Using Genetic Algorithms for Rapid Speaker Adaptation

Fabrice Lauri, Irina Illina, Dominique Fohr, Filipp Korkmazsky

Speech Group, LORIA - INRIA
B.P. 239 - 54506 Vandœuvre-lès-Nancy - FRANCE
{lauri,illina,fohr,korkmazs}@loria.fr

Abstract

This paper proposes two new approaches to rapid speaker adaptation of acoustic models by using genetic algorithms. Whereas conventional speaker adaptation techniques yield adapted models which represent local optimum solutions, genetic algorithms are capable to provide multiple optimal solutions, thereby delivering potentially more robust adapted models. We have investigated two different strategies of application of the genetic algorithm in the framework of speaker adaptation of acoustic models. The first approach (GA) consists in using a genetic algorithm to adapt the set of Gaussian means to a new speaker. The second approach ($GA + EV$) uses the genetic algorithm to enrich the set of speaker-dependant systems employed by the EigenVoices. Experiments with the *Resource Management* corpus show that, with one adaptation utterance, GA can improve the performances of a speaker-independant system as efficiently as EigenVoices. The method $GA + EV$ outperforms EigenVoices.

1. Introduction

Reducing acoustic mismatches due to speaker variability between the training conditions and the testing conditions is a major problem in automatic speech recognition. This problem is particularly difficult for rapid adaptation, when the available amount of adaptation data is small.

Among the speaker adaptation techniques which tackle this problem efficiently, EigenVoices [5], [6], [4] and methods combining *MLLR* and EigenVoices [7], [1], [2], [11] have shown to rapidly adapt to a new speaker the Gaussian means of the speaker-independant system (SIS).

EigenVoices can improve the performances of an ASRS even if only one adaptation utterance has been used. This outstanding result can be explained by the fact that EigenVoices employs *a priori* information about the inter-speaker variations, by using several well-trained speaker-dependant systems. *A priori* information enables EigenVoices to estimate much less parameters than *MLLR*.

In [7] a structural version of EigenVoices (SEV) is proposed to push back the early saturation encountered by the regular version of EigenVoices. Four different methods combining the concepts of both *Structural MLLR* and EigenVoices-based techniques (EV or SEV) are also presented. For a supervised batch adaptation, the four methods outperforms both $SMLLR$ and EV whatever the available amount of adaptation data.

The scheme presented in [1] extends the standard EigenVoices technique to large-vocabulary continuous speech recognition by training the acoustic models of each training speaker from SI models with the help of *MLLR* and *MAP*. In [2], the eigenspace representing the inter-speaker variations is built using *Principal Component Analysis (PCA)* from the parameters of the *MLLR* regression matrices obtained for each training speaker. The regression matrices computed for the adapted models of the new speaker are then constrained to be located in the space spanned by the first K eigen-matrices. This method thus solves the problem of huge memory requirements of the EigenVoices technique. Indeed the number of parameters of the regression matrices is much smaller than the parameters of a speaker-independant system. In [11], the authors propose three approaches which combine *MLLR* and EigenVoices adaptation. The Approach B exposed in [11] gives similar results to EigenVoices technique but requires far less online memory and computation load. In this approach, a new fast algorithm for maximum-likelihood coefficient estimation is used and the selection of the eigenspace includes SI-model information.

All of the adaptation techniques of acoustic models solve a numerical optimization problem. Such techniques try to estimate the best parameters of the acoustic models by maximizing a function of gain, the *log likelihood*. Yet all of the preceeding quoted methods are suboptimal in the sense that, because they are based on the E-M procedure to estimate the parameters of the acoustic models, they can only find a local optimum solution.

In the current study we propose to use genetic algorithms [8] in the framework of rapid speaker adaptation of acoustic models. Many reasons motivated the use of this family of algorithms.

