Enhanced Gaussian Selection in Medium Vocabulary Continuous Speech Recognition

Branislav Popović*, Dragiša Mišković*, Darko Pekar**, Stevan Ostrogonac*, Vlado Delić*
* University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia
** AlfaNum – Speech Technologies, Novi Sad, Serbia
bpopovic@uns.ac.rs, dragisa@uns.ac.rs, darko.pekar@alfanum.co.rs, ostrogonac.stevan@uns.ac.rs, vdelic@uns.ac.rs

***Abstract*—Eigenvalues Driven Gaussian Selection (EDGS) is used in this paper in order to reduce the computational complexity of an acoustic processing module of a medium vocabulary continuous speech recognition system for the Serbian language, based on Hidden Markov Models (HMMs) with the diagonal covariance matrices. The optimal values of five different parameters are discussed: overlap threshold and overlap percentage used for clustering, pruning threshold and pruning percentage used for decoding, as well as newly introduced discard threshold. Significant reduction of computational complexity is obtained, without noticeable degradation in error rate.**

I. Introduction

Gaussian Selection (GS) procedure is used in order to increase the speed of a Continuous Speech Recognition (CSR) system, with an acceptable degradation of the system performance, that consequently occurs as a trade-off. It was originally proposed in [1], stating that the likelihood of HMM state could be efficiently approximated by using only a small number of highly dominant Gaussian components, without significant degradation of the recognition accuracy. The method was later refined and efficiently applied to CSR system, using HMMs with the diagonal covariance matrices [2]. A novel method, addressing the problem of the GS in case of larger overlap between the baseline Gaussian components is proposed in [3]. It incorporates grouping algorithm, implemented as an initial step, before the actual GS clustering procedure. It was further enhanced in [4], by using iterative split and merge algorithms.

Calculation of acoustic state likelihoods contributes considerably to the total computational load of HMM-based recognition systems [2]. HMM states are represented by multiple mixture Gaussian state emitting distributions. Each Gaussian component has to be evaluated separately in order to determine the likelihood of a single state. The idea behind the GS is to generate a set of clusters during the training phase, i.e., to form hyper-Gaussians by clustering the baseline Gaussian components [5]. The Gaussians that are close to each other in terms of the appropriate clustering divergence are clustered into a single group, resulting in a division of the acoustic space into a set of vector quantized regions. Regions are represented by the parameters of their hyper-Gaussians. Each Gaussian component could be assigned to one or more regions, i.e., attached to one or more hyper-Gaussians. In the decoding phase, the Gaussian components associated to clusters with the corresponding hyper-densities, whose distance to the particular input speech frame is above the predefined threshold or percentage, are calculated exactly. The aim is to find the most significant components for calculating the overall state likelihood, based on a given input vector, and at the same time, to assign as few nonessential components as possible [2].

The paper is organized as follows. In Section II, EDGS is described in more details. In Section III, we describe the CSR system and the parameters used for training and testing purposes. The results are also given, confirming considerations from previous sections. Paper concludes with Section IV, providing conclusions.

II. Eigenvalues Driven Gaussian Selection

EDGS represents a variant of the GS procedure, driven by the eigenvalues of the covariance matrices of the baseline Gaussian components [3]. It was proposed in order to deal with the situation when there is a significant overlapping between the baseline Gaussian components. Prior to the execution of the appropriate clustering algorithm, a Gaussian is assigned to a group from a predefined set of groups. The assignment is based on a value aggregated from the eigenvalues of the covariance matrix of the particular Gaussian, using slightly modified Ordered Weighted Averaging (OWA) operators, represented in a form

$$OWA_{\omega}(\lambda_1,\ldots,\lambda_p)=\sum_{j=1}^{p}\omega_j\lambda_{\sigma(j)} \tag{1}$$

where $0 \le \lambda_{\sigma(1)} \le \ldots \le \lambda_{\sigma(\mathrm{p})}$ and the coefficients $\omega \in \mathrm{R}^p$ satisfy the constraints

$$0 \le \omega_j \le 1 \quad , \quad \sum_{j=1}^{p}\omega_j = 1 \tag{2}$$

EDGS combines the most significant eigenvalues of the baseline Gaussian components. The particular Gaussian with the eigenvalues $\lambda = (\lambda_1,\ldots,\lambda_\mathrm{p})$ is assigned to a g-th group, $g \in \{1,\ldots,\mathrm{G}\}$, iff $OWA_\omega(\lambda)$ is in the corresponding predefined interval $[\tau_{\mathrm{ming}}, \tau_{\mathrm{maxg}})$, where $\tau_{\mathrm{maxg}} = \tau_{\mathrm{ming}}+1$. We set the borders of intervals to $\tau(i+1) = c\tau(i)$, where c is a predefined constant.

