Delimited smoothing technique over pruned and not pruned syntactic language models: perplexity and WER

A. Varona, I. Torres
Dpto. Electricidad y Electrónica. Universidad del País Vasco
Apdo. 644 48080 Bilbao. SPAIN
e_mail (amparo, manes}@we.lc.ehu.es

ABSTRACT[1]

Continuous Speech Recognition (CSR) systems require a Language Model (LM) to represent the syntactic constraints of the language. A sub-class of the regular languages, the *k* Testable in the Strict Sense (*k*-TSS) languages, has been used to generate LMs. Then, a smoothing technique needs to be applied to also consider events not represented in the training corpus. In this work, a new syntactic backing off smoothing approach, the Delimited discounting, was applied to several pruned and no pruned *k*-TSS LMs. Delimited discounting deals with the Turing discounting problems while keeping the Katz' smoothing schema. The experimental evaluation was carried out over a Spanish speech application task, showing that an increase of the test set perplexity of a LM does not always mean a degradation in the model performance when integrated in a CSR system. Besides, there is a strong dependence between the amount of probability reserved by the smoothing technique to be assigned to *unseen* events and the value of the balance parameter applied to the LM probabilities in the Bayes´s rule needed to get the best system performance.

1. INTRODUCTION

Continuous Speech Recognition (CSR) systems require a Language Model (LM) to integrate the syntactic and/or semantic constraints of the language. Both statistical (typically *N*-grams) and syntactic approaches have been extensively used to generate LMs. It is well known that language constraints could be better modelled under a syntactic approach [1], but the LM generation from samples and the full integration with the acoustic models have been usually considered as difficult tasks under these formalisms [2]. The use of stochastic automata to represent statistical language models has been recently proposed [3] [4] [5] with the aim to handle accurate language models in a one-step decoding procedure. A syntactic approach based on regular grammars, the *k*-Testable in the Strict Sense (*k*-TSS) languages [6] has also been proposed in previous works [7] to generate LM. They are a subclass of regular languages and can be considered as a syntactic approach of the well-known *N*-grams models.

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.

A major problem to be solved when using a LM is the estimation of the probabilities to be assigned to those events not represented in the training corpus, that is, *unseen* events. Thus, a smoothing technique needs to be applied when integrating a LM in a CSR system. In previous works [8] a syntactic backing-off smoothing was proposed and evaluated using *k*-TSS language models. The recursive schema required by the smoothing procedure has been well integrated in the finite state formalism and, thus, an efficient implementation of the backing-off mechanism was achieved [7].

Smoothing techniques are based on a discounting-distribution schema: a mass of probability needs to be discounted from *seen* events to further be assigned to *unseen* events. In previous works, a Witten-Bell based discounting procedure was proposed and evaluated over no pruned [7] and pruned [9] models. In this approach, the discounting factor was applied to the whole set of *seen* events in the training corpus [10]. As a consequence, the mass of probability to be assigned to *unseen* events could be overestimated. Thus, a new proposal, based on the well-known Turing discounting

[1] Work partially supported by the Spanish CICYT under grant TIC98-0423-C06-03

[11], the Delimited discounting, has been developed and presented in this work. In this case, discounting factors were only applied to those events scarcely observed in the training corpus.

The must reliable way to evaluate the real performance of a LM is to measure the obtained Word Error Rates (%WER) after integrating it into a CSR system. However, the most common way to assess the goodness of different LMs is the evaluation of the test set perplexity, even if the relationship with the acoustic models is not considered. So that, some important points related to a LM generation (like smoothing techniques, pruning thresholds), could not be always well assessed by the perplexity [12]. Thus, in this work both, the test set perplexity and %WER, were considered and compared to evaluate the behavior of Delimited discounting, developed under the *k*-TSS language modeling formalism. After this evaluation, the ability of the test set perplexity to predict the real behavior of a smoothing technique when working in a CSR system could be questioned.

