Obtaining Word Phrases with Stochastic Inversion Transduction Grammars for Phrase-based Statistical Machine Translation*

J.A. Sánchez and J.M. Benedí

DSIC Universidad Politécnica de Valencia, 46022 Valencia, Spain

{jandreu | jbenedi}@dsic.upv.es

Abstract

Phrase-based statistical translation systems are currently providing excellent results in real machine translation tasks. In phrase-based statistical translation systems, the basic translation units are word phrases. An important problem that is related to the estimation of phrase-based statistical models is the obtaining of word phrases from an aligned bilingual training corpus. In this work, we propose obtaining word phrases by means of a Stochastic Inversion Transduction Grammar. Preliminary experiments have been carried out on real tasks and promising results have been obtained.

1 Introduction

Machine Translation is a problem that can be addressed by means of statistical techniques (Brown, Pietra, Pietra, & Mercer, 1993). In this approach, the process of human language translation is modeled statistically by means of statistical translation models.

In order to estimate these statistical translation models, several approaches have been proposed in the literature: finite-state techniques (Bangalore & Riccardi, 2001; Casacuberta & Vidal, 2004); alignment techniques (Brown et al., 1990, 1993; Zens, Och, & Ney, 2002; Vogel et al., 2003; Koehn, 2004; Och & Ney, 2004); and syntax-based techniques (Wu, 1997; Yamada & Knight, 2001). Phrase-based techniques are based on the alignment of word phrases (Marcu & Wong, 2002; Zens et al., 2002; Vogel et al., 2003; Koehn, 2004; Tomás, Lloret, & Casacuberta, 2005). Phrase-based statistical translation systems are currently providing excellent results in real machine translation tasks. In phrase-based statistical translation systems, the basic translation units are word phrases.

An important problem that is related to phrase-based statistical translation is to automatically obtain bilingual word phrases from parallel corpora. Several methods have been defined for dealing with this problem (Och & Ney, 2003). In this work, we study a method to obtain word phrases that is based on Stochastic Inversion Transduction Grammars that was proposed in (Wu, 1997).

Stochastic Inversion Transduction Grammars (SITG) can be viewed as a restricted Stochastic Context-Free Syntax-Directed Transduction Scheme (Aho & Ullman, 1972; Maryanski & Thomason, 1979; Casacuberta, 1995). SITGs can be used to carry out a simultaneous parsing of both the input string and the output string. In this work, we propose to apply this idea to obtain aligned word phrases to be used in phrase-based translation systems. Some works along this idea have been proposed elsewhere (Zhang & Gildea, 2005).

In Section 2, we review the phrase-based machine translation approach. SITGs are reviewed in Section 3. In Section 4, we present preliminary experiments with two real tasks.

*This work has been partially supported by the *Universidad Politécnica de Valencia* with the ILETA project.

2 Phrase-based Statistical Machine Translation

The translation units in a phrase-based statistical translation system are bilingual phrases rather than simple paired words. Several systems that follow this approach have been presented in recent works (Zens et al., 2002; Koehn, 2004; Tomás et al., 2005). These systems have demonstrated excellent translation performance in real tasks.

The word-based statistical machine translation systems present some problems. One of these problems is that the classical formulation presented in (Brown et al., 1993) does not have a direct translation method. Another of these problems is the reordering problem that occurs between languages with different word orders. Finally, the problem of the unit size which must be increased in order to improve the performance of the systems. These problems can be alleviated through the use of word phrases. These larger units allow us to represent bilingual contextual information in an explicit and easy way.

The basic idea of a phrase-based statistical machine translation system consists of the following steps (Zens et al., 2002):

1. The source sentence is segmented into phrases.
2. Each source phrase is translated into a target phrase.
3. The target phrases are reordered in order to compose the target sentence.

Bilingual translation phrases are an important component of a phrase-based system. Different methods have been defined to obtain bilingual translations phrases, mainly from word-based alignments and from syntax-based models (Yamada & Knight, 2001).

In this work, we focus on learning bilingual word phrases by using Stochastic Inversion Transduction Grammars (SITGs) (Wu, 1997). This formalism allows us to obtain bilingual word phrases in a natural way from the bilingual parsing of two sentences. In addition, the SITGs allow us to easily incorporate many desirable characteristics to word phrases such as length restrictions, selection according to the word alignment probability, bracketing information, etc. We review this formalism in the following section.