The second step is the GS clustering. It is an iterative procedure. At each iteration, the particular Gaussian component is assigned to a specified cluster, assuming that the "distance" between the Gaussian and the hyper-Gaussian that corresponds to that cluster is minimal. The parameters of hyper-Gaussians are obtained as maximum likelihood estimates, given in the closed form as functions

of the parameters of the belonging Gaussian components. In our previous research [6], we obtained optimal results by using the one-sided KL divergence as our clustering measure, and the Mahalanobis distance between the observation and the hyper-Gaussian in the decoding stage. The expression for the one-sided KL divergence for any two *d*-dimensional Gaussians exists in the closed form

$$\begin{aligned} a &= \log \frac{|\Sigma_2|}{|\Sigma_1|} \\ b &= Tr(\Sigma_2^{-1}\Sigma_1) \\ c &= (\mu_1 - \mu_2)^T \Sigma_2^{-1} (\mu_1 - \mu_2) \end{aligned} \qquad (3)$$

$$KL(h_1 \,||\, h_2) = (a + b + c - d)/2$$

(μ_1, Σ_1) and (μ_2, Σ_2) are the parameters of the corresponding Gaussians h_1 and h_2. The performance of the EDGS method is assessed in terms of the trade-off between the recognition performance and the reduction in the number of exactly evaluated hyper-Gaussians.

III. Experimental Results

A. System Setup

CSR system, developed for the Serbian language, is used in this paper for the purpose of our experiments [7]. Acoustic and linguistic model, together with a decoding module, constitute a decoder. The decoding module is independent of the acoustic model implementation. It allows the use of different scoring and optimization procedures, without modifications to the other parts of the system. The system is HMM-based, using Gaussian mixture models for representing HMM states.

The decoding module uses a sequence of input feature vectors in conjunction with the search space, in order to generate the recognition output. The decoder is based on a variant of the Viterbi algorithm, known as the token-passing algorithm [8]. The information about the path and the score is stored at the world level. Two types of pruning are supported, i.e. the beam search, where all the tokens whose score is lower than the current maximum, decreased by a predefined threshold, are discarded, as well as pruning by limiting the number of tokens with highest scores. Search space is created by the linguistic model, using the information from pronunciation dictionary and language model. Phonetic transcriptions of words are used for lexical tree creation. Afterwards, they become obsolete. If the full covariance matrices are used, the calculation of acoustic scores (CAS) is the critical part in terms of the computational complexity. Even in the case of the diagonal covariance matrices, the CAS produce a significant portion of the total computational load. The state emitting probability is calculated only for the states that correspond to the active tokens.

Medium-sized vocabulary is used for the purpose of our experiments, with the approximately 1250 words. The system operates on a set of 4792 states, and 30286 Gaussians, represented by the diagonal covariance matrices. The database is windowed using 30 ms Hamming windows, with 20 ms overlap between the adjacent frames. The system uses 32-dimensional feature vectors, containing 15 Mel-frequency cepstral coefficients and normalized energy, in combination with their first order derivatives. Significant improvements are obtained in terms of the trade-off between the speed and the accuracy, by applying the GS procedure, as shown by the experiments.

The values of five different parameters are examined in the paper. In case of the disjoint clustering, the overlap percentage determines the relative number of the "nearest" hyper-Gaussians to which a Gaussian component will be attached during the training GS phase, but only if the "distance" between the component and a hyper-Gaussian is below the minimum "distance" value for the given component and all of hyper-Gaussians, increased by the specified overlap threshold. The percentage of the baseline Gaussians shared between 1, 2, 3 or more hyper-Gaussians, is illustrated in Fig. 1. For 10% overlap between the clusters and overlap threshold set to 0.5, 81.48% of Gaussians are attached to only one hyper-Gaussian, 15.49% of Gaussians are shared between 2 hyper-Gaussians, less than 3% of Gaussians are shared between 3 hyper-Gaussians, and about 0.5% of Gaussians are shared between 4 or more hyper-Gaussians.

