Speaker Adaptation using Nonlinear Regression Techniques for HMM-based Speech Synthesis

Doo Hwa Hong, Shin Jae Kang, Joun Yeop Lee, Nam Soo Kim
Department of Electrical and Computer Engineering and INMC
Seoul National University
Seoul 151-742, Korea
{dhhong, sjkang, jylee}@hi.snu.ac.kr, nkim@snu.ac.kr

Abstract—The maximum likelihood linear regression (MLLR) technique is a well-known approach to parameter adaptation in hidden Markov model (HMM)-based systems. In this paper, we propose the maximum penalized likelihood kernel regression (MPLKR) approach as a novel adaptation technique for HMM-based speech synthesis. The proposed algorithm performs a nonlinear regression between the mean vector of the base model and the corresponding mean vector of adaptive data by means of a kernel method. In the experiments, we used various types of parametric kernels for the proposed algorithm and compared their performances with the conventional method. From experimental results, it has been found that the proposed algorithm outperforms the conventional method in terms of the objective measure as well as the subjective listening quality.

Keywords—maximum likelihood linear regression (MLLR); HMM-based speech synthesis; kernel; maximum penalized likelihood kernel regression (MPLKR);

I. INTRODUCTION

Maximum likelihood linear regression (MLLR) is one of the most popular techniques for parameter adaptation in hidden Markov model (HMM)-based systems [1][2]. In the MLLR approach, original parameters of the HMM-based system are mapped to their adapted values via a set of affine transformations which are estimated from a small amount of adaptation data. MLLR was first proposed for speaker adaptation in order to improve the performance of the speech recognition systems, and later a variety of extensions have been developed with applications to other areas [3]-[7].

MLLR employs a linear mapping to transform the base model parameters to the corresponding adapted parameters. Though simple and easy to handle, this linear mapping is considered too restrictive to approximate a sophisticated mapping between two model parameters. Especially in speech synthesis, there exist a lot of models which may not be mapped properly into the target speech with a small amount of adaptation data when a simple affine transform is applied. One of the promising ways to overcome this restriction is to employ non-linear mappings.

Recently in the area of speech recognition, Mak et. al. proposed the maximum penalized likelihood kernel regression (MPLKR) algorithm for fast speaker adaptation [8][9]. In MPLKR, kernels are employed in the MLLR framework as the weights of regression vectors, and a penalization term is appended to the likelihood formulation in order to avoid overfitting. The basic idea of this technique is to map the mean vector of the base model to a high-dimensional feature space via a nonlinear mapping before performing linear regression. In our previous work, we also proposed the factored maximum likelihood kernel regression (FMPLKR) technique for style-adaptive speech synthesis which conducts a nonlinear kernel regression between the mean vectors of the base model and the adaptation data obtained from different speaking styles [10][11].

In this paper, we propose the MPLKR technique as one of alternatives of general speaker-adaptive speech synthesis using nonlinear regression methods. One of the advantages of the proposed technique is that except for the nonlinear mapping of the base model, the rest of operations are linear. Hence, the optimal parameters can be obtained by solving a system of linear equations in a similar way as the conventional MLLR, which enables an easy implementation and does not require a heavy computational load. Performance of the proposed technique is evaluated in a series of experiments on speech synthesis, and compared with the conventional MLLR method.

II. MLLR

In conventional MLLR adaptation, a $p$-dimensional mean vector $\mu_s \in \mathbb{R}^p$ of a particular distribution $s$ of the HMM state is transformed to $\hat{\mu}_s$ via

$$\hat{\mu}_s = M \nu_s \quad (1)$$

where $M$ is a $p \times (p+1)$ regression matrix which can be decomposed into $M = [A \ b]$ with $A$ and $b$ indicating the parameters of the affine transformation, and $\nu_s$ denotes a $(p+1)$-dimensional augmented mean vector of the distribution $s$ defined by

$$\nu_s = [\mu_s \ 1]' \quad (2)$$

with the prime denoting the transpose of a matrix or a vector. In (2), it is seen that by appending a constant 1 to the mean vector $\mu_s$, the original affine transform $A \mu_s + b$ can be
written as a linear formulation as given by (1). The output probability density function (PDF) of the distribution \( s \) is assumed to be a single Gaussian with the mean \( \mu_s \) and covariance matrix \( \Sigma_s \). Generally, \( \Sigma_s \) is assumed to be diagonal [1].

