paper we model this distribution by a stochastic regular *bi-language*.

To this end, let $\mathbf{z}$ be a *bi-string* over the extended alphabet $\Gamma \subseteq \Sigma^{\leq m} \times \Delta^{\leq n}$ such as $\mathbf{z} : \mathbf{z} = z_1 \dots z_{|\mathbf{z}|}$, $z_i = (\tilde{s}_i : \tilde{t}_i)$ where $\tilde{s}_i = s_1 \dots s_{|\tilde{s}_i|} \in \Sigma^{\leq m}$ and $\tilde{t}_i = t_1 \dots t_{|\tilde{t}_i|} \in \Delta^{\leq n}$. Extended symbols $(\tilde{s}_i : \tilde{t}_i) \in \Gamma$ have been obtained through some alignment between $\Sigma^{\leq m}$ and $\Delta^{\leq n}$. String $\mathbf{s} \in \Sigma^*$ is a sequence of substrings $\tilde{s}_i$ such as $\mathbf{s} = \tilde{s}_1 \dots \tilde{s}_{|\mathbf{z}|}$ that has been obtained through a previously segmentation procedure. In the same way string $\mathbf{t} \in \Delta^*$ is a sequence of substrings $\tilde{t}_i$ such as $\mathbf{t} = \tilde{t}_1 \dots \tilde{t}_{|\mathbf{z}|}$. Then $Pr(\mathbf{s}, \mathbf{t})$ can be calculated as follows:

$$Pr(\mathbf{s}, \mathbf{t}) = \sum_{\forall \mathbf{z} \in \Gamma^* : (h_\Sigma(\mathbf{z}), h_\Delta(\mathbf{z})) = (\mathbf{s}, \mathbf{t})} Pr(\mathbf{z}) \quad (4)$$

In such a case, $Pr(\mathbf{s},\mathbf{t})$ can be modeled by a DPFBA $\mathcal{BA}$ such as the one defined in Definition 2.3. Thus, the probability $Pr(\mathbf{s},\mathbf{t})$ according to $\mathcal{BA}$ is defined as

$$\begin{aligned} Pr_{\mathcal{BA}}(\mathbf{s},\mathbf{t}) &= \sum_{\forall \mathbf{z}\in\Gamma^*:(h_\Sigma(\mathbf{z}),h_\Delta(\mathbf{z}))=(\mathbf{s},\mathbf{t})} Pr_{\mathcal{BA}}(\mathbf{z}) \\ &= \sum_{\forall\theta\in g(s,t)} Pr_{\mathcal{BA}}(\theta) \end{aligned}$$

where $g(s,t)$ denotes the set of all possible paths in $\mathcal{BA}$ matching $(\mathbf{s},\mathbf{t})$ and $Pr_{\mathcal{BA}}(\theta)$ is calculated according to Equation 2.

3.1 The search through a stochastic finite state bi-automaton

The main goal of SMT according to Equation 3 is to find the optimal target string $\hat{\mathbf{t}}$ given a source string $\hat{\mathbf{s}}$ and given a stochastic model of the involved joint probability. When $Pr(\mathbf{s},\mathbf{t})$ is modeled by a DPFBA $\mathcal{BA}$ we need to be able to get the string $\hat{\mathbf{t}} = \tilde{t}_1 \dots \tilde{t}_{|\mathbf{z}|}$ that corresponds to the source sequence $\mathbf{s} = \tilde{s}_1 \dots \tilde{s}_{|\mathbf{z}|}$, given $Pr_{\mathcal{BA}}(\mathbf{s},\mathbf{t})$ through Equation 5. A *bi-automaton* $\mathcal{BA}$ is ambiguous with respect to the input sequence $\mathbf{s}$. Thus, all pairs $(\mathbf{s},\mathbf{t})$ matching the given input sequence $\mathbf{s}$ are considered, i.e the maximization is carried out $\forall \mathbf{t} \in \Delta^*$ instead of $\forall(\mathbf{s},\mathbf{t}) \in \Gamma^*$. As a consequence $\hat{\mathbf{t}}$ is obtained as follows:

$$\begin{aligned} \hat{\mathbf{t}} &= \arg\max_{\mathbf{t}\in\Delta^*} Pr_{\mathcal{BA}}(\mathbf{s},\mathbf{t}) \\ &= \arg\max_{\mathbf{t}\in\Delta^*} \sum_{\forall\theta\in g(s,t)} Pr_{\mathcal{BA}}(\theta) \end{aligned}$$

