RAPID SPEAKER ADAPTATION USING MAXIMUM LIKELIHOOD NEURAL REGRESSION

Mohamad Hasan Bahari, Hugo Van hamme

Center for processing speech and images (PSI), Department of Electrical Engineering (ESAT),
Katholieke Universiteit Leuven, Belgium
MohamadHasan.Bahari@esat.kuleuven.be, hugo.vanhamme@esat.kuleuven.be

ABSTRACT

In this paper, a new method called Maximum Likelihood Neural Regression (MLNR) is introduced for Rapid Speaker Adaptation (RSA). MLNR, which is conceptually simple, adapts the Gaussian means of a speaker independent (SI) model to the data of a new speaker by assuming a non-linear mapping from the SI Gaussian means to the adapted Gaussian means. It performs a non-linear regression between maximum likelihood (ML) estimates of the means and the speaker independent means using General Regression Neural Networks (GRNN). Evaluation on the Wall Street Journal benchmark shows that the suggested scheme outperforms different conventional approaches.

Index Terms— rapid speaker adaptation, maximum likelihood, General regression neural networks, non-linear speaker adaptation

1. INTRODUCTION

In recent years, there has been a surge of interest in RSA due to its wide range of applications and its commercial importance [1]. Although different methods have been introduced for RSA, it is still a very challenging task. Lack of adaptation data may make conventional speaker adaptation (SA) approaches such as Maximum-a-Posteriori (MAP) [2] or Maximum Likelihood Linear Regression (MLLR) [3] inapplicable [1]. In MAP, each Gaussian is adapted based on data that can be assumed to be generated by it and consequently, MAP needs a large amount of adaptation data to influence the model effectively while very limited speech data is available in RSA. In MLLR (and its constrained and unconstrained variants), adaptation data is applied to estimate transformation matrices which map the means and covariances of the SI model to the means and covariances of the speaker adapted (SA) model. This linear affine model approximates the required non-linear mapping from SI to SA model, which limits the asymptotic performance as more adaptation data become available, even when working with multiple transforms, each valid in a subpart of the acoustic space. On the other side of the spectrum the elements of the aforementioned matrices are poorly estimated due to shortage of adaptation data in RSA. Consequently, we will have a poor mapping, which leads to an inaccurate adapted model [4]. Therefore, it is too risky to use conventional MLLR for RSA.

Researchers have put forward many suggestions to address these problems [4-9]. For example, Maximum-a-Posteriori Linear Regression (MAPLR), which is a Bayesian counter-part of MLLR, calculates the transformation matrices in a MAP setting. Therefore, it is more conservative than MLLR [9]. To insure the robustness of MLLR, Ding has developed a hybrid architecture of fuzzy logic and MLLR (FLC-MLLR) [5]. Furthermore, this method exploits different advantages of MLLR when there is enough training data. Maximum Penalized Likelihood Linear Regression (MPLLR) has been introduced recently for RSA [4]. This method, which shares many similarities with MAPLR, suggests an analytic solution for finding the transformation matrix and applies a regulation method to avoid over fitting. Although MAPLR, FLC-MLLR and MPLLR are more conservative than MLLR, none of them can capture the non-linear characteristics of the data since they use linear regressions. Maximum Penalized Likelihood Kernel Regression (MPLKR) represents a non-linear regression method for RSA [4]. A noticeable advantage of MPLKR is that the transformation matrix is calculated analytically by solving a linear system [4]. It is shown that the MPLKR outperforms different variants of MLLR such as MAPLR. However, it cannot be used in many real-time applications due to its large computational complexity. In fact, many other non-linear speaker adaptation algorithms

This work is supported by the European Commission as a Marie-Curie ITN-project (FP7-PEOPLE-ITN-2008), namely Bayesian Biometrics for Forensics (BBfor2), under Grant Agreement number 238803.

such as Kernel Ridge Regression (KRR) [6] suffer from the same drawback.

