Evaluating pruned *k*-TSS Language Models: perplexity and word recognition rates

A. Varona and I. Torres*

Dpto. Electricidad y Electrónica. Universidad del País Vasco.

Apdo. 644 48080 Bilbao. SPAIN

E-mail (amparo@we.lc.ehu.es, manes@we.lc.ehu.es)

Abstract

A syntactic approach based on regular grammars, the *k*-Testable in the Strict Sense (*k*-TSS) language models (LM), has been proposed in previous works to be integrated in Continuous Speech Recognition (CSR) Systems. In this work, a pruning procedure was applied to *k*-TSS models in order to reduce the size of the model while keeping its accuracy. An experimental evaluation of the pruned *k*-TSS models was carried out over a Spanish speech corpus in terms of both, perplexity and word recognition rates. Several pruning thresholds as well as several values of the well-known empirical scaling factor applied to the estimated LM probabilities were tested. These experiments showed that pruned models achieved a good system performance. They also showed that an increase of the test set perplexity of a language model does not always mean a degradation in the model performance when integrated in a CSR system.

Keywords: Continuos Speech recognition, Syntactic Language Models, N-grams, *k*-TSS, perplexity.

1.- Introduction.

Continuous Speech Recognition (CSR) Systems require a Language Model (LM) to integrate the syntactic and/or semantic constraints of the language. The generation of LM is a classical pattern recognition problem where both statistical (typically N-grams) and syntactic approaches have been extensively used. A syntactic approach based on regular grammars, the *k*-Testable in the Strict Sense (*k*-TSS) LMs [1] has also been proposed in previous works [2] [3] to generate LM. The use of regular grammars allowed to obtain a deterministic Stochastic Finite State Automaton (SFSA) integrating K k-TSS models (with k=1, 2..,K) into a self contained model. Such an Automaton can be efficiently represented and handled allowing its integration in a CSR system, even for high values of K [2] [3].

A major problem to be solved when using a Language Model is the estimation of the probabilities of events not represented in the training corpus. In previous works [4] a syntactic back-off smoothing technique was also proposed under the *k*-TSS formalism. The probability to be assigned to *unseen* events is recursively obtained from less accurate models (k-1, k-

* Work partially supported by the Spanish CICYT under grant TIC98-0423-C06-03

2,…1) in back-off smoothing. Then, the integration of different k-TSS models needed by this smoothing technique can be achieved in a compact and easy way [5]. The obtained smoothed SFSA integrating the K k-TSS models (with k=1, 2..,K) can be easily handled at decoding time by a simple search function trough the array allocating the model [5]. Additionally, beam search strategies can also be used to reduce the time required to decode each uttered sentence. However, the size of the model and thus the memory required to allocate the full smoothed SFSA still remains high, mainly for medium-large vocabulary speech recognition tasks.

In this work, a pruning procedure was applied to k-TSS models in order to reduce the size of the model and memory requirements, while keeping its accuracy. The pruning procedure consisted in removing infrequent k-grams from the model. However, the pruning thresholds should be experimentally evaluated since the goal of any pruning procedure is to find a correct balance between the memory requirements of the model and its performance.

The aim of this work was to evaluate several pruned k-TSS language models. The most common way to evaluate a Language Model is the test set perplexity. However, this measure does not take into account of the relationship to the acoustic models. New measures have been proposed to avoid this problem [6] [7]. Nevertheless the must reliable way to compare the performance of different Language Models that have to be integrated into a CSR system is to measure the obtained recognition rates. In this work, both measures were considered: the test set perplexity and the percentage of words correctly decoded.

CSR systems are invariably based on the well-known Bayes' rule. Bayes' rule maximises the product of the probability of a sequence of acoustic observations A for a given sequence of words Ω, $P(A/\Omega)$, and the probability $P(\Omega)$ that the word sequence Ω will be uttered. However, it is well known that empirical scaling factors need to be applied to $P(A/\Omega)$ or to $P(\Omega)$ in the Bayes' rule to obtain good word recognition rates in practical CSR systems [8] [9]. $P(A/\Omega)$ and $P(\Omega)$ are estimated distributions selected independently from speech and text sample sets respectively. Thus, a balance parameter has to be applied to lessen these effects and then obtain good system performances.