This search for the optimal $\hat{\mathbf{t}}$ through Equation 5 has proved to be a difficult computational problem (Casacuberta and de la Higuera, 2000). In practice Equation 5 can be computed by the so-called *maximum approximation*, which assume that the sum close the maximum term. In such a case we first estimate the optimal path $\hat{\theta}$ is obtained as:

$$\hat{\theta} = \arg\max_{\forall\theta\in g(\mathbf{s})} Pr_{\mathcal{BA}}(\theta)$$

where $g(\mathbf{s})$ denotes the set of possible paths in $\mathcal{BA}$ matching $\mathbf{s}$ and $Pr_{\mathcal{BA}}(\theta)$ is calculated by Equation 2. The approximate translation $\hat{\mathbf{t}}$ is then computed as the concatenation of the target substrings associated to the estimated path $\hat{\theta} : (q_0,(\tilde{s}_1 : \tilde{t}_1),q_1)(q_1,(\tilde{s}_2 : \tilde{t}_2),q_2).....(q_{m-1},(\tilde{s}_m : \tilde{t}_m),q_m)$ and $\hat{\mathbf{t}} = \tilde{t}_1,\tilde{t}_2,\dots,\tilde{t}_m$ by the recursive algorithm proposed in (Casacuberta and Vidal, 2004) adapted now to a *bi-automaton*.

4 Stochastic k-TSS bi-languages

Different stochastic regular *bi-languages* can be introduced to model $Pr_{\mathcal{BA}}(\mathbf{s},\mathbf{t})$ distribution in Equation 5. In particular we propose in this work the use stochastic k-TSS DPFBA. In this Section we deal with stochastic k-TSS *bi-languages* as a particular case of stochastic *bi-languages* defined in Section 2.

To this end, let us now turn to stochastic k-TSS languages which are a subclass stochastic regular languages. Stochastic k-TSS languages are defined in (Vidal et al., 2005a) and (Torres and Casacuberta, 2011) as a four-tuple $Z_k = (\Sigma, P_{I_k}, P_{F_k}, P_{T_k})$, where Σ is a finite alphabet; $P_{I_k} : \Sigma^{<k} \to [0,1]$ are the *initial* probabilities, i.e. the probability that a string $a_1 \dots a_j \in I_k \subseteq \Sigma^{<k}$ is a starting segment of a string in the language; $P_{F_k} : \Sigma^{<k} \to [0,1]$ are the *final* probabilities, i.e. the probability that a string $a_1 \dots a_j \in F_k \subseteq \Sigma^{<k}$ is a final segment of a string in the language and $P_{T_k} : \Sigma^k \to [0,1]$ are the allowed-segments probabilities, i.e. the probability that a string $a_1 \dots a_k \in (\Sigma^k - T_k)$ according to the corresponding normalization conditions. Thus, strings in the stochastic k-TSS language L_{Z_k} start with segments in I_k of length up to $k-1$, they end with segments in F_k of length up to $k-1$ and do not include segments in T_k of length k. This definition can be straightforwardly extended to consider *bi-languages* as follows:

Definition 4.1. *A stochastic k-TSS bi-language $Z_{B_k} = (\Gamma, P_{I_{B_k}}, P_{F_{B_k}}, P_{T_{B_k}})$ is a stochastic k-TSS language defined on a extended alphabet $\Gamma \subseteq \Sigma^{\leq m} \times \Delta^{\leq n}$.*

Z_{B_k} defines a probability distribution $\mathcal{B}_{Z_{B_k}}$ on Γ^*, simplified as $\mathcal{B}_k$ from now, such as for any string of *bi-strings* $\mathbf{z} \in \Gamma^*$ of size $|\mathbf{z}|$, i.e. $\mathbf{z} = z_1 \dots z_{|\mathbf{z}|}$ the probability $Pr_{\mathcal{B}_k}(\mathbf{z})$ is calculated according to:

$$\begin{cases} P_{I_k}(z_1 \dots z_{|\mathbf{z}|}) \cdot P_{F_k}(z_1 \dots z_{|\mathbf{z}|}) & \text{if } |\mathbf{z}| < k \\ P_{I_k}(z_1 \dots z_{k-1}) \cdot \prod_{i=k}^{|\mathbf{z}|} P_{T_k}(z_{i-k+1} \dots z_{i-1}, z_i) \cdot & \\ P_{F_k}(z_{|z|-(k-2)} \dots z_{|\mathbf{z}|}) & \text{if } |\mathbf{z}| \geq k \end{cases}$$