In this paper, a novel fast and reliable speaker adaptation method is developed based on General Regression Neural Networks (GRNN). This method, which is called Maximum Likelihood Neural Regression (MLNR), enjoys several advantages. First, in MLNR, unlike almost all other speaker adaptation methods, no optimization task or calculation of unknown parameters is needed. These characteristics make it very suitable for RSA since poor estimation of unknown parameters, which happens in many adaptation methods such as MLLR, is avoided completely. Second, MLNR uses a non-linear regression making it capable of capturing possible nonlinearities in the data. Finally, the computational complexity of MLNR is significantly lower than that of other non-linear speaker adaptation methods such as MPLKR and KRR.

The rest of this paper is organized as follows. In section 2 two conventional methods working based on ML criterion are described. In section 3, the MLNR is elaborated. Section 4 presents the experimental results and a discussion. The paper ends with conclusion in section 5**.**

2. CONVENTIONAL METHODS BASED ON THE ML CRITERION

Consider a Speaker Independent Hidden Markov Model (SI-HMM) with the following state probability density function (*pdf*).

$$f(o_t) = \sum_{i=1}^{N} c_i \Delta(o_t, \mu_i, \Sigma_i) \tag{1}$$

where O_t is the acoustic vector at time t, c_i is the mixture coefficient for the i^{th} mixture component, Δ is a Gaussian probability density function with mean μ_i and covariance matrix $\sum_i$ and N is the total number of Gaussians in the mixture.

To present MLNR, we need to briefly overview ML re-estimation (MLRE) and MPLKR.

2.1. MLRE [4]

In MLRE, the Gaussian means of SI-HMM are simply replaced by the corresponding re-estimated means which maximize the likelihood of the adaptation data, or equivalently maximize the following function over $\hat{\mu}_i$

$$Q(\mu, \hat{\mu}) = -\sum_{i=1}^{N}\sum_{t=1}^{T} \gamma_i(t)(o_t - \hat{\mu}_i)' \Sigma_i^{-1} (o_t - \hat{\mu}_i) \tag{2}$$

where $\gamma_i(t)$ is the posterior probability of the i^{th} mixture component at time t, and T is the number of adaptation frames.

The optimal value for the transformed means, which is shown by μ_i^*, can be easily obtained using the following relation.

$$\mu_i^* = \frac{\sum_{t=1}^{T} \gamma_i(t) o_t}{\sum_{t=1}^{T} \gamma_i(t)} \tag{3}$$

As can be interpreted from relation 3, a Gaussian is updated if the corresponding phonetic context of the new speaker is covered in the adaptation data. However, there are very limited speech data in RSA. Therefore, only M out of all N Gaussians can be updated. Consequently, the MLRE does not lead to an accurate adapted model.

2.2. MPLKR [4]

From the above discussion, we see that adaptation data should be used in more intelligent ways to reach more accurate adapted models. In MPLKR, the values of μ_i^* are used to setup a non-linear mapping from the means of the SI model to the means of the adapted model. The formulation of the aforementioned non-linear mapping is as follows.

$$\begin{aligned}
&\hat{\mu} = \Theta(\xi) = W\Psi(\xi) \\
&W = \left(U^* K + \alpha U K^{-1}\right)\left(K^2 + \alpha I\right)^{-1} \\
&\Psi(\xi) = \left[\psi(\xi, \xi_1), \ldots, \psi(\xi, \xi_N)\right]' \\
&U^* = \left[\mu_1^*, \ldots, \mu_N^*\right] \\
&U = \left[\mu_1, \ldots, \mu_N\right] \\
&K = \begin{bmatrix} \psi(\xi_1, \xi_1) & \cdots & \psi(\xi_1, \xi_N) \\ \vdots & \ldots & \vdots \\ \psi(\xi_N, \xi_1) & \cdots & \psi(\xi_N, \xi_N) \end{bmatrix} \\
&\xi_i = \left[\mu_i' \quad 1\right]'
\end{aligned} \tag{4}$$

where ψ is a kernel function and α is a regularization parameter, which can be determined empirically.

As can be interpreted from relation 4, MPLKR performs a non-linear mapping from the means of the SI model to the means of the adapted model based on the ML criterion. However, this method cannot be used in many practical applications due to its large computational complexity.