3 Stochastic Inversion Transduction Grammars

Stochastic Inversion Transduction Grammars (SITGs) (Wu, 1997) can be viewed as a restricted subset of Stochastic Syntax-Directed Transduction Grammars (Aho & Ullman, 1972; Maryanski & Thomason, 1979). They can be used to simultaneously parse two strings. SITGs are closely related to Stochastic Context-Free Grammars.

Formally, a SITG in Chomsky Normal Form[1] τ_s can be defined as a tuple (N, S, W_1, W_2, R, p), where: N is a finite set of non-terminal symbols; $S \in N$ is the axiom of the SITG; W_1 is a finite set of terminal symbols of language 1; and W_2 is a finite set of terminal symbols of language 2. R is a finite set of: lexical rules of the type $A \rightarrow x/\epsilon$, $A \rightarrow \epsilon/y$, $A \rightarrow x/y$; direct syntactic rules that are noted as $A \rightarrow [BC]$; and inverse syntactic rules that are noted as $A \rightarrow \langle BC \rangle$, where $A, B, C \in N$, $x \in W_1$, $y \in W_2$ and ϵ is the empty string. When a direct syntactic rule is used in a parsing, both strings are parsed with the syntactic rule $A \rightarrow BC$. When an inverse rule is used in a parsing, one string is parsed with the syntactic rule $A \rightarrow BC$, and the other string is parsed with the syntactic rule $A \rightarrow CB$. Term p of the tuple is a function that attaches a probability to each rule.

An efficient Viterbi-like parsing algorithm that is based on a Dynamic Programing Scheme is proposed in (Wu, 1997). The algorithm is similar to the stochastic version of the CYK algorithm for Stochastic Context-Free Grammars. An extension of this algorithm will be presented below. It allows us to obtain the most probable parsing tree that simultaneously analyzes two strings, x and y. The proposed algorithm has a time

[1] A Normal Form for SITGs can be defined (Wu, 1997) by analogy to the Chomsky Normal Form for Stochastic Context-Free Grammars.

complexity of $O(|x|^3|y|^3|R|)$. It is important to note that this time complexity restricts the use of the algorithm to real tasks with short strings.

If a bracketed corpus is available, then a modified version of the parsing algorithm can be defined in order to take into account the bracketing of the strings. The modifications are similar to those proposed in (Pereira & Schabes, 1992) for the *inside* algorithm. Following the notation that is presented in (Pereira & Schabes, 1992), we can define a partially bracketed corpus as a set of sentence pairs that is annotated with parentheses that mark constituent frontiers. More precisely, a bracketed corpus Ω is a set of tuples (x, B_x, y, B_y), where x and y are strings, B_x is the bracketing of x, and B_y is the bracketing of y. Let d_{xy} be a parsing of x and y with the SITG τ_s. If the SITG does not have useless symbols, then each non-terminal that appears in each sentential form of the derivation d_{xy} generates a pair of substrings $x_i \dots x_j$ of x, $1 \leq i \leq j \leq |x|$, and $y_k \dots y_l$ of y, $1 \leq k \leq l \leq |y|$, and defines a *span* (i, j) of x and a *span* (k, l) of y. A derivation of x and y is compatible with B_x and B_y if all the spans defined by it are compatible with B_x and B_y. This compatibility can be easily defined by the function:

$$c(i,j,k,l) = \begin{cases} 1 \text{ if } (i,j) \text{ does not overlap any } b \in B_x \\ \quad \text{and,} \\ \quad \text{if } (k,l) \text{ does not overlap any } b \in B_y, \\ 0 \text{ otherwise.} \end{cases}$$

This function filters those derivations (or partial derivations) whose parsing is not compatible with the bracketing defined in the sample.

The parsing algorithm is based on the definition of:

$$\delta_{ijkl}(A) = \Pr(A \stackrel{*}{\Rightarrow} x_{i+1} \cdots x_j / y_{k+1} \cdots y_l),$$

as the probability that the non-terminal symbol A simultaneously generates the substrings $x_{i+1} \cdots x_j$ and $y_{k+1} \cdots y_l$.

