Optimized State-Tying for Triphone-Based HMMs under Training Data Deficiency

Michal Borský, Petr Pollák
Czech Technical University in Prague
Faculty of Electrical Engineering
K13131 CTU FEE, Technická 2, 166 27 Prague 6, Czech Republic
Email: {borskmic,pollak}@fel.cvut.cz

***Abstract*—This paper deals with an optimization of state-tying for triphone-based HMM in the case of training data deficiency. The main goal is to analyse the importance of stopping threshold for criterial function in tree-based clustering. The log-likelihood measure was used as the criterial function, when a varying threshold with different sizes of training set was evaluated. Tied-state triphone HMMs with multiple Gaussian mixtures were trained under various setups. Realized experiments showed that the more complex AMs with less mixtures added could achieve better results that less complex models with more mixtures. The same conclusion was proved for even significantly reduced amount of training data.**

***Keywords*—speech recognition, acoustic modelling, tree-based clustering, tied-state HMM,**

I. Introduction

Automated speech recognition (ASR) systems has nowadays increasing number of real-time applications as personal dictation system, on-line automated subtitling, or automatic transcription and indexing of large audio archives. Hidden Markov Models (HMMs) are widely used standard for acoustic modelling (AM) in these systems and particular HMMs are built typically for context-dependent sub-word units (triphones) with multiple Gaussian distributions per model state. Such modelling means very high number of output probability density functions (pdfs) in the AM which put heavy requirements on the amount and quality of training data same as increasing the computational difficulty at the stage of decoding, especially in the case of large vocabulary continuous speech recognition (LVCSR).

A standard solution to these problems is to reduce the number of model parameters with the help of tying. It involves firstly the tying transition matrices but more significant reduction is achieved by a tying the state pdfs throughout models. The current trend in pdf reduction is achieved using two methods: agglomerative [1] or tree-based [2] clustering. The application of one of them is determined by the framework of the ASR system, when tree-based clustering is generally preferred in LVCSR systems, mainly due to the possibility to estimate the parameters of the triphone models which are not present in the training set.

The outcome of tree-based clustering is determined by the setting of two stopping thresholds: the criterial function and the occupation count threshold. Finding the optimal number for each threshold is generally very time-consuming [3]. For this reason, the setting of a stop threshold manually is the most common method used for this task, because the results for manually set thresholds are comparable to the optimized ones [4].

Since the decoding time is proportional directly to the complexity of AM used, the general attitude is to minimize the AM complexity while keeping the decoding accuracy as high as possible at the same time. Consequently, the complexity at the level of AM of any real-life LVCSR system must be optimized, i.e. the total number of Gaussians for all HMM models used should be properly estimated. To find this number two general approaches are used usually.

The first approach is the usage of rather low number of models commonly with an addition of higher number of mixtures in state emitting function. It means the usage of very strict clustering conditions for context-dependent phones or usage of monophone AM without context dependency. As typical example for the transcription of broadcast news [5], the authors used monophone AM with 41 Czech phonemes and 7 different noise models. At the end, up to 100 Gaussians were added into this AM which meant approximately 300 mixtures per model in average. Using such setup, nearly the same results were achieved as for AM based on context-dependent triphones, see [6].

Alternatively, higher number of models can be used with smaller number of mixtures per state mixtures. It means practically the usage of AM based on context-dependent phones with rather relaxed clustering conditions. As an example, for the online TV caption system described in [7] the authors used approximately 5k tied-states with 8 mixtures per state. It yielded to the amount of 40k Gaussians in this AM. In another application (audio archive transcription) [8] AM with 6k states and 107k Gaussians were used (it meant approximately 16 mixtures per state). A similar approach for English language can be seen in [9], with approximately 6k/7k physical states supplemented with 28/16 mixtures per state on average. These numbers are comparable to Czech language, especially when a very similar number of monophones in English language is taken into account [10]. In both cases a rather large amount of training data (400/370 hours) was used.