3. MAXIMUM LIKELIHOOD NEURAL REGRESSION

MLNR performs a novel non-linear mapping from the Gaussian means of SI model to the Gaussian means of

adapted model.

$$\hat{\mu} = F(\mu) = G(\mu) + \mu \quad (5)$$

where $G : \mathbf{R}^D \rightarrow \mathbf{R}^D$ is a non-linear function of μ. The idea of MLNR is to build the mapping G using GRNN with the ML estimates as targets.

3.1. Training dataset

Based on the ML criterion, as discussed in section 2, the best values for the means of the observed Gaussians can be calculated using relation 3. Therefore, the values of μ_i^* can be used to approximate G. Consequently, our training dataset is

$$S = \{(\mu_1, \mu_1^* - \mu_1) \cdots (\mu_m, \mu_m^* - \mu_m) \cdots (\mu_M, \mu_M^* - \mu_M)\}$$

Note that M is an integer between 1 and N, which represents the number of observed Gaussians, i.e. those i in (3) for which the denominator exceeds a threshold.

3.2. Function approximation

Our final goal is to approximate G such that it learns the training examples and generalizes well on new data points. There are different ways to reach this goal such as different classical linear and non-linear regression methods [10] or different soft computing approaches [11]. Among those, however, we have chosen GRNN due to different characteristics making it suitable for this problem. Advantages of GRNN over other function approximation methods will be discussed later in this paper. GRNN, which is a universal approximator, was introduced by Donald Specht [12]. It is an approach with a one-pass learning algorithm.

Figure 1 shows the network topology of the GRNN applied in MLNR. As can be seen in Fig. 1, the GRNN consists of four layers: the input layer, a layer of radial centers, a layer of regression units, and the output layer. The radial layer units are very similar to the hidden layer units of a Radial Basis Function (RBF) neural network [13]. The outputs of the neurons in this layer are calculated as follows.

$$\Omega_m(\mu) = \exp\left(-\frac{(\mu - \mu_m)'.(\mu - \mu_m)}{2\sigma^2}\right) \quad (6)$$

where σ is the width of the kernel function (the standard deviation of the Gaussian function assumed around each sample) and μ_m is the m^{th} normalized input training vector. The rationale for choosing a RBF is that the acoustic space in which o_t and μ_i live have obtained via linear discriminant analysis, so the intra-class covariance is spherical.

The regression layer has two types of processing: the summation units (Φ_d) and a single division unit ($\sum$). The output of these units is calculated using relation 7. Note that the number of summation units is always the same as the number of the GRNN output units.

$$\Phi_d(\mu) = \sum_{m=1}^{M} y_{dm}\Omega_m(\mu)$$
$$y_{dm} = (\mu_{dm}^* - \mu_{dm}) \quad (7)$$

where $\mu_{dm}^* - \mu_{dm}$ is the d^{th} element of the m^{th} output training vector.

The division unit only sums the weighted activation of the radial units. The output of this unit is calculated as follows.

$$\Sigma(\mu) = \sum_{m=1}^{M} \Omega_m(\mu) \quad (8)$$

The number of neurons in the output layer is equal to the dimension of the output vector. In the case where the output vector is $G(\mu)$, the output layer consists of D neurons and in each neuron the following calculation is performed:

$$G_d(\mu) = \frac{1}{\Sigma(\mu)}\Phi_d(\mu) \quad (9)$$

Therefore $F(\mu)$ is approximated as follows.

$$F(\mu) = \frac{\sum_{m=1}^{M} y_m \exp\left(-\frac{d(\mu - \mu_m)^2}{2\sigma^2}\right)}{\sum_{m=1}^{M} \exp\left(-\frac{d(\mu - \mu_m)^2}{2\sigma^2}\right)} + \mu$$
$$y_m = (\mu_m^* - \mu_m) \quad (10)$$

Equation (10) can also be interpreted as a smoothing operation of the observed mean corrections y_m involved in adaptation. The smoothing kernels are Radial Basis Functions with an extent controlled by σ.