The regression parameter \( M \) is estimated according to the maximum likelihood (ML) criterion, and the expectation maximization (EM) algorithm is applied to increase the likelihood iteratively. Let \( X = (x_1, x_2, \cdots, x_T) \) be the given adaptation data vectors that are used to adapt the mean vectors of the HMM. At the \( E \) (expectation) step of the EM algorithm, we first compute the posterior probability of the distribution \( s \) at each time defined by

\[
\gamma_t(s) = Pr(\theta(t) = s | X, \lambda) \quad (3)
\]

where \( \theta(t) \) indicates the distribution index at time \( t \) and \( \lambda \) represents the current adaptation parameters. Then, at the \( M \) (maximization) step, we update the regression parameter \( M \) so as to maximize the expectation of the complete data log likelihood as given by

\[
\hat{M} = \arg \max_M \left[ - \frac{1}{2} \sum_{s=1}^{S} \sum_{t=1}^{T} \gamma_t(s) \right. \\
\times (x_t - \mu_s)^T \Sigma_s^{-1} (x_t - \mu_s) \quad (4)
\]

where \( \hat{M} \) is the updated parameter and \( S \) denotes the number of distribution in the same regression class. The solution to (4) is computed by differentiation with respect to each row of \( M \). For more details, the readers are referred to [1].

III. MPLKR

The adaptation scheme given by (1) can be rewritten as follows:

\[
\hat{\mu}_s = \sum_{j=1}^{P+1} M(j) \nu_{s,j} \quad (5)
\]

where \( M(j) \) denotes the \( j \)-th column vector of \( M \) and \( \nu_{s,j} \) indicates the \( j \)-th element of \( \nu_s \). From (5), we can see that \( \hat{\mu}_s \) turns out to be a linear combination of a number of basis vectors. In this formalism, each \( M(j) \) acts as a basis vector and \( \nu_{s,j} \) is treated as a weight for the \( j \)-th basis vector.

Motivated by this viewpoint, we can extend (5) to a more generalized form as follows:

\[
\hat{\mu}_s = \sum_{j=1}^{P+1} M(j) \xi_j(\mu_s) = M \xi(\mu_s) \quad (6)
\]

where \( \{M(j) | j = 1, 2, \cdots, P+1\} \) represents a set of basis vectors and bias, and \( \xi_j(\mu_s) \) indicates the \( j \)-th element of the \( (P+1) \)-dimensional feature vector \( \xi(\mu_s) \) defined by

\[
\xi(\mu_s) = [\psi_1(\mu_s) \cdots \psi_P(\mu_s)]^T \quad (7)
\]

where \( \psi_j(\mu_s) \) is the weight associated with the \( j \)-th basis vector. In (6), \( P \) denotes the number of basis vectors, which can be selected freely. If \( P > p \), (6) implies an over-complete representation. Another point to note in (6) is that \( \psi_j(\cdot) \) is a nonlinear function which extracts the weight from the base model parameter \( \mu_s \).

A promising way to define the nonlinear function \( \psi_j(\cdot) \) is to apply a kernel map. Let \( \{c_1, c_2, \cdots, c_p\} \) denote a set of \( P \) vectors of dimension \( p \). Then a kernel map is defined by

\[
\psi_j(\mu_s) = \kappa(\mu_s, c_j) \quad (8)
\]

where \( \kappa(\cdot, \cdot) \) denotes a kernel function.