CSR systems are invariably based on the well-known Bayes' rule. Bayes' rule maximizes the product of the probability of a sequence of acoustic observations *A* given a sequence of words W, *P*(*A*/W), and the probability that the word sequence W will be uttered, *P*(W). However it is well known that the best performance of a CSR system is obtained when P(W) is modified by introducing a balance parameter α in the following way: $P(W)^{\alpha}$ [13]. The effect of scaling LM probabilities, using no pruned and pruned k-TSS language models, was also evaluated in this work showing a strong relationship between the behavior of the smoothing technique and the value of the scaling factor required to obtain the best CSR system performance.

In Section 2, the syntactic language model formalism is briefly described. In Section 3, the Delimited discounting is presented within the backing-off smoothing technique. Section 4 deals with the experimental evaluation of several pruned smoothed *k*-TSS LMs in terms of both, perplexity and %WER, along with the effect of scaling the LM probabilities. These experiments were carried out over a Spanish speech application task (1,208 words). Finally, some concluding remarks are presented in Section 5.

2. - THE SYNTACTIC LANGUAGE MODEL.

A syntactic approach of the well-known N-grams models, the *k*-Testable Language in the Strict Sense (*k*-TSS) has been 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 [7]. 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 labeled as w_{i-k}^{i-1}. Each transition represents a *k*-gram, $k = 1...K$; it is labeled by its last word w_i and connects two states labeled 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:

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

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:

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

where Σ 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 labeled by w_j coming from state labeled 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 [7]. As an example, Figure 1 represents the *K*-grams $w_{i-(K-1)}^{i-1}$ and $w_{i-(K-1)+1}^{i}$ labeling two states of the automaton. When w_i is observed an outgoing transition from the first to the second state is set and labeled by w_i.

This approach represents a syntactic formalism of the well-known N-grams derived from the formal languages theory [6]. Moreover, it has been shown [14] that the probability distribution obtained trough and N-gram model is equivalent to the distribution obtained by a stochastic grammar generating *k*-TSS language, where *k* play the same role as N does in N-grams. At this point, choosing k-TSS or N-grams could be just a matter of representation convenience [8] [14]. It is well known that the use of grammar formalism presents several advantages. However, the lack of syntactic smoothing techniques and the problems arising when integrating the smoothed SFSA with acoustic models have restricted their use in CSR applications [3] [4] [14]. In such a case, several approaches to obtain approximate N-grams probability distributions have been proposed and typically managed [3] [4] [10] [15]. Nevertheless, in previous works a very compact representation of the *k*-TSS SFSA has been proposed [7] [8] [9]. This compact structure is dynamically expanded at decoding time to implement the real back-off mechanism. This proposal, as well as the new

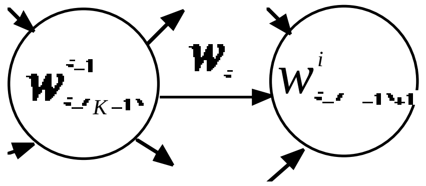

Figure 1: Two states of the K-TSS automaton labelled by K-grams $w_{i-(K-1)} w_{i-(K-1)+1} \ldots w_{(i-1)}$ and $w_{i-(K-1)+1} \ldots w_{i}$ labelling two states of the automaton. Transitions are labelled by words appearing in the training sample after K-grams labelling the source state.

syntactic smoothing presented in this work, allowed an efficient use of SFSA in CSR systems while keeping the probability distributions as they were defined by smoothing techniques.

3.- THE SMOOTHING TECHNIQUE

The probability associated to each transition representing *seen events* can be estimated under a maximum likelihood criterion (see Equation 1). However, a probability need also to be associated to those events not represented in the training corpus, i.e., *unseen events*. To deal with this problem, some probability mass should be discounted from observed events and then redistributed over *unseen* ones using some smoothing procedure. Backing-off smoothing was chosen in previous works [8] because the involved recursive scheme has been well integrated in the finite state formalism [7]. The syntactic approach suggested a state-dependent estimation of the total discount and, consequently, the symmetry principle was locally applied [8]. Thus the modified probability $P(w/q)$ to be associated to a transition $\delta^{k}(q, w) = (q', P(w/q))$ is estimated according to:

$$P(w/q) = \begin{cases} [1-\lambda]\dfrac{N(w/q)}{N(q)} & w \in \Sigma_q \\ \left(\displaystyle\sum_{\forall w_i \in \Sigma_q} \lambda \frac{N(w_i/q)}{N(q)}\right) \dfrac{P(w/b_q)}{1-\displaystyle\sum_{\forall w_i \in \Sigma_q} P(w_i/b_q)} & w \in \Sigma - \Sigma_q \end{cases} \qquad (2)$$

where Σ is the vocabulary of the task, that is, the set of words appearing in the training corpus; Σ_q is the vocabulary associated to state q and consists of the set of words appearing after the string labeling state q, i.e. words labeling the set of *seen* outgoing transitions from state q; $N(\mathrm{w}/q)$ is the number of times that word w_i appears after the string labeling state q and $N(q) = \sum_{\forall w \in \Sigma_q} P(w/q)$.

$P(w/b_q)$ is the estimated probability associated to the same event in the (k-1)-TSS model. In this schema, (1-λ) represents the discount factor, that is, the amount of probability to be subtracted and then be redistributed among *unseen* events. Figure 2 represents this schema for a state q labeled as $w_{i-(k-1)}^{i-1}$.

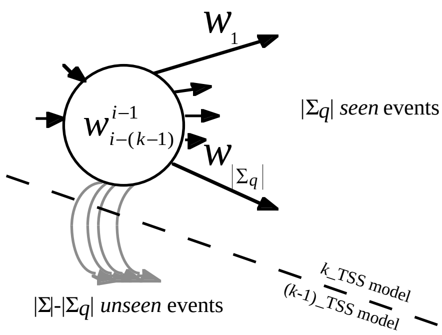

Figure 2: |Σq| *seen* events and |Σ|-|Σq| *unseen* events can be found at each state q labelled as $w_{i-(k-1)}^{i-1}$. The probability associated to unseen events is recursively obtained from less accurate models (k-1, k-2,…1) in back-off smoothing. Equation 2 is used to discount and redistribute a certain mass of probability.

Rigorous studies of several backing-off smoothing techniques could be found in [15] and [16]. On the one hand, the discounting factor (1-λ) could be applied to the whole set of seen events in the training corpus, as it is suggested in Equation 2. Within this classification, Absolute and Linear discounting are classical proposals in which the discounting factors depend on respective parameter values, whereas the Witten-Bell discounting is a proposal which does not depend on any parameter. On the other hand, discounting factor could be only applied to the scarcely seen events: Katz discounting.

In previous works [7] [8] [9], the Witten-Bell discounting was experimentally compared to other classical back-off methods leading to a significant decrease in test-set perplexity. However, because of applying the discounting factor to the whole set of *seen* events, i e, $\forall w \in \Sigma_q$, the mass of probability to be assigned to *unseen* events could be overestimated when using a smoothed k-TSS in a CSR system. This fact was observed using both no pruned [7] and pruned models [9]. Thus, in this paper a new proposal, the Delimited discounting was developed. Delimited discounting keeps the Katz discounting [11] philosophy subtracting a mass of probability only from scarcely *seen* events.

Delimited discounting (Dd).

The scheme devised by Katz [11] combines Turing discounting with backing-off. According to this formalism the probability associated to events occurring more than a fixed number of times, say r times, are estimated under a maximum likelihood criterion whereas events occurring less than r times, $N(w_i/q)<r$, are discounted a certain mass of probability. Thus:

$$P(w/q)=\begin{cases}\dfrac{N(w/q)}{N(q)} & w\in\Sigma_q \wedge N(w/q)>r\\ \left[1-\lambda\right]\dfrac{N(w/q)}{N(q)} & w\in\Sigma_q \wedge 1\le N(w/q)\le r\\ \sum\limits_{\substack{\forall w_i\in\Sigma_q\\ 1\le N(w_i/q)\le r}}\left[\lambda\dfrac{N(w_i/q)}{N(q)}\right]\dfrac{P(w/b_q)}{1-\sum\limits_{\forall w_i\in\Sigma_q}P(w_i/b_q)} & w\in\left(\Sigma-\Sigma_q\right)\end{cases} \qquad (3)$$

