Using Categories in the EuTrans System

J. C. Amengual[1] J. M. Benedí[2] F. Casacuberta[2] A. Castaño[1]
A. Castellanos[1] D. Llorens[2] A. Marzal[1] F. Prat[1]
E. Vidal[2] J. M. Vilar[1]

(1) Unidad Predepartamental de Informática
Campus Penyeta Roja
Universitat Jaume I
12071 Castellón de la Plana (Spain)

(2) Depto. de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
46071 Valencia (Spain)

Abstract

The EuTrans project, aims at developing Machine Translation systems for limited domain applications. These systems accept speech and text input, and are trained using an example based approach. The translation model used in this project is the Subsequential Transducer, which is easily integrable in conventional speech recognition systems. In addition, Subsequential Transducers can be automatically learned from corpora.

This paper describes the use of categories for improving the EuTrans translation systems. Experimental results with the task defined in the project show that this approach reduces the number of examples required for achieving good models.

1 Introduction

The EuTrans project[1] (Amengual et al., 1996a), funded by the European Union, aims at developing Machine Translation systems for limited domain applications. These systems accept speech and text input, and are trained using an example based approach. The translation model used in this project is the Subsequential Transducer (SST), which is easily integrable in conventional speech recognition systems by using it both as language and translation model (Jiménez et al., 1995). In addition, SSTs can be automatically learned from sentence aligned bilingual corpora (Oncina et al., 1993).

This paper describes the use of categories both in the training and translation processes for improving the EuTrans translation systems. The approach presented here improves that in (Vilar et al., 1995), the integration of categories within the systems is simpler, and it allows for categories grouping units larger than a word. Experimental results with the *Traveler Task*, defined in (Amengual et al., 1996b), show that this method reduces the number of examples required for achieving good models.

The rest of the paper is structured as follows. In section 2 some basic concepts and the notation are introduced. The technique used for integrating categories in the system is detailed in section 3. Section 4 presents the speech translation system. Both speech and text input experiments are described in section 5. Finally, section 6 presents some conclusions and new directions.

2 Basic Concepts and Notation

Given an alphabet X, X^* is the free monoid of strings over X. The symbol λ represents the empty string, first letters (a, b, c, ...) represent individual symbols of the alphabets and last letters (z, y, x, ...) represent strings of the free monoids. We refer to the individual elements of the strings by means of subindices, as in $x = a_1 \ldots a_n$. Given two strings $x, y \in X^*$, xy denotes the concatenation of x and y.

2.1 Subsequential Transducers

A *Subsequential Transducer* (Berstel, 1979) is a deterministic finite state network that accepts sentences from a given input language and produces associated sentences of an output language. A SST is composed of states and arcs. Each arc connects two states and it is associated to an input symbol and an output substring (that may be empty). Translation of an input sentence is obtained starting from the initial state, following the path corresponding to its symbols through the network, and concatenating the corresponding output substrings.

[1] Example-Based Understanding and Translation Systems (EuTrans). Information Technology, Long Term Research Domain, Open Scheme, Project Number 20268.

Formally, a SST is a tuple $\tau = (X, Y, Q, q_0, E, \sigma)$ where X and Y are the *input* and *output alphabets*, Q is a finite set of *states*, $q_0 \in Q$ is the *initial state*, $E \in Q \times X \times Y^* \times Q$ is a set of *arcs* satisfying the determinism condition, and $\sigma : Q \rightarrow Y^*$ is a *state emission function*[2]. Those states for which σ is defined are usually called *final states*. The *determinism condition* means that, if (p, a, y, q) and (p, a, y', q') belong to E, then $y = y'$ and $q = q'$. Given a string $x = a_1 \ldots a_n \in X^*$, a sequence $(q_0, a_1, y_1, q_1), \ldots, (q_{n-1}, a_n, y_n, q_n)$ is a *valid path* if (q_{i-1}, a_i, y_i, q_i) belongs to E for every i in $1, \ldots, n$, and q_n is a final state. In case there exists such a valid path for x, the translation of x by τ is $y_1 \ldots y_n \sigma(q_n)$. Otherwise, the translation is undefined. Note that due to the condition of determinism, there can be no more than one valid path, and hence at most one translation, for a given input string. Therefore, τ defines a function between an *input language*, $L_I \subseteq X^*$, and an *output language*, $L_O \subseteq Y^*$. Both L_I and L_O are regular languages and their corresponding automata are easily obtainable from the SST. In particular, an automaton for L_I can be obtained by eliminating the output of the arcs and states, and considering the final state set of the automaton being the same as in the SST. A state is *useless* if it is not contained in any valid path. Useless states can be eliminated from a SST without changing the function it defines.