Such algorithms can theoretically provide a global optimum solution, by exploring a population of solutions. Besides, genetic algorithms can estimate directly the parameters of the acoustic models without using some adaptation transformations (like linear regressions in *MLLR*). Thus no *a priori* constraint is assumed on the transformations which are applied to the parameters of the HMMs, so that a finest adaptation may be obtained.

The remainder of this paper is organized as follows. Section 2 presents the general principles of genetic algorithms. Section 3 gives the characteristics of the genetic algorithm we used for speaker adaptation. The regular version of EigenVoices is explained in section 4, and section 5 presents the new scheme which combine a genetic algorithm with EigenVoices. Section 6 evaluates both proposed methods using data from the *Resource Management (RM)* corpus. Section 7 discusses on the main drawbacks of techniques based on a genetic algorithm. Finally, concluding remarks and future research issues are given in Section 8.

2. Genetic Algorithms

Genetic algorithms are methods for solving numerical optimization problems. As most optimization techniques, genetic algorithms look for the best solution in a search space by maximizing a function of gain. The search in the space of solutions is yet inspired from the natural selection of Darwin, which associates the diversity begetted by chance and the surviving of the most fitted individuals.

Typically, genetic algorithms start from an initial population of solutions (*individuals*) and try to obtain after N_{IT} iterations a population which contains better solutions than the initial population. In the terminology of genetic algorithms, a solution is represented by a *chromosome* which is a vector consisting of *genes*. A gene is one of the parameters to estimate to solve the optimization problem. Each solution s is characterized by a *fitness function* $f(s)$ which represents its quality of adequation to the considered problem.

To create a new population of solutions, standard genetic algorithms use three genetic operators at each iteration : the operator of *reproduction*, the operator of *mutation* and the operator of *selection*. The reproduction operator can be seen as a way to provide an exchange of information eventually relevant between solutions. Once all of the children have been generated with the reproduction operator, they can be subjected to some mutations. The idea behind mutation holds in the introduction of some variations into the population. Finally, the selection operator enables the most fitted individuals of the current population to survive and thus to be able to perpetuate their genetic material if they are selected to belong to the population of the next iteration.

The characteristics of the genetic algorithm we used for speaker adaptation are more precisely defined in the next section.

3. Genetic Algorithms for Speaker Adaptation

In our case, a solution is a (super)vector consisting of all of the Gaussian mean vectors of all of the models of a speech recognition system; a gene is a Gaussian mean vector. A genetic population is represented by all such solution s. The *fitness function* $f(s)$ of a solution s is defined by :

$$f(s) = \frac{exp\left(\frac{log\ p(O/M_s)}{T}\right)}{exp\left(\sum_s \frac{log\ p(O/M_s)}{T}\right)}$$

where O, M_s and T represent respectively the adaptation data, the acoustic models of the solution s and the number of frames of the adaptation data.

The initial population is commonly made up of the supervectors extracted from the speaker-dependant systems and from the speaker-independant system.

3.1. Reproduction

This operator consists in (1) selecting among the individuals of the current population pairs of parents and (2) merging each pair of parents to generate two offsprings.

To be a member of a pair of parents, an individual is selected with a probability proportional to its fitness function. The higher the value of its fitness function, the more likely the corresponding individual will be selected as a parent. Of course, the parents of a pair must be different. Let N_I be the number of individuals in the current population, then $N_I/2$ pairs of parents will be defined in this step.

Once all of the $N_I/2$ pairs of parents have been defined, the parents of each pair are merged to generate the offsprings. The merging of two parents consists in swapping (*crossing-over step*) and combining (*interpolation step*) groups of genes to generate two offsprings. For example, if two parents p_1 and p_2 are represented by vectors containing 3 genes :

$$p_1 = (\ a_1, a_2, a_3\ )$$

and

$$p_2 = (\ b_1, b_2, b_3\ )$$

then crossing the chromosomes after the second gene and defining the interpolation factor as $i_f \in [0;1]$ would produce two offsprings o_1 and o_2 :

$$o_1 = (\ a_1 * i_f + b_1 * (1 - i_f), a_2 * i_f + b_2 * (1 - i_f), \\ b_3 * i_f + a_3 * (1 - i_f)\ )$$

and

$$o_2 = (\ b_1 * i_f + a_1 * (1 - i_f), b_2 * i_f + a_2 * (1 - i_f), \\ a_3 * i_f + b_3 * (1 - i_f)\ )$$

