一个基于 LB/IHMM 的高性能汉语连接数字语音识别系统*

王成友 汤叔祺 梁甸农

(国防科技大学电子技术系 长沙 410073)

摘 要 本文使用一种 IHMM 的简化训练算法, 讨论了一个基于 Level Building 搜索算法, 参考模式基于 IHMM 模型的高性能的汉语连接数字语音识别系统。文中详细描述了系统的特征提取、IHMM 模型、训练过程和识别过程, 以及系统用于特定人、多说话人和不认人语音识别的实验, 并且对实验结果进行了分析和讨论。

关键词 语音识别, Level Building, IHMM, 连接数字

分类号 TP391. 42

A High Performance Connected Digit Speech Recognition System Based LB/IHMM

Wang Chengyou Tang Shuqi Liang Diannong

(Department of Electronic Technology, NUDT, Changsha, 410073)

Abstract By using an IHMM easy-estimation training algorithm this paper discusses Chinese connected digit speech recognition system with efficient performance, and based on Level Building searching algorithm, whose reference patterns use IHMM mode. The paper describes the theory of the system. It includes its feature-extracting, IHMM mode, training procession, recognition procession, and the experiments that are used for speaker trained, multi-speaker and speaker independent mode. The results of the experiment are analyzed and discussed.

Key words speech recognition, level building, IHMM, connected digit

连接数字串的语音识别技术是实现象声音电话拨号、自动信贷卡声控人口等一些应用的关键技术。在前些年, 国外有着这方面比较成功的系统 (如: 文献 [1])。国内清华大学、中科院声学所在这方面都有过一些研究。

本文基于 LB/HMM, 建立了一个高性能的连接数字语音识别系统, 其中使用了一种独特的简化训练方法, 参考模式模型基于精度较高的 IHMM, 大大提高了训练速度和识别精度。文章在第二部分详细阐述这个系统结构、工作原理、算法过程, 第三部分讨论了系统用于特定人、多说话人、不认人三种情况的实验过程和实验结果, 并对实验结果进行了分析和讨论。

1 LB/HMM 语音识别系统原理

1.1 识别过程

整个 LB/HMM 语音识别系统识别过程框图显示在图 1, 其中主要几个步骤算法过程及其原理如下。

(1) 端点检测

本系统采用的是多门限率检测方法 (MCR)[2]。

* 1997 年 7 月 5 日收稿
第一作者: 王成友, 男, 1966 年生, 博士

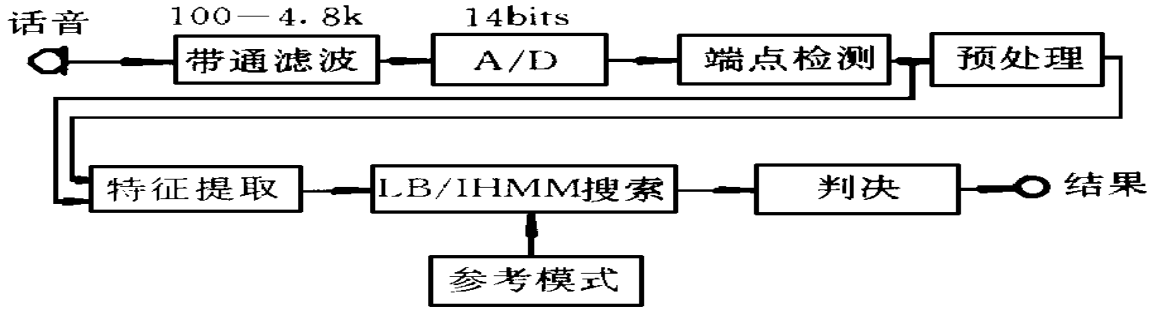

图 1 识别过程

(2) 预处理

预处理用冲激响应为 1-0.95Z^{-1}的滤波。

(3) 特征提取

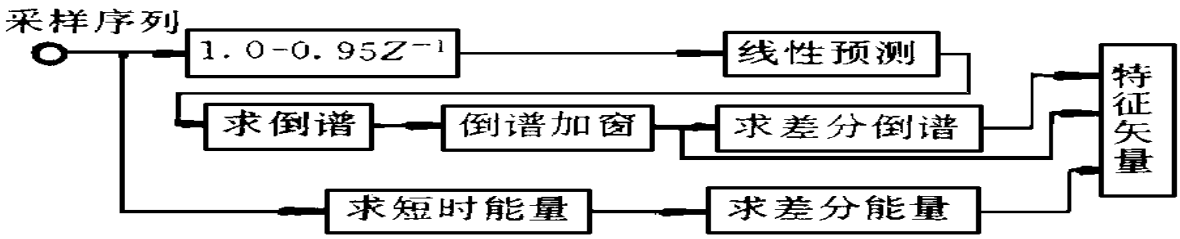

图 2 特征矢量计算过程

特征构成及其提取算法过程如图 2, 其中:

· 倒谱: LPC 倒谱。

· 差分过程: 设语音数据第 n 帧的倒谱系数为 $c_n(m)$。

$$\Delta c_n(m) = \left[\sum_{k=-K}^{K} k c_{n-k}(m)\right] G \qquad m = 1, 2, \cdots, 12$$

经验表明 K 取 2, G 取 0.375 是比较合适的。

· 加权窗函数:

$$w(m) = 1 + \frac{Q}{2}\sin\left(\frac{\pi m}{Q}\right)$$

其中 Q 是倒谱系数个数, 这里 $Q=12$。

(4) LB/IHMM 搜索

a. IHMM 模型

一个 HMM 模型一般具有三项特征参数:

模型各状态的初始概率 $\pi = [\pi]$, $i = 1, \cdots, N$

各状态之间的转移概率 $A = [a_{ij}]$, $i, j = 1, \cdots, N$

各状态观测概率 $B = [b_j(o)]$, $j = 1, \cdots, N$

记作 $\lambda(\pi, A, B)$, 其中: N 为状态数, $o = o_1 o_2 \cdots o_T$, T 为观察序列长度, 对应一个观察序列 $o_1 o_2 \cdots o_T$。

IHMM 模型中, a_{ij}变为 $a_{ij}(\tau)$, 表示 i 状态持续 τ时间转移到 j 状态的概率。

b. LB/IHMM 搜索算法

LB 是 Level-Building 的缩写。其主要思想是: 在第一层上对所有一词假设进行搜索, 算法仍采用 Viterbi 算法, 搜索终点包括 $t=1$ 至 $t=T$ 之间的任一点。在第一层搜索完后, 在每一终点上可以找到一个 $\delta_t(N, \tau)$ (V 是词条编号, N 是最后一个状态编号, τ是时长, $\tau \geq 1$) 的最大值及相应的词条编号。然后, 以这些点为起点构筑第二个搜索层。如果一个句子限定最多有 L 个词, 则共需构造 L 个搜

索层。具体如下:

设1为层数, 词汇表大小为 P, 词汇表可以表示为 $\{w_{[1]}, w_{[2]}, \cdots, w_{[p]}\}$, 观测矢量为 $\{o_1, o_2, \cdots, o_T\}$, q 为词条编号。

第一层计算:

(1) 令 $q=1$

(2) 初始化

$$\begin{cases} \delta_1(1,1) = \log b_1^q(o_1) \\ \delta_1(1,\tau_1) = -\infty & \tau_1 \neq 1 \\ \delta_1(i,\tau_1) = -\infty & i \neq 1 \end{cases}$$

(3) 迭代计算

$$\delta_t(j,\tau_j) = \begin{cases} \max\limits_{\tau_{j-1}}[\delta_{t-1}(j-1,\tau_{j-1}) + \log a_{j-1,j}(\tau_{j-1})] + \log b_j(o_t), & \tau_j = 1 \\ \delta_{t-1}(j,\tau_j-1) + \log a_{j,j}(\tau_j-1) + \log b_j(o_t), & \tau_j > 1 \end{cases}$$

$$1 \le j \le N, \quad 2 \le t \le T$$

(4) 终点

$$p(l,t,q) = \max_{\tau_N} \delta_t^q(N,\tau_N), \qquad B(l,t,q) = 0$$

(5) 如果 $q<p$, $q=q+1$, 回到第二步, 否则

$$\bar{p}(l,t) = \max_q[p(l,t,q)], \qquad \bar{B}(l,t) = B[l,t,\arg\max_q p(l,t,q)], \qquad \bar{w}(l,t) = \arg\max_q[p(l,t,q)]$$