In section 3, we will relax the model. Instead of imposing the determinism conditition, we will only enforce the existence of at most one valid path in the transducer for each input string (non-ambiguity). We will call them Unambiguous SSTs (USSTs). Standard algorithms for finding the path corresponding to a string in an unambigous finite state automaton (see for instance (Hopcroft and Ullman, 1979)) can be used for finding the translation in a USST. When the problem is the search for the best path in the expanded model during speech translation (see section 4), the use of the Viterbi algorithm (Forney, 1973) guarantees that the most likely path will be found.

2.2 Inference of Subsequential Transducers

The use of SSTs to model limited domain translation tasks has the distinctive advantage of allowing an automatic and efficient learning of the translation models from sets of examples. An inference algorithm known as OSTIA (Onward Subsequential Transducer Inference Algorithm) allows the obtainment of a SST that correctly models the translation of a given task, if the training set is *representative* (in a formal sense) of the task (Oncina et al., 1993). Nevertheless, although the SSTs learned by OSTIA are usually good translation models, they are often poor input language models. In practice, they very accurately translate correct input sentences, but also accept and translate incorrect sentences producing meaningless results. This yields undesirable effects in case of noisy input, like the one obtained by OCR or speech recognition.

To overcome this problem, the algorithm OSTIA-DR (Oncina and Varó, 1996) uses finite state domain (input language) and range (output language) models, which allow to learn SSTs that only accept input sentences and only produce output sentences compatible with those language models. OSTIA-DR can make use of any kind of finite state model. In particular, models can be n-testable automata, which are equivalent to n-grams (Vidal et al., 1995) and can be also automatically learned from examples.

3 Introducing Word Categories in the Learning and Translation Processes

An approach for using categories together with SSTs was presented in (Vilar et al., 1995), proving it to be useful in reducing the number of examples required for learning. However, the approach presented there was not easily integrable in a speech recognition system and did not provide for the case in which the categories included units larger than a word.

For the EuTrans project, the approach was changed so that a single USST would comprise all the information for the translation, including elementary transducers for the categories. These steps were followed:

- Category identification. The categories used in EuTrans were seven: masculine names, femenine names, surnames, dates, hours, room numbers, and general numbers. The election of these categories was done while keeping with the example based nature of the project. In particular, the categories chosen do not need very specific rules for recognising them, the translation rules they follow are quite simple, and the amount of special linguistic knowledge introduced was very low.
- Corpus categorization. Once the cate-

[2] In this paper, the term function refers to *partial* functions. We will use $f(x) = \emptyset$ to denote that the function f is undefined for x.

