Modelling and Decoding of Crossword Context Dependent Phones in the Philips Large Vocabulary Continuous Speech Recognition System

Peter Beyerlein, Meinhard Ullrich, Patricia Wilcox

Philips GmbH Forschungslaboratorien Aachen,
P.O. Box 50 01 45, D-52085 Aachen, Germany

ABSTRACT

The performance of the Philips system for large vocabulary continuous speech recognition has been improved significantly by crossword N-phone modelling, enhanced clustering of HMM-states during training, consistent handling of untrained HMM-states during decoding and a new efficient crossword N-phone M-gram decoding strategy. We report word error rate reductions of up to 18% on various ARPA test sets as compared to our best within-word triphone system, based on Laplacian densities, Viterbi decoding and filterbank-LDA features. The following two issues are addressed:

- Transformation of a tree-organized bigram beam-search decoder into an efficient tree-organized decoder capable of handling long-span acoustic contexts as well as long-span language model contexts.
- State-clustering and generalizing of unseen contexts for the case of Laplacian emission probability density functions.

1 INTRODUCTION

When working on a large vocabulary continuous speech recognition task, proper modelling of frequent word sequences and coarticulation effects is crucial. This calls for integration of long-span context modelling at two levels. At the word level we employ M-gram language models (with $M > 2$). At the phone level accurate long-span acoustic models are constructed by training crossword and within-word context dependent phones. Crossword models cause the number of parameters that have to be estimated during training to become prohibitively large. On top of this there will always be too little data to robustly estimate models for some of the very rare context dependent phones. Thus clustering has to be applied. We investigated two approaches both having the additional advantage of providing an easy way of modelling unseen context dependent phones during recognition. As the handling of crossword context dependent phones during decoding requires the anticipation of all possible successor phones, the resulting complexity poses a severe problem where computing resources are concerned. Our decoding strategy takes care of this problem.

2 CROSSWORD N-PHONE M-GRAM DECODER

2.1 Score Propagation

The decoder is an extension of the *one-pass beam search* algorithm [1] and is able to handle a M-gram language model history together with N-phone crossword models. In the literature crossword N-phone M-gram decoders have already been described in a somewhat informal way [5]. We present a formal framework for such a decoder based on the approach described in [1] .
For a one-pass bigram Viterbi-decoder using a word conditioned lexical tree search the following recursive equations describe the time-synchronous propagation of scores through the search space:
Let $Q_v(t,s,w)$ denote the score (log-likelihood value) of the best Viterbi-path reaching state s of the HMM for word w at time t given the predecessor word v. Then

$$Q_v(t,s,w) = \max_{s'}\{Q_v(t-1,s',w) + \log p(x_t, s|s', w)\},$$

where $p(x_t, s|s', w)$ consists of the likelihood of the observation x_t given state s at time t and the probability for the transition from state s' to state s. The initial value $Q_v(t,0,w)$ for the recursion is

$$Q_v(t,0,w) = \max_{v'}\{Q_{v'}(t-1, S_E(v), v) + \log p(v|v')\},$$

where $S_E(v)$ is the last state of the predecessor word v and word v' in turn is predecessor of word v. The best hypothesis can be identified after adding the language model score $\log p(v|v')$. The language model score cannot be considered any earlier due to the tree organization of the pronunciation lexicon, unless a *language model look-ahead* is applied. For the case of a bigram language model, using above equations guarantees that optimal alignments will be found. Now substituting the language model history $V_1^{M-1} = [v_1, ..., v_{M-1}]$ for the word identity v in $Q_v(t,s,w)$ we immediately obtain the respective score propagation equations for a M-gram language model. Thus by modifying the predecessor word identity in the above equations we can formally construct a M-gram decoder:

$$Q_{[v_1,\ldots,v_{M-1}]}(t,s,w) = \max_{s'}\{Q_{[v_1,\ldots,v_{M-1}]}(t-1,s',w) + \log p(x_t, s|s', w)\}$$

$$Q_{[v_1,...,v_{M-1}]}(t,0,w) = \max_{[v',v_1,...,v_{M-2}]} \{Q_{[v',v_1,...,v_{M-2}]}(t-1, S_E(v_{M-1}), v_{M-1}) + \log p(v_{M-1}|[v', v_1, ..., v_{M-2}])\}.$$