其中 $\bar{p}$ 是输出层最大概率, B 是输出层回溯点, $\bar{w}$ 输出层词条指示。其它层, 由于是从上一层开始, 因而初始计算过程与第一层并不一致, 但其它部分基本相同。下面仅介绍不同之处:

$$\delta_1(i,\tau_i) = -\infty$$

$$\delta_t(1,1) = \bar{p}(l-1,t-1) + \log b_1^q(o_t) \qquad 2 \le t \le T$$

$$\alpha_t(1,1) = t-1$$

α 是回溯函数, 其它 α 可如下计算:

$$\alpha(i,\tau_i) = \begin{cases} \alpha_{t-1}[i-1, \arg\max\limits_{\tau_{i-1}} \delta_t(i-1,\tau_{i-1})] & \tau_i = 1, i \neq 1 \\ \alpha_{t-1}(i,\tau_t-1) & \tau_t \neq 1 \end{cases}$$

在每一层的最后, 有

$$p(l,t,q) = \max_{\tau_N} \delta_t(N,\tau_N), \qquad B(l,t,q) = \alpha[N, \arg\max_{\tau_N} \delta_t(N,\tau_N)]$$

令 n_w 为语音串中的词条数目, 所有层都计算完后, 则有 $n_w = \arg\max_l \bar{p}(l, T)$, 并且第 n_w 个词条为 $\bar{w}(n_w, T)$, 第 n_w-1 个词条为 $\bar{w}(n_w-1, \bar{B}(n_w, T))$, 依次反推, 则所有词条均可找出来。

1.2 训练过程

如图3, 训练过程需要两个训练库, 一个是孤立词训练库, 一个是连接词训练库。整个训练过程的初始模式参数是由孤立词训练库训练得来的, 然后根据得到的参考模式、利用LB词条切割算法, 将连接词训练库中的连接词串分割成单个的词条, 然后用IHMMEE算法对分割后的词条进行训练, 得到的参数与上一次的参数比较。若不收敛, 则以新的参数用LB算法分割词串, 再用IHMMEE重新训练。如此反复直到收敛。其中IHMMEE算法原理见下节。

1.3 IHMMEE算法

本系统观测密度函数是混合高斯连续密度函数:

$$b_j(o) = \sum c_{jm} \mathrm{N}(o, \mu_{jm}, U_{jm})$$

用 N 表示模型中的状态数目, M 表示高斯密度混合数量, D 表示观测矢量维数, $C=[c_{jm}]$ 表示混合增益矩阵, $\mu=[\mu_{jmd}]$ 表示混合分量均值, $U=[U_{jmd}]$ 表示混合分量方差矩阵。

如图4, IHMMEE算法首先是模型初始化, 它包括设定 N、M 值以及 A、μ、U、C 的初始值。继

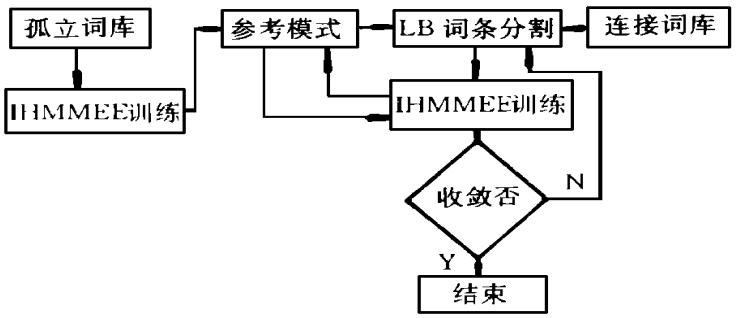

图 3 训练过程

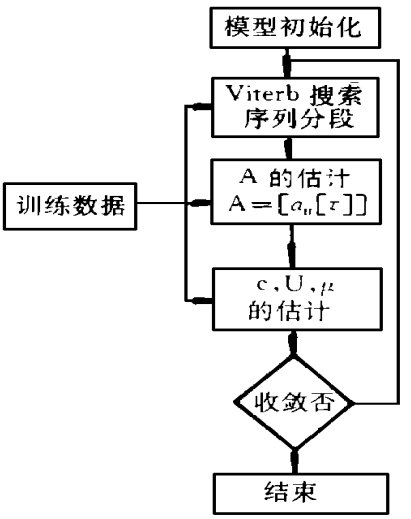

图 4 IHMM EE 算法框图

而是 Viterbi 状态序列分段, 依据 A、μ、U、C 的初始值或者是上一次估计值, 通过 Viterbi 搜索算法找到最佳状态序列; 然后每段对应一个状态, 将观测序列分成 N 段, 直到将所有训练序列总共分成 N 堆数据。下一步是根据分段结果估计矩阵 A, 通过聚类算法将每一堆数据分成 M 类, 再依此聚类结果估计每堆的 μ、U、C 的值。