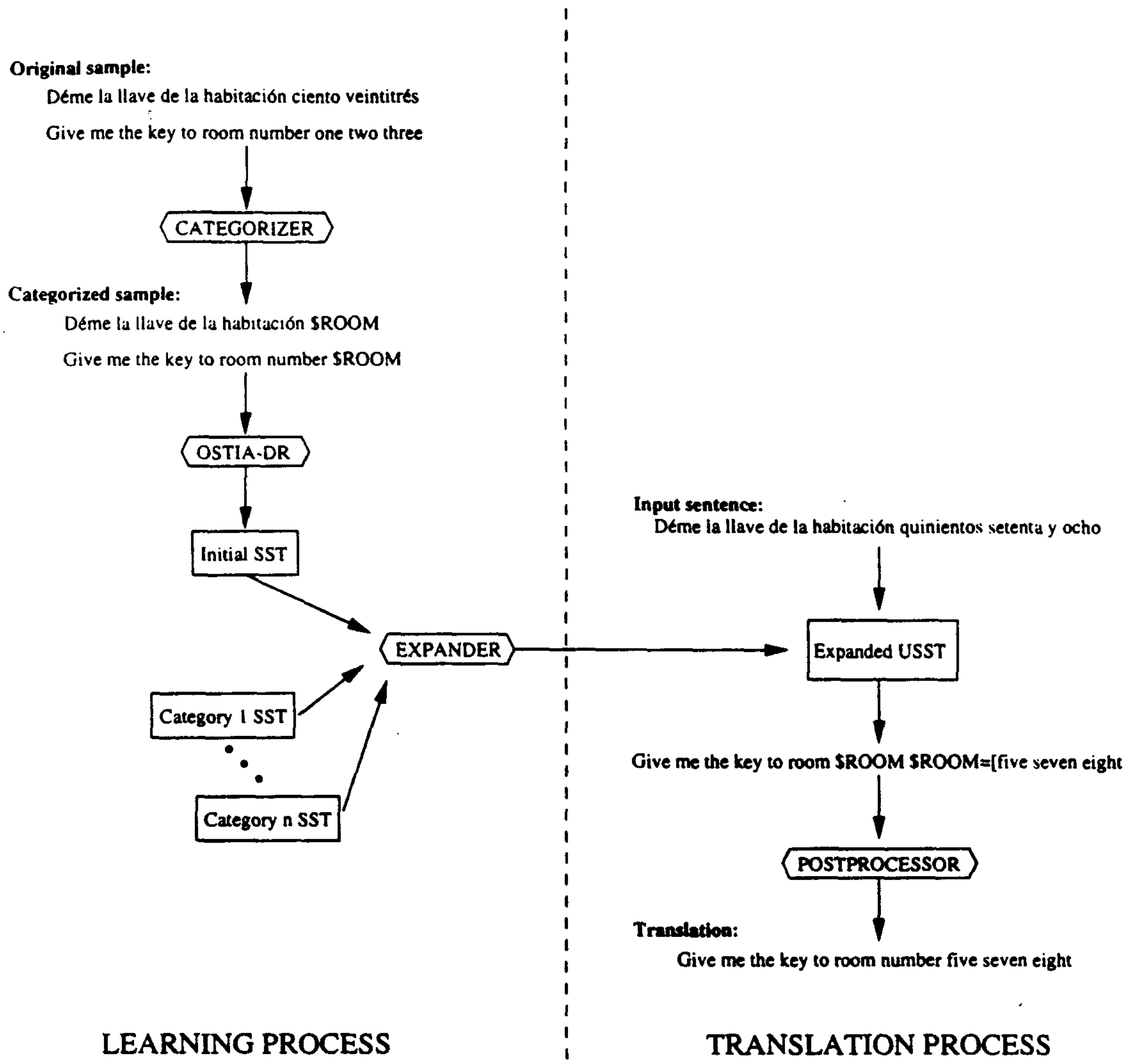


Figure 1: General schema of the treatment of categories in the learning and translation processes.

gories were defined, simple scripts substituted the words in the categories by adequate labels, so that the pair (*déme la llave de la habitación ciento veintitrés – give me the key to room one two three*) became (*déme la llave de la habitación $ROOM – give me the key to room $ROOM*), where *$ROOM* is the category label for room numbers.

- INITIAL MODEL LEARNING. The categorised corpus was used for training a model, the *initial SST*.
- CATEGORY MODELLING. For each category, a simple SST was built: its *category SST* (cSST).
- CATEGORY EXPANSION. The arcs in the initial SST corresponding to the different categories were expanded using their cSSTs.

A general view of the process can be seen in Figure 1. The left part represents the elements involved in the learning of the expanded USST, exemplified with a single training pair. The right part of the diagram gives a schematic representation of the use of this transducer.

The category expansion step is a bit more complex than just substituting each category-labeled arc by the corresponding cSST. The main problems are: (1) how to insert the output of the cSST within the output of the initial transducer; (2) how to deal with more than one final state in the cSST; (3) how to deal with cycles in the cSST involving its initial state.

The problem with the output had certain subtelities, since the translation of a category label can appear before or after the label has been seen in the input. For example, consider the transducer in Figure 2(a) and a Spanish sentence categorised as *me voy a $HOUR*, which corresponds to the categorised English one *I am leaving at $HOUR*. Once *me voy a* is seen, the continuation can only be *$HOUR*, so the initial SST, before seeing this category label in the input, has already produced the whole output (including $HOUR). Taking this into account, we decided to keep the output of the initial SST and to include there the information necessary for removing the category labels. To do this, the label for the category was considered as a variable that acts as a placeholder in the output sentence and whose contents are also fixed by an assignment appearing elsewhere within that sentence. In our example, the expected output for *me voy a las tres y media* could be *I am leaving at $HOUR $HOUR = [half past three]*. This assumes that each category appears at most once within each sentence.

The expanded model is obtained by an iterative procedure which starts with the initial SST. Each time the procedure finds an arc whose input symbol is a category label, it expands this arc by the adequate cSST producing a new model. This expansion can introduce non-determinism, so these new models are now USSTs. When every arc of this kind has been expanded, we have the *expanded USST*. The expansion of each arc follows these steps:

- Eliminate the arc.
- Create a copy of the cSST corresponding to the category label.
- Add new arcs linking the new cSST with the USST. These arcs have to ensure that the output produced in the cSST is embraced between *c=[* and *]*, *c* being the category label.
- Eliminate useless states.

Formally, we have an USST $\tau = (X, Y, Q, q_0, E, \sigma)$, a cSST $\tau_c = (X, Y, Q_c, q_{0c}, E_c, \sigma_c)$, where we assume that $\sigma_c(q_{0c} = \emptyset$, and an arc $(p, c, z, q) \in E$. We will produce a new USST $\tau' = (X, Y, Q \cup Q'_c, q_0, (E - (p, c, z, q)) \cup E'_c, \sigma')$. The new elements are:

- The set Q'_c is disjoint with Q and there exists a bijection $\phi : Q_c \rightarrow Q'_c$.
- The new set of arcs is:

$$\begin{aligned} E'_c &= \{(\phi(r), a, y, \phi(s)) \mid (r, a, y, s) \in E_c)\} \\ &\cup \{(p, a, zc{=}[y, \phi(s)) \mid (q_{0c}, a, y, s) \in E_c)\} \\ &\cup \{(\phi(r), a, y\sigma_c(s)], q) \mid (r, a, y, s) \in E_c) \\ &\quad \wedge \sigma_c(s) \neq \emptyset\} \\ &\cup \{(p, a, zc{=}[y\sigma_c(s)], q) \mid (q_{0c}, a, y, s) \in E_c) \\ &\quad \wedge \sigma_c(s) \neq \emptyset\} \end{aligned}$$

  Note that this solves the problems deriving from the cSST having multiple final states or cycles involving the initial state. The price to pay is the introduction of non-determinism in the model.

