EFFICIENT SEARCH WITH POSTERIOR PROBABILITY ESTIMATES IN HMM-BASED SPEECH RECOGNITION

Daniel Willett, Christoph Neukirchen, Gerhard Rigoll

Department of Computer Science
Faculty of Electrical Engineering
Gerhard-Mercator-University Duisburg, Germany
e-mail: {willett,chn,rigoll}@fb9-ti.uni-duisburg.de

ABSTRACT

In this paper we present the methods we developed to estimate posterior probabilities for HMM states in continuous and discrete HMM-based speech recognition systems and several ways to speed up decoding by using these posterior probability estimates. The proposed pruning techniques are State Deactivation Pruning (SDP), similar to an approach proposed for hybrid recognition systems, and a novel posteriori-based lookahead technique, Posteriori Lookahead Pruning (PLP), that evaluates future posteriors in order to exclude unlikely HMM states as early as possible during search. By applying the proposed methods we managed to vastly reduce the decoding time consumed by our time-synchronous Viterbi-decoder for recognition systems based on the Verbmobil and the Wall Street Journal database with hardly any additional search error.

1. INTRODUCTION

With the introduction of long-span language models, very large vocabularies and context-dependent acoustic models, the problem of an efficient search for the most probable sentence for a spoken utterance became increasingly important. Several techniques became state-of-the-art like the usage of a tree lexicon, tree copies, language model lookahead and others. The integration of these techniques in our time-synchronous Viterbi-decoder, gave us the expected remarkable speed-ups [5]. Additionally, we developed some novel pruning techniques, based on the estimation of posterior probability estimates that we will report about in this paper.
Posterior probabilities were first exploited for pruning by Renals et al. [7] in the NOWAY stack-decoder for decoding with hybrid recognition systems. Contrary to standard continuous HMM systems that are based on the estimation of probabilistic distribution functions for the observations' likelihoods, hybrid recognition systems utilize neural networks to estimate the HMM states' probability given an acoustic observation as input [1]. These posterior probabilities are discriminative by nature as they sum up to unity over all HMM states. For this reason, Phone Deactivation Pruning proved to be a very effective method for speeding up decoding with this type of system architecture. Phone Deactivation Pruning simply limits the possible HMMs for a specific observation to those whose posterior probability does exceed a certain threshold. The possibility of applying this pruning technique is often referred to as one of the major advantages of these NN/HMM hybrids.
Standard HMM systems are based on the estimation of the observations' likelihood functions $p(\mathbf{x}|w)$ for each HMM state w, with $\mathbf{x}$ denoting an arbitrary feature vector extracted from the acoustic observation. However, the Bayes' Formula can be applied to derive local posterior state probabilities from the probabilistic distribution functions (pdfs) according to

$$P(w|\mathbf{x}) = \frac{p(\mathbf{x}|w)P(w)}{p(\mathbf{x})} \tag{1}$$

This requires the two additional probability estimates $P(w)$ and $p(\mathbf{x})$. $P(w)$, the state w's prior probability, can be estimated easily on the training data, just like it is done in hybrid systems for transforming posterior probability estimates into likelihood estimates. The efficient estimation of the observation's prior $p(\mathbf{x})$ will be discussed in the following sections for several types of HMM systems.

2. ESTIMATING POSTERIOR PROBABILITIES IN CONTINUOUS HMM SYSTEMS

In continuous HMM systems the pdfs are modeled as weighted sums of multi-variat basic distribution functions, usually Gaussians or Laplacians [3].

$$p(\mathbf{x}|w) := \sum_{i=1}^{C_w} d_{g_{wi}} \cdot g_{wi}(\mathbf{x}) \tag{2}$$