$Pr_{\mathcal{B}_k}(\mathbf{z})$ is the probability of the string $z \in \Gamma^*$ under the k-TSS distribution $\mathcal{B}_k$. Thus:

$$\sum_{\mathbf{z}\in\Gamma^*} Pr_{\mathcal{B}_k}(\mathbf{z}) = 1 \tag{5}$$

Let us now fall back to classical k-TSS to bear in mind some important theorems. An interesting subclass of k-TSS is the class of 2-TSS languages, which are known as *local languages*. There is an important generative property which relates local languages and general regular languages given by the morphism theorem (García et al., 1987), which establish that any regular language can be generated by a local language. A stochastic extension of the morphism theorem was introduced in (Vidal et al., 2005b). A stochastic regular *bi-language* is a particular case of stochastic regular languages for an *extended* alphabet $\Gamma \subseteq (\Sigma^{\leq m} \times \Delta^{\leq n})$. As a consequence, we can apply the stochastic extension of the morphism theorem in (Vidal et al., 2005b) to stochastic regular *bi-languages* and then write a corollary for this theorem as follows:

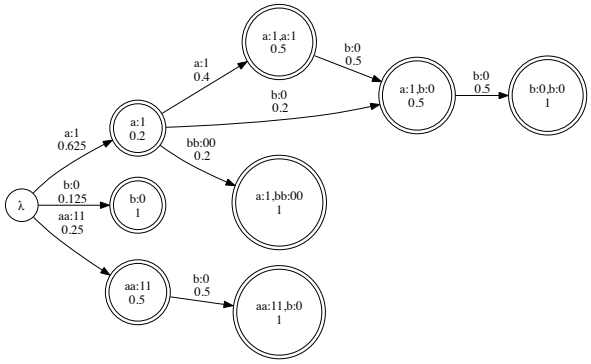
Corollary 4.1. *Let Σ and Δ be two finite alphabets, $\Gamma \subseteq (\Sigma^{\leq m} \times \Delta^{\leq n})$ be an extended alphabet and $\mathcal{B}$ a stochastic regular bi-language on Γ^*. There exists then a finite alphabet Γ', an alphabetic morphism $h : \Gamma'^* \rightarrow \Gamma^*$ and a stochastic local language $\mathcal{D}_2$ over Γ'^* such that $\mathcal{B} = h(\mathcal{D}_2)$; i.e.,*

$$\begin{aligned} Pr_{\mathcal{B}}(\mathbf{z}) &= Pr_{\mathcal{D}_2}(h^{-1}(\mathbf{z})) \\ &= \textstyle\sum_{\mathbf{y}\in h^{-1}(\mathbf{z})} Pr_{\mathcal{D}_2}(\mathbf{y}) \quad \forall \mathbf{z} \in \Gamma^* \end{aligned}$$

where $h^{-1}(\mathbf{z}) = \{\mathbf{y} \in \Gamma'^* | \mathbf{z} = h(\mathbf{y})\}$. Thus, any stochastic regular *bi-language* defined over Γ^* can be generated by a local language over some Γ'^* where Γ and Γ' are finite alphabets of *extended symbols* such that $\Gamma, \Gamma' \subseteq \Sigma^{\leq m} \times \Delta^{\leq n}$

We need now to deal with stochastic k-TSS *bi-automata* as well as with the way to get them from a training corpus. The inference of k-TSS automata was first addressed in (García and Vidal, 1990). Given a set of positive sample set R^+ of an unknown language, an efficient algorithm obtains a deterministic finite-state automaton that recognizes the smallest k-TSS language containing the sample set R^+. A preliminary form of a stochastic extension was presented in (Segarra, 1993) and then fully formalized in (Torres and Casacuberta, 2011). In that work a k-TSS DPFA is defined as a class of DPFA able to generate stochastic k-TSS languages where the unambiguity of the automaton allowed for a maximum likelihood estimation of each transition probability. This algorithm, can be easily adapted to infer a k-TSS DPFBA, $\mathcal{BA}_k$, generating a stochastic k-TSS *bi-language* by considering an *extended* alphabet of *bi-strings* $\Gamma \subseteq (\Sigma^{\leq m} \times \Delta^{\leq n})$. Example 4.1 shows the way to infer a k-TSS DPFBA $\mathcal{BA}_k$ that generates a k-TSS *bi-language* containing a previously defined sample R^+.