An experimental evaluation of the pruned k-TSS models was carried out over a Spanish speech corpus. Several pruning thresholds as well as several values of the balance parameter α applied to the estimated LM probabilities $(P(\Omega))^{\alpha}$ were tested. These experiments showed that pruned models achieved a good system performance. These experiments also showed that an increase of the test set perplexity of a language model does not always mean a degradation in the model performance when integrated in a CSR system.

In Section 2, we resume the k-TSS language models used along the work as well as the syntactic back-off smoothing technique. Section 3 deals with the experimental evaluation of several Pruned k-TSS Language Models in terms of both, perplexity and recognition rates. Section 4 shows the effect of the balance parameter α when scaling Language Model probabilities. Finally, some concluding remarks are presented in Section 5.

2.- The Language Model.

2.1.- The Stochastic Finite State Automaton (SFSA).

A syntactic approach of the well-known N-grams models, the k-Testable Language in the Strict Sense (k-TSS) [1] [2] [3], was used in this work to be integrated in a CSR system. The use of k-TSS regular grammars allowed to obtain a deterministic Stochastic Finite State

Automaton (SFSA) integrating K k-TSS models (with k=1, 2..K) into a self-contained model [2] [3]. In such a model, each state of the automaton represents a string of words $w_{i-k}w_{i-(k-1)}...w_{i-1}$, k = 1...K-1, with a maximum length of K-1, where i stands for a generic index in any string $w_1...w_i...$ appearing in the training corpus. Such a state is labelled as w_{i-k}^{i-1}. Each transition represents a k-gram, k = 1...K; it is labelled by its last word w_i and connects two states labelled up to with K-1 words. As an example, transitions corresponding to strings of words of length K connecting states associated to string lengths K-1 are defined as:

$$\delta^K\left(w_{i-(K-1)}^{i-1}, w_i\right) = \left(w_{i-(K-1)+1}^{i}, P(w_i / w_{i-(K-1)}^{i-1})\right) \quad (1)$$

The probability to be associated to each transition $\delta^K\left(w_{i-(K-1)}^{i-1}, w_i\right)$ can be estimated under a maximum likelihood criterion as:

$$P_{ML}(w_i / w_{i-(K-1)}^{i-1}) = \frac{N(w_i / w_{i-(K-1)}^{i-1})}{\sum_{\forall w_j \in \Sigma} N(w_j / w_{i-(K-1)}^{i-1})} \quad (2)$$

where $\Sigma$is the vocabulary, that is, the set of words appearing in the training corpus, $N(w_j / w_{i-(K-1)}^{i-1})$ is the number of times the word w_j appears at the end of the K-gram $w_{i-(K-1)}...w_{i-1}w_j$, that is the count associated to the transition labelled by w_j coming from state labelled as $w_{i-(K-1)}^{i-1}$.

The whole and detailed definition of the Automaton, i.e. initial and final states, unigram representation, etc., can be found in [2] and [5]. As an example, Figure 1a) represents the K-grams $w_{i-(K-1)}^{i-1}$ and $w_{i-(K-1)+1}^{i}$ labelling two states of the automaton. When w_i is observed an outgoing transition from the first to the second state is set and labelled by w_i.

2.2.- The Smoothing technique

Smoothing techniques are usually based on a discounting and distribution scheme of the LM estimated probabilities. A probability mass of the observed events is discounted and then assigned to unobserved events. A syntactic back-off smoothing procedure was developed under the k-TSS formalism, in previous works [4]. The syntactic approach suggested a state-dependent estimation of the total discount and, consequently, the symmetry principle was only locally applied [4]. Thus, the modified probability $P(w_i/q)$ to be associated to a transition $\delta^K(q, w_i) = (q', P(w_i / q))$ representing an observed event at state q is estimated according to:

$$P(w_i / q) = P_{ML}(w_i / q) d(q) \quad (3)$$

The discount coefficient $d(q)$ was calculated as:

$$d(q) = \frac{N(q)}{N(q) + \left|\Sigma_q\right|} \quad (4)$$

where Σ_q is the vocabulary associated to state $q \in Q^K$ and consists of the set of words appearing after the string labelling state q in the training corpus, $|\Sigma_q|$ is the size of Σ_q, $N(w/q)$ is the number of times that word w appears after the string labelling state q and $N(q) = \sum_{\forall w \in \Sigma_q} N(w / q)$.