In Turing discounting the discounted mass of probability depends on n_1, n_2, ...,n_{r+1}, (being n_i the number of events which occur i times). The lower the count $N(w/q)$ is, the bigger discounting is applied, because higher counts are supposed to be better estimated. This approach puts some constrains to the relative values of n_1, n_2, ...,n_{r+1}, which are not always satisfied by *k*-grams models with medium and high values of *k*, due to the lack of an adequate distribution of the samples.

To avoid the Katz discounting problems, we proposed the Delimited discounting. As in the Katz model, the discounting operation was limited to low counts, i.e., $N(w/q)\le r$ in the following way:

$$1-\lambda = d-\tau(r-N(w/q)) \qquad \tau,d<1\wedge\tau <<< d \qquad (4)$$

Discounting depends on d and τ parameters' values, which must be minor than one. The bigger the count was ($N(w/q)\le r$) the lower discounting was applied. When $N(w/q)=r$, the discounting was the minimum (only depends on d parameter), and when $N(w/q)=1$, the discounting applied was the maximum (d-t(r-1)).

Another problem to be addressed when using Katz' discounting is that additional checks are required for those states for which all the events are *seen* more then *r* times. The remedy used in the CMU toolkit [10] to solve this problem is to increase the count at eat state $N(q)$ by one using the gained probability mass $1/(N(q)+1)$ to be redistributed over *unseen* events. However, the discount is applied to all the *seen* events' probabilities, which does not agree with Katz´s discounting philosophy. In our proposal, only the minimum counts were decremented (discounting only depends on *d* parameter) in the following way:

$$P(w/q)=\begin{cases}\dfrac{N(w/q)}{N(q)} & \begin{matrix}w\in\Sigma_q\wedge\\ N(w/q)>min(N(w/q))\end{matrix}\\ d\,\dfrac{N(w/q)}{N(q)} & \begin{matrix}w\in\Sigma_q\wedge\\ N(w/q)=min(N(w/q))\end{matrix}\\ \left(\sum\limits_{\substack{\forall w_i\in\Sigma_q\\ 1\le N(w_i/q)\le r}}\left[[1-d]\dfrac{N(w_i/q)}{N(q)}\right]\right)\dfrac{P(w_i/b_q)}{1-\sum\limits_{\forall w_i\in\Sigma_q}P(w_i/b_q)} & w\in\left(\Sigma-\Sigma_q\right)\end{cases}$$

As a consequence, events occurring a high number of times are estimated under a maximum likelihood criterion as in Katz proposal [11].

4. - EXPERIMENTAL EVALUATION OF PRUNED K-TSS LM.

The Delimited discounting was evaluated over a set of k-TSS language models integrated in a CSR system [17]. This evaluation was carried out in terns of both, the test set perplexity and the Word Error Rates (%WER) obtained by the CSR system.

The syntactical model integrating K k-TSS models (with k=1, 2..,K) can be easily handled at decoding time [7]. 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. So that, 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.

For these experiments a task-oriented Spanish speech corpus [18], consisting in 82,000 words and a vocabulary of 1,208 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 that composed the speech test set. Uttered sentences were decoded by the time-synchronous Viterbi algorithm with a fixed beam-search to reduce the computational cost. A chain of Hidden Markov models representing the acoustic model of the word phonetic chain replaced each transition of the k-TSS automaton.