Example 4.1. *Let $\Sigma = \{a, b\}$ and $\Delta = \{1, 0\}$ be two finite alphabets and let $\Gamma \subseteq (\Sigma^{\leq m} \times \Delta^{\leq n})$ be the extended alphabet such as: $\Gamma = \{(a:1), (aa:11), (b:0), (bb:00)\}$. Let now R^+ be a positive sample set of a stochastic k-TSS bi-language $\mathcal{B}$ consisting of strings in Γ^* such that: $R^+ = \{(a:1), (b:0), (aa:11), (a:1)(a:1), (aa,11)(b:0), (a:1)(a:1)(b:0), (a:1)(b:0)(b:0), (a:1)(bb:00)\}$*
Then for $k = 3$
$I_3 = \{(a:1), (b:0), (aa:11), (a:1)(a:1), (aa:11)(b:0), (a:1)(b:0), (a:1)(bb:00)\}$
$P_{I_k} = \{0.125, 0.125, 0.125, 0.25, 0.125, 0.125, 0.125\}$
$F_3 = \{(a:1), (b:0), (aa:11), (a:1)(a:1), (aa:11)(b:0), (a:1)(b:0), (b:0)(b:0), (a:1)(bb:00)\}$
$P_{f_k} = \{1, 1, 1, 0.5, 1, 0.5, 1, 1\}$
The inferred bi-automaton $\mathcal{BA}_3$ is represented as:



where each state $q \in Q_k$ is labelled by a bi-string $(\tilde{s}_1 : \tilde{t}_1 \ldots \tilde{s}_i : \tilde{t}_i) \in \Gamma^i$ $i < k$ along with the probability $P_f(q)$ and each edge is labelled by a pair $\tilde{s}_i : \tilde{t}_i \in \Gamma$ such that $(q, \tilde{s}_i : \tilde{t}_i, q') \in \delta_k$ along with the probability $P_k(q, \tilde{s}_i : \tilde{t}_i, q')$.

5 Inference of k-TSS bi-automata for machine translation

In Section 3 we have propose to compute the joint probability distribution $P(\mathbf{s}, \mathbf{t})$ through some stochastic *bi-automaton* according to definitions in Section 2. Then in Section 4 we have shown how to get an stochastic k-TSS *bi-automaton* from a positive sample set of *bi-strings*. Thus, we can now propose a technique for the inference of stochastic k-TSS *bi-automata* for machine translation purposes based on GIATI methodology, which takes advantage of theoretical background previously (Casacuberta and Vidal, 2004) (Vidal et al., 2005b) (Torres and Casacuberta, 2011).

Given a finite sample set $\mathcal{S}^+$ of strings pairs $(\mathbf{s}, \mathbf{t}) \in \Sigma^* \times \Delta^*$ from a bilingual (*parallel*) corpus then

- **Step 1**: Given a pair of strings $(\mathbf{s}, \mathbf{t})$ get a *bi-string* $\mathbf{z} \in \Gamma^*$ according to some particular alignment and segmentation procedures. As a result, the sample set $\mathcal{S}^+$ of bilingual sentences $(\mathbf{s}, \mathbf{t}) \in \Sigma^* \times \Delta^*$ is transformed into a set $\mathcal{R}^+$ of *bi-strings* $\mathbf{z} \in \Gamma^*$.

$$\mathcal{S}^+ \subseteq \Sigma^* \times \Delta^* \rightarrow \mathcal{R}^+ \subseteq \Gamma^*$$

- **Step 2**: From the set of *bi-strings* $\mathcal{R}^+ \subseteq \Gamma^*$ infer the k-TSS DPFBA $\mathcal{BA}_k$ generating a stochastic k-TSS *bi-language* that includes $\mathcal{R}^+$.

$$\mathcal{R}^+ \subseteq \Gamma^* \rightarrow \mathcal{BA}_k : \mathcal{R}^+ \subseteq \mathcal{B}_{\mathcal{BA}_k}$$