The number N_{CP} of crossing points (in our case $N_{CP} = 1$) and the interpolation factor i_f are parameters of the algorithm and remain unchanged for all iterations. The position of each crossing point is randomly generated for each pair of parents.

3.2. Mutation

Let p_m be the probability of mutation of a gene, μ_g the mean of the gaussian (*gene*) g and σ_g the variance related to the gaussian g in the speaker-independant system. Then mutation consists in generating a random number $r \in [0;1[$ for each gene g of each children's chromosome and modifying this gene g if $r < p_m$. In this case, the new value $\hat{\mu}_g$ of the gene g is $\hat{\mu}_g = \mu_g + s * \sigma_g$ where s is a random number generated within the range $[-\gamma_m; \gamma_m]$. γ_m is the coefficient of mutation. It represents the degree of conservation of a gene : the higher γ_m, the more radically a gene may be altered by mutations and inversely. p_m and γ_m are parameters of the algorithm.

3.3. Selection

The N_I parents of the next population are selected as follows. The N_E best individuals in the current total population (parents + generated children) are first selected to belong to the next population according to their fitness function. N_E is another parameter of the algorithm. The higher N_E, the more parents are likely to be chosen to be a part of the next population. The $N_I - N_E$ best generated children are then selected to be members of the next population.

4. EigenVoices

EigenVoices (EV) technique constrains the adapted models to be located in a dimensionality reduced speaker-space. The speaker space reduced in dimension is obtained by applying a dimensionality reduction technique [1] to a set of T supervectors of dimension D extracted from T well-trained speaker-dependant (SD) models. A supervector μ is made up of the

---

[1]Principal Component Analysis (PCA) for instance

parameters of the acoustic models that have to be adapted. Typically, it consists of the concatenation of all of the Gaussian mean vectors of all of the models of a speaker-dependant system, if only Gaussian means need to be adapted. Thus :

$$\mu = (\mu_1, \mu_2, \cdots, \mu_i, \cdots, \mu_N)$$

where N is the total number of gaussians of a speaker-dependant system.

This offline step yields T supervectors of dimension D, called the eigenvectors. To build the reduced speaker-space, only the K first eigenvectors $\{e_1, e_2, \cdots, e_K\}$ with $K < T << D$ are kept. Related to an origin e_0 [2], these K eigenvoices, which capture most of the variation of the training data, span the reduced speaker-space of dimension K.

A new speaker is then located in the reduced speaker-space by a vector of $K + 1$ weights $\{w_0, w_1, \cdots, w_K\}$.

The supervector $\hat{\mu}$ of the adapted models is then obtained using the equation $\hat{\mu} = \sum_{k=0}^{K} w_k \ e_k$. The $K + 1$ weights are generally estimated using *Maximum Likelihood Eigen-Decomposition (MLED)* [9] to maximize the likelihood of the adaptation data. The other HMM parameters are obtained from the SI-model parameters.

5. Combining GA with EV

This approach consists in, first, using the genetic algorithm to get a final population of N_I potential systems adapted to the new speaker. Among these N_I systems, the N_S best systems are selected to be included into the set of the T SD systems used by the regular version of EigenVoices. The EigenVoices technique is then applied (as explained in the previous section) to the speaker-independant system using an initial speaker space of $T + N_S$ systems.

We assume that the inclusion of some systems adapted to the new speaker into the initial speaker space of T systems will make it closer to the new speaker. Hence the estimation of the weights by the EigenVoices will be more robust.