In the equation above w denotes the HMM state, $d_{g_{wi}}$ are w's mixture weights and C_w resembles the number of basic functions used in the pdf of state w. g_{wi} are the basic functions. For matter of simplicity we will use the term Gaussian in the further text, knowing that all that will be stated also holds true for other types of basic functions. For the estimation of $p(\mathbf{x})$ in general, two methods are possible:
On the one hand, a separate pdf can be trained on all the training data to model the general distribution of the acoustic observations. In order to achieve a good resolution, usually a large number of mixture components is needed, so that the estimation of $p(\mathbf{x})$ with this independent modeling is computationally not cheap.
On the other hand, $p(\mathbf{x})$ can be estimated as the sum over all the HMM states' likelihood functions weighted by the states' priors according to

$$p(\mathbf{x}) \approx \sum_{w} P(w)p(\mathbf{x}|w) \tag{3}$$

Unfortunately though, this computation requires the evaluation of all states' pdfs which is computationally very expensive. Thus, the estimation of $p(\mathbf{x})$ with this formula does not seem to be worth it. In continuous systems, however, this estimation of $p(\mathbf{x})$ can be simplified according to

$$p(\mathbf{x}) \approx \sum_{w} P(w)p(\mathbf{x}|w) = \sum_{w} \left(P(w) \sum_{i=1}^{C_w} d_{g_{wi}} g_{wi}(\mathbf{x}) \right)$$

$$= \sum_{j=1}^{C} \left(\sum_{\text{all } w \text{ tied to } g_j} P(w) d_{g_{wj}} \right) g_j(\mathbf{x}) = \sum_{j=1}^{C} D_j \cdot g_j(\mathbf{x}) \tag{4}$$

where C denotes the total number of Gaussian mixture components. Hence, $p(\mathbf{x})$ estimated according to Eqn. (3) turns out to be a weighted sum of the pdfs' mixture components, too.

2.1. CHMM

In classical continuous systems that do not make use of any mixture tying, the number C of mixture components is usually very large and the evaluation of Eqn. (3) according to Eqn. (4) is too expensive. In this case the proposed training of a separate pdf for modeling $p(\mathbf{x})$ is more adequate. However, if some kind of mixture tying is used, Eqn. (4) simplifies drastically and its evaluation can be accomplished with lower computational cost.

2.2. SCHMM

Semi-continuous HMM systems offer the ideal structure for estimating the observations' priors according to Eqn. (4). In SCHMM systems [4] all pdfs share a common "codebook" of Gaussian distributions. In this case, the equation simplifies to

$$p(\mathbf{x}) \approx \sum_{i=1}^{C} \left(\sum_{all\ w} P(w) d_{g_{wi}} \right) g_i(\mathbf{x}) \tag{5}$$

with a reasonably small number C of mixture components. Hence, $p(\mathbf{x})$ turns out to be another weighted sum of the Gaussian codebook. Its evaluation is as cheap as the evaluation of a single conditional likelihood $p(\mathbf{x}|w)$.

2.3. GENERALIZED MIXTURE TYING

Besides the classical CHMM and SCHMM systems several more generalized types of tying of mixture components have been proposed [2, 8]. They aim to find the ideal tradeoff between the robustness of SCHMM systems and the resolution of CHMM systems. In general, they have more Gaussian components than SCHMM systems. Most commonly, however, the number is small enough, so that the estimation of $p(\mathbf{x})$ as a weighted sum of all the system's mixture components according to Eqn. (4) is usually worth the computation. Nevertheless, the more mixture components there are and the less tied the whole system is, the more useful gets an independent modeling of $p(\mathbf{x})$ as proposed for CHMM systems.

3. ESTIMATING POSTERIOR PROBABILITIES IN DISCRETE HMM SYSTEMS

Discrete HMM systems map the (continuous) feature vector $\mathbf{x}$ onto discrete labels m with some kind of vector quantization $m(\mathbf{x})$. The pdfs are modeled by discrete probabilities according to

$$p(\mathbf{x}|w) \propto p(m(\mathbf{x})|w) \tag{6}$$

and the prior likelihood $p(\mathbf{x})$ can be estimated as

$$p(\mathbf{x}) \approx \sum_{w} P(w) p(\mathbf{x}|w) \propto \sum_{w} P(w) p(m(\mathbf{x})|w) \approx p(m(\mathbf{x})) \tag{7}$$

and turns out to be just another discrete distribution on the labels m. Thus, the estimation of posterior probabilities according to

$$P(w|\mathbf{x}) \approx \frac{p(m(\mathbf{x})|w) P(w)}{p(m(\mathbf{x}))} \tag{8}$$

can be implemented as a mere table lookup. This way, the estimation of posterior probabilities in discrete HMM systems is extremely cheap.