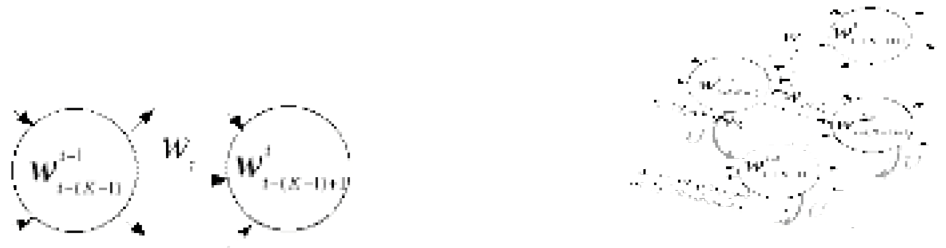

Figure 1: a) Two states of the SFSA representing two K-grams. Transitions are labelled by words appearing in the training sample after the K-gram labelling the source state. b) Transitions labelled by seen events($w_j \in \Sigma_q$) connect each state to states in the same k-TSS submodel. Transition labelled by unseen events (U) connect it to its back-off state in the (k-1)-TSS submodel

Therefore, the discount coefficient $d(q)$ depends on the number of different *seen* events, $|\Sigma_q|$, appearing after the string labelling state q and on $N(q)$. This approach is similar to the Witten-Bell discounting scheme used by the CMU Statistical Language Modelling toolkit [10].
The transitions probabilities corresponding to those events not represented in the training corpus, i.e. *unseen* events, are estimated according to more general probability distributions in k-TSS models, with $k<K$. These transitions can be grouped into a unique transition to a back-off state b_q associated to each state q. This back-off state b_q associated to each state q can be found in the (k-1)-TS submodel. A complete presentation of the syntactic back off scheme can be found in [4]. Figure 1b shows such a structure for a state q labelled as $w_{i-(K-1)}^{i-1}$.

3.- Pruning the *k*-TSS Language Models.

A pruning procedure was applied to k-TSS models in order to reduce the size of the model while keeping its accuracy. The pruning procedure consisted in eliminating states with a probability under a certain threshold. Thus, infrequent k-grams w_{i-k}^{i-1}, k = 1...K-1 were removed from the model. However, the pruning thresholds should be experimentally evaluated since the goal of any pruning procedure is to find a correct balance between the memory requirements of the model and its performance.
An experimental evaluation of the pruned k-TSS models was carried out over a Spanish speech corpus. The model degradation was first evaluated in terms of a test set perplexity (PP). Then the smoothed SFSA was integrated in a CSR system. Each transition of the automaton was replaced by a chain of Discrete Hidden Markov models, with four observation codebooks, representing the acoustic model of each phonetic unit of the word. The time-synchronous Viterbi algorithm, along with a beam-search procedure reducing the involved computational cost, was used to decode uttered sentences. The beam-search factor (bf) was optimised in previous experiments [2]. Thus, the same fixed value, bf=0.7, was used for all the experiments in this work. The experiments were carried out by a Silicon Graphics O2 with a R10000 processor. This CSR system was then used to evaluate the pruned k-TSS models over the same test set in terms of recognition performance, i.e. percentage of words correctly decoded (%W). This value was calculated as: $\%W=c/(i+s+d+c)*100$ where c accounts for the number of words correctly decoded, s and d for the number of substitution and deletion errors respectively and i for the number of inserted words. Finally, both evaluation measures, perplexity and recognition performance, were compared.
For those experiments, a task-oriented Spanish speech corpus [11], consisting in 82,000 words and a vocabulary of 1,213 words, was used. This corpus represents a set of queries to a Spanish geography database. The training corpus used to obtain the k-TSS models, consisted

in 9150 sentences. The text test set consisted in 200 different sentences. These sentences were then uttered by 12 speakers resulting in a total of 600 sentences and 5655 words that composed the speech test set.

Table 1 shows the experimental evaluation of several *k*-TSS models (*k*=2,...,6) when different pruning factors (*pf*) were considered. Pruning factors (*pf*) represent the count threshold bellow which *k*-grams were discarded, so that, *pf*=1 represents the no pruned models. Table 1 also shows the number of states (*LMst*) of the *k*-TSS language models, and the size of memory required to allocate them. The average number of active nodes per frame (*NPF*) in the lattice (consisting of acoustic and LM states) and the average time per frame (*TPF*) needed to decode the sentences are also represented in Table 1.