The main purpose of this article is to analyse the influence of various state-tying conditions on triphone based AM within basic ASR systems. This analysis will be done on the basis of speech recognition accuracy for loop-digit and control command recognizer.

II. Acoustic modelling of triphones

As it was mentioned above, standard acoustic modelling of current ASR systems is based on Hidden Markov Models typically. The most frequently modelled units are context dependent phones, i.e. triphones (cross-word or word-internal respectively) with proper state-tying to decrease the model complexity.

A. Creation of tied-state triphones

As this method is generally well known, it is not necessary to describe it in details which can be found in [11]. We present here just brief summary of a creation of tied-state triphone AM in the following particular steps.

- Initially, monophone AM model were trained using Baum-Welch reestimation.
- Similar states of triphone AM were found and clustered into particular subgroups. Two above mentioned state tying algorithms are used typically. The first one is data-driven and it uses Euclidean distance between the means of Gaussian pdfs as a criterial function. The second one is tree-based clustering which is based on phonetic decision trees. The later is generally preferred since it allows also the modelling of unseen triphones as well.
- At the end, such AM were iteratively reestimated with Gaussian mixture components added until desired number of mixtures is achieved.

B. Strategy of tree-based clustering

Within our AM 43 Czech monophone-set wereused with two additional models for silence and short-pause. It originated from defined SAMPA set for Czech language [10] but several allophones were not used, exactly syllabic consonants `m=`, `l=`, `r=`, same as rather very rare `G`, `@`, `?`, `P\`.

This number of monophones gave a possibility to create 83250 triphones. Using exact training database, only a smaller number of triphones could be always found and appropriately trained.

Concerning state-tying, tree-based clustering algorithm was chosen as our approach in this study. As the evaluation function for tree-based clustering a log-likelihood was used as criterial function. In each case, the situation of too little training data must be avoided, so minimum required occupation count for all states of the HMM set must be set at some exact value. In our case the minimal occupation count for a leaf was set at 100 frames. Around 400 questions were defined, asking only about the immediate left and/or right context.

Five different state-tying thresholds were selected, summarized in the Tab. I and Tab. II

III. Experiments

Described various level of state-tying in triphone AM was analysed in the experiments within two small vocabulary recognition tasks connected-digit recognition and control command recognition. The experimental setup is described in details in the following paragraphs.

TABLE I
SUMMARY FOR FULL TRAINING SET

Name	TB Thresh.	Init. Gauss.	Mixtures	End Gauss.
Tri_360	360	6665	5	33225
Tri_720	720	3893	7	27251
Tri_1800	1800	1926	15	28890
Tri_2800	2800	1410	20	28200
Tri_3800	3800	1138	22	25036

TABLE II
SUMMARY FOR REDUCED TRAINING SET A AND B

Reduced_A			
Name	Init. Gauss.	Mixtures	End Gauss.
Tri_360	3583	7	25081
Tri_720	1905	13	24765
Tri_1800	923	27	24921
Tri_2800	665	40	26600
Tri_3800	555	46	25530
Reduced_B			
Name	Init. Gauss.	Mixtures	End. Gauss.
Tri_360	2435	10	24350
Tri_720	1303	20	26060
Tri_1800	630	40	25200
Tri_2800	465	55	25575
Tri_3800	376	65	24440

A. General setup of recognizer

A rather standard setup for recogniton was used. Concerning *front-end processing*, mel-frequency cepstral coefficients (MFCC) were computed with the following parameters:

- 12 cepstral coefficients with c[0];
- frame length of 30 ms,
- segmentation step of 10 ms,
- pre-emphasis and Hamming window weighting,
- 24 filters in frequency band $100 \div 3600$ Hz,
- static, dynamic, and acceleration features used.

Acoustic model had the following parameters:

- originally 43 different monophones + silence and short-pause,
- standard left-right HMMs with 3 state per group of clustered triphones,
- total number of generalized triphones, same as a the number of mixture added, depended on state-tying level,
- static, dynamic, and acceleration features in 1 stream.