Accurate decoding based on crossword context dependent phone models requires the left and right phonetic contexts of the word w to be taken into account.
The left phonetic context is integrated into above equations by supplementing the language model history with additional *pause words*. In order to indicate a non-coarticulated transition a pause word is inserted in between the respective words. Two consecutive non-pause words now define a crossword transition. Since the predecessor word sequence can contain pause words, it may obviously be longer than the original language model history.
In order to incorporate the right phonetic context into the term Q we have to consider all possible sequences of successor phones $U_1^K = [u_1, ..., u_K]$.
Let $Q_{V_1^L, U_1^K}(t, s, w)$ be the score of the best Viterbi-path, reaching state s of the HMM for word w with the right context (*phone sequence*) $U_1^K = [u_1, ..., u_K]$ at time t given the left (acoustic and language) context (*word sequence*) $V_1^L = [v_1, ..., v_L]$. Note that the scope of the phonetic context may exceed the scope of the language model context and vice versa. Thus the length L of the left context and the length K of the right context have to be chosen as short as possible while at the same time accommodating the crossword N-phone M-gram constraint. For this purpose we use a *context matching function* $m(A|B)$. It equals one if the contexts A and B match (overlap) both, phonetically and with respect to the language model. Otherwise it equals zero. The equations of the score propagation can now be written as:

$$Q_{V_1^L, U_1^K}(t, s, w) = \max_{s'} \{Q_{V_1^L, U_1^K}(t-1, s', w) + \log p(x_t, s|s', w, V_1^L, U_1^K)\}$$

$$Q_{V_1^L, U_1^K}(t, 0, w) = \max_{V'^{L'}_1, v_L, U'^{K'}_1} m(V_1^L, w, U_1^K | V'^{L'}_1, v_L, U'^{K'}_1) \cdot \{Q_{V'^{L'}_1, U'^{K'}_1}(t-1, S_E(v_L), v_L) + \log p(v_L|V'^{L'}_1)\}. \quad (1)$$

Note that the term $p(x_t, s|s', w, V_1^L, U_1^K)$ includes the entire phonetic context w, V_1^L, U_1^K. The term $p(v_L|V'^{L'}_1)$ implies the application of a language model containing pause words. Thus equation (1) suggests to model a priori probabilities for pause transitions. This may be rather important for the decoding of spontaneous and strongly coarticulated speech. Deleting the pause words from the history $V'^{L'}_1$ and using a zero-gram language model probability for the pause word itself is consistent with the conventional language modelling approach.
Starting from equation (1) a tree-organized crossword N-phone M-gram decoder is constructed.

2.2 Reducing Complexity

The decoder is realized with help of an efficient tree-copy scheme. A tree copy for each occuring acoustic/language-model context $\{V_1^L; U_1^K\}$ is required to guarantee optimality during recombination of the search hypotheses (see equation (1)). Thus the number of hypotheses will grow exponentially with time t. Language-model look-ahead [9] and histogram pruning are used to decrease the number of hypotheses. In order to limit the number of tree copies we constrain the number of wordend hypotheses per time instance t.

2.3 Killer Heuristics

We employ a strategy from *game-searching theory* known as *killer heuristics* to further reduce the search space. The optimal state hypothesis $s^*(t)$ at time instance t is likely to be found in the vicinity of the optimal state hypothesis $s^*(t-1)$. Thus computing the optimal score of the states s in the neighborhood $\mathcal{N}(s^*(t-1))$ gives an estimate of the overall optimal score at time t:

$$\hat{Q}^*(t) = \max_{s \in \mathcal{N}(s^*(t-1))} Q(s,t).$$

In the original one-pass beam-search paradigm, pruning is applied after the computation of the optimal score of all active hypotheses at time t. Now, using the estimate $\hat{Q}^*(t)$ of the optimal score, pruning is interwoven with the score propagation at time instance t. The method described reduces the number of state hypotheses and even the computation time by 20% without loss of optimality. A similar strategy is employed in the RWTH-Aachen system [2].

