特征空间本征音说话人自适应

屈 丹 [1] 杨绪魁 [1] 张文林 [1]

摘 要 提出了特征空间本征音说话人自适应算法, 该方法首先借鉴 RATZ 算法的思想, 采用高斯混合模型对特征空间中的说话人信息进行建模; 其次利用子空间方法实现对特征补偿项的估计, 减少估计参数的数量, 在对特征空间精确建模的同时, 降低了算法对自适应数据量的需求. 基于微软语料库的中文连续语音识别实验表明, 该算法在自适应数据量极少时仍能取得较好的性能, 配合说话人自适应训练能够进一步降低词错误率, 其实时性优于本征音说话人自适应算法.

关键词 连续语音识别, 说话人自适应, 多高斯倒谱规整, 本征音

引用格式 屈丹, 杨绪魁, 张文林. 特征空间本征音说话人自适应. 自动化学报, 2015, **41**(7): 1244−1252

DOI 10.16383/j.aas.2015.c140644

Feature Space Eigenvoice Speaker Adaptation

QU Dan[1] YANG Xu-Kui[1] ZHANG Wen-Lin[1]

Abstract A speaker adaptation method at feature level named feature-space eigenvoice adaptation method is proposed. In this method, similar to RATZ, the information of speakers in the feature space is modeled by a Gaussian mixture model. Moreover, the number of parameters to be estimated is decreased by taking the dependency of these parameters into account. This method can use very little data to construct a more accurate feature space model. Experimental results of continuous speech recognition on Microsoft speech database show that this method can still achieve good performance even when the adaptation data is limited. And speaker adaptive training based on this method can further decrease the word error rate with a superior real-time performance to that of eigenvoice methods.

Key words Continuous speech recognition, speaker adaptation, multivariate Gaussian-based cepstral normalization, eigenvoice

Citation Qu Dan, Yang Xu-Kui, Zhang Wen-Lin. Feature space eigenvoice speaker adaptation. *Acta Automatica Sinica*, 2015, **41**(7): 1244−1252

在现代语音识别系统中, 自适应模块是其不可或缺的一个重要组成部分. 它通过消除声学模型参数与测试语音声学特征之间的不匹配, 提高系统识别性能. 说话人自适应可以在模型层上进行, 即通过调整说话人无关 (Speaker independent, SI) 声学模型的参数, 使之与目标说话人语音的声学特性相匹配. 在这类算法中, 基于子空间的说话人自适应算法[1−4], 如本征音 (Eigenvoice, EV)[4] 自适应, 在自适应数据量较少时仍能取得不错的性能, 是连续语音识别研究中的热点之一.

但是, 对于模型层的说话人自适应算法, 自适应时必须针对每一个测试说话人调整 SI 声学模型参数, 构造新的说话人自适应 (Speaker adapted, SA) 模型, 客观上增加了解码的复杂度, 在一些实时性要求较高的场合不甚适用. 事实上, 通过对测试语音的声学特征进行规整, 使之与 SI 声学模型相匹配, 同样可以达到自适应的目的, 且其实时性优于模型层的说话人自适应算法. 这类技术即为特征层说话人自适应技术, 大致可以分为特征参数自适应和特征参数归一化.

特征参数自适应通过对目标说话人的语音特征进行一定的变换, 使之与 SI 声学模型相匹配. 这类方法的典型代表是特征空间最大似然线性回归 (Feature-space maximum likelihood linear regression, FMLLR)[5−6] 及其在不同模型下的应用算法[7−9], 它通过线性变换矩阵对目标说话人的特征矢量进行变换以实现特征自适应的目的.

在基于数据驱动 (Data-driven) 的特征参数归一化算法中, 假设语音声学特征 (如梅尔频率倒谱系数 (Mel frequency cepstrum coefficient, MFCC)) 为随机变量, 训练和识别的不匹配在特征域上就体现为训练声学特征概率分布和测试声学特征概率分布的差异. 因此, 对特征参数进行规整, 使得训练和识别过程中特征参数的分布趋于一致, 系统的性能

收稿日期 2014-09-12 录用日期 2015-01-24
Manuscript received September 12, 2014; accepted January 24, 2015
国家自然科学基金 (61175017, 61403415, 61302107) 资助
Supported by National Natural Science Foundation of China (61175017, 61403415, 61302107)
本文责任编委 吴玺宏
Recommended by Associate Editor WU Xi-Hong
1. 解放军信息工程大学信息系统工程学院 郑州 450000
1. Institute of Information Systems Engineering, Information Engineering University, Zhengzhou 450000

将得到改善. 最简单的特征参数规整技术是倒谱均值规整 (Cepstral mean normalization, CMN)[10−12], 其基本思想是对倒谱特征的一阶矩 (即均值) 进行规整, 减小训练语音和测试语音倒谱特征的概率密度函数之间的差异, 进而补偿语音识别系统中特征参数不匹配造成的影响.

CMN 用一个固定的均值矢量来补偿所有的声学特征, 从理论上看, CMN 只能够对特征域中的卷积噪声进行补偿. 因此, 这样的特征规整太过简单, 存在很大限制. 在鲁棒性语音识别中, 基于数据驱动的特征变换技术[13−14], 如多高斯倒谱规整 (Multivariate Gaussian-based cepstral normalization, RATZ) 算法[15], 用高斯混合模型 (Gaussian mixture model, GMM) 对纯净语音特征空间进行建模, 而带噪语音特征概率分布同样假设为 GMM. 相关研究表明, 在倒谱域, 噪声对语音信号分布的影响表现为均值的偏移和方差的变化. 因此, 带噪语音的概率分布可以近似为在纯净语音 GMM 的均值和方差上叠加一定的补偿项. 如果将不同说话人之间的差异视为某种噪声, 这种特征补偿方法也可以用来进行特征层的说话人自适应. 为了精确描述声学特征的概率分布, GMM 模型的混元数目一般较高, 因此 RATZ 算法对补偿项的估计中, 需要较多的数据量才能得到稳健的参数估计.

本文借鉴 RATZ 算法的思想, 提出了特征空间本征音说话人自适应 (Feature-space eigenvoice, FEV) 算法. 该算法采用与 RATZ 算法类似的方法对特征参数进行规整; 同时为了解决 RATZ 算法估计参数过多、数据需求量较大的问题, 该算法采用子空间方法对估计参数进行建模, 极大地减少了估计参数的数目, 降低了算法对自适应数据量的需求. 基于微软中文语料库的说话人自适应实验表明, 该方法在自适应数据较少时, 性能优于 FMLLR 算法; 同时 FEV 与说话人自适应训练 (Speaker adaptive train, SAT)[16−17] 相配合, 性能较 EV 算法有了一定的提高, 更重要的是, FEV 算法的实时性明显优于 EV 算法.