Following the notation of (Wu, 1997), the parsing algorithm can be adequately modified in order to take into account only those partial parses that are compatible with the bracketing defined on the strings:

1. Initialization

$$\begin{aligned} \delta_{i-1,i,k-1,k}(A) &= p(A \to x_i/y_k) \\ &\quad 1 \leq i \leq |x|, 1 \leq k \leq |y|, \\ \delta_{i-1,i,k,k}(A) &= p(A \to x_i/\epsilon) \\ &\quad 1 \leq i \leq |x|, 0 \leq k \leq |y|, \\ \delta_{i,i,k-1,k}(A) &= p(A \to \epsilon/y_k) \\ &\quad 0 \leq i \leq |x|, 1 \leq k \leq |y|, \end{aligned}$$

2. Recursion. For all $A \in N$, and i, j, k, l such that $0 \leq i < j \leq |x|$, $0 \leq k < l \leq |y|$ and $j - i + l - k > 2$:

$$\begin{aligned} \delta_{ijkl}(A) &= c(i+1, j, k+1, l) \\ &\quad \max(\delta^{[]}_{ijkl}(A), \delta^{\langle\rangle}_{ijkl}(A)) \end{aligned}$$

where

$$\delta^{[]}_{ijkl}(A) = \max_{\substack{B,C \in N \\ i \leq I \leq j, k \leq K \leq l \\ (I-i)(j-I)+(K-k)(l-K) \neq 0}} p(A \to [BC]) \delta_{iIkK}(B) \delta_{IjKl}(C)$$

$$\delta^{\langle\rangle}_{ijkl}(A) = \max_{\substack{B,C \in N \\ i \leq I \leq j, k \leq K \leq l \\ (I-i)(j-I)+(K-k)(l-K) \neq 0}} p(A \to \langle BC \rangle) \delta_{iIKl}(B) \delta_{IjkK}(C).$$

This algorithm can be implemented to compute only those subproblems in the Dynamic Programing Scheme that are compatible with the bracketing. Thus, the time complexity is $O(|x|^3|y|^3|R|)$ for an unbracketed string, while the time complexity is $O(|x||y||R|)$ for a full bracketed string. It is important to note that the last time complexity allows us to work with real tasks with longer strings.

By keeping the argument of the maximization, the parse tree can be efficiently obtained. Each node in the tree relates two word phrases of the strings being parsed. The related word phrases can be considered to be the translation of each other. These word phrases can be used to compute the translation table of a phrase-based machine statistical translation system.

4 Experiments

In this section, we describe preliminary experiments that were carried out using SITGs. Two different corpora were used in the experiments, the EuTrans-I corpus (Casacuberta & Vidal, 2004) and the XRCE corpus (TT2, 2002). The EuTrans-I is a corpus with a small vocabulary that has been semi-automatically generated. This corpus allowed us to carry out a comprehensive set of experiments. The XRCE corpus is a real corpus that has been taken from manuals of Xerox printers.

A SITG was obtained for every experiment in this section. The SITG was used to parse paired sentences in a training sample by using the parsing algorithm described in Section 3. All pairs of word phrases that were derived from each internal node in the parse tree, except the root node, were considered for the phrase-based machine translation system. A translation table was obtained from paired word phrases, by counting the number of times that each pair appeared in the phrases. These values were then appropriately normalized.

In all the experiments in this section, the Pharaoh software (Koehn, 2004) was used as phrase-based translation system. The default values were used for the translation process, and a trigram model was used as language model. This trigram model was trained with the SRILM toolkit using the same parameters described in the Pharaoh system manual. We used the word error rate (WER) and the BLEU score to measure the results.

4.1 Experiments with the EuTrans-I corpus

The EuTrans-I corpus consists of queries, requests, and complaints made at the reception desk of a hotel (Casacuberta & Vidal, 2004). The corpus was semi-automatically generated using travel booklets. This corpus has a small vocabulary and a lot of repeated strings. For these experiments, the translation was from Spanish to English. The main characteristics of this corpus can be seen in Table 1.