2.4 Wordgraph Constrained Decoding

Operating the decoder in a wordgraph-constrained mode reduces the set of possible hypotheses by orders of magnitude without significant loss of accuracy as compared to the one-pass mode. A lattice is obtained in a preprocessing step with acoustic and language models of low complexity [1]. In [1] bigram-constrained wordgraphs were suggested for the evaluation of more complex acoustic and language models. Using M-gram N-phone crossword models this may lead to a suboptimal decoder. Instead of applying this constraint, here a *N-best algorithm* [10] is applied to thin out the lattice. From this lattice a compact wordgraph is derived by discarding scores, time information and even the predecessor information. This wordgraph is transformed into a finite state network to constrain the set of allowed word sequences. Now dynamically creating tree copies for accurate crossword N-phone M-gram decoding (eq. (1)) is tractable.

2.5 Fast Likelihood Computation

As decoding still has to deal with a large number of hypotheses, the application of fast but approximative likelihood computation techniques ([11],[12]) is of advantage. However the impact of such approximations cannot be forseen when evaluating highly accurate acoustic models.

Yet they can be used in the first pass, if a wordgraph constrained decoding with an accurate likelihood computation will follow the generation of the lattice.

3 TRAINING CONTEXT DEPENDENT PHONES

After estimation of context independent phone models an automatic selection of pronunciation variants takes place. Thus a crossword N-phone script is derived and standard HMM training is performed, yielding a single density for each state. We investigate two different methods for clustering state models, a top-down and a bottom-up approach. Either approach is capable of handling long-span acoustic contexts.

One of the approaches is the recently proposed *generalized bottom-up state-clustering strategy* [8]. In a first pass bottom-up state clustering is carried out. In a second pass the state clusters derived are used to estimate similarities between different contexts. When a state model for an unseen context dependent phone is needed during decoding the state model of the most similar context dependent phone seen during training is substituted for it.

The top-down approach uses *decision-tree* based clustering and generalization to estimate *continuous Laplacian mixture emission probability densities*. In other systems the acoustic modelling is typically associated with Gaussian mixture densities ([3],[4],[6],[7]). Bahl et al.[3] for instance, presented a goodness-of-split criterion for Gaussian densities which aims directly at maximizing the log-likelihood of the training data. Following their work we derived an analogous goodness-of-split criterion G for a decision-tree based clustering of HMM states with Laplacian densities.

We consider the training samples of each state of a monophone HMM as constituting a class. The parameters of a Laplacian density function are estimated from these samples. Regarding this class as the root node of a decision tree, we subsequently split the leaves of the tree by asking binary questions about the phonetic context of the respective samples. We used the set of questions proposed by Odell [5].

The likelihood of the set of samples Y modelled by the decision tree with leaves $\lambda = 1, \ldots, \Lambda$ can be written as

$$\mathcal{P}(Y) \quad = \quad \Pi_{\lambda=1}^{\Lambda} \Pi_{y \in Y_\lambda} \mathcal{P}_\lambda(y),$$

where $\mathcal{P}_\lambda$ is the Laplacian density modelling the set of samples Y_λ associated with leaf λ.

Obviously, $Y = \bigcup_{\lambda=1}^{\Lambda} Y_\lambda$ and $Y_m \cap Y_n = \emptyset$ if $m \neq n$. When splitting a node n into two successor nodes l and r the change of the overall log-likelihood can be determined locally. The gain $G(q,n)$ when applying question q at node n is given by

$$G(q,n) \quad = \quad \log\left((\mathcal{P}_l(Y_l)\mathcal{P}_r(Y_r))/\mathcal{P}_n(Y_n)\right).$$

We model $\mathcal{P}_\lambda$ using D-dimensional Laplacians with diagonal covariance matrices

$$\mathcal{P}_\lambda(y) = \Pi_{d=1}^{D} \frac{1}{2b_{\lambda,d}} \exp\left(-\frac{|y_d - \mu_{\lambda,d}|}{b_{\lambda,d}}\right).$$

The maximum likelihood estimates $\hat{\mu}_{\lambda,d}$ and $\hat{b}_{\lambda,d}$ are found to be the median $\hat{\mu}_{\lambda,d}$ and its average distance $\left(\sum_{y \in Y_\lambda} |y_d - \tilde{\mu}_{\lambda,d}|\right)/N_\lambda$ from the N_λ observations.