本文的组织如下: 第 1 节对基于数据驱动的特征变换技术进行了介绍; 第 2 节对特征空间本征音进行了介绍; 第 3 节给出实验设置及实验结果; 最后为结论部分.

1 基于数据驱动的特征变换技术

基于数据驱动的特征变换技术直接从观测数据中得到环境噪声对纯净语音倒谱特征分布的影响. 这类技术并没有明确地对环境噪声进行建模, 而是用现有的观测数据描述环境特征.

1.1 倒谱均值规整

倒谱均值规整, 也叫倒谱域减均值 (Cepstral mean subtraction, CMS), 是在特征补偿技术中最简单的一种, 也是其中的典型代表. 其基本假设是训练和识别的特征参数概率分布相同, 只是相差一个常数. 因此, 可以通过特征参数减去其均值来去除常数偏移的影响. 但 CMN 一般只能用来补偿信道差异带来的卷积畸变. 其处理步骤如下: 首先计算整段语音倒谱特征的均值 $\bar{\boldsymbol{x}}$; 然后将各帧特征 $\boldsymbol{x}_t$ 都减去该均值, 得到新的特征参数 $\hat{\boldsymbol{z}}_t$, 即

$$\hat{\boldsymbol{z}}_t = \boldsymbol{x}_t - \bar{\boldsymbol{x}} \tag{1}$$

最初, CMN 是为了解决传输信道中卷积噪声的影响而提出来的. 但是在无卷积噪声干扰时, CMN 仍能有效地提高系统性能. 因此, 倒谱均值不仅反映了信道传输函数的特性, 同时也描绘了语音中说话人的特性. 自然而然, CMN 可以在说话人归一化中得到应用.

1.2 RATZ 算法

CMN 用一个固定的修正向量对所有的特征向量进行补偿, 这样的补偿在实际环境中显得过于简单. RATZ 算法对语音的倒谱特征分布进行了更为精细的建模, 即假设纯净语音倒谱特征 $\boldsymbol{z}$ 和带噪语音倒谱特征 $\boldsymbol{x}$ 的概率密度函数可以用混元数相同的 GMM 来拟合, 分别如式 (2) 和式 (3) 所示:

$$p(\boldsymbol{z}) = \sum_{c=1}^{C} w_c^{(\boldsymbol{z})} \mathrm{N}\left(\boldsymbol{z}|\boldsymbol{m}_c^{(\boldsymbol{z})}, \boldsymbol{\Sigma}_c^{(\boldsymbol{z})}\right) \tag{2}$$

$$p(\boldsymbol{x}) = \sum_{c=1}^{C} w_c^{(\boldsymbol{x})} \mathrm{N}\left(\boldsymbol{x}|\boldsymbol{m}_c^{(\boldsymbol{x})}, \boldsymbol{\Sigma}_c^{(\boldsymbol{x})}\right) \tag{3}$$

其中, C 为高斯混合模型的混元数, 带噪语音分布函数的均值 $\boldsymbol{m}_c^{(\boldsymbol{x})}$ 和方差 $\boldsymbol{\Sigma}_c^{(\boldsymbol{x})}$ 可以近似为纯净语音均值 $\boldsymbol{m}_c^{(\boldsymbol{z})}$ 和方差 $\boldsymbol{\Sigma}_c^{(\boldsymbol{z})}$ 的基础上叠加一定的补偿项 $\Delta\boldsymbol{m}_c$ 和 $\Delta\boldsymbol{\Sigma}_c$, 即

$$\boldsymbol{m}_c^{(\boldsymbol{x})} = \boldsymbol{m}_c^{(\boldsymbol{z})} + \Delta\boldsymbol{m}_c \tag{4}$$

$$\boldsymbol{\Sigma}_c^{(\boldsymbol{x})} = \boldsymbol{\Sigma}_c^{(\boldsymbol{z})} + \Delta\boldsymbol{\Sigma}_c \tag{5}$$

因此, 带噪语音倒谱特征 $\boldsymbol{\chi} = \{\boldsymbol{x}_t\}_{t=1}^{T}$ 的对数似然函数为

$$L(\boldsymbol{\chi}) = \sum_{t=1}^{T} \log\left(p(\boldsymbol{x}_t)\right) = \sum_{t=1}^{T} \log\left(\sum_{c=1}^{C} w_c^{(\boldsymbol{x})} \mathrm{N}\left(\boldsymbol{x}_t|\boldsymbol{m}_c^{(\boldsymbol{x}_t)}, \boldsymbol{\Sigma}_c^{(\boldsymbol{x}_t)}\right)\right) \tag{6}$$

则补偿项 $\Delta\boldsymbol{m}_c$ 和 $\Delta\boldsymbol{\Sigma}_c$ 的最大似然估计可以通过 EM 算法迭代求解[15].

最终, 对纯净语音倒谱特征向量的 RATZ 估计为

$$\hat{\boldsymbol{z}}_t = \boldsymbol{x}_t - \sum_{c=1}^{C} \mathrm{N}\left(\boldsymbol{x}_t|\boldsymbol{m}_c^{(\boldsymbol{x})}, \boldsymbol{\Sigma}_c^{(\boldsymbol{x})}\right) \Delta\boldsymbol{m}_c \tag{7}$$

2 特征空间本征音自适应算法

从特征分布的角度来分析, CMN 是一种与模型无关的特征层说话人自适应方法, 它用一个单高斯概率密度函数描述倒谱特征在特征空间的分布, 从统计意义上看该分布的均值可以一定程度上反映出说话人的特性. CMN 用均值对每一帧特征向量进行规整, 可以减小不同特征集合之间的分布差异, 进而缩小不同说话人特征分布之间的差异. 但是, 仅用单高斯概率密度函数进行建模, 难以精确地描述特征空间中说话人信息的分布情况. RATZ 算法采用 GMM 拟合倒谱特征的概率分布, 精确描述特征空间分布情况的同时, 也给算法的应用带来了限制. RATZ 需要为 GMM 中每一个混元估计一个均值偏移, 在数据量较少时难以得到稳定的估计值. 本文提出的 FEV 算法采用与 RATZ 相同的方法对声学特征进行规整, 同时充分利用了估计参数之间的相关性, 采用子空间方法对其进行建模, 极大地减少了估计参数的数量, 降低了算法对自适应数据量的需求.