First, Table 1 shows the memory requirements for several k-TSS models (k=2,...,5) 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 shows an important reduction of the

Table 1: Number of states (LMst) and memory requirements when several pruned k-TSS language models (k=2,..,5) when different pruning factors (*pf*) were considered.

k	*pf*	*LMst*	*memory (Mb)*
2	1	1,213	0.13
3	1	7,479	0.43
	2	3,854	0.20
	3	2,845	0.14
	4	2,336	0.11
	5	1,999	0.09
4	1	21,551	0.95
	2	9,360	0.38
	3	6,366	0.25
	4	4,993	0.19
	5	4,139	0.16
5	1	42,849	1.69
	2	16,086	0.58
	3	10,260	0.36
	4	7,795	0.26
	5	6,308	0.22

number of states (LMst) of the k-TSS language models when higher *pf* were applied. These reductions were even more important for higher values of k.

In a first series of experiments, the proposed Delimited (Dd) discounting was applied to several no pruned smoothed *k*-TSS language models, with *k*=2,...,6. Different values of *d* parameter in Equation 4 were tested while keeping fixed values for parameter τ (τ=0.01). The minimum number of times *r* required for a maximum likelihood estimation of event probabilities (Equation 3) was also set to *r*≈7.

Figure 3 shows the obtained Perplexity (PP) results when not pruned models were evaluated. The more mass of probability (<<*d*) was assigned to the unseen events by the Delimited discounting procedure (up to a maximum *d*=0.70), the better (lower) value of perplexity was observed. Thus, the best perplexity results were obtained when the higher mass of probability was reserved to be applied to unseen events, even if the differences around the optimum were not very meaningful.

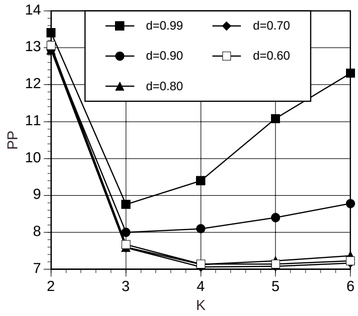

Figure 3. - PP obtained by several no pruned smoothed k-TSS LM after applying Delimited discounting with several values of *d* parameter in Equation 4.

Two different *d* parameter values were chosen to evaluate pruned models: those which got a "good" LM (low PP), i.e., *d*=0.70 and a "bad" LM (high PP), i.e., *d*=0.99. Figure 4 shows the experimental evaluation in terms of perplexity for those pruned k-TSS language models presented in Table 1, using Delimited discounting *d*=0.70 and *d*=0.90 respectively.

Figure 4 shows that: when *d*=0.70 the values of the perplexity were almost constant for values of *k* higher than 3 and that behavior was not depending on *pf*. Perplexity values went up when *pf* was higher but they were always under a fixed value (around 15). On the contrary, when *d*=0.99 the perplexity reached very high values with *pf*, especially when higher values of k were considered.

Figure 5 shows the %WER obtained by the same pruned and not pruned smoothed LM in Figure 4, when they were integrated in the CSR system. When *d*=0.70 the obtained %WER were also almost constant for values of k higher than 3. However, the most surprising results reported in Figure5 is that the %WER significantly decreased (from 35.5% with not pruned models to 33% with pruned models) when *pf* increased (up to *pf*=4). Values of *pf* higher than 4 led to excessive model degradation and, as a consequence, to a lower system performance.

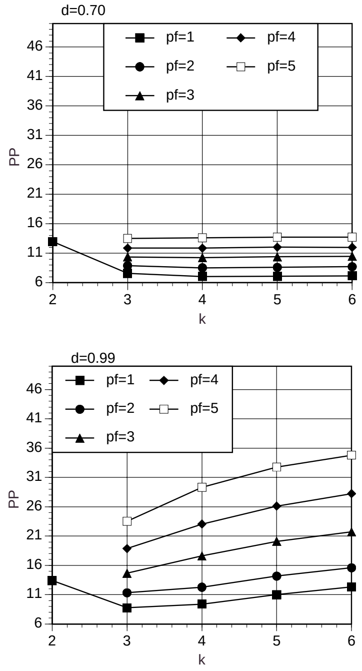

Figure 4. - PP obtained by the pruned k-TSS LMs of Table 1 using Delimited (Dd) discounting methods, *d*=0.70 and *d*=0.99 respectively.