Resubstitution of these estimates into the above equations yields:

$$G(q,n) \quad = \quad \sum_{d=1}^{D} \left[N_n \log \frac{\sum_{y \in Y_n} |y_d - \tilde{\mu}_{n,d}|}{N_n} - \right.$$

$$\left. \left(N_l \log \frac{\sum_{y \in Y_l} |y_d - \tilde{\mu}_{l,d}|}{N_l} + N_r \log \frac{\sum_{y \in Y_r} |y_d - \tilde{\mu}_{r,d}|}{N_r} \right) \right],$$

where N_n, N_l and N_r are the number of samples assigned to the parent node n and its successor nodes l and r. For Gaussian densities it is not necessary to compute models directly from the samples of the state clusters as all of the information required is included in the statistics of the associated states. This does not hold for Laplacians. Yet, experiments showed that without loss of recognition accuracy a simpler criterion G^* for splitting a node n into two nodes l and r can be applied:

$$G^*(q,n) \quad = \quad \sum_{d=1}^{D} \left[\sum_{p \in P_n} N_p |\bar{y}_{p,d} - \bar{y}_{n,d}| \right.$$

$$\left. - \left(\sum_{p \in P_l} N_p |\bar{y}_{p,d} - \bar{y}_{l,d}| + \sum_{p \in P_r} N_p |\bar{y}_{p,d} - \bar{y}_{r,d}| \right) \right],$$

where P_n, P_l and P_r are the sets of states represented by the nodes n, l and r; N_p is the number of samples belonging to state p; $\bar{y}_p$ denotes the mean of these samples; $\bar{y}_n$, $\bar{y}_l$ and $\bar{y}_r$ are the means associated with nodes n, l and r. After completion of the clustering process a continuous Laplacian mixture density is estimated for each cluster.

4 RESULTS

We compared our best within-word triphone system (*ww-gbut*) with the new crossword-triphone system, clustering states with either the generalized state-clustering approach (*cw-gbut*) or the decision-tree approach (*cw-dtree*). Results were determined for various ARPA test sets with vocabularies of 5000 and 64000 words. The training of the triphone models was carried out gender dependently on the WSJ0 and the WSJ0+1 corpus. Averaged over all test sets we obtained a relative improvement of 10 %. For the male part of the si_et_h2 set we observed a relative improvement of 18 % (see table 1).

On the NAB'94 task the decision tree approach led to a degradation in performance for the within-word setup. When switching from within-word to crossword models we observed that the decision tree approach clearly outperformed the bottom-up clustering method described in [8] (see table 1, table 2). Thus the inclusion of phonetic knowledge into the recognition system is of advantage when dealing with a large number of crossword context dependent phones as its generalization capabilty is superior to that of the bottom-up approach. When applying decision-trees to pentaphones, we found no significant improvement in the error rate on the WSJ0 setup. This came as no surprise as the amount of training data in this

setup is insufficient for a robust estimation of pentaphone models. Moreover we observed that pentaphone-specific questions had little impact on decision tree construction. In addition to long-span acoustic models, long-span language models, trained on 38 million words of the WSJ corpus, were integrated. The use of a trigram instead of a bigram language model pays off (see table 3), going one step further still yields a small decrease in word error rate. Saturation for the 5000 word task is observed when switching to a pentagram language model.

Table 1: Word error rates (in %) for within-word (ww) and crossword (cw) models using generalized bottom-up tying (gbut) or decision trees (dtree) for a vocabulary of 5000 words (ARPA sets: si_dt_05'92, si_et_05'92, si_dt_05'93, si_et_h2'93) bigram language model, WSJ0 training

	ww-gbut	cw-gbut	cw-dtree
male si_dt_05 92	9.1	8.7	8.3
male si_et_05 92	5.4	5.9	5.0
male si_dt_05 93	11.7	10.5	10.6
male si_et_h2 93	13.5	12.2	11.0
males 92/93	9.9	9.3	8.8 (-11%)
female si_dt_05 92	7.0	6.9	6.3
female si_et_05 92	7.1	6.9	6.3
female si_dt_05 93	10.5	9.7	10.1
female si_et_h2 93	7.6	8.0	7.0
females 92/93	8.4	8.1	7.6 (-10%)
all 92/93	9.2	8.7	8.2 (-11%)