Particular *recognition tasks* were:

- connected digit recognizer - grammar for 10 different words (digits) with unlimited possible loop repetitions in one utterance,
- control command recognizer - 468 different commands without a repetition in one utterance.

Widely used tools from HTK Toolkit [11] were used for the implementation of particular steps of above described ASR system.

B. Training and testing database

The training and testing data originated from SPEECON database.The utterances were selected from rather clean environments (OFFICE and ENTERTAINMENT) recorded by headset microphone which meant signals with rather high SNR.

TABLE III
TRAINING SETS SUMMARY

	Signals	Rich signals	Triphones	Occur
Full	54278	6498	15392	472
Reduced_A	11906	6498	13451	148
Reduced_B	8676	3249	12131	109

The experiments were performed with the maximum training set containing approx. 55k signals from 190 speakers, with and overall length of 51 hours. Concerning the coverage of triphones, this maximum training database contained only 15392 different triphones.

The testing sets were composed from utterances of other 21 speakers, for digit recognition we had at the end 190 utterances with isolated or connected digit sequences of an overall length of 13.5 minutes. For command recognition 370 utterances were used, which contained commands for a device control of an overall length of 15 minutes.

The reduced training sets were composed of the fraction of the original training set, with a special focus on selecting signals containing phonetically rich content. This approach resulted in a very similar number of triphones found, but reduced the amount of training data for each triphone. The details about particular training sets are in the Tab. III

C. ASR performance evaluation

The performance of used ASR systems was analysed on the basis of standard criteria used for speech recognition, i.e.

$$WER = \frac{S + D + I}{N} 100\ [\%]\,, \qquad (1)$$

where N is the total number of words (phrases) and S, D, I represent the number of substituted, deleted, and inserted words respectively.

IV. RESULTS

In the first experiment the full training set was used to train the cross-word triphones. The number of mixtures added depended on the initial number of tied-state triphones, so that the resulting number of Gaussians was roughly the same.

The mixtures were added two at a time and models were reestimated three-times afterwards. Only the best results were plotted for each set of models containing the same number of mixtures.

For both recognition tasks the accuracy of models with more initial states generally outperformed the models with less states. It can be attributed to the fact that the training set contained enough data, which in consequence allowed for a sufficient training of more complex models. On the other hand, the less complex models showed signs of over-training, especially in the digit task. The state-tying TB_720 yielded the best results for both recognition tasks with the full training set.

The same experiment were conducted on reduced training sets. The results for reduced training set A are summarized in Tab. IV, the tab. V contains the results for reduced set B. The overall results of recognition for digit and commands tasks are shown in Fig. 1 and 2.

TABLE IV
RESULTS FOR REDUCED TRAINING SET A

	$WER\ Digits$ [%]			$WER\ Commands$ [%]		
Name	Start	End	Best	Start	End	Best
TB_360	4.17	2.69	2.69	3.51	2.16	2.16
TB_720	4.4	3.5	2.4	2.97	1.89	1.89
TB_1800	3.9	4.17	2.96	4.05	2.97	2.43
TB_2800	4.44	4.04	3.36	5.41	2.7	2.43
TB_3800	5.52	3.77	3.5	5.52	3.77	3.5

TABLE V
RESULTS FOR REDUCED TRAINING SET B

	$WER\ Digits$ [%]			$WER\ Commands$ [%]		
Name	Start	End	Best	Start	End	Best
TB_360	4.98	3.23	3.23	4.32	2.97	2.43
TB_720	4.71	3.5	2.83	3.51	2.97	1.89
TB_1800	6.06	4.85	4.31	4.86	2.43	1.89
TB_2800	7	4.98	4.58	6.22	3.78	2.43
TB_3800	8.08	3.9	3.5	4.86	3.51	2.7

V. CONCLUSIONS

In this paper the difference in recognition accuracy for varying state-tied conditions in triphone based acoustic modelling was analysed. The most important conclusions can be summarized in the following points.