Table 1: Evaluation of pruned *k*-TSS language models (*k*=2,..,6) when different pruning factors (*pf*) were considered. Both, perplexity and percentage of correctly decoded words are represented.

K	*pf*	*LMst*	*memory (Mb)*	*PP*	*NPF*	*TPF(msec)*	*%W*
2	1	1,213	0.13	**13.10**	3,397	69	**58.38**
3	1	7,479	0.43	**7.53**	4,270	91	**61.15**
	2	3,854	0.20	**8.51**	2,755	59	**63.56**
	3	2,845	0.14	**9.76**	2,286	43	**64.54**
	4	2,336	0.11	**11.03**	1,960	35	**65.17**
	5	1,999	0.09	**12.14**	1,753	30	**65.05**
4	1	21,551	0.95	**6.95**	4,392	96	**61.50**
	2	9,360	0.38	**8.07**	2,809	60	**63.81**
	3	6,366	0.25	**9.62**	2,325	46	**64.78**
	4	4,993	0.19	**10.08**	1,988	36	**65.25**
	5	4,139	0.16	**12.41**	1,776	31	**64.76**
5	1	42,849	1.69	**6.90**	4,411	98	**61.37**
	2	16,086	0.58	**8.10**	2,818	60	**63.74**
	3	10,260	0.36	**9.80**	2,332	46	**64.82**
	4	7,795	0.26	**11.28**	1,994	36	**64.83**
	5	6,308	0.22	**12.63**	1,781	32	**64.66**
6	1	69,616	2.55	**6.90**	4,418	99	**61.23**
	2	22,839	0.77	**8.18**	2,822	61	**63.75**
	3	13,784	0.45	**9.91**	2,333	46	**64.81**
	4	10,179	0.33	**11.47**	1,996	37	**64.93**
	5	8,089	0.26	**12.78**	1,783	32	**64.68**

In order to show the statistical significance of the differences among the *%W* reported in Table 1, a summary of the best results along with their confidence intervals $[W^-,W^+]_{95\%}$ is shown in Table 2. The confidence interval $[W^-,W^+]_{95\%}$ was calculated as:

$$W^{\pm} = \frac{p + \frac{z^2}{2N} \pm z\sqrt{\frac{p(1-p)}{N} + \frac{z^2}{4N^2}}}{1 + \frac{z^2}{N}} * 100$$

where p=%W/100, z=1.96 and N is the number of words in the test set (5655 in these experiments).

Table 2: Summary of the best %W results of Table 1, along with the confidence interval $[W^-,W^+]_{95\%}$ values were included to measure the statistical significance of the obtained results.

K	*pf*	*W*	$[W^-,W^+]_{95\%}$
2	1	58.38	[57.08, 59.65]
3	1	61.15	[59.87, 62.41]
	4	65.17	[63.93, 66,40]
4	1	61.50	[60.22, 62.76]
	4	65.25	[63.99, 66.47]
5	1	61.37	[60.09, 62.63]
	4	64.83	[63.57, 66,06]
6	1	61.23	[59.95, 62.49]
	4	64.93	[63.71, 66,16]

Important reductions of the number of states and memory requirements were observed when the pruned *k*-TSS models (pf>1) were considered (Table 1). These reductions were even more important for higher values of *k*. On the other hand, the test set perplexity (PP) increased when *pf* did showing the normal degradation of the pruned models' structure.

Figure 2 represents the perplexity (*PP*) and the obtained *%W* for the *k*-TSS models and pruning factors evaluated in these experiments (Table 1). Figure 3 shows the %W and the average number of actives nodes (*NPF*) in the lattice for the same experiments (Table 1). Figure 2 shows that the values of the perplexity are almost constant for values of *k* higher than 3. This behaviour is managed thanks to the use of the back-off smoothing procedure [4]. However, the most surprising results reported in Table 1, Figure 2 and Figure 3 is that the *%W* significantly increased when *pf* did. Thus, pruned models achieved better performance and require less memory to be allocated that not pruned models!. Moreover, they also need less time to decode each sentence since the average active nodes per frame also decrease with *pf* (see Figure 3).