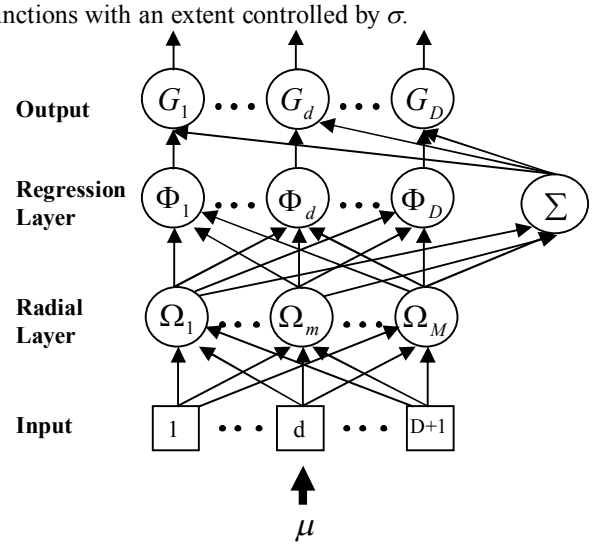


Fig. 1. The architecture of GRNN applied in MLNR.

3.3. Expectation-Maximization algorithm

In MLNR, just like MLLR, we need several Expectation-Maximization (EM) iterations to maximize the ML function. In each iteration the old values of μ in ML function are replaced by the estimated values $\hat{\mu}$ of equation 5. Then, the values of $\gamma_i(t)$ are re-computed. Hence, in principle, we should do multiple iterations of MLNR.

3.4. Determining hyperparameters

The value of the standard deviation of Gaussian functions (σ), which is also called smoothing parameter, significantly impacts the function approximation accuracy [14,15].

It is shown in [14] that there exists a close relationship between the variance of the input data, var(μ), and the smoothing parameter .Therefore, the following assumption is adopted in this paper.

$$\sigma^2 = \alpha \left\| \mathrm{var}(\mu) \right\|_2 \qquad (11)$$

where α is an unknown constant that is determined on an independent development set.

3.4. Advantages of MLNR over other speaker adaptation methods

Many adaptation schemes such as kernel eigenvoice [7-8] and kernel eigenspace-based MLLR [9] involve gradient-based optimization methods. An evident disadvantage pertaining to all gradient based optimization techniques is the possibility of local extrema. In contrast, MLNR involves no optimization task or calculation of unknown parameters. Another important advantage of MLNR is its computational simplicity. Table 1 shows the computational complexity of MLNR along with two other non-linear adaptation methods published recently.

Table 1. Computational Complexity of KRR, MPLKR, and MLNR. *D* is the vector dimension, *N* is the number of Gaussians, *M* is the number of observed Gaussians and *T* is the number of adaptation frames

Method	Computational Complexity
KRR	$O(DN^3T^3)$
MPLKR	$O(N^3)$
MLNR	$O(DMN)$

As can be interpreted from the table 2, the computational complexity of MLNR is significantly lower than that of MPLKR or KRR. This characteristic makes the MLNR suitable for real time applications.

Finally, for large amounts of adaptation data, the limiting behavior of MLNR can still be that of MAP by letting σ tend to zero.

3.5. Advantages of GRNN over other NNs

Different advantages of GRNN over other NNs, which are pointed out in [12-13], [16], are:

- GRNN is not involved with iterative learning algorithms. Instead, it has a one pass and fast learning. The standard supervised neural network architectures such as Multilayer Perceptron (MLP) and RBF infer a parameterized model (the weights forming the parameters) from available training data. These networks use the back-propagation algorithm for training, where it may take a large number of iterations to converge to the desired solution.
- GRNN requires only a fraction of the training samples a back propagation based neural network would need. In other words, GRNN can be effectively applied in the case of encountering very sparse data.

4. EXPERIMENTAL RESULTS

In this section, the efficacy of proposed method in comparison with three conventional methods, namely full transform MLLR (F-MLLR), diagonal transform MLLR (D-MLLR) and MPLLR, is shown.

4.1. Experiment setup

The proposed adaptation method was investigated on the WSJ benchmark large vocabulary recognition task: 5k word closed vocabulary (so no out-of-vocabulary words) using the standard bigram language model on the November 92 evaluation set. The feature extraction is MEL-spectrum based, including mean subtraction on the log spectrum, and a linear discriminant analysis resulting in 39 features. The baseline model doesn't incorporate Vocal Tract Length Normalization (VTLN). Training of the acoustic models is based on the standard SI- 84 WSJ0 database which contains about 15 hours of speech in total from 84 speakers. Speaker independent, cross word context and position dependent acoustic models were generated with 1938 tied HMM states, defined by an automatic phonetic decision tree for 45 phones. In total 32064 Gaussians with diagonal covariance were estimated. There is no sharing of Gaussians across states.