2.1 模型假设

本文仅讨论基于隐马尔科夫模型的语音识别系统中高斯均值矢量的自适应. 设训练集中共有 S 个说话人, 采用 F 维声学特征矢量, 声学模型共包含 C 个高斯分量. 对于说话人无关的声学特征 $\boldsymbol{z}$ 和说话人 s 的语音声学特征 $\boldsymbol{\chi}(s)$, 令 $\boldsymbol{m}_c$ 和 $\boldsymbol{m}_c(s)$ 分别为 SI 模型和第 s 个说话人 SD 模型中第 c 个高斯混元的均值矢量, $\boldsymbol{\Sigma}_c$ 和 $\boldsymbol{\Sigma}_c^{(s)}$ 为其相应的协方差矩阵, $c = 1, 2, \cdots, C$, 且两者仅差一个偏移项. 为了简化算法, 仅仅只对均值矢量进行自适应. 因此

$$\boldsymbol{m}_c(s) = \boldsymbol{m}_c + \Delta\boldsymbol{m}_c(s) \tag{8}$$

由于对系统中的每个状态的每个高斯混元都需要估计 $\Delta\boldsymbol{m}_c(s)$, 待估计的参数数量为 CF 个. 当自适应语料有限时, 无法得到稳健的参数估计. 为了解决这个问题, 定义均值偏移超矢量 $\Delta\boldsymbol{m}(s) = \left[\Delta\boldsymbol{m}_1(s)^{\mathrm{T}}, \Delta\boldsymbol{m}_2(s)^{\mathrm{T}}, \cdots, \Delta\boldsymbol{m}_C(s)^{\mathrm{T}}\right]^{\mathrm{T}}$, 假设其存在一个低维子空间, 即

$$\Delta\boldsymbol{m}(s) = \boldsymbol{V}\boldsymbol{y}(s) = \left[\boldsymbol{v}_1^{\mathrm{T}}, \boldsymbol{v}_2^{\mathrm{T}}, \cdots, \boldsymbol{v}_C^{\mathrm{T}}\right]^{\mathrm{T}} \boldsymbol{y}(s) \tag{9}$$

其中, $\boldsymbol{V}$ 为子空间基矢量矩阵, 其维数为 $CF \times R$, R 为子空间维数, $\boldsymbol{y}(s)$ 为 $R \times 1$ 向量.

当 $\boldsymbol{V}$ 已知时, 只需估计 $\boldsymbol{y}(s)$ 即可得到均值偏移 $\Delta\boldsymbol{m}_c(s)$. 由于 $R \ll CF$, 需要估计的参数少, 因此算法对数据的需求量更低.

2.2 算法推导

2.2.1 子空间基矩阵的估计

对于说话人 s 的声学特征 $\boldsymbol{\chi}(s) = \{\boldsymbol{x}_t^s\}_{t=1}^{\mathrm{T}}$, 假设 $\gamma_t^s(c)$ 为说话人 s 的第 t 帧特征 $\boldsymbol{x}_t^s$ 在模型 $\lambda_{\boldsymbol{z}}$ 第 c 个混元上的后验概率, $\gamma_t^s(c)$ 可以通过两种方法得到, 一种是给定自适应数据的标注, 通过 Baum-Welch 算法计算得到, 另外一种是通过 Viterbi 算法将训练数据强制对齐到 SI 模型后, 令

$$\gamma_t^s(c) = \begin{cases} 1, & \boldsymbol{x}_t^s \text{ 对齐到混元 } c \\ 0, & \boldsymbol{x}_t^s \text{ 不对齐到混元 } c \end{cases} \tag{10}$$

则有 $\boldsymbol{\chi}(s) = \{\boldsymbol{x}_t^s\}_{t=1}^{\mathrm{T}}$ 的似然函数为

$$\begin{aligned} l(\boldsymbol{V}|\boldsymbol{\chi}(s), \boldsymbol{y}(s)) &= P\left(\boldsymbol{\chi}(s)|\boldsymbol{V}, \boldsymbol{y}(s)\right) = \\ &\prod_{t=1}^{T}\sum_{c=1}^{C} \gamma_t^s(c)\mathrm{N}(\boldsymbol{x}_t^s|\boldsymbol{m}_c(s), \boldsymbol{\Sigma}_c(s)) = \\ &\prod_{t=1}^{T}\sum_{c=1}^{C} \gamma_t^s(c)\mathrm{N}(\boldsymbol{x}_t^s|\boldsymbol{m}_c + \boldsymbol{v}_c\boldsymbol{y}(s), \boldsymbol{\Sigma}_c) \end{aligned} \tag{11}$$

若 $\boldsymbol{y}(s)$ 的先验分布服从标准正态分布, 则 $\boldsymbol{v}_c$ 的估计可以通过最大化式 (12) 的似然函数来实现, 即

$$P\left(\boldsymbol{\chi}|\boldsymbol{v}_c\right) = \prod_{s=1}^{S}\int_{\boldsymbol{y}(s)} P\left(\boldsymbol{\chi}(s)|\boldsymbol{V}, \boldsymbol{y}(s)\right) P\left(\boldsymbol{y}(s)\right) \mathrm{d}\boldsymbol{y}(s) \tag{12}$$

该优化问题可以通过 EM 算法求解.

假设第 i 次迭代后的子空间基矩阵元素 c 的参数为 $\boldsymbol{v}_c^i$, 则第 $i+1$ 次迭代中 EM 算法的辅助函数 $Q\left(\boldsymbol{v}_c, \boldsymbol{v}_c^i\right)$ 如式 (13) 所示, 即

$$\begin{aligned} Q\left(\boldsymbol{v}_c, \boldsymbol{v}_c^i\right) &= E_{\boldsymbol{y}(s)}\left[\log l(\boldsymbol{V}|\boldsymbol{\chi}(s), \boldsymbol{y}(s))|\boldsymbol{\chi}(s), \boldsymbol{v}_c^i\right] = \\ &\sum_{s=1}^{S}\{\int_{\boldsymbol{y}(s)}\left[\log P\left(\boldsymbol{\chi}(s), \boldsymbol{y}(s)|\boldsymbol{v}_c\right) \times \right. \\ &\left. P\left(\boldsymbol{y}(s)|\boldsymbol{\chi}(s), \boldsymbol{v}_c^i\right)\right] \mathrm{d}\boldsymbol{y}(s)\} \end{aligned} \tag{13}$$

其中, $P\left(\boldsymbol{y}(s)|\boldsymbol{\chi}(s), \boldsymbol{v}_c^i\right)$ 为给定说话人 s 的声学特征 $\boldsymbol{\chi}(s)$ 和当前参数 $\boldsymbol{v}_c^i$ 条件下 $\boldsymbol{y}(s)$ 的后验概率, 其

表达式如下:

$$P\left(\boldsymbol{y}(s)|\boldsymbol{\chi}(s),\boldsymbol{v}_c^i\right)\propto P\left(\boldsymbol{\chi}(s)|\boldsymbol{v}_c^i,\boldsymbol{y}(s)\right)P\left(\boldsymbol{y}(s)\right)=\prod_{t=1}^{T}\sum_{c=1}^{C}\left[\gamma_t^s(c)\mathrm{N}(\boldsymbol{x}_t^s|\boldsymbol{m}_c+\boldsymbol{v}_c^i\boldsymbol{y}(s),\boldsymbol{\Sigma}_c)\times \mathrm{N}(\boldsymbol{y}(s)|0,\boldsymbol{I})\right] \tag{14}$$