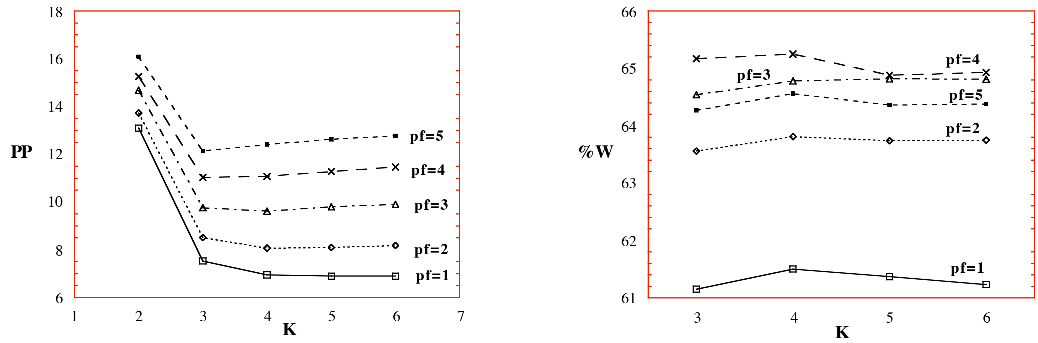

Figure 2: Perplexity (PP) and percentage of words correctly decoded (*%W*) obtained by the pruned *k*-TSS language models (*k*=2 to 6) evaluated in the experiments reported in Table 1, when different pruning factors (pf = 1 to 5) were considered.

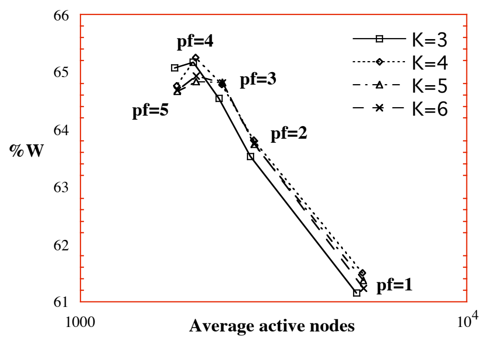

Figure 3.- Percentage of words correctly decoded for %W and average number of active nodes per frame (NPF) obtained for the pruned *k*-TSS language models evaluated (*k*=2 to 6) evaluated in the experiments reported in Table 1, when different pruning factors were considered.

This peculiar behaviour can be explained by analysing the smoothing technique presented in Section 2.2. When pruning factors were applied many infrequent *k*-grams disappeared from the model and, as a consequence, the number of *seen* events at each state ($|\Sigma_q|$) could strongly decrease. In this case, those infrequent *k*-grams were decoded as *unseen* events when appearing in the test set. An important reduction of the $|\Sigma_q|$ value was then achieved. The value of *N(q)* also decreased but this reduction was not so significant. Therefore, the distribution of the probability mass between observed and unobserved events made by the syntactic smoothing technique has been seriously modified. The discount coefficient $d(q)$ is smaller (see Equation 4) for pruned models than for not pruned ones and thus, the probability mass assigned to the backoff transition, i.e. *unseen* events, is then also smaller. So that, the gap between high and low probabilities in pruned models is bigger than in no pruned models. Consequently, the beam search technique needs to keep a lower number of active nodes in the lattice (the average number of active nodes (*NPF*) decreased when *pf* increased as shown in Figure 3). The syntactic backoff smoothing technique seems to overestimate the unseen events' probability in these experiments since the smoothed pruned models got better recognition rates with less number of active nodes per frame.