4.2. Tuning the hyperparameters

The optimal value for α for MLNR is found by a line search using a word error rate (WER) criterion on the November 92 development set. Each of the 10 speakers in this set has about 40 sentences. 30 of these were used for testing and of the remaining sentences, those with a maximum length of 10 seconds were used for adaptation. The average WER

over all speakers and retained adaptation sentences for different values of α can be seen in table 2.

As can be read from table 1, 0.5 yields the lowest WER in comparison with other tested values. Therefore, we adopt this value in rest of our experiments.

Table 2. Average WER in % for different values of α

α	1/8	2/8	4/8	6/8	8/8
Average WER	26.54	11.53	**7.97**	8.29	8.41

4.3. Results

To evaluate the proposed method, two experiments have been performed as follows.

Experiment 1:

In this experiment, our aim is to assess MLNR when the duration of the adaptation speech is less than 5 seconds. The initial SI model is adapted to five different speakers. For each speaker, the SI model is adapted using an utterance with a maximum length of 5 seconds. After the training phase, each adapted model is tested on about thirty sentences of the evaluation data. To ensure the evaluation process, the training and testing phases are repeated four times for each speaker using four different adaptation utterances. Table 3 indicates the average Word Error Rate (WER) for each speaker for which a sufficient number of short adaptation sentences is available. Note that for each method 10 adaptation iterations were run.

Table 3. Average WER in % for experiment 1

METHOD→ / SPEAKER ID ↓	SI	F-MLLR	D-MLLR	MPLLR	MLNR
440	10.22	14.05	8.15	8.74	7.12
441	13.98	59.66	10.88	12.14	10.97
445	4.70	26.34	6.04	4.22	3.90
446	4.89	18.12	4.76	4.76	4.35
447	11.07	34.03	7.62	7.77	8.01
AVERAGE	8.97	30.44	7.49	7.52	**6.87**

As it can be interpreted from table 3, MLNR is the most accurate method in rapid speaker adaptation in comparison with the other three methods. It yields 23.41%, 77.43%, 8.28%, 8.64% relative word error rate reduction compared to SI, F-MLLR, D-MLLR and MPLLR respectively. This experiment also shows that F-MLLR can fail due to poor estimation of the transformation matrix when the adaptation data is very scarce.

Experiment 2:

In this experiment, our aim is to assess MLNR when the duration of adaptation speech is between 5 seconds to 10 seconds. Therefore, all adaptation utterances of this experiment are in this interval. The initial SI model is adapted to six different speakers. For each speaker, the SI model is adapted using an adaptation utterance. After the training phase, each adapted model is tested on the same thirty sentences used experiment 1. To ensure the evaluation process, the training and testing phases are repeated five times for each speaker using five different adaptation utterances. Table 4 illustrates the average WER for each speaker. Note that for each method 10 adaptation iterations were run. Again, MLNR outperforms the other methods.

Table 4. Average WER in % for experiment 2

METHOD→ / SPEAKER ID ↓	SI	F-MLLR	D-MLLR	MPLLR	MLNR
441	13.98	17.67	11.07	12.15	10.49
442	11.23	11.27	10.51	11.23	10.73
443	8.25	6.39	6.30	7.57	6.00
444	10.86	6.82	9.44	8.61	7.64
445	4.70	8.46	5.47	4.23	3.93
446	4.89	4.86	4.96	4.78	4.86
AVERAGE	8.99	9.24	7.96	8.10	**7.27**

Table 4 suggests MLNR achieves relative word error rate reduction of 19.13%, 21.32%, 8.66% and 10.24% over SI, F-MLLR, D-MLLR and MPLLR respectively.