6. Experimental Evaluation

6.1. Experimental Conditions

EigenVoices and the speaker adaptation techniques based on genetic algorithms have been implemented into the automatic speech recognition system ESPERE [3] [3] and evaluated on the *Resource Management (RM)* corpus.

The speech signals in *RM* are sampled at 16 kHz and were parameterized into the 11 MFCCs $C1$ to $C11$ and the 12 first and second order time derivatives of $C0$ to $C11$, yielding a 35-dimensional feature vector.

The speaker-independant training set of RM1 was used to train the acoustic models of both the speaker-independant system and the speaker-dependant systems. This set groups together 23 female and 49 male american native speakers. Each speaker pronounced 40 training utterances, for a total of 2880 utterances. The acoustic models of the speaker-independant system were trained by performing 20 iterations of the Baum-Welch algorithm ; each speaker-dependant system was trained by adapting the speaker-independant system using 10 iterations of *Structural Maximum A Posteriori (SMAP)* [10]. We used the

---

[2] e_0 can be the average supervector of all of the SD models or the supervector extracted from the SI models.

[3] ESPERE is a first order HMM-based speech recognition toolbox developed at LORIA.

speech data from 16 speakers (7 female and 9 male speakers) of the speaker-dependant set RM2 for the adaptation phase and the recognition phase. Each speaker uttered 600 training sentences used for the adaptation phase only. For the recognition phase, a total of 1280 utterances were tested : 120 sentences per speaker for four of them and 100 utterances per speaker for eight of them.

The acoustic units in the speaker-independant system and in each speaker-dependant system are represented by 45 HMMs with 3 states and a HMM with one state to handle silence and short pause. The probability density function of each state is modelled by a mixture of 8 gaussians. Speech recognition experiments were conducted by using the regular *word-pair* grammar of *RM*.

EigenVoices was parameterized to estimate 31 weights. 30 weights are related to the 30 first eigenvectors and one weight is associated to the supervector s_{SIS} extracted from the SIS. The supervector s_{SIS} is used as the origin of the reduced speaker space.

The initial population of the genetic algorithm is made up of the 72 speaker-dependant systems and the speaker-independant system. The genetic algorithm was parameterized with $N_{IT} = 20$, $N_{CP} = 1$, $i_f = 0.2$, $p_m = 0.0001$, $\gamma_m = 0.01$ and $N_E = 73$. This parameterization seemed to provide the best results.

6.2. Experimental Results

The subsequent results represents the average word accuracy (*WA*) for sixteen test speakers, by taking a confidence interval of $\pm 1\%$, with a risk of 5%. The average *WA* of the speaker-dependant systems is of 94.1%; the *WA* of the speaker-independant system is of 83.8% [4].

The table 1 presents the results of the two proposed schemes GA and $GA + EV$ compared to the EigenVoices, for a supervised batch adaptation with one adaptation utterance.

Baseline	83.8 %
EV	84.3 %
GA	84.3 %
GA+EV ($N_S = 2$)	84.5 %
GA+EV ($N_S = 5$)	84.6 %
GA+EV ($N_S = 10$)	**84.7 %**

Table 1: Comparison of the proposed genetic algorithm based approaches with EigenVoices for one adaptation utterance

EV and GA give the same improvement of performances of the speaker-independant system. Although the genetic algorithm had to estimate a huge number of parameters (about 38000 coefficients) with a small number of adaptation data (about 500 frames), it was capable to find good solutions.

The versatility of the genetic algorithms is emphasized by the results of $GA + EV$. Indeed $GA + EV$ outperforms Eigen-Voices. We explain this result by the fact that this method is able to deliver some new acoustic models which can refine the initial speaker space. The new initial speaker space which is

---

[4] We obtained 87.3% in *WA* with a speaker-independant system using a mixture of 32 gaussians per state, but the results for GA and $GA + EV$ were not wholly available.

used by EigenVoices to build the reduced speaker space is then located closer to the new speaker. The estimation of the weights is, hence, carried out more accurately.