The pruning percentage represents the percentage of the hyper-Gaussians with the highest likelihoods for a given input speech frame. The Gaussian components attached to hyper-Gaussians within the specified range, have to be evaluated exactly, assuming that the likelihood of hyper-Gaussian to which they are attached is above the predefined value, determined as the difference between the maximum likelihood value for all hyper-Gaussians, for

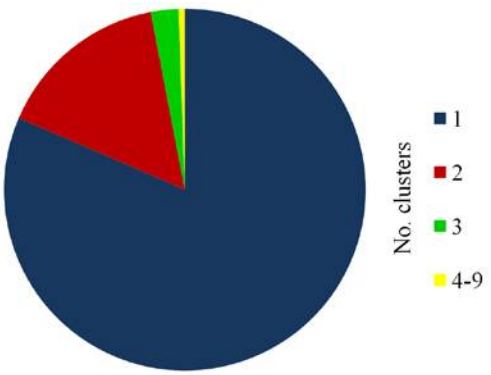

Figure 1. Mixture sharing percentage per number of clusters for 10% overlap and the overlap threshold set to 0.5

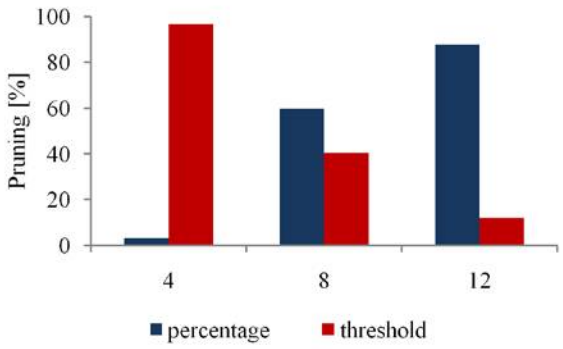

Figure 2. Pruning by threshold or percentage, the pruning percentage set to 10

TABLE I.
DISJOINT CLUSTERING, DISCARD THRESHOLD DEACTIVATED

No.	Pruning percentage	Pruning threshold	Discard threshold	Overlap percentage	Overlap threshold	CSR time gain [%]	CAS time gain [%]	WER	PER
-	-	-	-	-	-	-	-	13.60	3.50
1	20	-	-	-	-	1.97	19.79	13.70	3.60
2	10	-	-	-	-	3.32	33.33	14.00	3.80
3	5	-	-	-	-	3.73	37.50	14.30	4.00
4	20	4	-	-	-	3.01	30.21	15.10	4.10
5	10	4	-	-	-	4.25	42.71	15.60	4.10
6	5	4	-	-	-	3.83	38.54	15.60	4.30
7	20	8	-	-	-	3.01	30.21	13.90	3.70
8	10	8	-	-	-	3.21	32.29	14.10	3.80
9	5	8	-	-	-	3.73	37.50	14.50	4.00
10	20	12	-	-	-	2.49	25.00	13.80	3.70
11	10	12	-	-	-	3.01	30.21	13.90	3.80
12	5	12	-	-	-	3.83	38.54	14.30	4.00

TABLE II.
SMALL OVERLAPPING, DISCARD THRESHOLD DEACTIVATED

No.	Pruning percentage	Pruning threshold	Discard threshold	Overlap percentage	Overlap threshold	CSR time gain [%]	CAS time gain [%]	WER	PER
-	-	-	-	-	-	-	-	13.60	3.50
1	20	-	-	10.0	0.5	1.35	13.54	13.50	3.60
2	10	-	-	10.0	0.5	2.69	27.08	13.70	3.60
3	5	-	-	10.0	0.5	3.11	31.25	14.10	3.90
4	20	4	-	10.0	0.5	2.38	23.96	14.40	3.90
5	10	4	-	10.0	0.5	2.80	28.13	14.60	4.00
6	5	4	-	10.0	0.5	3.11	31.25	14.40	4.00
7	20	8	-	10.0	0.5	1.76	17.71	13.90	3.60
8	10	8	-	10.0	0.5	2.07	20.83	13.80	3.70
9	5	8	-	10.0	0.5	2.69	27.08	14.20	3.90
10	20	12	-	10.0	0.5	1.87	18.75	13.60	3.60
11	10	12	-	10.0	0.5	2.59	26.04	13.70	3.60
12	5	12	-	10.0	0.5	2.49	25.00	14.10	3.90

a given input speech frame, and the pruning threshold. The percentage of hyper-Gaussians pruned by the threshold or the percentage, for the pruning percentage set to 10, and the values of pruning threshold set to {4,8,12} respectively, is presented in Fig. 2.