5. CONCLUSIONS AND FUTURE RESEARCHES

In this paper, a novel non-linear approach for speaker adaptation was introduced and evaluated on an RSA task. In this method, which is called Maximum Likelihood Neural Regression or MLNR, a GRNN trained based on the ML criterion is applied to adapt the SI Gaussian means. The proposed adaptation approach enjoys four important advantages. First, since the MLNR does not involve estimating unknown parameters, this method can be applied even if the available adaptation data is very limited. Second, due to non-linear nature of this method, it can capture any possible nonlinearity in the data. Third, this method is very simple and computationally more efficient than many other non-linear approaches such as MPLKR or KRR. Fourth, it can be tuned to converge to MAP adaptation with increasing adaptation set sizes.

Simulation results over WSJ benchmark confirm that the proposed scheme enjoys higher recognition accuracy in comparison with F-MLLR, D-MLLR and MPLLR.

In the near future, we will extend the method to take the reliability of the ML estimates into account and propose a method for automatically tuning σ.

6. ACKNOWLEDGMENTS

The authors would like to thank Ms. Xueru Zhang, Katholieke Universiteit Leuven, for advice and support during the course of this work.

7. REFERENCES

[1] R. Kuhn, F. Perronnin, and J. Junqua, "Time is money: Why very rapid adaptation matters," *in proc. ISCA Tutorial and Research Workshop on Adaptation Methods for Speech Recognition*, France, pp. 33-36, 2001.

[2] J. L. Gauvain and C. H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observation of Markov chains," *IEEE Trans. Speech Audio Process.*, vol. 2, pp. 291–298, Apr. 1994.

[3] C. J. Leggeter and P. C. Woodland, "Maximum Likelihood Linear Regression for Speaker Adaption of Continuous Density Hidden Markov Models," *Computer Speech and Language*, vol. 9, pp. 171–185, 1995.

[4] B. K. Mak, T. Lai, I. W. Tsang, and J. T. Kwok, "Maximum Penalized Likelihood Kernel Regression for Fast Adaptation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 7, pp. 1372–1381, 2009.

[5] I. Ding, "Incremental MLLR speaker adaptation by fuzzy logic control," *Pattern Recognition*, vol. 40, pp. 3110-3119, 2007.

[6] G. Saon, "A non-linear speaker adaptation technique using kernel ridge regression," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2006, vol. I, pp. 225-228.

[7] B. Mak, J. T. Kwok, and S. Ho, "Kernel eigenvoice speaker adaptation," *IEEE Trans. Speech Audio Process.*, vol. 13, no. 5, pp. 984-992, Sep. 2005.

[8] B. Mak, R. Hsiao, S. Ho, and J. T. Kwok, "Embedded kernel eigenvoice speaker adaptation and its implication to reference speaker weighting," *IEEE Trans. Speech Audio Process.*, vol. 14, no. 4, pp. 1267-1280, Jul. 2006.

[9] B. Mak and R. Hsiao, "Kernel eigenspace-based MLLR adaptation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 3, pp. 784-795, Mar. 2007.

[10] D. A. Freedman, "Statistical Models: Theory and Practice," Cambridge University Press, 2005.

[11] R. A. Aliev and R. R. Aliev, "Soft computing and its applications," World Scientific, 2001.

[12] D. F. Specht, "A general regression neural network," *IEEE Trans. Neural Networks*, vol. 2, no. 6, pp. 568-576, November 1991.

[13] C. Christodoulou and M. Georgiopoulos, "Application of Neural Network in Electromagnetics," Boston: Artech House, pp. 149-160, 2001.

[14] M. Zhong, et. al., "Gap-Based Estimation: Choosing the Smoothing Parameters for Probabilistic and General Regression Neural Networks" *Int. Conf. Neural Networks*, Canada, pp. 1870-1877, 2006.

[15] D. F. Specht, "Experience with Adaptive Probabilistic Neural Networks and Adaptive General Regression Neural Networks," in *Proc. IEEE World Congress on Computational Intelligence*, vol. 2, pp. 1203-1208, *1994*.

[16] B. Erkmena and T. Yıldırım, "Improving classification performance of sonar targets by applying general regression neural network with PCA" *Expert Sys. Appl.*, vol. 35, no. 1-2, pp. 472-475, 2008.