- The new state emission function is:

$$\sigma'(s) = \begin{cases} \sigma(s) & \text{if } s \in Q \\ \emptyset & \text{if } s \in Q'_c \end{cases}$$

Finally, the useless states that may appear during this construction are removed.

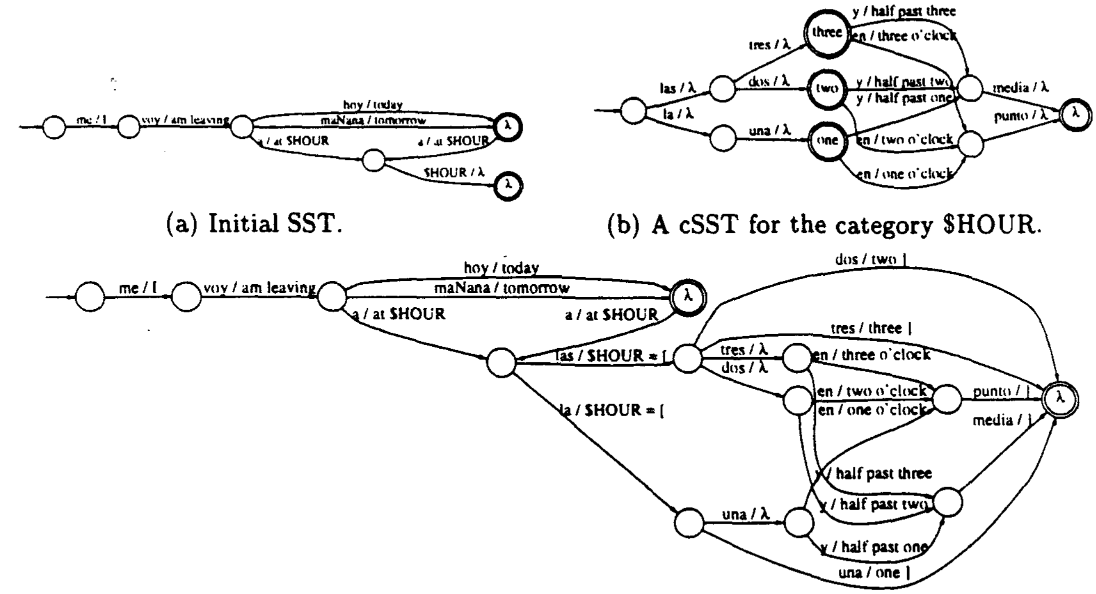
A simple example of the effects of this procedure can be seen on Figure 2. The drawing (a) depicts the initial SST, (b) is a cSST for the hours between one and three (in o'clock and half past forms), and the expanded USST is in (c).

4 Overview of the Speech Translation System

A possible scheme for speech translation consists in translating the output of a conventional Continuous Speech Recognition (CSR) front-end. This implies that some restrictions present in the translation and the output language, which could enhance the acoustic search, are not taken into account. In this sense, it is preferable to integrate the translation model within a conventional CSR system to carry out a simultaneous search for the recognised sentence and its corresponding translation. This integration can be done by using a SST as language and translation model, since it has included in the learning process the restrictions introduced by the translation and the output language. Experimental results show that better performance is achieved (Jiménez et al., 1994; Jiménez et al., 1995).

Thus, our system can be seen as the result of integrating a series of finite state models at different levels:

- Acoustic level. Individual phones are represented by means of Hidden Markov Models (HMMs).
- Lexical level. Individual words are represented by means of finite state automata with arcs labeled by phones.



(a) Initial SST. (b) A cSST for the category $HOUR.

(c) Expanded USST.

Figure 2: An example of the expansion procedure.

- SYNTACTIC AND TRANSLATION LEVEL. The syntactic constrains and translation rules are represented by an USST.

In our case, the integration means the substitution of the arcs of the USST by the automata describing the input language words, followed by the substitution of the arcs in this expanded automata by the corresponding HMMs. In this way, a conventional Viterbi search (Forney, 1973) for the most likely path in the resulting network, given the input acoustic observations, can be performed, and both the recognised sentence and its translation are found by following the optimal path.