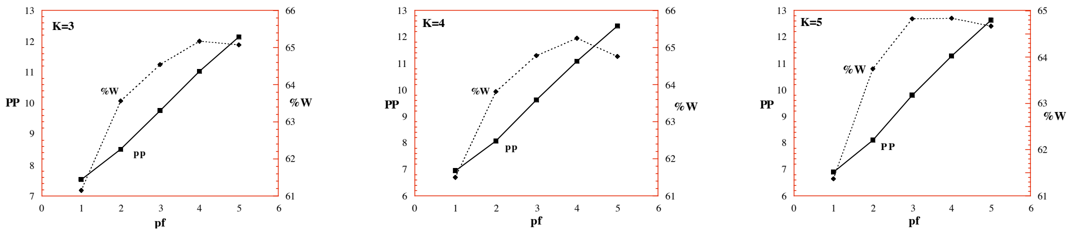

Figure 4: Evolution of the percentage of words correctly decoded (*%W*) and the perplexity values with the pruning factor (*pf*), for *k*=3, 4 and 5 *k*-TSS models

However, the test set perplexity did not report this behaviour. Figure 4 represents the perplexity and the recognition rates obtained for some pruned *k*-TSS language models evaluated in these experiments (Table 1). As it was previously reported in [5] [6], an increase of the LM perplexity does not always means a degradation in the system performance. Figure 4 shows that perplexity and recognition rates increased when *pf* did. The minimum values of both, *%W* and *PP* values were reached by the not pruned (*pf*=1) model. The perplexity is practically linear with the pruned factor. However, the *%W* reached its maximum value for *pf*= 3 or 4. Values of *pf* higher than 4 led to an excessive model degradation and, as a consequence, to a lower system performance.

4.- Scaling probabilities: effect of the balance parameter α.

A new experimental evaluation of pruned models was then carry out over the same Spanish corpus and CSR system. The Bayes's rule was then applied but raising the language model probability to a power α: $(P(\Omega))^{\alpha}$. Several values of the balance parameter α were tested to optimise the percentage of words correctly decoded by the system.
Figure 5 shows the experimental evaluation of several *k*-TSS models (*k*=3,...,6) when different pruning factors (*pf*) and balance parameter values (α) were considered. The continuous line represents the not pruned (*pf*=1) model whereas dotted lines represent different pruned models (*pf*>1). Points at the top left corner of each Figure show the best system performances: the highest *%W* values and the lowest values of the average number of active nodes.
For any *k*-TSS model, an important increase in recognition rates along with a notable decrease in the average number of active nodes in the lattice can be observed (Figure 5) when the balance parameter α increased (up to a maximum). Thus, higher values of α reduced the word insertion rate and, consequently, the word error rate. The effect of the balance factor α is the attenuation of all the LM probabilities, but this attenuation is higher for lower probability values. So that, the gap between high and low probabilities is also bigger and thus the beam search technique needs to keep a low number of active nodes in the lattice (see Figure 5). Therefore, the use of α values greater than one lead to higher recognition rates for lower average numbers of active nodes in the lattice.
However a different behaviour of pruned and not pruned models was observed:

a) The value of α optimising the *%W* was slightly higher for not pruned *k*-TSS models (6 for *k*=4, 5 and 6) than for pruned *k*-TSS models (4 or 5).

b) Pruned models got better performances than not pruned models when the balance parameter values remains under 5. A particular case, α=1, of this surprising behaviour was shown in the experiments reported in previous Section (Table 1 and Figure 2). However, for high values of α, high values of the pruning factor produced worse *%W*. Table 3 shows the experimental evaluation of several pruned *k*-TSS models for a fixed scaling factor, α=5, leading to a good system performance. In this case, not pruned models got the best *%W* as is usually expected.