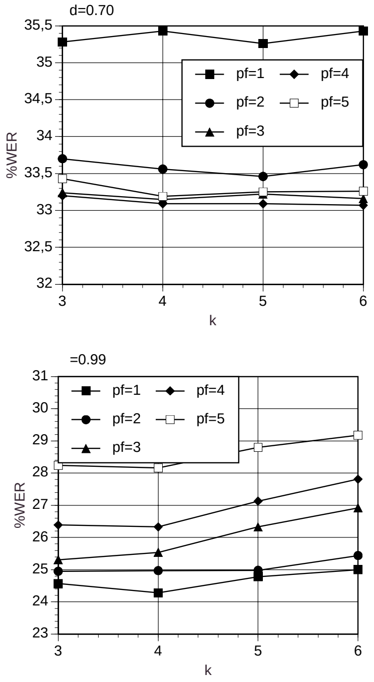

Figure 5. - %WER obtained by the pruned k-TSS LMs of Table 1 (*d*=0.70 and *d*=0.99).

When *d*=0.99, the best %WER results were obtained with not pruned models (*pf*=1) (from 23% with not pruned models to 28% with pruned models). But, even then, lower smoothed models (*d*=0.99) achieved better performances, even if they are pruned, that higher smoothed models (*d*=0.70).

In all cases, pruned models require less memory to be allocated that not pruned models (see Table 1). Moreover, they also need less time to decode each sentence since the average active nodes per frame also decrease with *pf* (see Figure 6).

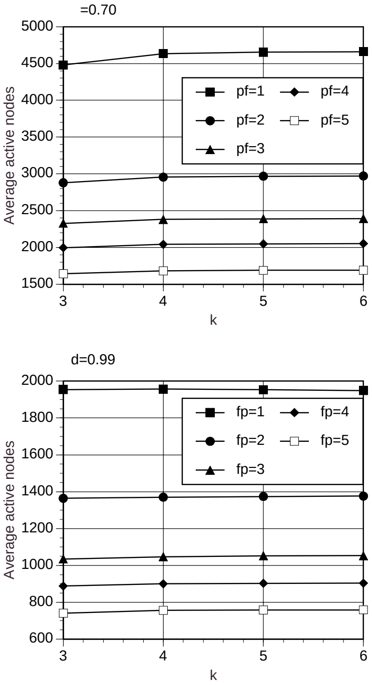

Figure 6. - Average active nodes obtained by the pruned *k*-TSS LMs in Figures 4 and 5 (*d*=0.70 and *d*=0.99).

This behavior can be explained by analyzing the smoothing technique. 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 decrease. In this case, those infrequent k-grams were decoded as unseen events when appearing in the test set. Therefore, the distribution of the probability mass between observed and unobserved events made by the syntactic smoothing technique has been seriously modified. The probability mass assigned to the back-off transition, i.e. unseen events, is then smaller for pruned models. 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.

To summarize the obtained results it could be said that the syntactic back-off smoothing technique with *d*=0.70 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 and lower smoothed models (*d*=0.99) got better performances.

However, the test set perplexity did not report this behavior. As it was previously reported in [12] [9], an increase of the LM perplexity does not always means degradation in the system performance. Figures 4 and 6 show that when high smoothed models (*d*=0.70) were evaluated, perplexity increased when *pf* did, but %WER decreased. Besides, when a lower smoothed model was evaluated (*d*=0.99) better %WER were obtained in spite of the bad perplexity results.

4.1. - SCALLING 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: $(P(\mathrm{W}))^{\alpha}$ [13]. Several values of the balance parameter α were tested to optimize the percentage of words correctly decoded by the system.

Figure 7 shows the experimental evaluation for k=3 and Figure 8 for k=4 when different pruning factors (*pf*) and balance parameter values (α) were considered. Points at the bottom left corner of each plot show the best system performances: the lowest %WER values and the lowest values of the average active nodes in the lattice.