因此,

$$\begin{aligned}&\log P\left(\boldsymbol{y}(s)|\boldsymbol{\chi}(s),\boldsymbol{v}_c^i\right)\propto\\&\sum_{t=1}^{T}\sum_{c=1}^{C}\gamma_t^s(c)\left\{\boldsymbol{y}^{\mathrm{T}}(s){\boldsymbol{v}_c^i}^{\mathrm{T}}\boldsymbol{\Sigma}_c^{-1}(\boldsymbol{x}_t^s-\boldsymbol{m}_c)-\frac{1}{2}\boldsymbol{y}^{\mathrm{T}}(s){\boldsymbol{v}_c^i}^{\mathrm{T}}\boldsymbol{\Sigma}_c^{-1}\boldsymbol{v}_c^i\boldsymbol{y}(s)\right\}-\frac{1}{2}\boldsymbol{y}^{\mathrm{T}}(s)\boldsymbol{y}(s)=\\&\boldsymbol{y}^{\mathrm{T}}(s)\sum_{c=1}^{C}{\boldsymbol{v}_c^i}^{\mathrm{T}}\boldsymbol{\Sigma}_c^{-1}\sum_{t=1}^{T}\boldsymbol{\gamma}_t^s(c)\left(\boldsymbol{x}_t^s-\boldsymbol{m}_c\right)-\frac{1}{2}\boldsymbol{y}^{\mathrm{T}}(s)\left(\boldsymbol{I}+\sum_{c=1}^{C}\sum_{t=1}^{T}\gamma_t^s(c){\boldsymbol{v}_c^i}^{\mathrm{T}}\boldsymbol{\Sigma}_c^{-1}\boldsymbol{v}_c^i\right)\boldsymbol{y}(s)=\\&-\frac{1}{2}(\boldsymbol{y}(s)-\boldsymbol{a}(s))^{\mathrm{T}}\boldsymbol{l}(s)\left(\boldsymbol{y}(s)-\boldsymbol{a}(s)\right)\end{aligned} \tag{15}$$

其中,

$$\boldsymbol{l}(s)=\boldsymbol{I}+\sum_{c=1}^{C}\sum_{t=1}^{T}\gamma_t^s(c){\boldsymbol{v}_c^i}^{\mathrm{T}}\boldsymbol{\Sigma}_c^{-1}\boldsymbol{v}_c^i \tag{16}$$

$$\boldsymbol{a}(s)=\boldsymbol{l}^{-1}(s)\sum_{c=1}^{C}{\boldsymbol{v}_c^i}^{\mathrm{T}}\boldsymbol{\Sigma}_c^{-1}\sum_{t=1}^{T}\gamma_t^s(c)\left(\boldsymbol{x}_t^s-\boldsymbol{m}_c\right) \tag{17}$$

所以, $\boldsymbol{y}(s)$ 的后验概率服从均值为 $\boldsymbol{a}(s)$, 方差为 $\boldsymbol{l}^{-1}(s)$ 的高斯分布.

以上即为 EM 算法中 E-step, M-step 通过最大化辅助函数 $Q\left(\boldsymbol{v}_c,\boldsymbol{v}_c^i\right)$ 实现, 最终 $\boldsymbol{v}_c$ 的更新公式如式 (18) 所示[4].

$$\boldsymbol{v}_c^{i+1}=\left(\sum_{s=1}^{S}\sum_{t=1}^{T}\gamma_t^s(c)\left(\boldsymbol{x}_t^s-\boldsymbol{m}_c\right)\boldsymbol{a}^{\mathrm{T}}(s)\right)\times\left(\sum_{s=1}^{S}\sum_{t=1}^{T}\gamma_t^s(c)\left(\boldsymbol{l}^{-1}(s)+\boldsymbol{a}(s)\boldsymbol{a}^{\mathrm{T}}(s)\right)\right)^{-1} \tag{18}$$

2.2.2　均值偏移的估计

估计得到 $\boldsymbol{v}_c$ 后, $\boldsymbol{y}(s)$ 的估计可以通过如下优化问题实现, 即

$$\boldsymbol{y}(s)=\arg\max P\left(\boldsymbol{y}(s)|\boldsymbol{\chi}(s),\boldsymbol{v}_c\right) \tag{19}$$

由式 (15) 可知, 该优化问题的解为

$$\boldsymbol{y}(s)=\boldsymbol{l}^{-1}(s)\sum_{c=1}^{C}\boldsymbol{v}_c^{\mathrm{T}}\boldsymbol{\Sigma}_c^{-1}\sum_{t=1}^{T}\gamma_t^s(c)\left(\boldsymbol{x}_t^s-\boldsymbol{m}_c\right) \tag{20}$$

因此, 均值偏移 $\Delta\boldsymbol{m}_c(s)$ 为

$$\Delta\boldsymbol{m}_c(s)=\boldsymbol{v}_c\boldsymbol{y}(s) \tag{21}$$

2.2.3　说话人无关的声学特征的估计

说话人无关的声学特征的估计为

$$\hat{\boldsymbol{z}}_t=\boldsymbol{x}_t^s-\sum_{c=1}^{C}{\gamma'}_t^s(c)\boldsymbol{v}_c\boldsymbol{y}(s) \tag{22}$$

其中, ${\gamma'}_t^s(c)$ 说话人 s 的第 t 帧特征 $\boldsymbol{x}_t^s$ 在模型 $\lambda_{\boldsymbol{x}}$ 第 c 个混元上的状态占有概率, 即

$${\gamma'}_t^s(c)=\frac{w_c\mathrm{N}(\boldsymbol{x}_t^s|\boldsymbol{m}_c+\boldsymbol{v}_c\boldsymbol{y}(s),\boldsymbol{\Sigma}_c)}{\sum_{l=1}^{C}w_l\mathrm{N}(\boldsymbol{x}_t^s|\boldsymbol{m}_l+\boldsymbol{v}_l\boldsymbol{y}(s),\boldsymbol{\Sigma}_l)} \tag{23}$$