5 Experiments

5.1 The Traveler Task

The *Traveler Task* (Amengual et al., 1996b) was defined within the EUTRANS project (Amengual et al., 1996a). It is more realistic that the one in (Castellanos et al., 1994), but, unlike other corpora such as the Hansards (Brown et al., 1990), it is not unrestricted.

The general framework established for the *Traveler Task* aims at covering usual sentences that can be needed in typical scenarios by a traveler visiting a foreign country whose language he/she does not speak. This framework includes a great variety of different translation scenarios, and thus results appropriate for progressive experimentation with increasing level of complexity. In a first phase, the scenario has been limited to some human-to-human communication situations in the reception of a hotel:

- Asking for rooms, wake-up calls, keys, the bill, a taxi and moving the luggage.
- Asking about rooms (availability, features, price).
- Having a look at rooms, complaining about and changing them.
- Notifying a previous reservation.
- Signing the registration form.
- Asking and complaining about the bill.
- Notifying the departure.
- Other common expressions.

The *Traveler Task* text corpora are sets of pairs, each pair consisting in a sentence in the input language and its corresponding translation in the output language. They were automatically built by using a set of Stochastic, Syntax-directed Translation Schemata (Gonzalez and Thomason, 1978) with the help of a data generation tool, specially developed for the EUTRANS project. This software allows the use of several syntactic extensions

Table 1: Some examples of sentence pairs from the *Traveler Task*.

Spanish:	*Por favor, ¿quieren pedirnos un taxi para la habitación trescientos diez?*
English:	*Will you ask for a taxi for room number three one oh for us, please?*
Spanish:	*Desearía reservar una habitación tranquila con teléfono y televisión hasta pasado mañana.*
German:	*Ich möchte ein ruhiges Zimmer mit Telefon und Fernseher bis übermorgen reservieren.*
Spanish:	*¿Me pueden dar las llaves de la habitación, por favor?*
Italian:	*Mi potreste dare le chiavi della stanza, per favore?*

Table 2: Main features of the Spanish to English, Spanish to German and Spanish to Italian text corpora.

	Spanish to English		Spanish to German		Spanish to Italian	
Vocabulary size	689	514	691	566	687	585
Average sentence length	9.5	9.8	8.9	8.2	12.7	11.8
Test set perplexity	13.8	7.0	13.2	9.0	13.6	10.6

to these schema specifications in order to express optional rules, permutation of phrases, concordance (of gender, number and case), etc. The use of automatic corpora generation was convenient due to time constrains of the first phase of the EuTrans project, and cost-effectiveness. Moreover, the complexity of the task can be controlled.

The languages considered were Spanish as input and English, German and Italian as output, giving a total of three independent corpora of 500,000 pairs each. Some examples of sentence pairs are shown in Table 1. Some features of the corpora can be seen in Table 2. For each language, the test set perplexity has been computed by training a trigram model (with simple flat smoothing) using a set of 20,000 random sentences and computing the probabilities yielded by this model for a set of 10,000 independent random sentences. The lower perplexity of the output languages derives from a design decision: multiple variants of the input sentences were introduced to account for different ways of expressing the same idea, but they were given the same translation.

Finally, a multispeaker speech corpus for the task was acquired. It consists of 2,000 utterances in Spanish. Details can be found in (Amengual et al., 1997a).

5.2 Text Input Experiments

Our approach was tested with the three text corpora. Each one was divided in training and test sets, with 490,000 and 10,000 pairs, respectively. A sequence of models was trained with increasing subsets of the training set. Each model was tested using only those sentences in the test set that were not seen in training. This has been done because a model trained with OSTIA-DR is guaranteed to reproduce exactly those sentences it has seen during learning. The performance was evaluated in terms of *Word Error Rate* (WER), which is the percentage of output words that has to be inserted, deleted and substituted for they to exactly match the corresponding expected translations.

The results for the three corpora can be seen on Table 3. The columns labeled as "Different" and "Categ.", refer to the number of different sentences in the training set and the number of different sentences after categorization. Graphical representations of the same results are on Figures 3, 4 and 5. As expected, the use of lexical categories had a major impact on the learning algorithm. The differences in WER attributable to the use of lexical categories can be as high as about a 40% in the early stages of the learning process and decrease when the number of examples grows. The large increase in performance is a natural consequence of the fact that the categories help in reducing the total variability that can be found in the corpora (although sentences do exhibit a great deal of variability, the underlying syntactic structure is actually much less diverse). They also have the advantage of allowing an easier extension in the vocabulary of the task without having a negative effect on the performance of the models so obtained (Vilar et al., 1995).

5.3 Speech Input Experiments

A set of Spanish to English speaker independent translation experiments were performed integrating in our speech input system (as described in

Table 3: Text input results: Translation word error rates (WER) and sizes of the transducers for different number of training pairs.