4. ESTIMATING POSTERIOR PROBABILITIES IN SYSTEMS OF MULTIPLE STREAMS

It is a common practice in HMM-based speech recognition to group the extracted features into streams that are modeled independently. This is the more useful the less correlated the divided features are. When having multiple feature streams the total likelihood of an observation x computes to

$$p_\lambda(\mathbf{x}|w) := \prod_{s=1}^{M} p_{s\lambda}(\mathbf{x}|w)^{w_s} \tag{9}$$

with the stream weights w_s usually set to unity. The multi-stream prior distribution $p(x)$ can be modeled as

$$p_\lambda(\mathbf{x}) \approx \prod_{s=1}^{M} \left(\sum_{w} P(w) p_{s\lambda}(\mathbf{x}|w) \right)^{w_s} \tag{10}$$

With this prior distribution the posterior probability estimates can be set up according to Eqn. (1). Nevertheless, it is possible to use only a subset of the streams for estimating posteriors. The advantage is a less expensive computation. Or it is possible to use each stream's posterior probability estimate independently for pruning. We made some experiments using stream specific estimates, the results of which can be found in Section 7.

5. STATE DEACTIVATION PRUNING

In the previous sections we showed, that it is possible to gain posterior probability estimates in likelihood-based HMM systems with only little computation, too. These estimates can be used similar to the way they were exploited in [7], with a global threshold that is used to prune those states w for an observation $\mathbf{x}$ whose posterior probability estimate falls below.

$$P(w|\mathbf{x}) = \frac{p(\mathbf{x}|w) P(w)}{p(\mathbf{x})} < \text{thresh} \Rightarrow \textbf{prune state } w \tag{11}$$

Experimental Results will be presented in Section 7.
However, we found it to be even more useful to have individual thresholds for each HMM state. These thresholds thresh_w can be easily computed as the minimum posteriors on the training data. Experiments with individual thresholds are displayed in Section 7 as well.

$$P(w|\mathbf{x}) = \frac{p(\mathbf{x}|w) P(w)}{p(\mathbf{x})} < \text{thresh}_w \Rightarrow \textbf{prune state } w \tag{12}$$

Another advantage of these individual thresholds is that the priors $P(w)$ can be omitted in the calculation, and thresholds thresh' can be set up for only the quotient $p(\mathbf{x}|w)/p(\mathbf{x})$.

$$\frac{p(\mathbf{x}|w)}{p(\mathbf{x})} < \frac{\text{thresh}_w}{P(w)} \Rightarrow \frac{p(\mathbf{x}|w)}{p(\mathbf{x})} < \text{thresh'}_w \tag{13}$$

Phone Deactivation Pruning was first proposed and evaluated for hybrid systems by Renals in [7]. Using a fixed threshold for the frame-based HMM posteriors, speed-ups by an order of magnitude were reported for the NOWAY stack-decoder. In our Viterbi-decoder, we integrated the posteriori estimation and the posteriori pruning at the state level. Therefore, it is referred to as State Deactivation Pruning. However, we believe, that the two approaches do not differ much in computational effort as well as in the effect on decoding speed.

6. POSTERIORI LOOKAHEAD PRUNING

Figure 1 shows the posterior probability estimates (normalized by dividing with $P(w)$) of the four HMMs $/th/$,$/ae/$,$/t/$ and $/s/$ in an utterance of the words "That's" extracted from an utterance of the WSJ0 corpus. It is obvious that the posterior probability estimates reach their maximum values at the phones' centers while close to the phonetical borders the discrimination among the posterior probability estimates is only weak. This corresponds to our observation that usually State Deactivation Pruning cannot prevent an HMM node from being expanded at all, but only comes into effect after several frames have passed, during which the posterior probability did not drop below the threshold.
Ney et al. [6] developed a likelihood-based lookahead technique,