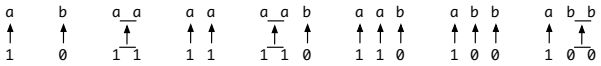
5.1 Step 1- Segmentation

The goal of this step is to get a corpus of *bi-strings* from a bilingual corpus. Let $(\mathbf{s}, \mathbf{t}) : \mathbf{s} \in \Sigma^*, \mathbf{t} \in \Delta^*$ be a pair of strings in $\mathcal{S}^+$ such that each string $\mathbf{s} \in \Sigma^*$ and each string $\mathbf{t} \in \Delta^*$ is a sequence of substrings $\tilde{s}_i$ and $\tilde{t}_i$. Then a *segmentation* procedure is required to get a *bi-string* $\mathbf{z} \in \Gamma^* : \mathbf{z} = (\tilde{s}_1, \tilde{t}_1) \dots (\tilde{s}_{|\mathbf{z}|}, \tilde{t}_{|\mathbf{z}|})$ such that string $\mathbf{s}$ is a sequence of substrings $\tilde{s}_i$ and string $\mathbf{t}$ is a sequence of substrings $\tilde{t}_i$. The *segmentation* is *monotone* if $\mathbf{s} = \tilde{s}_1 \dots \tilde{s}_{|\mathbf{z}|}$ and $\mathbf{t} = \tilde{t}_1 \dots \tilde{t}_{|\mathbf{z}|}$.

Then a relation between substrings $\tilde{s}_i \in \Sigma^*$ and substrings $\tilde{t}_i \in \Delta^*$ need also be defined. This relation was called *alignment* in (Kornai, 2008) and depends on the the application task. In this context the aim of the *alignment* is to synchronize sequences of features from two different finite alphabets (Kornai, 1995). Correspondences between source and target strings could be complex, could include long-distance and/or not consecutive associations, etc, such that the choice of a suitable *alignment* is a difficult problem to be solved. One way to deal with this problem in the machine translation framework is the use of statistical *alignments* models (Brown et al., 1993) (Och and Ney, 2003).

The choice of an adequate alignment/segmentation procedure is also related with the parsing procedure based on the *bi-automaton*. In the translation procedure, the target sentence $\hat{\mathbf{t}}$ is obtained as the concatenation of target substrings matching a given source sentence that also consists of a sequence of source substrings. A monotonic segmentation guaranties that the procedure to transform pairs of strings in $\mathcal{S}^+$ into *bi-strings* in Γ^* is reversible.

Example 5.1. *Let $\Sigma = \{a, b\}$ and $\Delta = \{0, 1\}$ be two finite alphabets. Let now $\mathcal{S}^+$ be a bilingual corpus of translations consisting in pairs of strings $(\mathbf{s}, \mathbf{t})$ such that $\mathbf{s} \in \Sigma^*$ and $\mathbf{t} \in \Delta^*$ and $\mathcal{S}^+ = \{(a, 1), (b, 0), (aa, 11), (aab, 110), (aab, 110)\}$. From this corpus we can obtain, among others, the following alignments:*



From these alignments we get the alphabet of bi-strings $\Gamma = \{(a : 1), (aa : 11), (b : 0), (bb : 00)\}$. Thus the positive sample set R^+ consisting of bi-strings in Γ^* is: $R^+ = \{(a : 1), (b : 0), (aa : 11), (a : 1)(a : 1), (aa, 11)(b : 0), (a : 1)(a : 1)(b : 0), (a : 1)(b : 0)(b : 0), (a : 1)(bb : 00)\}$

Let us to note that symbols of the general form $(\tilde{s}_i : \tilde{t}_i)$, relate strings in Σ^m, $m \geq 0$ with strings in Δ^n, $n \geq 0$. Alternatively, some machine translation models deal with pairs $(s_i : \tilde{t}_i)$ where the relation is established between symbols $s_i \in \Sigma \cup \{\lambda\}$ and strings $\tilde{t}_i \in \Delta^n$, $n \geq 0$. In such a case, the *bi-string* is defined as composed by pairs $(s_i : \tilde{t}_i) \in (\Sigma \cup \{\lambda\} \times \Delta^n), n \geq 0$.