Table 2: Word error rates (in %) for within-word (ww) and crossword (cw) models using generalized bottom-up tying (gbut) or decision trees (dtree) for a vocabulary of 64000 words (male part of evaluation set NAB'94) with bigram and trigram language models, WSJ0+1 training

	ww-gbut	cw-gbut	cw-dtree
bigram	13.2 (0%)	12.9 (-2%)	11.9 (-11%)
trigram	10.4 (-21%)	10.0 (-24%)	9.6 (-27%)

5 CONCLUSION

We presented an efficient method for handling long-span acoustic and long-span language models in our decoder. We found that the use of such long-span models merits the additional effort in a large vocabulary continuous speech recognition system, as we were able to decrease the word error rates by up to 27% on various ARPA test sets. When modelling crossword context dependent phones the decision-tree approach performed better than the generalized bottom-up tying approach. It is interesting to note that our system as opposed to other systems ([3],[4],[6],[7]), uses filterbank-LDA features, Laplacian instead of Gaussian densities and a Viterbi alignment instead of Baum-Welsh training. Nevertheless, proper handling of acoustic contexts and language model contexts leads to a comparable improvement of our system.

Table 3: Word error rates (in %) for a vocabulary of 5000 words (ARPA sets: si_dt_05'92, si_et_05'92, si_dt_05'93, si_et_h2'93) using various N-gram language models, decision tree clustering, WSJ0 training

	females 92/93	males 92/93
ww-bigram	8.4	10.0
ww-trigram	6.8 (-19%)	8.0 (-20%)
ww-tetragram	6.5 (-23%)	8.0 (-20%)
ww-pentagram	6.5 (-23%)	8.0 (-20%)
cw-bigram	7.6 (-10%)	8.8 (-12%)
cw-trigram	6.4 (-24%)	7.5 (-25%)
cw-tetragram	6.1 (-27%)	7.3 (-27%)
cw-pentagram	6.1 (-27%)	7.3 (-27%)

6 ACKNOWLEDGEMENT

Special thanks to Dietrich Klakow for providing the M-gram language models.

7 REFERENCES

1. Ney H., Aubert X., "Dynamic Programming Search Strategies: From Digit Strings To Large Vocabulary Word Graphs", in "Automatic Speech and Speaker Recognition", edited by Lee C.-H., Soong F. K., Paliwal K.K., Kluwer Academic Publishers, Boston, 1996, pp. 385-411
2. Ney H., private communication
3. Bahl L.R., de Souza P.V., Gopalakrishnan P.S., Picheny M.A. : "Context Dependent Vector Quantization for Continuous Speech Recognition", Proc. ICASSP'93, Minneapolis, MN, USA, pp. 632-635
4. Young S.J., Odell J.J., Woodland P.C.: "Tree-based Tying for High Accuracy Acoustic Modelling", Proc. SLT-HLT'95 Workshop, Austin, Texas, USA, pp. 286-291
5. Odell J.J.: "The Use of Context in Large Vocabulary Speech Recognition", Ph.D. thesis, University of Cambridge 1995, England.
6. Hwang M.-Y., Huang X., Alleva F.: "Predicting Unseen Triphones with Senones", Proc. ICASSP'93 II, Minneapolis, MN, USA, pp. 311-314
7. Digalakis V., Weintraub M., Sankar A., Franco H., Neumeyer L., and Murveit H., "Continuous Speech Dictation on ARPA's North American Business News Domain", Proc. SLT-HLT'95 Workshop, Austin, Texas, USA, pp. 88-93
8. Aubert, X., Beyerlein, P., Ullrich, M., "A Bottom-Up Approach for Handling Unseen Triphones in Large Vocabulary Continuous Speech Recognition", Proc. ICSLP'96, Philadelphia, PA, pp. 14-17
9. Steinbiss V., Tran B.-H., "Improvements in Beam Search", Proc. ICSLP'94, pp. 2143-2146
10. Tran B.-H., Seide F., Steinbiss V., "A word graph based N-best search in continuous speech recognition", Proc. ICSLP'96, pp. 2127-2130
11. Boccieri E. "Vector Quantization for the Efficient Computation of Continuous Density Likelihoods", Proc. ICASSP'93, Minneapolis, USA, pp. 692-695
12. Beyerlein P., Ullrich M. "Hamming Distance Approximation for a Fast Log-Likelihood Computation for Mixture Densities", Proc. Eurospeech'95, Madrid, Spain, pp. 1083-1086