2.3　算法比较

本征音自适应算法假设说话人 s 相关的模型参数超矢量 $\boldsymbol{m}(s)$ 是在一个 SI 模型参数超矢量 $\boldsymbol{m}$ 上叠加一定的偏移得到的, 这个偏移代表的就是该说话人的个性信息. 这些信息可以认为存在于一个低维流形中, 称该流形为说话人子空间. 即

$$\boldsymbol{m}(s)=\boldsymbol{m}+\boldsymbol{V}\boldsymbol{y}(s) \tag{24}$$

其中, $\boldsymbol{m}=[\boldsymbol{m}_1^{\mathrm{T}},\boldsymbol{m}_2^{\mathrm{T}},\cdots,\boldsymbol{m}_C^{\mathrm{T}}]^{\mathrm{T}}$, $\boldsymbol{m}(s)=[\boldsymbol{m}_1(s)^{\mathrm{T}},\boldsymbol{m}_2(s)^{\mathrm{T}},\cdots,\boldsymbol{m}_C(s)^{\mathrm{T}}]^{\mathrm{T}}$, $\boldsymbol{V}=[\boldsymbol{v}_1^{\mathrm{T}},\boldsymbol{v}_2^{\mathrm{T}},\cdots,\boldsymbol{v}_C^{\mathrm{T}}]^{\mathrm{T}}$ 为说话人子空间基矩阵, 维数为 $CF\times R$, $\boldsymbol{y}(s)$ 为说话人因子 (Speaker factor), 且 $\boldsymbol{y}(s)$ 服从标准正态分布, 即 $\boldsymbol{y}(s)\sim\mathrm{N}\left(0,\boldsymbol{I}\right)$.

可以看出 EV 算法与 FEV 算法对均值偏移超矢量的假设是一致的, 同时这两种算法对 $\boldsymbol{V}$ 和 $\boldsymbol{y}(s)$ 的估计方法也相同.

假设 $\boldsymbol{V}$ 和 $\boldsymbol{y}(s)$ 已知, 则说话人 s 的声学特征 $\boldsymbol{\chi}(s)$ 在 EV 算法自适应得到第 s 个说话人 SD 模型上的对数似然概率为

$$\begin{aligned}&P\left(\boldsymbol{\chi}(s)|\boldsymbol{m}(s)\right)=\\&-\frac{1}{2}\sum_{t=1}^{T}\sum_{c=1}^{C}\gamma_t^s(c)(\boldsymbol{x}_t^s-\boldsymbol{m}_c(s))^{\mathrm{T}}\boldsymbol{\Sigma}_c^{-1}\left(\boldsymbol{x}_t^s-\boldsymbol{m}_c(s)\right)=\\&-\frac{1}{2}\sum_{t=1}^{T}\sum_{c=1}^{C}[\gamma_t^s(c)(\boldsymbol{x}_t^s-\boldsymbol{m}_c-\boldsymbol{v}_c\boldsymbol{y}(s))^{\mathrm{T}}\times\\&\boldsymbol{\Sigma}_c^{-1}(\boldsymbol{x}_t^s-\boldsymbol{m}_c-\boldsymbol{v}_c\boldsymbol{y}(s))]\end{aligned} \tag{25}$$

在 FEV 算法中, 说话人无关的声学特征 $\hat{\boldsymbol{z}}(s)$ 在 SI 声学模型上的对数似然概率为

$$\begin{aligned} &P\left(\hat{\boldsymbol{z}}(s)\,|\boldsymbol{m}\right) = \\ &\quad -\frac{1}{2}\sum_{t=1}^{T}\sum_{c=1}^{C}[\gamma_t^s(c)(\boldsymbol{x}_t^s-\boldsymbol{m}_c-\Delta\bar{\boldsymbol{m}}(s))^{\mathrm{T}}\times \\ &\quad \boldsymbol{\Sigma}_c^{-1}(\boldsymbol{x}_t^s-\boldsymbol{m}_c-\Delta\bar{\boldsymbol{m}}(s))] = \\ &\quad P\left(\boldsymbol{\chi}(s)\,|\hat{\boldsymbol{m}}(s)\right) \end{aligned} \tag{26}$$

其中,

$$\hat{\boldsymbol{m}}(s)=\boldsymbol{m}_c+[\Delta\bar{\boldsymbol{m}}_1(s),\Delta\bar{\boldsymbol{m}}_2(s),\cdots,\Delta\bar{\boldsymbol{m}}_C(s)]^{\mathrm{T}} \tag{27}$$

$$\Delta\bar{\boldsymbol{m}}_c(s)=\sum_{l=1}^{C}\gamma'^{s}_{t}(c)\boldsymbol{v}_c\boldsymbol{y}(s) \tag{28}$$

从式 (26) 可以看出, 说话人无关的声学特征 $\hat{\boldsymbol{z}}(s)$ 在 SI 声学模型 $\boldsymbol{m}$ 上的对数似然概率等于说话人 s 的声学特征 $\boldsymbol{\chi}(s)$ 在自适应得到第 s 个说话人 SD 模型 $\hat{\boldsymbol{m}}(s)$ 上的对数似然概率, 因此 FEV 算法可等价于在一种新的模型层上的自适应算法.

但与模型层上自适应算法不同的是, 特征层的自适应方法不需要调整模型参数, 因此在解码的过程中更具实时性, 这在实际应用中是相当重要的.

2.4 说话人自适应训练

在特征参数自适应技术中, 训练声学模型的训练语料含有多个训练说话人的语音数据, 因此训练所得到的 SI 声学模型实际上是一个与训练说话人相关的 "多说话人" 声学模型, 并不是真正意义上的说话人无关模型. 如果同样对训练语音的声学特征规整, 去除说话人的个性信息之后再进行声学模型训练, 则训练得到的声学模型更具有说话人无关性, 称这样的声学模型训练方法为 "说话人自适应训练", 其声学模型为 SAT 声学模型.

SAT 方法配合 FEV 自适应方法能够进一步提升系统性能. 图 1 给出了进行基于特征空间本征音的说话人自适应训练的完整流程.

从图 1 可以看出, SAT 首先用训练集中所有说话人的语料训练得到一个 SI 声学模型; 然后在这个 SI 模型基础上, 对训练集中各说话人的训练语料采用 FEV 自适应算法, 去除说话人相关性, 得到说话人无关的声学特征; 最后, 利用这些说话人无关的声学特征进行模型训练, 得到 SAT 声学模型. 重复进行 FEV 自适应和声学模型训练, 最终得到一个稳健的 SAT 声学模型.

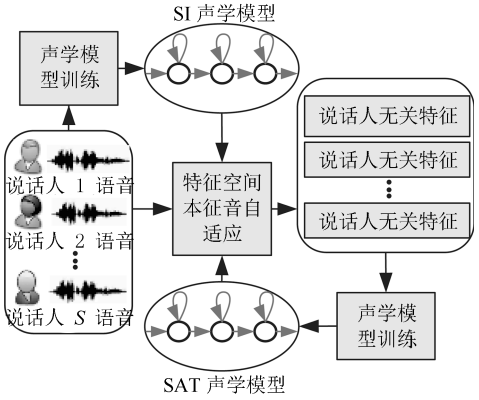


图 1 说话人自适应训练示意图

Fig. 1 Illustration of the speaker adaptive training procedure

3 实验

为了验证 FEV 算法的性能, 本文使用 Kaldi 语音识别工具箱[18] 搭建基于 HMM-GMM 的中文连续语音识别系统, 进行了说话人自适应相关的实验.