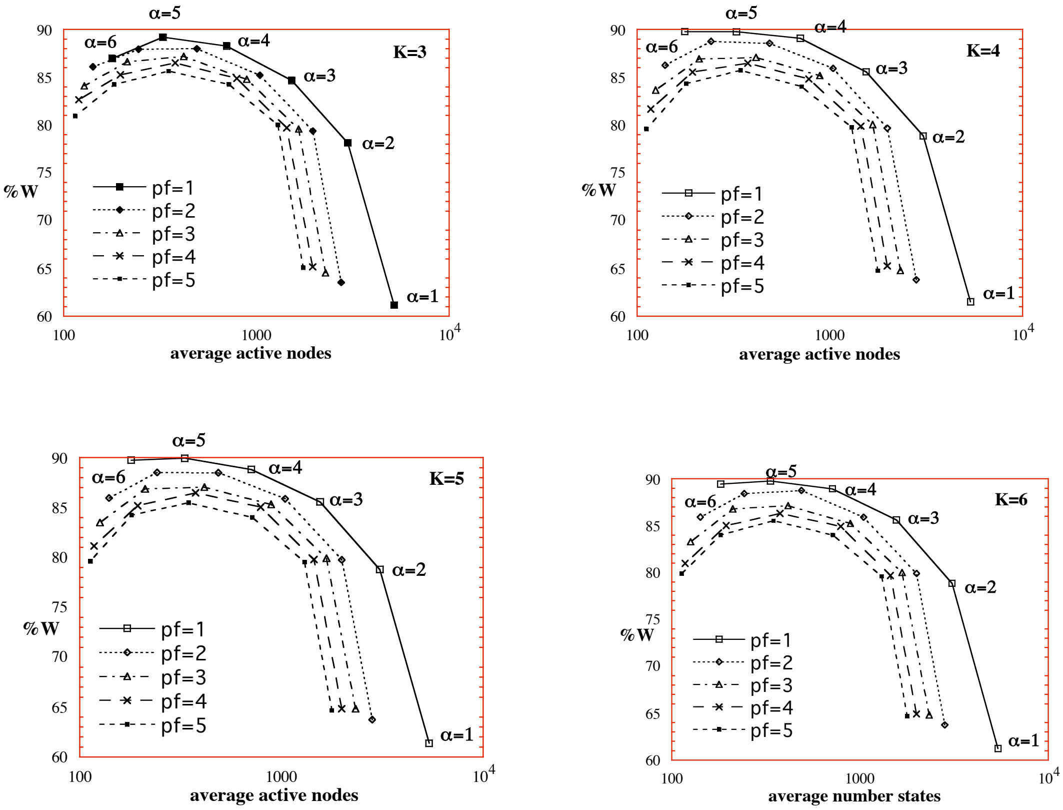

Figure 5. Evaluation of *k*-TSS models for different pruning factors (*pf*) and scaling parameter α.

Table 3: Evaluation of pruned *k*-TSS language models (*k*=2,..,6) for α=5 (good system performance).

K	*pf*	*NPF*	*TPF (msec)*	*%W*
3	1	328	16	**89.18**
	2	245	10	**87.93**
	3	214	8	**86.62**
	4	197	7	**85.25**
	5	183	6	**84.23**
4	1	328	16	**89.76**
	2	242	10	**88.76**
	3	210	8	**86.92**
	4	194	7	**85.55**
	5	180	6	**85.72**
5	1	332	17	**89.94**
	2	242	10	**88.52**
	3	211	8	**86.88**
	4	194	7	**85.19**
	5	181	6	**84.24**
6	1	335	18	**89.77**
	2	243	10	**88.42**
	3	211	8	**86.79**
	4	195	7	**85.02**
	5	182	6	**84.01**

The use of a balance factor α can be understood as a new smoothing of the LM probabilities. This effect is not exactly the same but it is very similar to the one produced by the syntactic

back-off smoothing over pruned models observed in previous Section. Both procedures, pruning k-TSS models and scaling the LM probabilities, produced similar effects: increasing the *%W* while decreasing the average number of active nodes in the lattice. This could explain why pruned models reach their best performance with a lower value of α applying the syntactic smoothing technique after pruning the model is similar to apply a value of α>1 in the recognition scheme.
Figure 6 represents the perplexity and the recognition rates obtained for some pruned *k*-TSS language models evaluated in these experiments when a balance parameter α=5 was applied (Table 3). This Figure shows that an increase in perplexity (obtained with pruned models *pf*>1) means a decrease in recognition rates when an appropriate factor α was applied. This Figure shows a behaviour of the perplexity and *%W* completely different to the one observed in Figure 4, when a value of α=1 was used. Now *k*-TSS language models with lower perplexity values lead to better *%W* when integrated in a CSR system. This fact could simply means that the test set perplexity is not the most adequate measure to predict the behaviour of a smoothing technique when the LM has to be integrated in a CSR system since the final performance fundamentally depend on empirical factors as α.