Figure 7 and Figure 8 show that there were not significative differences among the obtained results with different values of k. 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 when the balance parameter α increased (up to a maximum).

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 7 and 8). Therefore, the use of α values greater than one lead to lower %WER and average numbers of active nodes in the lattice.

However a different behavior of pruned and not pruned models was observed:

a) The value of α optimising the *%WER* was slightly higher for not pruned *k*-TSS models (α around 5 when *d*=0.70 and around 4 when *d*=0.99) than for pruned *k*-TSS models (a around 4 when *d*=0.70 and around 3 when *d*=0.99).

b) Pruned models got better performances than not pruned models when a high smoothed model was used and the balance parameter values remains under 4. A particular case, α=1, of this surprising behavior was shown in the experiments previously reported.

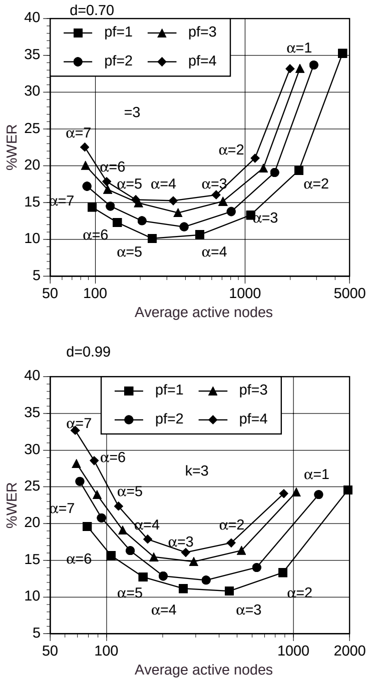

Figure 7. - %WER obtained by the pruned 3-TSS LMs in Figures 4 and 5 using different values of the α parameter (d=0.70 and d=0.99).

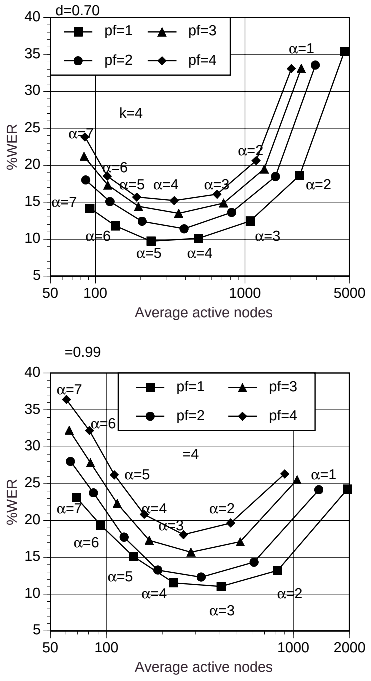

Figure 8. - %WER obtained by the pruned 4-TSS LMs in Figures 4 and 5 using different values of the α parameter (d=0.70 and d=0.99).

However, for high values of a, high values of the pruning factor produced worse *%WER*. For low smoothed models (d=0.99) pruned models got worse results for any α value.

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 previously observed. Both procedures, pruning k-TSS models and scaling the LM probabilities, produced similar effects: decreasing both the %WER (with high smoothed models) and 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 α value of α>1 in the recognition scheme.

At the end of each word at the Viterbi trellis there is an accumulated probability. The gap between these accumulated probabilities is usually bigger than the gap between the LM probabilities due to the acoustic probabilities (whose values are smaller and are applied much more times in the Viterbi trellis). The immediate consequence is that the values of LM probabilities are irrelevant in most part of the situations to decide the best way to follow. However, when the LM probabilities are raised to a power α: $(P(w))^{\alpha}$, all of them are attenuated, but this attenuation is higher for lower probability values. So that, the gap between the high and low probabilities is also bigger and then the LM probabilities are more and more competitive with the increase of a values, up to a maximum where LM probabilities are overvalued.