5.2 Step 2 - Inferring a k-TSS DPFBA

Next, a stochastic finite-state *bi-automaton*, such as the one defined in Section 4, is inferred from the corpus of *bi-stings* $\mathcal{R}^+$. In particular we propose the inference of a k-TSS DPFBA $\mathcal{BA}_k$. To this end, the inference algorithm for k-TSS DPFA summarized in (Torres and Casacuberta, 2011) and then extended to get k-TSS DPFBA in Section 4 need to be applied. Example 4.1 shows the k-TSS DPFBA inferred from the positive sample set R^+ get in Example 5.1

Notice that in this case a smoothed model is required since the model has to generate any *bi-string* $\mathbf{z} \in \Gamma^*$ with a non-zero probability, even for *bi-strings* not in the stochastic *bi-language* generated by the inferred *bi-automaton*. Specific smoothing schemas has been proposed for stochastic k-TSS automata for speech recognition purposes in (Torres and Varona, 2001) and in (Llorens et al., 2002). Under a back-off scheme, these techniques adjust the maximum likelihood estimation of transition probabilities to recursively obtain probabilities to be assigned to *unseen* combinations of strings from models with decreasing the value of k, i.e. less accurate (Torres and Varona, 2001) (Llorens et al., 2002). These procedures are now straightforward extended to get *smoothed* k-TSS DPFBA. However let us to note that this procedure does not assign a non-zero probability to *bi-strings* in Γ^* which does not consists of sequences of *extended* symbols in Γ. Thus, it does not guarantee that any target string $\mathbf{t} \in \Delta^*$ could be obtained (with either high or small probability) as a liable translation of a given source string. To this end the smoothing should be applied to get a non-zero probability for any pair $(\mathbf{s}, \mathbf{t}) \in (\Sigma^* \times \Delta^*)$. This problem is similar to the one of smoothing transducers, which is still an open problem (Llorens et al., 2002).

The k-TSS DPFBA $\mathcal{BA}_k$ models the joint probability distribution $P(\mathbf{s}, \mathbf{t})$ for machine translation purposes. Thus the string $\hat{\mathbf{t}} = \tilde{t}_1 \dots \tilde{t}_{|\mathbf{z}|}$ that corresponds to the source sequence $\mathbf{s} = \tilde{s}_1 \dots \tilde{s}_{|\mathbf{z}|}$, given $Pr_{\mathcal{BA}_k}(\mathbf{s}, \mathbf{t})$ can be directly obtained parsing with the *bi-automaton* using Equation 5 according to the procedure described in Section 3.1. As a consequence this procedure does not need any final step aimed to transform back extended symbols into pairs of strings in $\Sigma^* \times \Delta^*$ since any SFST is inferred. Thus, the morphism theorems which are the basis of the classical GIATI methodology (Casacuberta and Vidal, 2004) are not now required.

6 Conclusions and future work

Machine translation can be viewed as the problem of computing the joint probability distribution of some *bi-language* inferred from a bilingual corpus. In such a context, we have proposed to represent translation models by stochastic regular *bi-languages*. To this end we have provided some specific definitions. Moreover, stochastic *bi-automata* can directly obtain the target string corresponding to a given source string.

On the other hand, we have specifically considered the stochastic k-TSS *bi-languages* to model joint probability distributions. The morphism theorem relating stochastic local languages and stochastic regular languages can now be extended to stochastic k-TSS *bi-languages* through a corollary. Moreover, stochastic k-TSS *bi-automaton* can also be inferred from a positive sample set through an extension of the inference algorithm for classical stochastic k-TSS languages.

With this basis we have reformulated the GIATI methodology to infer stochastic stochastic k-TSS *bi-languages* for machine translation purposes, which takes advantage of the knowledge about stochastic k-TSS languages and their application to natural language tasks. Moreover, the finite-state formalism allows easy integration of other automata representing target language models or acoustic models in speech translation tasks. However, the monotonic segmentation does not allow to deal with long-distance alignments which is a problem when the distance between the pair of languages is large. On the other hand smoothing techniques dealing with any pair of strings need also to be further explored.

Finally let us notice that relationship between stochastic k-TSS *bi-languages* and a subclass of stochastic regular translations, i.e. between stochastic k-TSS *bi-automata* and a subclass of stochastic finite state transducers, is going to be explored in the future.

Acknowledgments.

We would like to acknowledge support for this work to the Spanish Ministry of Sci-

ence and Innovation under the Consolider Ingenio 2010 programme (MIPRCV CSD2007-00018), grant TIN2008-06856-C05-01 and grant TIN2009-14511; to the the Basque Government under grant GIC10/158 IT375-10 and to the Generalitat Valenciana under grant Prometeo/2009/014.

References

S. Bangalore and G. Riccardi. 2002. Stochastic finite-state models for spoken language machine translation. *Machine Translation*, 17(3):165–184.