Table 1: Characteristics of the EuTrans-I corpus

Training		
	Spanish	English
Sentence pairs	10,000	
Running words	97,131	99,292
Vocabulary	683	513
Test		
Sentence pairs	3,000	
Running words	35,067	35,630
3-gram test-set perp.	3.7	3.0

4.1.1 Obtaining a SITG from an aligned corpus

For this experiment, a SITG was constructed as follows: the GIZA++ toolkit (Och & Ney, 2000) was used to obtain a translation table and the corresponding probability $\Pr(f|e)$. The alignment was carried out in both directions in order to have both insertions and deletions available. This table was used to compose lexical rules of the form $A \rightarrow e/f$. Then, two additional rules of the form $A \rightarrow [AA]$ and $A \rightarrow \langle AA \rangle$ with low probability were added. The rules were then adequately normalized. This SITG was used to obtain word phrases from the training corpus by parsing each pair of aligned sentences. Then these word phrases were used by the Pharaoh system to translate the test set. The results obtained for this experiment were 19.1% WER and 0.72 BLEU.

It is important to point out that the constructed SITG did not parse all the training sentences. Even the insertions and deletions included in the SITG did not solve this problem. Therefore, the model was *smoothed* by adding all the remaining rules of the form $A \rightarrow e/\epsilon$ and $A \rightarrow \epsilon/f$ with low probability, so that all the training sentences could be parsed. The results obtained with this new SITG were 14.6% WER and 0.79 BLEU. Note that the WER results decreased notably. The reason for this was that more phrases were obtained (an increase of 100%) and their probability was better estimated. The following experiments were carried out with only *smoothed* SITGs.

4.1.2 Using bracketing information in the parsing

As Section 3 shows, the parsing algorithm for SITGs can be adequately modified in order to take bracketed sentences into account. If the bracketing respects linguistically motivated structures, then aligned phrases with linguistic information can be used. Note that this approach requires having quality parsed corpora available. This problem can be reduced by using automatically learned parsers.

This experiment was carried out to determine the performance of the translation when some kind of structural information was incorporated in the parsing. Since the training data was not bracketed, we parsed the English part of the corpus with the Charniak parser (Charniak, 2000). Only the bracketing was kept in the corpus and the other information (POStags and syntactic tags) was removed. We then obtained word phrases according to the bracketing by using the same SITG that was described in the previous section. The obtained phrases were used with the Pharaoh system. The results in this experiment were 10.7% WER and 0.83 BLEU. The results improved notably by incorporating bracketing information in the training corpus. This suggests that using some structural information could lead to important improvements.

4.1.3 Increasing the number of non-terminal symbols in the SITG

Note that the SITG described in Section 4.1.1 was very restricted since only one non-terminal symbol should be modeling the structural relations of both strings. In this experiment, we tried to determine whether moderately increasing the number of non-terminal symbols would lead to improvements since the SITG could have more flexibility to model structural relations.

Given the complexity of the parsing algorithms, only small values were tested. We generated all the syntactic rules (direct and inverse) that could be generated with a fixed number of non-terminal symbols, except for one non-terminal symbol that only generated lexical rules. Probabilities of the syntactic rules were randomly generated and were then conveniently normalized.

First, we parsed the corpus that did not include any linguistic information. Second, we parsed the corpus that included bracketing information. The results obtained are shown in Table 2. Note that the first row corresponds to the experiments in Sections 4.1.2 and 4.1.3.

Note that the results improved as the number of non-terminal symbols increased. These results confirm our hypothesis in the sense that better phrases were obtained when more flexibility in modeling structural relations was given to the model.

It should also be noted that better results were obtained when the phrases were obtained from the non bracketed corpus. The reason for this could be that in the case of phrases obtained from the non-bracketed corpus, the model had more flexibility to pair word phrases. This way, the number of different phrases decreased (see column *# param.* in Table 2). Thus, the probabilities of the phrases were better estimated. In the case of phrases obtained from the bracketed corpus, the bracketing may be imposing hard restrictions and many phrases were paired in a forced manner. Thus, the number of different phrases did not decrease as the number of non-terminal symbols increased. Therefore, the probabilities of the phrases were not well estimated.

Finally, we considered the combination of both kinds of segments. The results can be seen in the *Combined* column in Table 2. This table shows that the results improved in all cases. The reason for this could be that both kinds of segments were different in nature, and, therefore, the number of segments (column *# param.*) increased notably.