Training pairs			Without categories			With categories		
Generated	Different	Categ.	WER	States	Arcs	WER	States	Arcs
10,000	6,791	5,964	60.72	3,210	10,427	30.51	4,500	32,599
20,000	12,218	9,981	54.86	4,119	15,243	22.46	4,700	35,585
40,000	21,664	16,207	47.92	5,254	22,001	13.70	4,551	34,879
80,000	38,438	25,665	38.39	6,494	31,017	7.74	4,256	37,673
160,000	67,492	39,747	26.00	6,516	36,293	3.71	4,053	34,045
320,000	119,048	60,401	17.38	6,249	41,675	1.42	4,009	33,643
490,000	168,629	77,499	13.33	5,993	47,151	0.74	3,854	29,394

(a) Spanish to English corpus.

Training pairs			Without categories			With categories		
Generated	Different	Categ.	WER	States	Arcs	WER	States	Arcs
10,000	6,679	5,746	66.17	3,642	11,410	35.21	5,256	76,582
20,000	11,897	9,535	58.45	4,892	16,956	23.41	8,305	148,881
40,000	21,094	15,425	53.87	6,486	25,358	16.06	11,948	245,293
80,000	37,452	24,580	48.74	8,611	37,938	9.85	12,530	255,294
160,000	66,071	38,656	42.06	11,223	56,432	5.17	11,724	227,667
320,000	115,853	59,510	33.93	14,772	82,434	2.55	9,919	174,208
490,000	163,505	77,053	29.86	16,914	101,338	1.23	10,055	178,312

(b) Spanish to German corpus.

Training pairs			Without categories			With categories		
Generated	Different	Categ.	WER	States	Arcs	WER	States	Arcs
10,000	6,698	5,795	58.29	2,857	9,650	29.86	3,094	30,010
20,000	12,165	9,716	52.96	3,774	14,176	22.29	3,581	38,370
40,000	21,670	15,741	47.39	4,629	19,864	14.30	4,151	52,482
80,000	38,408	25,119	36.40	5,403	26,989	7.66	4,599	61,575
160,000	67,355	39,281	26.98	5,598	32,588	4.68	5,109	76,007
320,000	118,257	60,286	20.72	5,827	40,754	3.06	6,143	100,099
490,000	166,897	77,877	17.60	6,399	49,430	2.54	7,467	123,900

(c) Spanish to Italian corpus.

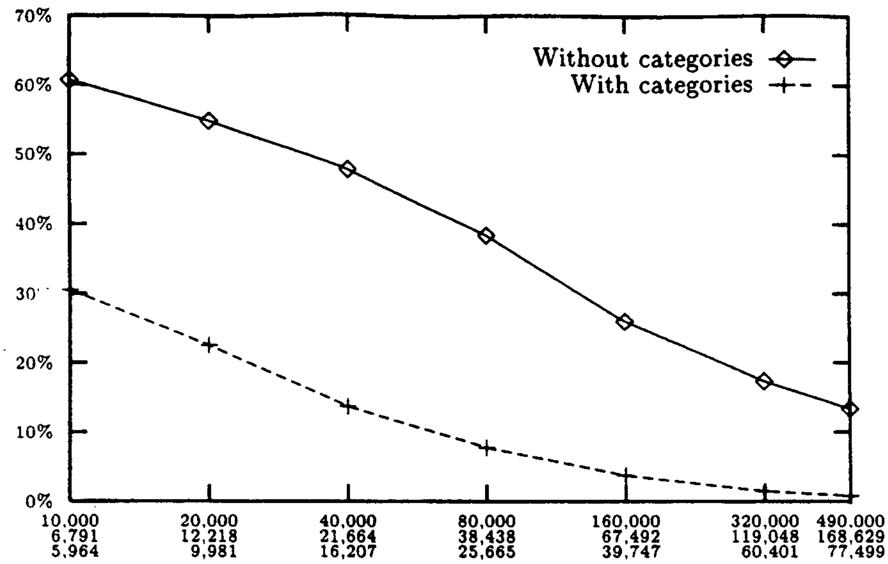


Figure 3: Evolution of translation WER with the size of the training set: Spanish to English text corpus. The sizes in the horizontal axis refer to the first three columns in Table 3(a).

Table 4: Speech input results:Translation word error rates (WER) and real time factor (RTF) for the best Spanish to English transducer.

Number of Gaussians	HMM Beam Width	WER	RTF
1,663	300	2.3 %	5.9
1,663	150	6.4 %	2.2
5,590	300	1.9 %	11.3
5,590	150	6.3 %	5.6

section 4) the following models:

- ACOUSTIC LEVEL. The phones were represented by context-independent continuous-density HMMs. Each HMM consisted of six states following a left-to-right topology with loops and skips. The emission distribution of each state was modeled by a mixture of Gaussians. Actually, there were only three emission distributions per HMM since the states were tied in pairs (the first with the second, the third with the fourth, and the fifth with the sixth). Details about the corpus used in training these models and its parametrization can be found in (Amengual et al., 1997a).