J. Berger and C. Pair. 1978. Inference for regular bilanguages. *Journal of Computer and System Sciences*, 16(1):100–122.

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

G. Blackwood, A. de Gispert, J. Brunning, and W. Byrne. 2009. Large-scale statistical machine translation with weighted finite state transducers. In *Proceeding of the 2009 conference on Finite-State Methods and Natural Language Processing*, pages 39–49, Amsterdam, The Netherlands, The Netherlands. IOS Press.

P. F. Brown, S. A. Della Pietra, V. 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 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 and E. Vidal. 2007. Learning finite-state models for machine translation. *Machine Learning*, 66(1):69–91.

I. Galiano and E. Segarra. 1993. The application of k-testable languages in the strict sense to phone recognition in automatic speech recognition. In *Proceedings of the IEE Colloquium of Grammatical Inference. Theory, Applications and Alternatives*. IEE.

P. García and E. Vidal. 1990. Inference of k-testable languages in the strict sense and application to syntactic pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(9):920–925.

P. García, E. Vidal, and F. Casacuberta. 1987. Local languages, the succesor 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.

J. González and F. Casacuberta. 2009. GREAT: a finite-state machine translation toolkit implementing a Grammatical Inference Approach for Transducer Inference (GIATI). In *Proceedings of the EACL Workshop on Computational Linguistics Aspects of Grammatical Inference*, pages 24–32, Athens, Greece, March 30.

V. Guijarrubia and M.I. Torres. 2010. Text and speech based phonotactic models for spoken language identification of basque and spanish. *Pattern Recognition Letters*, 31(6):523–532.

R. Justo and M.I. Torres. 2009. Phrase classes in two-level language models for asr. *Pattern Analysis and Applications*, 12:427–437.

K. Knight and Y. Al-Onaizan. 1998. Translation with finite-state devices. In *Lecture Notes in Computer Science*, volume 1529, pages 421–437. Springer-Verlag.

A. Kornai. 1995. *Formal Phonology*. Outstanding Dissertations in Linguistics. Garland Publishing, NewYork.

A. Kornai. 2008. *Mathematical Linguistics*. Advanced Information and Knowledge Processing. Springer, Cambridge, MA - USA.

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

J.B. Mariño, R. E. Banchs ans J.M. Crego, A.de Gispert, P. Lambert, J.A.R. Fonollosa, and M.R. Costa-juss. 2006. N-gram based machine translation. *Computational Lingustics*, 32(4):527–549.

F. Och and H. Ney. 2003. A systematic comparison of various statistical alignments models. *Computational Linguistics*, 29(1):19–51.

J. Oncina, P. García, and E. Vidal. 1993. Learning subsequential transducers for pattern recognition interpretation tasks. *pami*, 15(5):448–458.

C. Pair and A. Quere. 1968. Définition et etude des bilangages réguliers. *Information and Control*, 13(6):565–593.

A. Pérez, M.I. Torres, and F. Casacuberta. 2008. Joining linguistic and statistical methods for Spanish-to-Basque speech translation. *Speech Communication*, 50:1021–1033.

E. Segarra. 1993. *Una aproximación Inductiva a la Comprensión del Discurso Continuo*. Ph.D. thesis, Universidad Politécnica de Valencia. Advisors: Dr. P. García and Dr. E. Vidal.

K. Shankar, Y. Deng, and W. Byrne. 2005. A weighted finite state transducer translation template model for

statistical machine translation. *Natural Language Engineering*, 12:35–75, December.

C. E. Shannon. 1948. A mathematical theory of communication. *Bell Systems Technical Journal*, 27(3):379–423, July.

M. Inés Torres and F. Casacuberta. 2011. Stochastic k-tss languages. Technical report, PR&Speech Technologies Group, Depto. Electricidad y Electrónica, Universidad del País Vasco, April.

M. I. Torres and A. Varona. 2001. k-tss language models in speech recognition systems. *Computer, Speech and Language*, 15(2):127–149.

E. Vidal, F. Thollard, C. de la Higuera, F. Casacuberta, and R. Carrasco. 2005a. Probabilistic finite-state machines - part I. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 27(7):1013–1025.

E. Vidal, F. Thollard, C. de la Higuera, F. Casacuberta, and R. Carrasco. 2005b. Probabilistic finite-state machines - part II. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 27(7):1025–1039.