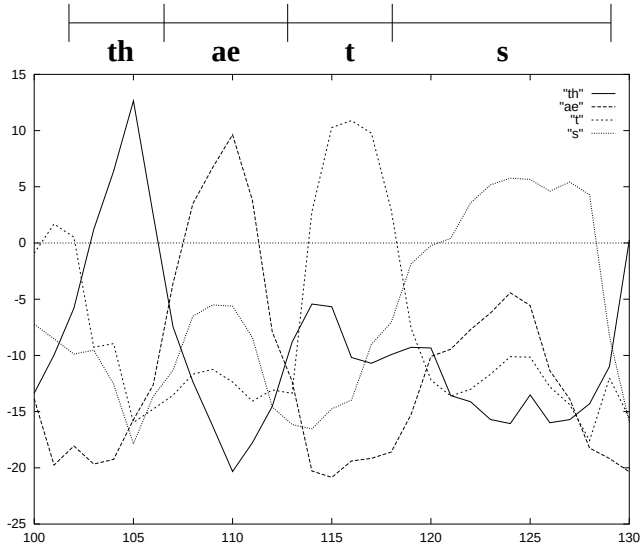

Figure 1. Normalized posterior probability estimates and alignment of the phones in an utterance "That's"

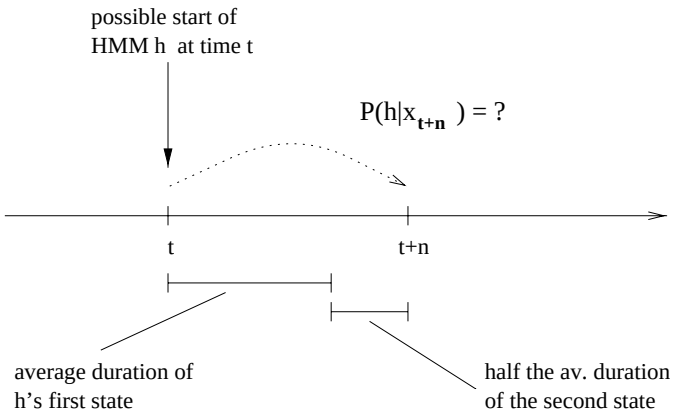

Figure 2. Looking ahead into the probable phone center

that evaluates several future frames with simplified acoustic models whenever a possible HMM start occurs. This procedure is computationally quite expensive and the discrimination among the likelihoods is often very weak. Furthermore is it usually not possible to reuse the computations made for the phonetical lookahead when evaluating the accurate acoustic models. Based on the proposed methods for estimating posterior probabilities, we developed a novel lookahead technique that is considerably simpler:
At any possible start of a new HMM h, the decoder looks ahead in time for an HMM-specific number of frames, in order to check the posterior probability for the frame that is most likely to be close to the specific HMM's center. Once the HMM's posterior probability at this point turns out to fall below a certain lookahead threshold lah_h, the whole HMM is pruned. This means that the HMM h must not start at the actual frame. Again, the thresholds can be computed on the training data. In our recognition systems that use HMMs of three emitting states each, the HMM posterior probability is estimated by its second state. Figure 2 illustrates this procedure. The start of HMM h is omitted if the HMM's posterior probability at $t+n$ falls below the HMM-specific lookahead threshold lah_h.

$$P(h|x_{t+n}) < \text{lah}_h \Rightarrow \textbf{forbid start of HMM h at time t} \quad (14)$$

n resembles the average duration of h's first state plus half the average duration of the second. Thus, it is the average duration to the center of the second state in the standard linear three-state HMM system that we use. With a different HMM topology, different formulas for estimating the average duration to the HMM's center have to be applied.

7. IMPLEMENTATIONAL REMARKS

In addition to the phone models, common recognition systems use a dedicated model for silence and a word boundary model. Our Verbmobil system additionally uses an HMM that models several types of noise. The duration of such non-phoneme models has an extremely large variation. Therefore, we found that Posteriori Lookahead Pruning is inadequate for these models. In the State Deactivation Pruning, however, those models can be included without restrictions.
As stated before, the estimation of thresholds for State Deactivation Pruning as well as for the Posteriori Lookahead Pruning can be performed on the training data. For our experiments we simply computed the minimum posterior probabilities for each HMM state on a Viterbi-alignment and used them as thresholds for State Deactivation Pruning. The lookahead thresholds were computed as each HMM's minimum posterior probability at the specific distance from its beginning.