- For the complete training set the initial one mixture models for all state-tying conditions achieved roughly the same results.
- As the number of mixtures increased, the overall *WER* was dropping, but in both cases the *TB_360* and *TB_720* model outperformed the *TB_1800*, *TB_2800* and *TB_3800* models. This can be attributed to the fact that with GMM, the amount of data in training set still allowed for sufficient training of the great number of model parameters.
- The more complex, but still appropriately trained, tied-state models showed better results than a less complex one. For both recognition tasks this happened when models reached around 10k+ Gaussians.
- Results with the reduced training set followed a similar trend. The less strict state-tying showed worse results for initial one mixture models. As

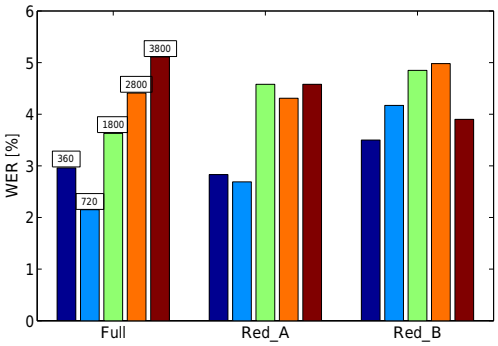

Fig. 1. Final WER for various tied-state conditions and target approx. 20k Gaussians