The smoothing technique reserves a different mass of probability to be assigned to *unseen* events depending on the *d* valued. When this mass of probability is higher (*d*=0.70), the probability assigned to seen events is lower (flat probability distribution). In such a case, a bigger value of the parameter α was needed to get the best performance (see Figure 7 and Figure 8).

Figures 7 and 8 show a behavior of *%WER* completely different to the one observed in Figure 6, when a value of α=1 was used. Now *k*-TSS language models with lower perplexity values lead to better *%WER* 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 behavior 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.

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. In this work, a new syntactic backing off smoothing approach, the Delimited discounting, was applied to also consider events not represented in the training corpus. Delimited discounting deals with the Turing discounting problems while keeping the Katz' smoothing schema.

An experimental evaluation of pruned and not 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, high smoothed pruned models achieved better performance (better percentage of words correctly decoded and less time to decode a sentence) that not pruned models. Besides, low smoothed pruned and no pruned models got better performances that high smoothed 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 behavior of pruned and not pruned models was observed: pruned models got better performances in terms of average active nodes and %WER 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 degradation in the model performance. The test set perplexity seems not to be an adequate measure to predict the behavior 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.

6. REFERENCES

[1] F. Jelinek, (1991): "Up from trigrams: the struggle for improved language models," *proc. of the Eurospeech 91*; Genova, Italy, pp.1037-1039.

[2] P. Placeway, R. Schwartz, P., Fung, L. and Nguyen, (1993): "The estimation of powerful Language Models from small and large corpora," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing* , vol. II, pp. 33-36

[3] G. Riccardi, R. Pieraccini, E. Bocchieri, (1996): “Stochastic automata for language modeling”. *Computer Speech and Language* 10, pp. 265-293.

[4] M. Suzuki, H. Aso, (1999): “An automatic acquisition method of statistic finite-state automaton for sentences”. *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*

[5] D. Llorens (2000): “Suavizado de aut\'omatas y traductores finitos estoc\'asticos” PhD thesis, Universidad Politécnica de Valencia.

[6] P. García, and E. Vidal, (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.

[7] A. Varona and I. Torres (1999) “Using Smoothed K-TSS Language Models in Continuous Speech Recognition”. Procc. IEEE Int. Conf. Acoust, Speech, Signal Processing. Vol II pp. 729-732.

[8] G. Bordel, I. Torres and E. Vidal (1994): “Back-off smoothing in a syntactic approach to Language Modeling”. Proc.ICSLP-94, pp. 851-854.

[9] A. Varona and I. Torres (2000) “Evaluating pruned k-TSS language models: perplexity and word recognition rates”. In *Pattern Recognition and Applications*, Frontiers in Artificial Intelligence series, Ios Press Publisher, The Netherlandas.

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

[11] S. M.Katz. (1987). “Estimation of Probabilities from Sparce Data for The Language Model Component of a Speech Recognizer”. IEEE Trans. on Acoustics, Speech and Signal Processing,. vol. ASSP-35, n 3, pp. 400-401.

[12] P. Clarkson, T. Robinson (1999). ”Towards improved language model evaluation measures”. Procc of EUROSPEECH.99. Vol 5. pp 1927-1930

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

[14] E. Segarra (1993): “Una Aproximación Inductiva a la Comprensión del Discurso Continuo”. PhDthesis, Universidad Politécnica de Valencia.

[15] F. S. Chen, J. Goodman, (1999). “An empirical study of smoothing techniques for language modeling”. *Computer Speech and Language*. Vol 13. pp 359-394.

[16] H. Ney, S., Martin, F Wessel, (1997): “Statistical Language Modeling using leaving-one-out”. In S. Young and G. Bloothooft (eds.). Corpus-based methods in Language and Speech processing, pp. 174-207. Kluwer Academic Publishers.

[17] L.J. Rodriguez, I.Torres, J.M Alcaide. A. Varona, K. López de Ipiña, M.Peñagarikano. G. Bordel (1999). “An Integrated System for Spanish CSR Tasks”. Procc of EUROSPEECH.99. Vol 2. pp 951-954

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