3.1 实验数据

2001 年, 微软亚洲研究院发布了用于搭建、测试中文连续语音识别系统的中文语料库[19]. 语料库训练集由 100 个男性说话人的语音构成, 每个说话人的语音约 200 段, 每段语音时长约 5 秒, 共有 19 688 段、总时长约为 33 小时的语音数据. 测试集由 25 个说话人的语音构成, 每个说话人的语音各 20 段, 每段语音时长约 5 秒. 训练集中的说话人均为北京某广播传媒学校的学生, 年龄分布覆盖 18 到 40 岁, 平均年龄为 25 岁, 说话人籍贯分布全国 26 省, 录音时用标准普通话发音; 测试集中的说话人是为来自北京的志愿者, 北京方言口音. 语料库共包含 1 677 个汉字, 用汉语有调音节 (184 个) 进行标注, 其中包含 27 个声母和 157 个有调韵母. 语音内容均来源于报纸新闻, 均衡覆盖所有汉语有调音节.

3.2 评测指标

连续语音识别结果一般为词序列, 采用动态规划算法将识别结果与正确的标注序列对齐后进行比较, 错误的类型可以分为三类: 插入错误即在两个相邻的标注之间插入了其他词, 删除错误即在识别结果中找不到与某个标注对应的词, 替代错误即识别得到的词与对应标注不相符.

假设某个测试集中标注的总个数为 N, 插入错误个数为 I, 删除错误个数为 D, 替代错误个数为 R, 则 WER 的定义如下:

$$\mathrm{WER}=\frac{I+D+R}{N}\times 100\,\% \tag{29}$$

该指标越低则系统性能越好.

3.3 实验设置

实验中, 特征参数采用原始的 13 维 MFCC 特征及其一阶、二阶差分参数, 总的矢量维数为 39 维, 帧长 25 ms, 帧移 10 ms. 使用 CMN 对 MFCC 参数进行规整.

使用 Kaldi 工具箱搭建 SI 基线系统, 首先以有调音节为建模单元训练一个单音子模型, 然后依照发音词典进行上下文扩展, 最终将单音子扩展为 295 180 个三音子, 其中 95 534 个三音子在训练语料中出现. 用三状态 (三个发射状态)、自左向右无跨越的 HMM 对每一个三音子有调音节进行建模, 用五状态 (一个起始状态、一个终止状态、三个发射状态)、自左向右无跨越的 HMM 对静音进行建模. 对这些三音子模型采用决策树进行状态聚类后, 系统最终包含 1 885 个不同的上下文相关的状态. 用 GMM 对各状态进行建模, 根据数据量的不同, 各状态 GMM 的混元数不同. 最终, 声学模型中包含 5 015 个高斯混元. 使用训练语料的标注文本训练得到一个包含 126 000 个二元文法和 40 000 个三元文法的语言模型. 使用 Kaldi 工具箱中的 WSFT 解码器进行解码识别. 最终在测试集上, SI 基线系统的 WER 为 25.23 %.

3.4 实验结果

3.4.1 特征空间本征音自适应实验

为了比较本文算法的自适应性能, 在测试数据上进行了无监督说话人自适应实验. 实验中, 我们构造了两套基线系统和一套本文算法的系统.

1) SI: 直接利用说话人无关的系统进行识别, 不采用说话人自适应技术;

2) FMLLR: 采用特征空间最大似然线性回归自适应技术;

3) EV: 采用本征音自适应技术, 子空间维数 R 从 20 变化到 100;

4) FEV: 本文提出的特征空间本征音技术, 子空间维数 R 从 20 变化到 100.

为了验证 FEV 算法在不同长度的自适应数据量下的性能, 分别对 1 句、2 句、5 句、10 句和 20 句的自适应数据进行了实验.

从表 1 的实验结果可以看出, FEV 算法在不同数据量下都能提升系统识别率, 因此该算法是有效的. 从 EV 算法和 FEV 算法的对比实验结果可以看出, 在各种数据量下 EV 算法的性能都优于 FEV 算法. 对比 FMLLR 算法与 FEV 算法的实验结果, 可以看出 FEV 算法在自适应数据量极少 (只有 1 句) 的情况下, 性能明显优于 FMLLR 算法, 这是由于 FMLLR 算法在自适应过程中需要估计的参数较多, 自适应数据不足时难以得到稳健的估计值. 但是自适应数据量增加后, FMLLR 算法性能优于 FEV 算法. 同时, 观测不同自适应数据量下 FEV 算法的实验结果, 可以看出, 随着自适应数据量的增加, FEV 算法的性能迅速趋于饱和. 因此, FEV 算法适用于自适应数据量极少的情况.

表 1　各自适应算法在不同自适应数据量下实验结果

Table 1　Experimental results of adaptive algorithm with different adaptive data volume

自适应方法		自适应句数				
		1	2	5	10	20
FMLLR		25.03	23.54	**22.65**	**22.24**	**21.97**
EV	$R=20$	23.85	23.49	23.31	23.02	22.92
	$R=40$	**23.70**	**23.43**	23.12	22.72	22.71
	$R=60$	23.81	23.66	23.09	22.67	22.60
	$R=80$	24.14	23.80	**23.05**	**22.64**	22.56
	$R=100$	24.27	24.05	23.14	22.78	**22.44**
FEV	$R=20$	24.16	23.97	23.82	23.68	23.52
	$R=40$	23.91	23.83	23.79	23.59	23.31
	$R=60$	24.22	24.01	23.77	23.52	23.27
	$R=80$	24.33	24.04	23.76	23.40	23.15
	$R=100$	24.43	24.19	23.85	23.51	23.27

3.4.2 说话人自适应训练实验

为了提升 FEV 算法的性能, 将 FEV 算法与第 2.4 节中给出的算法结合进行说话人自适应训练, 并与基于 FMLLR 的自适应训练方法进行比较. 表 2 给出了相应的实验结果.