4.1.4 Using a SITG from an improved translation table

One possible way to improve the quality of the translation table consists of aligning the source and target sentences in both directions, and then choosing the alignments that appear in both directions (Och & Ney, 2003). Alignments that appear in the intersection are assumed to be of better qual-

Table 2: Results obtained when the number of non-terminal symbols ($|N|$) in the SITG was increased.

	Non bracketed			Bracketed			Combined		
$\|N\|$	WER	BLEU	# param.	WER	BLEU	# param.	WER	BLEU	# param.
1	14.6	0.79	37,508	10.7	0.83	35,300	10.5	0.84	63,292
5	9.5	0.88	28,028	10.1	0.86	37,828	8.3	0.89	58,452
10	8.7	0.89	30,260	9.2	0.86	37,372	7.6	0.88	59,833

ity. This heuristic has demonstrated to improve the results in phrase-based translation systems (Tomás et al., 2005). We tested this heuristic by computing the alignments that appeared in the intersection, and then we *smoothed* the model as described in Section 4.1.1. The obtained results are shown in Table 3.

It should be pointed out that similar results were obtained using the bracketed corpus and using the non bracketed corpus. This behavior suggests that this approach can be very useful when a bracketed corpus is not available. Note that no improvements were obtained when the number of non-terminal symbols was increased. The results from this table did not improve the results obtained in Section 4.1.3. The reason for this could be that if the number of lexical rules is reduced, then fewer word phrases are obtained and they are not well estimated.

The best result reported for this task was 4.4% WER, which was obtained by using the alignment templates approach (Och & Ney, 2000). However, that result cannot be compared exactly with the results achieved in this work because the statistical templates approach used an explicit (automatic) categorization of the source and the target words and our approach used only the raw word forms. A comparable result to the ones obtained here can be seen in (Casacuberta & Vidal, 2004), which was 6.7% WER and 0.90 BLEU. However, it should be noted that we carried out all the experiments by using the default parameters of the Pharaoh system. When we slightly tuned the parameters for the experiment in Table 2, the *Combined* column, row *10*, we obtained a WER of 7.3%.

4.2 Experiments with the XRCE corpus

This corpus consisted of manuals of Xerox printers. This is a reduced-domain task that has been defined in the TransType2 project (TT2, 2002). The usage manuals were originally written in English and were then translated to Spanish, German, and French. For these experiments, the translation was from Spanish to English. The main characteristics of this corpus are shown in Table 4.

Table 4: Characteristics of the XRCE corpus

Training		
	Spanish	English
Sentence pairs	55,761	
Running words	752,469	665,388
Vocabulary	11,051	7,957
Test		
Sentence pairs	1,125	
Running words	10,106	8,370
3gram test-set perp.	31	45

Given the size of this corpus and the complexity of the algorithms, only preliminary experiments that were analogous to those of Section 4.1.4 were carried out. The lexical rules of the model were obtained by aligning the source sentence and the target sentence in both directions and then choosing the alignments that appear in the intersection. Word phrases were then obtained with the SITG constructed from bracketed sentences and the SITG constructed from unbracketed training sentences, which had both been used in the Pharaoh system. The results obtained are shown in Table 5.

Several results for this task were reported in (Tomás et al., 2005). In that work, several ways of obtaining word phrases were

Table 3: Results obtained with the SITGs.

	Non bracketed			Bracketed			Combined		
$\|N\|$	WER	BLEU	# param.	WER	BLEU	# param.	WER	BLEU	# param.
1	10.4	0.87	31,413	10.0	0.85	35,257	8.2	0.87	57,963
5	10.0	0.87	28,095	10.1	0.86	37,781	8.6	0.89	58,368

Table 5: Results obtained with the XRCE corpus.

Non bracketed			Bracketed			Combined		
WER	BLEU	# param.	WER	BLEU	# param.	WER	BLEU	# param.
32.6	0.57	397,284	33.2	0.57	389,286	32.9	0.57	661,479

described. The best reported result was 26.2% WER and the number of different word phrases that were used for the best result was about 2.5M. The obtained word phrases were used in a phrase-based machine translation system that is different to the one used in our work. With our proposal, the number of different word phrases was about 0.4M. The different number of parameters might explain the better results obtained in Tomás's work. In his experiment, a WER of about 31% was obtained when a number of parameters of about 0.4M was used. When we slightly tuned the parameters of the Pharaoh system for the experiment in Table 5, the *Combined* column, we obtained a WER of 31.5%.