Further experiments with genetic algorithm are also in progress in unsupervised and incremental mode with several adaptation utterances and higher values of N_S. We hope that they will show further improvement in recognition performance.

7. Discussion

The improving of the recognition performances using GA or $GA + EV$ goes along with an increase of the computation load and of the memory needs required by such techniques based on a genetic algorithm. These main drawbacks can be explained by the fact that, unlike EigenVoices, which estimates one solution in T iterations, a technique based on a genetic algorithm estimates several solutions in N iterations, with $N > T$ generally and keeping in mind that one iteration in GA is much longer than one iteration in EV. For instance, in our experiments, EigenVoices was about 6 times faster than GA.

8. Conclusions

We have proposed in this paper two approaches based on a genetic algorithm for speaker adaptation of acoustic models in supervised batch mode. It has been shown experimentally that the GA technique which uses a genetic algorithm to estimate the Gaussian means of a speaker-independant system improves its performances as well as EigenVoices. Moreover, the scheme $GA + EV$ outperforms EigenVoices by providing to EigenVoices a speaker space that is located closer to the new speaker. This implies that the estimation of the weights by EigenVoices can be carried out more precisely.

Our future work will be focused on the estimation of the weights with the help of a genetic algorithm. The weights in EigenVoices are currently carried out by the *Maximum Likelihood Eigen-Decomposition* procedure [5] and represent a linear combination of acoustic models. The simplicity and versatility of the genetic algorithms can then be used to estimate weights which represent a *polynomial combination of acoustic models*. We anticipate that such a polynomial combination would produce adapted models which will be more accurate than adapted models built from a linear combination of acoustic models. Besides, such a technique would require less memory and a reduced computation load compared to GA and $GA + EV$.

9. References

[1] H. Botterweck. Very Fast Adaptation for Large Vocabulary Continuous Speech Recognition using Eigenvoices. *ICSLP'2000*, pages 354–357, 2000.

[2] K.-T. Chen, W.-W. Liau, H.-M. Wang, and L.-S. Lee. Fast Speaker Adaptation using Eigenspace-based Maximum Likelihood Linear Regression. *ICSLP'2000*, pages 742–745, 2000.

[3] D. Fohr, O. Mella, and C. Antoine. The automatic speech recognition engine ESPERE : experiments on telephone speech. *ICSLP, Pkin, Chine*, pages 246–249, 2000.

[4] R. Kuhn, J.-C. Junqua, P. Nguyen, and N. Niedzielski. Rapid Speaker Adaptation in Eigenvoice Space. *IEEE Trans. Speech Audio Proc.*, 8(6):695–707, 2000.

[5] R. Kuhn, P. Nguyen, J.-C. Junqua, and al. Eigenvoices for Speaker Adaptation. *ICSLP'1998*, 1998.

[6] R. Kuhn, P. Nguyen, J.-C. Junqua, R. Boman, N. Niedzielski, S. Fincke, K. Field, and M. Contolini. Fast Speaker Adaptation using A Priori Knowledge. *ICASSP'1999*, pages 1587–1590, 1999.

[7] F. Lauri, I. Illina, and D. Fohr. Combining Eigenvoices and Structural MLLR for Speaker Adaptation. *ICASSP'2003*, 2003.

[8] Zbigniew Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, 1996.

[9] P. Nguyen. Fast speaker adaptation. Technical report, Speech Technology Laboratory, 1998.

[10] K. Shinoda and C.-H. Lee. A structural bayes approach to speaker adaptation. *IEEE Transactions on Speech and Audio Processing*, 9(3):276–287, 2001.

[11] N.J.-C. Wang, S. S.-M. Lee, F. Seide, and L.-H. Lee. Rapid Speaker Adaptation using A Priori Knowledge by Eigenspace Analysis of MLLR Parameters. *ICASSP'2001*, 2001.

---

[5]This procedure is based on the E-M algorithm and hence can only find a local solution.