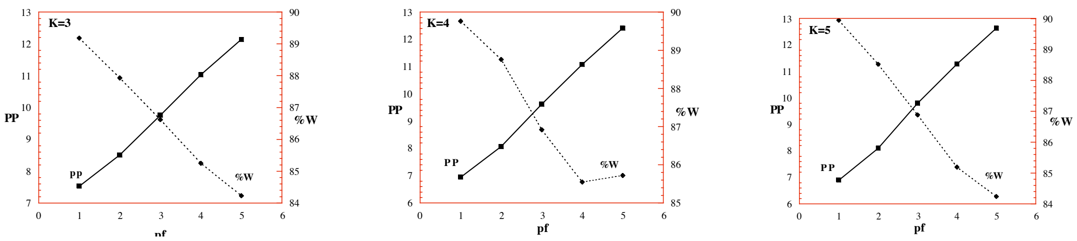

Figure 6: Evolution of the percentage of words correctly decoded (*%W*) and the perplexity values with the pruning factor (*pf*), for *k*=3, 4 and 5 *k*-TSS models (α=5).

5.- Concluding Remarks.

A syntactic approach based on regular grammars, the *k*-Testable Language Models in the Strict Sense (*k*-TSS), has been used in this work to be integrated in a CSR System. A pruning procedure was then applied to the *k*-TSS models in order to reduce the size of the model while keeping its accuracy.
An experimental evaluation of the pruned *k*-TSS models was carried out over a Spanish speech corpus. In these experiments, important reductions of the number of states were observed for pruned *k*-TSS models (pf>1). Moreover, pruned models achieved better percentage of words correctly decoded but require less memory to allocate them and less time to decode a sentence that not pruned models. This behaviour is due to the state-dependent back-off smoothing technique applied to pruned *k*-TSS models.
Then, several scaling factors where applied to the estimated LM probabilities. Important increases of recognition rates along with a notable decrease of the average number of active nodes in the lattice were observed in this case. However a different behaviour of pruned and not pruned models was observed: pruned models got better performances in terms of average active nodes and %W than not pruned models when low values of the balance parameter α were used. Thus, the use of a balance factor α can be understood as a new smoothing of the LM probabilities.
The experiments carried out in this work also show that an increase of the test set perplexity of a language model does not always means a degradation in the model performance. The test

set perplexity seems not to be an adequate measure to predict the behaviour of a smoothing technique when the LM has to be integrated in a CSR system.

6.- References.

[1] García, P. and Vidal, E. (1990): "Inference of *k*-testable languages in the strict sense and application to syntactic pattern recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, nº 9, pp. 920-925.

[2] Bordel, G., Varona, A. and Torres, I. (1997): "*k*-TLSS(S) Language Models for Speech Recognition". *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol 2, pp 819-822.

[3] Varona, A. and Torres, I. (1999): "Using Smoothed *k*-TLSS(S) Language Models in Continuos Speech Recognition". *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*. vol II. pp 729-732

[4] Bordel, G., Torres, I. and Vidal, E. (1994): "Back-off smoothing in a syntactic approach to Language Modelling". *Proc. International Conference on Speech and Language Processing*, pp. 851-854.

[5] Torres, I , Varona, A. "An efficient representation of k-TSS language models". *Proc. of the IV Simposio Iberoamericano de Reconocimiento de Patrones, La Habana, Cuba, march 1999, pp. 645-654.*

[6] Ferretti, M. Maltese, G. Scarci, S. (1990), ."Measuring information provided by language model and acoustic model in probabilistic speech recognition: Theory and experimental results". *Speech Communication* 9. 531-539.

[7] Rubio, A. J., Segura, J. C., Garcia, p. Diaz, J.E. (1994); "Automatic Speech Recognition at UGR" *Proceedings in Attificial Inteligence CRIMFORWISS workshop on Progress and Prospects of Speech Researh and Technology.pp 158-165*

[8] Bourlard, H., Hermansky, H, Morgan, N. (1996): "Towards increasing speech recognition error rates." *Speech Communication* 18, pp 205-231.

[9] Jelinek, F. (1996): "Five speculations (and a divertimento) on the themes of H. Bourlard, H. Hermansky and N. Morgan". *Speech Communication* 18, pp 242-246.

[10] Clarkson, P. Rosenfeld, R. "Statistical language modeling using the CMU-CAMBRIDGE toolkit", (1997) *Proceedings of EUROSPEECH 97* pp- 2707-2710

[11] Diaz, J. E., Rubio, A. J., Peinado, A. M., Segarra, E., Prieto, N. and Casacuberta, F. (1993); "Development of Task Oriented Spanish Speech Corpora," *Proceedings of EUROSPEECH 93*.