表 2　说话人自适应训练在不同自适应数据量下实验结果

Table 2　Experimental results of the speaker adaptive algorithm with different adaptive data volume

自适应方法		自适应句数				
		1	2	5	10	20
EV (最优 R)		23.70	23.43	23.05	22.64	22.44
SAT + FMLLR		30.12	23.25	**21.85**	**21.70**	**21.51**
SAT + FEV	$R=20$	23.35	23.21	23.14	23.01	22.85
	$R=40$	**23.23**	**23.17**	23.09	22.89	22.74
	$R=60$	23.36	23.24	23.01	22.76	22.59
	$R=80$	23.59	23.27	22.96	22.63	22.41
	$R=100$	23.67	23.32	23.14	22.87	22.65

从表 2 中可以看出, 使用说话人自适应训练后, FEV 自适应方法性能有了明显提升, 且优于本征音自适应方法. 同时, 在自适应数据量较少的情况下

(1∼2 句), 基于 FEV 的说话人自适应训练性能明显优于基于 FMLLR 的说话人自适应训练. 需要注意的是, 当自适应数据量极少时 (只有 1 句), 基于 FMLLR 的说话人自适应训练不但没有起到自适应应有的效果, 反而降低了系统性能, 文献 [6] 也指出了这个问题.

3.4.3 解码速度对比实验

从式 (22) 可以看出, 由于混元数 C 较大, 因此估计说话人无关的声学特征效率并不高. 但是, 对于 $\gamma'^s_t(c)$ 而言, 一般只有少数几个状态占有率较大, 其他的几乎可以忽略, 因此为了提高特征变换的效率, 可以对式 (22) 所示估计式进行简化, 采用 TOP-C' 策略, 即

$$\hat{\boldsymbol{y}}_t = \boldsymbol{x}^s_t - \sum_{c=1}^{C'} \gamma'^s_t(c)\boldsymbol{v}_c\boldsymbol{y}(s) \tag{30}$$

其中, C' 为一个固定的整数, 且 $C' \ll C$. 为了对比 FEV 算法与 EV 算法对解码速度的影响, 我们进行了如下实验, 其中 SI 基线系统的 WER 为 25.23 %, 当系统不进行说话人自适应, 直接识别时, 识别一句语音的平均耗时为 1.696 秒.

1) EV: 令说话人子空间维数 $R = 40$, 分别进行自适应数据量为 1 句、2 句、5 句、10 句和 20 句的本征音说话人自适应, 统计系统的 WER 和平均耗时. 选择两种平均耗时进行实验, 其中 TIME1 为识别一句语音的时间, 包括估计说话人因子用时、调整声学模型用时和识别用时; TIME2 为模型自适应时间, 包括估计说话人因子和调整声学模型的平均时间 (声学模型调整需要先估计说话人因子).

2) SAT + FEV: 令说话人子空间维数 $R = 40$, 分别进行自适应数据量为 1 句、2 句、5 句、10 句和 20 句的特征空间本征音说话人自适应, 且声学模型为 SAT 声学模型. 统计系统 WER 和平均耗时. 同样采用两个时间, 一个是平均识别一句语音的耗时 (TIME1) 包括估计说话人因子用时、特征变换用时和识别用时; 另外一个是特征变换平均时间 (TIME2), 包括特征变换所需的说话人因子计算用时和变换用时. 令 $C' = 2$、$C' = 10$ 和 $C' = C$ 分别进行实验.

从表 3 中可以看出, 进行说话人自适应之后, 降低了系统的识别效率. 而且自适应语料越短, 效率越低, 这是由于自适应语料越短, 对模型参数或特征参数调整的越频繁, 因此效率越低. 同时, 当进行 FEV 自适应时, 不对算法进行简化 ($C' = C$), 系统效率与基于 EV 自适应的系统的效率相近, 甚至在自适应语料较多时 (10∼20 句), 低于基于 EV 自适应的系统. 对 FEV 简化后, 当 $C' = 10$ 时, 系统的性能略微有所下降, 但是明显提升了自适应语料较少 (1∼5 句) 时系统的效率. 当 $C' = 2$ 时, 系统的 WER 进一步变差, 但是效率基本保持不变.

为了更好地比较两种算法的性能时, 采用 NIST 公布的开源工具包 SCTK[1] 进行显著性水平测试 (Significance test), 以检验识别结果之间的差异在统计上是否显著. 三种显著性测试 (MP 测试、SI 测试及 WI 测试) 结果均表明在 5 % 的显著性水平之下, 在 1 句话自适应语料时, 两种方法的的实验结果之间差异是显著的, 表明 SAT+FEV 算法优于 EV 算法; 而在 2 句话自适应语料时, SAT+FEV 算法的 MP 测试相对更优一些, 而其他两种测试显示其差异是不显著的; 在自适应语料为 5 句至 20 句时, 三种测试方法的性能均不显著, 这就说明二者的性

表 3 解码速度对比 ($R = 40$)

Table 3 Comparison of decoding speed

自适应方法		自适应句数 1 WER	1 TIME1 (TIME2)	2 WER	2 TIME1 (TIME2)	5 WER	5 TIME1 (TIME2)	10 WER	10 TIME1 (TIME2)	20 WER	20 TIME1 (TIME2)
EV		23.70	3.326 (1.630)	23.43	2.511 (0.815)	23.12	2.022 (0.326)	**22.72**	1.859 (0.163)	**22.71**	1.777 (0.081)
SAT	$C' = 2$	23.49	1.751 (0.055)	23.38	1.725 (0.029)	23.27	1.709 (0.013)	23.04	1.704 (0.008)	22.83	1.701 (0.005)
+	$C' = 10$	23.31	**1.751 (0.055)**	23.24	**1.725 (0.029)**	23.15	**1.709 (0.013)**	22.91	**1.704 (0.008)**	22.79	**1.701 (0.005)**
FEV	$C' = C$	**23.23**	3.247 (1.551)	**23.17**	3.221 (1.525)	**23.09**	3.205 (1.509)	22.89	3.200 (1.504)	22.74	3.198 (1.502)

[1] ftp://jaguar.ncsl.nist.gov/pub/sctk-2.4.0-20091110-0958.tar.bz2

能从统计上讲几乎是相同的.

因此, 简化后的 FEV 自适应算法, 在自适应数据较少的情况下, 不仅提升了系统的性能, 而且提高了系统的识别效率, 这在实际应用中是十分有意义的.

4 结论

本文提出了一种特征层的说话人自适应方法, 该方法借鉴 RATZ 算法, 采用 GMM 对特征空间中的说话人信息进行建模, 同时充分利用估计参数之间的相关性, 减少估计参数的数量, 在对特征空间精确建模的同时, 降低了算法对自适应数据量的需求. 由于该方法在参数估计时的理论基础与实现步骤与本征音自适应相类似, 因此称之为特征空间本征音自适应方法. 在基于微软语料库的中文连续语音识别实验中, FEV 在自适应数据量极少时仍能取得较好的性能, 同时配合说话人自适应训练能够进一步降低词错误率, 而将本方法与正则化本征音自适应、正交拉普拉斯自适应相结合, 可以取得更加优异的性能.