The value for all the other Gaussian components, attached to hyper-Gaussians outside the specified range, have to be approximated in order to reduce the computational complexity. In case that the likelihood of their hyper-Gaussian is above the difference between the maximum likelihood value for all hyper-Gaussians, for a given input speech frame, and the discard threshold, their likelihood values will be floored with the value determined for the hyper-Gaussian to which they are attached. Otherwise, the Gaussians will be "discarded", and their likelihood will be specified as the likelihood of the first hyper-Gaussian, whose likelihood value is above the specified difference.

In order to catch the order of magnitude of the particular eigenvalue which has a multiplicative nature [3], the vector of thresholds was set to $\tau_{vec} = [0.7\ 1.4\ 2.8]$. We set the borders of the intervals to $\tau(i+1) = c\tau(i)$, c is a constant ($c = 2$ in our case). The approximate number of Gaussians per cluster was set to 200. Larger hyper-Gaussians further reduce the computational complexity, but they also have more significant impact on error rate.

B. Parameter Values

In Table I, the results are presented for the system using the disjoint EDGS clustering vs. the baseline system (the

TABLE III.
DISJOINT CLUSTERING, DISCARD THRESHOLD ACTIVATED

No.	Pruning percentage	Pruning threshold	Discard threshold	Overlap percentage	Overlap threshold	CSR time gain [%]	CAS time gain [%]	WER	PER
-	-	-	-	-	-	-	-	13.60	3.50
1	20	-	16	-	-	3.73	37.50	13.50	3.50
2	10	-	16	-	-	5.08	51.04	13.70	3.70
3	5	-	16	-	-	5.60	56.25	14.20	3.90
4	20	4	16	-	-	5.39	54.17	15.10	4.10
5	10	4	16	-	-	5.80	58.33	15.20	4.10
6	5	4	16	-	-	5.49	55.21	15.40	4.30
7	20	8	16	-	-	4.77	47.92	13.90	3.60
8	10	8	16	-	-	5.08	51.04	13.80	3.70
9	5	8	16	-	-	5.49	55.21	14.30	3.90
10	20	12	16	-	-	3.73	37.50	13.60	3.50
11	10	12	16	-	-	4.77	47.92	13.80	3.70
12	5	12	16	-	-	5.18	52.08	14.20	3.90

TABLE IV.
SMALL OVERLAPPING, DISCARD THRESHOLD ACTIVATED

No.	Pruning percentage	Pruning threshold	Discard threshold	Overlap percentage	Overlap threshold	CSR time gain [%]	CAS time gain [%]	WER	PER
-	-	-	-	-	-	-	-	13.60	3.50
1	20	-	16	10.0	0.5	2.69	27.08	13.50	3.50
2	10	-	16	10.0	0.5	4.56	45.83	13.60	3.60
3	5	-	16	10.0	0.5	4.46	44.79	14.10	3.80
4	20	4	16	10.0	0.5	4.15	41.67	14.40	3.90
5	10	4	16	10.0	0.5	4.77	47.92	14.60	4.00
6	5	4	16	10.0	0.5	4.35	43.75	14.40	4.00
7	20	8	16	10.0	0.5	3.94	39.58	13.80	3.60
8	10	8	16	10.0	0.5	4.46	44.79	13.80	3.60
9	5	8	16	10.0	0.5	4.15	41.67	14.20	3.90
10	20	12	16	10.0	0.5	3.52	35.42	13.50	3.50
11	10	12	16	10.0	0.5	4.46	44.79	13.60	3.60
12	5	12	16	10.0	0.5	4.66	46.88	14.00	3.80

system working directly with the baseline Gaussians, i.e. without the Gaussian selection procedure). The results are given for the combinations of values for the pruning percentage set to {20,10,5}, and the pruning threshold set to {4,8,12} respectively, or without the pruning threshold, i.e., when only the pruning percentage is used. For the pruning percentage set to 10, and the pruning threshold set to 8, about 60% of Gaussians are pruned by the percentage, and another 40% are pruned by the threshold. Computational time needed in order to calculate the acoustic score was decreased by 30%, without significant degradation of word (WER) and phoneme (PER) error rate (less than 0.5 in both cases).