- LEXICAL LEVEL. Spanish Phonetics allows the representation of each word as a sequence of phones that can be derived from standard rules. This sequence can be represented by a simple chain. There were a total of 31 phones, including stressed and unstressed vowels plus two types of silence.

- SYNTACTIC AND TRANLATION LEVEL. We used the best of the transducers obtained in the Spanish to English text experiments. It was enriched with probabilities estimated by parsing the same training data with the final model and using relative frequencies of use as probability estimates.

The Viterbi search for the most likely path was speeded up by using beam search at two levels: independent beam widths were used in the states of the SST (empirically fixed to 300) and in the states of the HMMs. Other details of the experiments can be found in (Amengual et al., 1997a).

Table 4 shows that good translation results (a WER of 6.4%) can be achieved with a Real Time Factor (RTF) of just 2.2. It is worth noting that these results were obtained in a HP-9735 workstation without resorting to any type of specialised hardware or signal processing device. When translation accuracy is the main concern, a more detailed acoustic model and a wider beam in the search can be used to achieve a WER of 1.9%, but with a RTF of 11.3.

6 Conclusions

In the EUTRANS project, Subsequential Transducers are used as the basis of translation systems that accept speech and text input. They can be

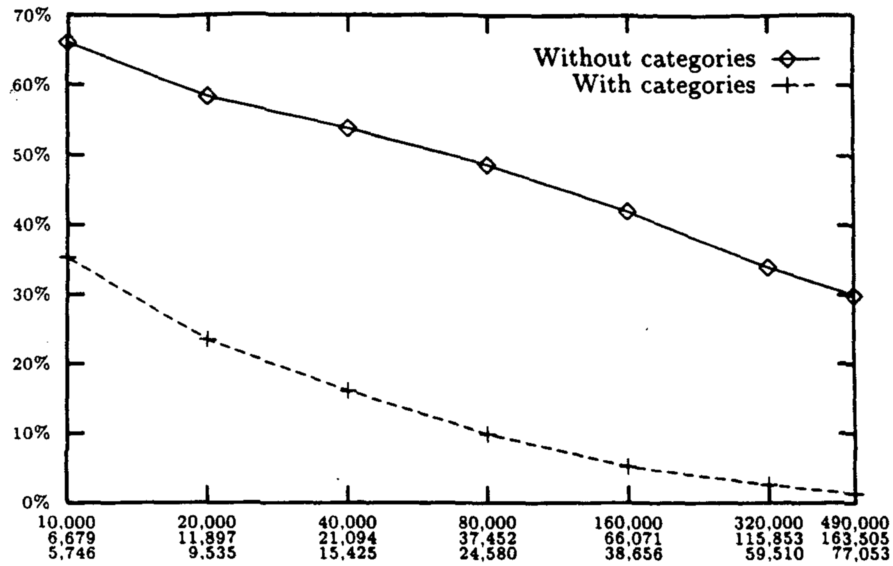


Figure 4: Evolution of translation WER with the size of the training set: Spanish to German text corpus. The sizes in the horizontal axis refer to the first three columns in Table 3(b).

automatically learned from corpora of examples. This learning process can be improved by means of categories using the approach detailed in this paper.

Experimental results show that this approach reduces the number of examples required for achieving good models, with good translation results in acceptable times without using specialised hardware.

Our current work concentrates in further reducing the number of examples necessary for training the translation models in order to cope with spontaneous instead of synthetic sentences. For this, new approaches are being explored, like reordering the words in the translations, the use of new inference algorithms, and automatic categorization.

Results obtained with a different enhancement of our text input system, the inclusion of error correcting techniques, can be found in (Amengual et al., 1997b).

References

Juan-Carlos Amengual, José-Miguel Benedí, Francisco Casacuberta, Asunción Castaño, Antonio Castellanos, Víctor M. Jiménez, David Llorens, Andrés Marzal, Federico Prat, Héctor Rulot, Enrique Vidal, Juan Miguel Vilar, Cristina Delogu, Andrea Di Carlo, Hermann Ney, Stephan Vogel, José Manuel Espejo, and Josep Ramón Freixenet. 1996a. EuTrans: Example-based understanding and translation systems: First-phase project overview. Technical Report D4, Part 1, EuTrans (IT-LTR-OS-20268). (Restricted).

Juan-Carlos Amengual, José-Miguel Benedí, Asunción Castaño, Andrés Marzal, Federico Prat, Enrique Vidal, Juan Miguel Vilar, Cristina Delogu, Andrea Di Carlo, Hermann Ney, and Stephan Vogel. 1996b. Definition of a machine translation task and generation of corpora. Technical Report D1, EuTrans (IT-LTR-OS-20268). (Restricted).