由于 FEV 算法估计的参数数量较少, 当自适应数据量增多时, 其性能将逐渐趋于饱和, 此时应采用 MLLR 或 MAP 等算法, 以进一步提高其自适应性能. 近年来, 深度神经网络正成为连续语音识别声学建模的主流技术, 本文方法也可以与深度神经网络框架相结合, 如针对 BN (Bottle neck) 特征进行自适应, 或者将经过 FEV 算法自适应过的特征矢量作为 DNN 的输入特征等, 这将是我们进一步的重点研究方向之一.

References

1 Teng W X, Gravier G, Bimbot F, Soufflet F. Speaker adaptation by variable reference model subspace and application to large vocabulary speech recognition. In: Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing. Taiwan, China: IEEE, 2009. 4381−4384

2 Zhang W L, Zhang W Q, Li B C, Qu D, Johnson M T. Bayesian speaker adaptation based on a new hierarchical probabilistic model. *IEEE Transactions on Audio, Speech, and Language Processing*, 2012, **20**(7): 2002−2015

3 Zhang W L, Qu D, Zhang W Q, Li B C. Rapid speaker adaptation using compressive sensing. *Speech Communication*, 2013, **55**(10): 950−963

4 Kenny P, Boulianne G, Dumouchel P. Eigenvoice modeling with sparse training data. *IEEE Transactions on Speech and Audio Processing*, 2005, **13**(3): 345−354

5 Varadarajan B, Povey D, Chu S M. Quick FMLLR for speaker adaptation in speech recognition. In: Proceedings of the 2008 International Conference on Acoustics, Speech, and Signal Processing. Las Vegas, Nevada, USA: IEEE, 2008. 4297−4300

6 Ghoshal A, Povey D, Agarwal M, Akyazi P, Burget L, Kai Feng, Glembek O, Goel N , Karafiat M, Rastrow A, Rose R C, Schwarz P, Thomas S. A novel estimation of feature-space MLLR for full-covariance models. In: Proceedings of the 2010 IEEE International Conference on Acoustics, Speech, and Signal Processing. Dallas, TX, USA: IEEE, 2010. 4310−4313

7 Rath S P, Povey D, Vesely K, Cernocky J. Improved feature processing for Deep Neural Networks. In: Proceedings of the 14th Annual Conference of the International Speech Communication Association, Lyon, France: ISCA, 2013. 109−113

8 Rath S P, Burget L, Karafiát M, Glembek O, Cernocký J. A region-specific feature-space transformation for speaker adaptation and singularity analysis of Jacobian matrix. In: Proceedings of the 2013 Annual Conference of International Speech Communication Association. Lyon, France: ISCA, 2013. 1228−1232

9 Ghalehjegh S H, Rose R C. Two-stage speaker adaptation in subspace gaussian mixture models. In: Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing. Florence, Italy: IEEE, 2014. 6324−6328

10 Chen S, Kingsbury B, Mangu L, Povey D, Saon G, Soltau H, Zweig G. Advances in speech transcription at IBM under the DARPA EARS program. *IEEE Transactions on Audio, Speech, and Language Processing*, 2006, **14**(5): 1596−1608

11 Saon G, Chien J T. Large-vocabulary continuous speech recognition systems: a look at some recent advances. *IEEE Signal Processing Magazine*, 2012, **29**(6): 18−33

12 Joshi V, Prasad V N, Umesh S. Modified cepstral mean normalization-transforming to utterance specific non-zero mean. In: Proceedings of the 2013 Annual Conference of International Speech Communication Association. Lyon, France: ISCA, 2013. 881−885

13 Buera L, Lleida E, Miguel A, Ortega A, Saz O. Cepstral vector normalization based on stereo data for robust speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 2007, **15**(3): 1098−1113

14 Droppo J, Deng L, Acero A. Uncertainty decoding with SPLICE for noise robust speech recognition. In: Proceedings of the 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing. Orlando, FL, USA: IEEE, 2002. I-57−I-60

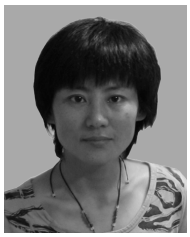
15 Moreno P J, Raj B, Stern R M. Data-driven environmental compensation for speech recognition: a unified approach. *Speech Communication*, 1998, **24**(4): 267−285

16 Wang Y Q, Gales M J F. Model-based approaches to adaptive training in reverberant environments. In: Proceedings of the 2012 Annual Conference of International Speech Communication Association. Portland, Oregon: ISCA, 2012. 959−963

17 Ochiai T, Matsuda S, Lu X G, Hori C, Katagiri S. Speaker adaptive training using deep neural networks. In: Proceedings of the 2014 IEEE International Conference on Acoustics, Speech, and Signal Processing. Florence, Italy: IEEE, 2014. 6349−6353

18 Povey D, Ghoshal A, Boulianne G, Burget L, Glembek O, Goel N, Hannemann M, Motlicek P, Qian YM, Schwarz P, Silovsky J, Stemmer G, Vesely K. The Kaldi speech recognition toolkit. In: Proceedings of the 2011 IEEE Workshop on Automatic Speech Recognition and Understanding. Hawaii, USA: IEEE, 2011.

19 Eric C, Zhou J L, Shi Y, Huang C. Speech lab in a box: a Mandarin speech toolbox to jumpstart speech related research. In: Proceedings of the 2001 European Conference on Speech Communication and Technology. Scandinavia, Aalborg, Denmark: ISCA, 2001. 2799−2782



屈 丹 中国人民解放军信息工程大学信息系统工程学院副教授. 2005 年获解放军信息工程大学博士学位. 主要研究方向为语音信号处理, 模式识别, 自然语言处理和机器学习. 本文通信作者.
E-mail: qudanqudan@sina.com
(**QU Dan** Associate professor at the Institute of Information Systems Engineering, PLA Information Engineering University. She received her Ph. D. degree from PLA Information Engineering University in 2005. Her research interest covers speech signal processing, pattern recognition, natural language processing, and machine learning. Corresponding author of this paper.)

杨绪魁 中国人民解放军信息工程大学信息系统工程学院博士研究生. 主要研究方向为语音信号处理, 语音识别.
E-mail: gzyangxk@163.com
(**YANG Xu-Kui** Ph. D. candidate at the Institute of Information Systems Engineering, PLA Information Engineering University. His research interest covers speech processing and speech recognition.)

张文林 中国人民解放军信息工程大学信息系统工程学院讲师. 2013 年获解放军信息工程大学博士学位. 主要研究方向为语音信号处理, 语音识别, 机器学习.
E-mail: zwlin_2004@163.com
(**ZHANG Wen-Lin** Lecturer at the Institute of Information Systems Engineering, PLA Information Engineering University. He received his Ph. D. degree from PLA Information Engineering University in 2013. His research interest covers speech signal processing, speech recognition, and machine learning.)