In Table II, the results are given by using the same pruning settings, with only small overlapping between the clutters. Our previous research showed that small overlapping between clusters provides favorable results in comparison to the disjoint clustering or larger overlapping case [6]. Therefore, we used 10% overlapping, and the overlap threshold set to 0.5. The values are determined intuitively, in order to get about 80% of Gaussians attached to only one hyper-Gaussian, and another 10 to 20% shared between 2 "nearest" clusters, as shown in Fig. 1. Similar computational gain of about 30% is obtained by using the lower pruning percentage and threshold values. However, we also obtained better accuracy.

In Table III, we used previously described settings, given in Table I, but in this case we also used the discard threshold. For a given value of the discard threshold, more than 50% of Gaussian components, that were selected to

be floored, will be "discarded". The values of "far away" hyper-Gaussians will be floored by using the greater likelihood value determined for the first hyper-Gaussian, whose likelihood is above the specified difference, instead of smaller likelihood value determined for the hyper-Gaussian to which they are attached. Therefore, they have more chance to be selected in the decoding phase, for a given input speech frame. We obtained larger speed gain of about 55% and slightly better accuracy.

In Table IV, we used the settings given for Table II, i.e., small overlapping between the clusters, but we also used the discard threshold. Better results are obtained in terms of the accuracy in comparison to the results given in Table III, i.e., the disjoint case. In terms of both, the speed and the recognition performance, better results are obtained in comparison to the results given in Table II. We obtain the computational gain of about 45%, followed by the increase of WER and PER by no more than 0.2, as a trade-off between speed and accuracy.

IV. Conclusion

Significant reduction in the computational complexity of acoustic scores calculations is obtained by using the appropriate values of five different parameters, examined in this paper. In terms of trade-off between speed and accuracy, the optimal results were obtained by calculating no more 10% of hyper-Gaussians, and for the values of pruning threshold that provide close, but not equal pruning border, to the one obtained by using the above mentioned pruning percentage. Additional improvements in terms of the accuracy were obtained by introducing small overlapping between clusters, in combination with the appropriate overlap threshold. Another discard threshold was also introduced, providing optimal results.

Acknowledgment

The work described in this paper was supported in part by the Ministry of Education, Science and Technological Development of the Republic of Serbia, within the project TR32035: “Development of Dialogue Systems for Serbian and Other South Slavic Languages”.

References

[1] E. Bocchieri, “Vector quantization for efficient computation of continuous density likelihoods,” in *Proc. ICASSP*, 2:II-692-II-695, Minneapolis, MN, 1993.

[2] K. M. Knill, M. J. F. Gales, S. J. Young, “Use of Gaussian selection in large vocabulary continuous speech recognition using HMMs,” in *Proc. ICSLP*, vol. 1, pp. 470–473, 1996.

[3] M. Janev, D. Pekar, N. Jakovljević, V. Delić, “Eigenvalues driven Gaussian selection in continuous speech recognition using HMM’s with full covariance matrices,” in *Appl. Intel.*, vol. 33, no. 2, pp. 107–116, 2010.

[4] B. Popović, M. Janev, D. Pekar, N. Jakovljević, M. Gnjatović, M. Sečujski, V. Delić, “A novel split-and-merge algorithm for hierarchical clustering of Gaussian mixture models,” in *Appl. Intel.*, vol. 37, no. 3, pp. 377-389, 2012.

[5] B. Popović, M. Janev, V. Delić, “Gaussian Selection Algorithm in Continuous Speech Recognition,” in *Proc. TELFOR 2012*, pp. 705-712, Belgrade, Serbia, 2012.

[6] D. Pekar, M. Janev, N. Jakovljević, B. Popović, V. Delić, “Improving the performance of Gaussian selection algorithm,” in *Proc. SPECOM 2011*, pp.89–95, Kazan, Russia, 2011.

[7] N. Jakovljević, D. Mišković, M. Janev, D. Pekar, “A Decoder for Large Vocabulary Speech Recognition,” in *Proc. IWSSIP 2011*, pp.1–4, Sarajevo, Bosnia and Herzegovina, 2011.

[8] S. J. Young, N. H. Russell, J. H. S. Thornton, “Token passing: a simple conceptual model for connected speech recognition,” Cambridge University Engineering Department, Cambridge, UK, Tech. Rep. CUED/FINFENG/TR-38, July 1989.