Combining (6) and (8), the MPLKR approach adapts the model parameter in the following way:

\[
\hat{\mu}_s = \sum_{j=1}^{P} M(j) \kappa(\mu_s, c_j) + b \quad (9)
\]

where \( b \) indicates the bias term identical to \( M(P+1) \). There is a prominent distinction between the MPLKR and MLLR approaches. In the MPLKR technique, the number of basis vectors is not restricted to \( p + 1 \), the dimension of the model parameter including the bias term. This distinction enables MPLKR to give a more flexible way of parameter adaptation.

A. Parameter estimation

Generally, nonlinear regression with a large number of parameters may usually suffer from the problem of overfitting. In order to alleviate this problem, a regularization technique is usually applied. Given some prior knowledge of the expected values of \( M \), we can add a regularization term to (4) to penalize the estimated parameters deviating from the expected value, then the maximum penalized likelihood criterion with (9) can be written as

\[
\tilde{M} = \arg \max_M \left[ - \frac{1}{2} \sum_{s=1}^{S} \sum_{t=1}^{T} \gamma_t(s) \right. \\
\times \left[ \Sigma_s^{-1} (x_t - \mu_s) \right]^T \Sigma_s^{-1} (x_t - \mu_s) \\
+ \frac{1}{2} \sum_{s=1}^{S} \sum_{t=1}^{T} \frac{1}{\sigma^2_{s,i}} x_{t,i}^2 \\
- \beta \sum_{i=1}^{P} \sum_{j=1}^{P+1} (M(i,j) - M_0(i,j))^2 \quad (10)
\]

in which \( x_{t,i} \) and \( b_i \) denote the \( i \)-th elements of \( x_t \) and \( b \), respectively, \( M(i,j) \) and \( M_0(i,j) \) are the \( i \)-th row and \( j \)-th column elements of the matrix \( M \) and its prior \( M_0 \), respectively, and \( \beta \) indicates a regularization parameter. In (10), we have assumed that the covariance matrix of each distribution \( s \) is diagonal.

The maximization step of parameter estimation using an EM algorithm is performed as follows. Setting the derivative of the objective function shown in the right hand side of (10)
with respect to $M(i, j)$ to zero for each row $i$, we are led to
\[
\sum_{s=1}^{S} \sum_{t=1}^{T} \gamma_t(s) \frac{1}{\sigma_{s,i}} \left( x_{t,i} - \sum_{k=1}^{P} M(i, k) \kappa(\mu_s, c_k) - b_j \right) \\
\times \kappa(\mu_s, c_j) - \beta \left( M(i, j) - M_0(i, j) \right) = 0. \tag{11}
\]
To replace $M_0(i, j)$ in (11), the parameters of $M_0$ can be chosen in order that will reproduce the original mean vector
\[
M_0 = \arg \min_M \sum_{s=1}^{S} \sum_{i=1}^{P} \| \mu_{s,i} - \sum_{j=1}^{P} M(i, j) \kappa(\mu_s, c_j) - b_j \|^2 \tag{12}
\]
where $\mu_{s,i}$ indicates the $i$-th element of $\mu_s$. Then the solution is given by
\[
\hat{m}_t = (k^{(i)} + \beta k_0^{-1}) (G^{(i)} + \beta I)^{-1}
\]
where $\hat{m}_t$ represents the $i$-th row vector of $\hat{M}$, $I$ denotes a $(P + 1) \times (P + 1)$ identity matrix, and the other symbols are defined as
\[
G_0 = \sum_{s=1}^{S} \xi(\mu_s) \xi(\mu_s)', \tag{14}
\]
\[
G^{(i)} = \sum_{s=1}^{S} \sum_{t=1}^{T} \frac{\gamma_t(s)}{\sigma^2_{s,i}} \xi(\mu_s) \xi(\mu_s)', \tag{15}
\]
\[
k_0^{(i)} = \sum_{s=1}^{S} \mu_{t,i} \xi(\mu_s)', \tag{16}
\]
and
\[
k^{(i)} = \sum_{s=1}^{S} \sum_{t=1}^{T} \frac{\gamma_t(s)}{\sigma^2_{s,i}} x_{t,i} \xi(\mu_s)', \tag{18}
\]

IV. EXPERIMENTS

In order to evaluate the performance of the proposed technique when applied to speech synthesis, we conducted several experiments on objective measurement and subjective listening test. All the speech data collected for speech synthesis were Korean spoken language.