8. EXPERIMENTS AND RESULTS

We evaluated the proposed pruning techniques using our Viterbi-decoder that was first presented in [5]. It performs a time-synchronous beam search in a network of partial tree copies in order to incorporate N-gram language models into the search. Within the tree copies language model smearing [6] is performed in order to apply the language model as early as possible during search.
Both recognition systems, that we evaluated the pruning techniques on, use an ordinary semi-continuous acoustic modeling with 200 Gaussian mixture distributions in each of four streams. The extracted features are 12 Cepstrum coefficients, first and second order delta coefficients of these values, and log energy with delta coefficients. The experiments were run on a DEC alpha 366 MHz workstation.
The German spontaneous speech recognition system was trained on parts of the Verbmobil speech database. The test-set that was used consists of 265 sentences. It is the test-set from the Verbmobil evaluation of 1995. Using this system, we only conducted a few preliminary experiments. The best results are summarized in Table 1. The pruning thresholds were set as explained in the previous section. The time displayed in the table is the time consumed for decoding all sentences with a beam width set to the minimum value that causes no additional search error. Using State Deactivation and Posteriori Lookahead Pruning, we achieved a reduction of the time consumed for decoding by an order of magnitude with hardly any decrease in recognition accuracy.
The evaluation of the proposed pruning techniques on a recognition system for the WSJ0 5000-words task are displayed in more detail. The tests were performed using the Nov. 92 test-set of 330 sentences and the 5k bigram language model of perplexity 110. Table 2 lists the evaluation of State Deactivation Pruning with fixed and individual thresholds according to Eqn. (11) and (13) with a global posterior probability estimation for all streams. Again, the individual thresholds were estimated as the minimum posteriors on the training data. The first row (thresh $= 0.0$) displays the decoding time and recognition accuracy of the baseline system without any posteriori-based pruning. It turned out that the individual thresholds, estimated as minimum posteriors on the training data, have to be slightly increased for best performance. This is probably due to some extreme outliers among the WSJ0 training

System	word error [%]	time [100s]
baseline	28,78	140
State Deactivation Pruning	28,80	102
SDP & PLP	28,66	74

Table 1. Experiments with a German spontaneous speech recognition system

threshold	correctness [%]	word error [%]	time [100s]
0.0	**90.68**	**10.76**	**220**
e^{-10}	90.58	10.84	162
e^{-5}	88.70	15.02	128
e^{-3}	85.04	18.42	82
min	90.82	10.64	240
$e^{log(min)+1.0}$	90.80	10.68	155
$e^{log(min)+2.0}$	90.73	10.82	144
$e^{log(min)+3.0}$	90.32	11.47	135
$e^{log(min)+4.0}$	89.91	12.11	123
$e^{log(min)+5.0}$	89.01	13.22	113

Table 2. SDP with global and individual thresholds

data. This observation differs from the experiments with the German recognition system where the use of these mere minimum posteriors as thresholds improved the decoding speed.
The slight increasing of the thresholds was performed by adding fixed values in the log-domain as displayed in rows 6 to 10 of Table 2, where min denotes the minimum posteriors estimated on the training data. Table 3 displays the results measured using individual posterior estimates and thresholds for several of the feature streams. In this table the increasing of the thresholds in the log-domain is indicated by $+$ and the additional value. The result in the forth row that only has 1% of additional word error with a remarkable speed-up in decoding might be the best result we obtained without Posteriori Lookahead on the WSJ system. The experiments with this novel lookahead technique are listed in Table 4. Here, it is probably the 10. and the 13. row that are most interesting. Row 13 shows an additional error of about 1% consuming about half the time for decoding. In Row 10 an additional decoding error of less than 2% was measured with a decoding time of about 40% of the baseline system. For all the improvements in decoding time, it has to be taken into account that the contribution of the likelihood computation procedure in the overall decoding time is at about 30% when applying no posteriori-based pruning and that it is even higher when the pruning techniques are applied. Therefore,

threshhold	streams	correctness [%]	word error [%]	time [100s]
min	1	90.38	11.13	161
min	1,2	90.27	11.32	150
min	1,2,3	90.26	11.55	143
min	**1,2,3,4**	**90.20**	**11.77**	**129**
+1.0	1	90.20	11,77	119
+1.0	1,2	88.46	14,40	101
+1.0	1,2,3	86.65	17.43	91
+1.0	1,2,3,4	84.91	20.09	77