5 Conclusions

In this work, we have explored the problem of obtaining word phrases for phrase-based machine translation systems from SITGs. We have presented how the parsing algorithms for this formalism can be modified in order to take into account a bracketed corpus. Experiments were reported for two different tasks, and the results obtained were very promising.

For future work, we propose to work along different lines. First, to incorporate new linguistic information in both the parsing algorithm and in the aligned corpus. Second, to obtain better SITGs from aligned bilingual corpora. Third, to improve the SITG by estimating the syntactic rules. In addition, we also intend to address other machine translation tasks.

References

Aho, A., & Ullman, J. (1972). *The theory of parsing, translation, and compiling. volumen i: parsing.* Prentice-Hall.

Bangalore, S., & Riccardi, G. (2001). A finite-state approach to machine translation. In *Proc. of the naacl.*

Brown, P., Cocke, J., Pietra, S. D., Pietra, V. D., Jelinek, F., Lafferty, J., Mercer, R., & Roossin, P. (1990). A statistical approach to machine translation. *Computational Linguistics*, *16*(2), 79–85.

Brown, P., Pietra, S. D., Pietra, V. D., & Mercer, R. (1993). The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, *19*(2), 263–311.

Casacuberta, F. (1995). Probabilistic estimation of stochastic regular syntax-directed translation schemes. In A. Calvo & R. Medina (Eds.), *Proc. vi spanish symposium on pattern recognition and image analysis* (pp. 201–207). Córdoba, España.

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

Charniak, E. (2000). A maximum-entropy-inspired parser. In *Proc. of naacl-2000* (pp. 132–139).

Koehn, P. (2004). Pharaoh: a beam search

decoder for phrase-based statistical machine translation models. In *Proc. of amta.*

Marcu, D., & Wong, W. (2002). A phrase-based, joint probability model for statistical machine translation. In *Proc. of the conference on empirical methods in natural language processing.*

Maryanski, F., & Thomason, M. (1979). Properties of stochastic syntax-directed tranlation schemata. *Journal of Computer and Information Sciences*, *8*(2), 89–110.

Och, F., & Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, *29*(1), 19–52.

Och, F., & Ney, H. (2004). The alignment template approach to statistical machine tranlation. *Computational Linguistics*, *30*(4), 417–450.

Och, F. J., & Ney, H. (2000). Improved statistical alignment models. In *Proc. of acl* (pp. 440–447). Hongkong, China.

Pereira, F., & Schabes, Y. (1992). Inside-outside reestimation from partially bracketed corpora. In *Proceedings of the 30th annual meeting of the association for computational linguistics* (pp. 128–135).

Tomás, J., Lloret, J., & Casacuberta, F. (2005). Phrase-based alignment models for statistical machine translation. In *Iberian conference on pattern recognition and image analysis* (Vol. 3523, pp. 605–613). Estoril (Portugal): Springer-Verlag.

TT2. (2002). Transtype2 computer assisted translation (tt2). technical report. information society technologies (ist) program. ist-2001-32091.

Vogel, S., Zhang, Y., Huang, F., Tribble, A., Venugopal, A., Zhao, B., & Waibel, A. (2003). The cmu statistical machine translation system. In *Proc. of the ninth machine translation summit.*

Wu, D. (1997). Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, *23*(3), 377–404.

Yamada, K., & Knight, K. (2001). A syntax-based statistical translation model. In *Proc. of the 39th annual meeting of the association of computational linguistics* (pp. 523–530).

Zens, R., Och, F., & Ney, H. (2002). Phrase-based statistical machine translation. In *Proc. of the 25th annual german conference on artificial intelligence* (pp. 18–32).

Zhang, H., & Gildea, D. (2005). Stochastic lexicalized inversion transduction grammar for alignment. In *Proceedings of the 43rd annual conference of the association for computational linguistics (acl-05).* Ann Arbor, MI.