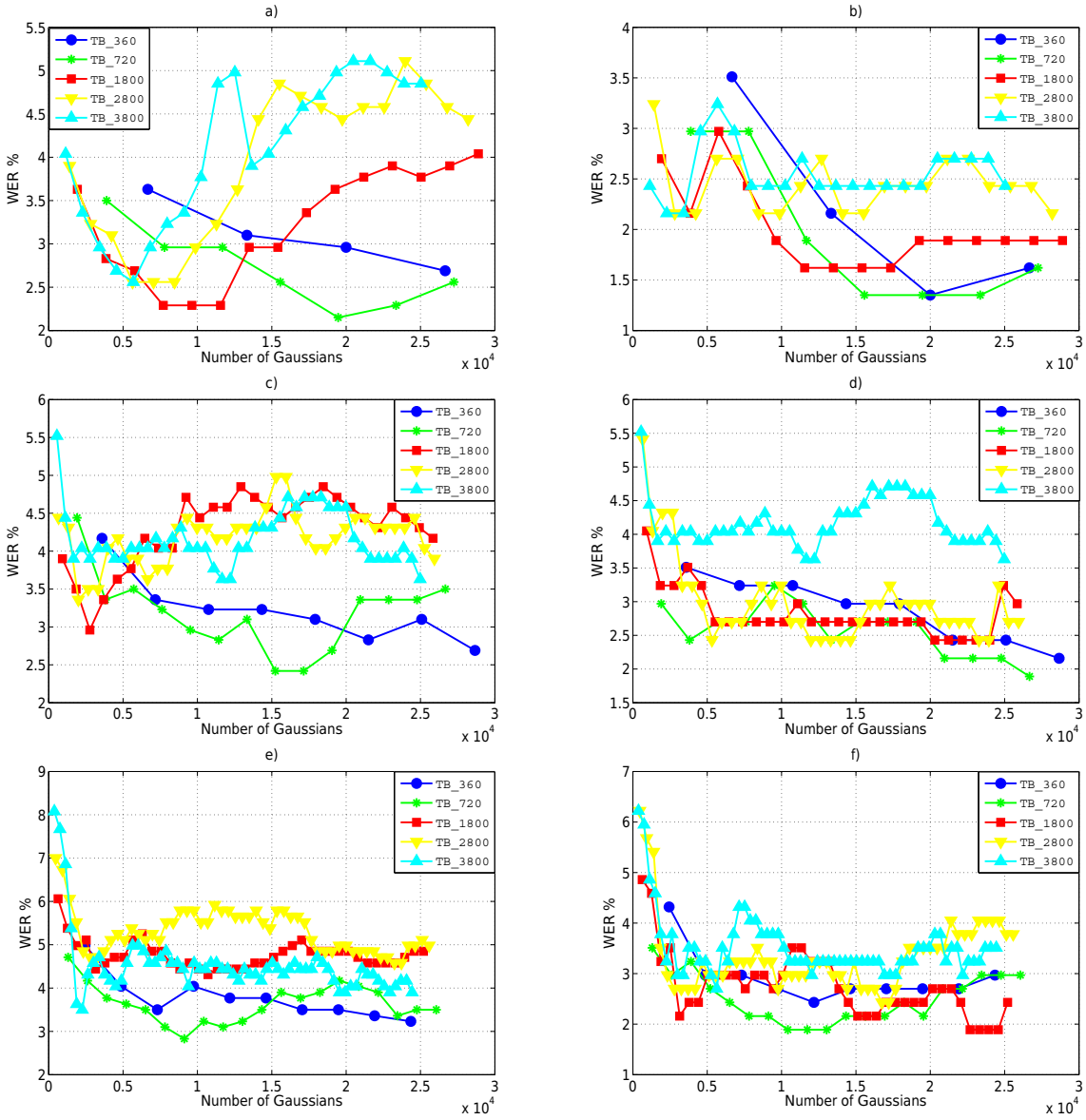

Fig. 2. Dependency of WER for various tied-state condition on number of Gaussians: a) digit recognition full training set - b) command recognition full training set - c) digit recognition reduced training set A - d) command reduced training set A - e) digit recognition reduced training set B - f) command reduced training set B

the number of mixtures increased, the *WER* for *TB_360* and *TB_720* still achieved better results.

Acknowledgements

Research described in the paper was supported by internal CTU grant SGS12/143/OHK3/2T/13 "Algorithms and Hardware Realizations of Digital Signal Processing".

References

[1] S. J. Young and P. C. Woodland, "The use of state tying in continuous speech recognition.," in *EUROSPEECH*, ISCA, 1993.

[2] S. J. Young, J. J. Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modelling," 1994.

[3] K. Shinoda and T. Watanabe, "Mdl-based context-dependent subword modeling for speech recognition," *Acoustical Science and Technology*, vol. 21, no. 2, pp. 79–86, 2000.

[4] H. Nock, M. Gales, and S. Young, "A comparative study of methods for phonetic decision-tree state clustering," in *In: Proceedings European Conference on Speech Communication and Technology*, pp. 111–114, 1997.

[5] J. Nouza, D. Nejedlová, J. Zdánský, and J. Kolorenc, "Very large vocabulary speech recognition system for automatic transcription of Czech broadcast programs," in *INTERSPEECH*, 2004.

[6] J. Nouza, J. Zdansky, P. Cerva, and J. Silovsky, "Challenges in speech processing of slavic languages (case studies in speech recognition of Czech and Slovak)," in *Proceedings of the Second international conference on Development of Multimodal Interfaces: active Listening and Synchrony*, COST'09, (Berlin, Heidelberg), pp. 225–241, Springer-Verlag, 2010.

[7] T. Jan, P. Aleš, L. Zdeněk, and J. Psutka, "Online TV captioning of Czech parliamentary sessions," vol. 6231 of *Lecture Notes in Computer Science*, (Springer Berlin / Heidelberg), pp. 416–422, 2010.

[8] P. Ircing, J. Psutka, and V. Radová, "Automatic transcription of audio archives for spoken document retrieval," (Anaheim), pp. 448–452, ACTA Press, 2006.

[9] X. Liu, M. J. F. Gales, K. C. Sim, and K. Yu, "Investigation of acoustic modeling techniques for lvcsr systems," in *In Proc. ICASSP*, pp. 849–852, 2005.

[10] J. C. Wells, "Sampa home page." `http://www.phon.ucl.ac.uk/home/sampa/home.htm`.

[11] S. Young and et al., *The HTK Book, Version 3.4.1*. Cambridge, 2009.