Table 3. Stream-wise State Deactivation Pruning

thresh	lah	streams	correctness [%]	word error [%]	t [100s]
min	min	global	90.82	10.64	256
min	+5.0	global	90.50	11.13	206
min	+8.0	global	90.01	11.80	170
min	+10.0	global	89.66	12.41	147
min	+12.0	global	88.36	14.43	129
min	+15.0	global	84.81	19.63	102
+2.0	min	global	90.73	10.82	171
+2.0	+5.0	global	90.44	11.19	135
+2.0	+8.0	global	89.88	11.94	111
+2.0	**+10.0**	**global**	**89.46**	**12.63**	**89**
+2.0	+12.0	global	88.20	14.62	76
+2.0	+15.0	global	84.46	19.92	68
+1.0	**+1.0**	**1**	**90.11**	**11.70**	**107**
+1.0	+1.0	1,2	88.38	14.48	93
+1.0	+1.0	1,2,3	86.90	17.68	82
+1.0	+1.0	1,2,3,4	84.58	20.35	70

Table 4. Experiments with Posteriori Lookahead

the measured gain in the total time consumed for decoding has to be rated as an even bigger improvement in the mere search process.

9. CONCLUSION

The paper has shown that estimates of the posterior state probabilities can effectively be computed and utilized for an efficient search in continuous HMM-based recognition systems. State Deactivation Pruning proved to vastly speed up the decoding procedure with ordinary time-synchronous Viterbi-decoders. In addition to that, a novel lookahead technique has been proposed that proved to be capable of providing another remarkable speed-up. With the proposed pruning techniques we were able to double the speed of a German spontaneous speech recognition system with hardly any additional error. On the WSJ system the same speed-up comes along with an additional error of 1%. This slight gap is most probably due to badly set thresholds. Some more work will have to be spent on the estimation of useful state-dependent posteriori thresholds. Several smoothing and clustering techniques could be applied in order to get better thresholds for sparsely represented HMM states and to gain resistance to extreme outliers among the posterior probability estimates on the training data.

REFERENCES

[1] H. Bourlard, N. Morgan, "Connectionist Speech Recognition - A Hybrid Approach", *Kluwer Academic Press*, 1994.

[2] V. V. Digalakis, P. Monaco, H. Murveit: "Genones: Generalized Mixture Tying in Continuous Hidden Markov Model-Based Speech Recognition", IEEE Transactions on ASSP, Vol. 4, 1996, pp. 281–289.

[3] X. D. Huang, Y. Ariki, M. A. Jack: "Hidden Markov Models for Speech Recognition", Edinburgh University Press, 1990.

[4] X. D. Huang, K. F. Lee, H. W. Hon: "On Semi-Continuous Hidden Markov Models", Proc. ICASSP'90, pp. 689–692.

[5] C. Neukirchen, D. Willett, G.. Rigoll: "Reduced Lexicon Trees for Decoding in a MMI-Connectionist/HMM Speech Recognition System", EUROSPEECH '97, Rhodes, pp. 2639–2642

[6] H. Ney, R. Haeb-Umbach, B.-H. Tran, M. Oerder, "Improvements In Beam Search For 10000-Word Continuous Speech Recognition", Proc. ICASSP'92, pp. I 9–12.

[7] S. Renals, M. Hochberg, "Efficient Search Using Posterior Phone Probability Estimates", Proc. ICASSP'95, pp. 596–599.

[8] D. Willett, G. Rigoll: "A New Approach To Generalized Mixture Tying For Continuous HMM-Based Speech Recognition", EUROSPEECH '97, Rhodes, pp. 1175–1178