For the construction of the baseline speech synthesizer, a Korean speech database spoken by a female speaker (YMK) was applied. The speaker provided 4,000 utterances of reading-style speech data amounting to 507 minutes, which were used to train a baseline reading-style speech synthesizer. We also collected the speech data for adaptation from the utterances of the other female speaker (SKJ).

![Figure 1. Average mel-cepstral distance between the original and synthesized speech.](image)

The adaptation data consisted of 130 utterances amounting to 11 minutes. Among these 130 utterances, we used 80 utterances to train the regression matrices and the remaining 50 utterances to evaluate the performance.

The model parameters were trained using HTS version 2.3 alpha [13]. Each utterance was sampled at 16 kHz and a 20 ms Hamming window was applied with 5 ms frame shift for speech feature extraction. As for the spectrum feature, a 25th-order mel-scaled cepstrum vector was extracted at each frame. By attaching the $\Delta$- and $\Delta\Delta$-cepstra derived from the extracted mel-scaled cepstrum sequence, the spectrum feature could be represented by a 75-dimensional vector at each frame. We also extracted the pitch from each frame for the generation of voiced excitation signals. As the basic unit of speech synthesis, we applied quinphones followed by context-dependent reading-style text analysis described in [12]. Each quinphone was modeled by a 5 state left-to-right structured HMM where the observation distribution at each state was given by a single Gaussian PDF with diagonal covariance matrix.

A. Objective performance evaluation

First, we compared the proposed algorithm with the conventional MLLR. For the nonlinear map of the proposed algorithm, we used three different types of kernel functions: linear, quadratic, and Gaussian kernels. Based on preliminary experiments, we set the constant of the linear or quadratic kernel function to 1, the width $\sigma^2$ of Gaussian kernel to $\{100.0, 1.0, 5.0\}$, for each static, $\Delta$ and $\Delta\Delta$ block transformation, respectively, and $\beta$ in (10) to 1.0.

The results of objective performance test obtained for various values of $P$ are shown in Fig. 1. We evaluated the average mel-cepstral distance which is the squared Euclidean norm of difference between the mel-cepstra obtained from the original and synthesized speech. From the results, we can find that the proposed approach using the Gaussian
kernel function is more effective in reducing the mel-cepstral discrepancy than the conventional algorithm.

B. Subjective performance evaluation

Next, we performed a subjective listening test to compare the proposed algorithm using the Gaussian kernel to the conventional MLLR technique, for which 15 listeners participated and 10 sentences were used. The value of $P$ was set to 200 to generate synthesized speech samples for the proposed approach. In the test, each listener was provided with speeches synthesized through different methods, and the speech quality was measured in terms of the comparative mean opinion score (CMOS) [14], where for each test a pair of two speech files were given and each subject provided his/her preference in speech quality in the range of $[-3, 3]$ with a positive value indicating that the former shows a better quality than the latter, and vice versa. The results are shown in Fig. 2 from which we can confirm that the proposed approach produced a better speech quality than the conventional method.

V. Conclusions

In this paper, we have proposed the MPLKR algorithm for HMM-based adaptive speech synthesis. The proposed approach provides nonlinear regression between the mean vector of the base model and the corresponding mean vector of adaptation data with the use of a kernel method. From the experimental results, it has been found that the proposed algorithm outperformed the conventional MLLR method in terms of the objective measure as well as the subjective listening quality.

Acknowledgment

This work was supported in part by Ministry of Science, ICT (Information and Communication Technologies) and Future Planning by the Korean Government (NRF-2011K2A2A6A00002), and by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (NIPA-2014-H0301-14-1019) supervised by the NIPA (National IT Industry Promotion Agency).

References