J. C. Amengual, J. M. Benedí, K. Beulen, F. Casacuberta, A. Castaño, A. Castellanos, V. M. Jiménez, D. Llorens, A. Marzal, H. Ney, F. Prat, E. Vidal, and J. M. Vilar. 1997a. Speech translation based on automatically trainable finite-state models. To appear in Proceedings of EUROSPEECH'97.

Juan C. Amengual, José M. Benedí, Francisco Casacuberta, Asunción Castaño, Antonio Castellanos, , David Llorens, Andrés Marzal, Federico Prat, Enrique Vidal, and Juan M. Vilar. 1997b. Error correcting parsing for text-to-text machine translation using finite state models. To appear in Proceedings of TMI 97, July.

J. Berstel. 1979. *Transductions and Context-Free Languages*. Teubner.

Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Frederick Jelinek, John D. Lafferty, Robert L. Mercer,

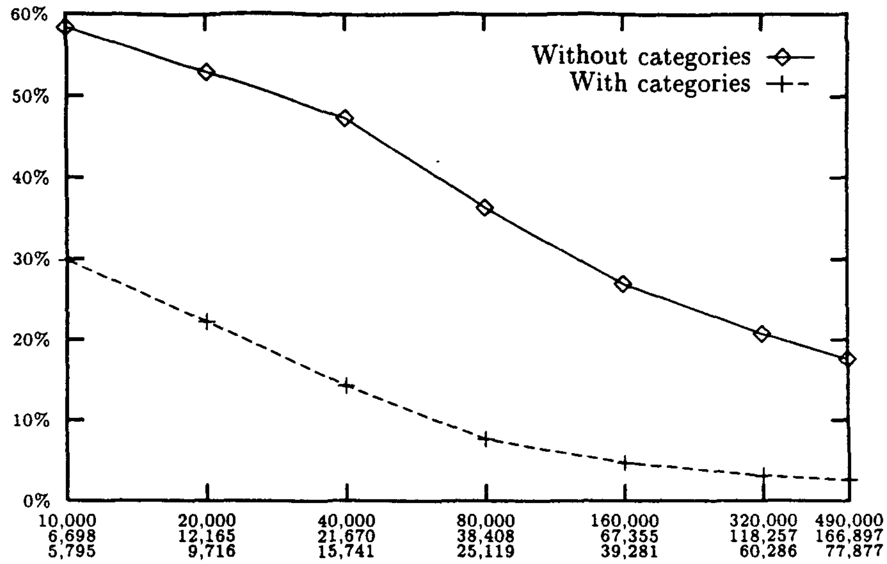


Figure 5: Evolution of translation WER with the size of the training set: Spanish to Italian text corpus. The sizes in the horizontal axis refer to the first three columns in Table 3(c).

and Paul S. Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85, June.

A. Castellanos, I. Galiano, and E. Vidal. 1994. Application of OSTIA to machine translation tasks. In Rafael C. Carrasco and José Oncina, editors, *Grammatical Inference and Applications*, volume 862 of *Lecture Notes in Computer Science*, pages 93–105. Springer-Verlag, September.

G. D. Forney, Jr. 1973. The Viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, March.

R. C. Gonzalez and M. G. Thomason. 1978. *Syntactic Pattern Recognition: An Introduction.* Addison-Wesley, Reading, Massachusetts.

J. E. Hopcroft and J. D. Ullman. 1979. *Introduction to Automata Theory, Languages and Computation.* Addison-Wesley, Reading, Mass., USA.

V.M. Jiménez, E. Vidal, J. Oncina, A. Castellanos, H. Rulot, and J. A. Sánchez. 1994. Spoken-language machine translation in limited-domain tasks. In H. Niemann, R. de Mori, and G. Hanrieder, editors, *Proceedings of the CRIM/FORWISS Workshop on Progress and Prospects of Speech Research and Technology*, pages 262–265. Infix.

V. M. Jiménez, A. Castellanos, and E. Vidal. 1995. Some results with a trainable speech translation and understanding system. In *Proceedings of the ICASSP-95*, Detroit, MI (USA).

José Oncina and Miguel Ángel Varó. 1996. Using domain information during the learning of a subsequential transducer. In Laurent Miclet and Colin de la Higuera, editors, *Grammatical Inference: Learning Syntax from Sentences*, volume 1147 of *Lecture Notes in Computer Science*, pages 301–312. Springer-Verlag.

José Oncina, 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, may.

Enrique Vidal, Francisco Casacuberta, and Pedro García. 1995. Grammatical inference and automatic speech recognition. In A. J. Rubio and J. M. López, editors, *Speech Recognition and Coding, New Advances and Trends*, NATO Advanced Study Institute, pages 174–191. Springer-Verlag, Berlin.

Juan Miguel Vilar, Andrés Marzal, and Enrique Vidal. 1995. Learning language translation in limited domains using finite-state models: some extensions and improvements. In *4th European Conference on Speech Communication and Technology*, Madrid (Spain), September. ESCA.