(1) 模型初始化

除 N、M 是已经选择好的值外, A、μ、U、C 需要设定一个初始值。A 的设定比较简单, 令 $a_{ii}(\tau) = 1.0/\tau$, $a_{ii+1}(\tau) = 1.0 - a_{ii}(\tau)$。而 μ、U、C 的选择复杂一点, 首先将所有训练序列均分成 N 段, 然后对每段聚类, 分成 M 类, 这样第 i 段第 m 类的类中心 z_{im} 就是第 i 个状态第 m 个混合的均值矢量 μ_{im}, 而对应的该类方差矢量就是第 i 个状态第 m 个混合的方差矢量 U_{im}。每类在段中的观测矢量数目比例就是混合权系数 c_{im}。其计算可以简单地用公式表示。设 $o_{l_{im}}$ ($1 \le i \le N$, $1 \le l_{im} \le L_{im}$, $1 \le m \le M$, L_{im} 是该类总数目) 是第 i 段第 m 段的某一观测矢量, 则有

$$\mu_{im} = z_{im} \qquad (1 \le i \le N, 1 \le m \le M) \tag{1}$$

$$U_{im} = \frac{1}{L_{im}} \sum_{l_{im}}^{L_{im}} (o_{l_{im}} - z_{im})(o_{l_{im}} - z_{im})^T \qquad (1 \le i \le N, 1 \le m \le M) \tag{2}$$

$$c_{im} = L_{im} \Big\backslash \sum_{k=1}^{M} L_{ik} \qquad (1 \le i \le N, 1 \le m \le M) \tag{3}$$

这里 z_{im}, $o_{l_{im}}$, μ_{im}, U_{im} 均是 D 维矢量。

(2) Viterbi 算法分段和参数 A 的估计

Viterbi 是一种搜索算法, 根据最大似然准则找到最好的状态序列。然后根据该最好状态序列将序列分成 N 段, 每一个状态对应一段。由于是多序列训练, 因此每一个观测序列都需要进行一次 Viterbi 算法, 最后总共分成 N 堆数据。

令 $p_i(\tau)$ 表示第 i 个状态时长为 τ 时的概率, 再令 $\bar{P}_i(\tau)$ 表示第 i 个状态时长不超过 τ 时的概率, 则有

$$\bar{p}_i(\tau) = \sum_{k=0}^{\tau} p_i(k)$$

如此, $a_{ij}(\tau)$ 可这样估计

$$\begin{cases} a_{ii}(\tau) = 1 - \bar{p}_i(\tau) = 1 - \sum_{k=0}^{\tau} p_i(k) \\ a_{ii+1}(\tau) = 1 - a_{ii}(\tau) \end{cases}$$

其中$p_i(k)$如下近似计算:

$$p_i(k) = \text{第 } i \text{ 个状态时长为 } k \text{ 的序列数目} / \text{总序列数目}$$

(3) μ、U、C的估计

此处μ、U、C的估计方法和(1)中的计算方法基本一致，所不同的是此处所用N堆数据是根据Viterbi搜索算法所得来的，而非均分各序列。其步骤依然是利用聚类方法将各堆分成M类，再依式(1)～(3)式计算μ、U、C。其中聚类算法是“长轴”类分裂方法和MKM方法的结合。

2 实验与讨论

2.1 实验

实验过程中设置系统IHMM Model的状态数为9，每个状态的混合数为9，我们考查了系统对特定人、多说话人和不认人的识别性能。

对特定人实验表明，该系统的性能是很好的。实验发音人为男性，训练数据库包括10个数字的1字串、2字串、3字串、4字串、5字串和6字串各100个，识别数据库由该男性间隔两星期后的发音，库中同样包括1字串、2字串、3字串、4字串、5字串和6字串各100个。系统对训练集和识别集的识别结果如表2。实验过程中数字串长度未知。

表2 系统对特定人识别结果

训练集							识别集						
音数	字串长度	误识率		误识别次数			音数	字串长度	误识率		误识别次数		
		字串	数字	替代	删除	增补			字串	数字	替代	删除	增补
100	1	1%	1%	0	0	1	100	1	1%	1%	0	0	1
100	2	2%	1%	0	2	0	100	2	2%	1%	0	1	1
100	3	2%	0.7%	0	2	0	100	3	3%	0.7%	0	1	2
100	4	3%	0.75%	0	2	1	100	4	4%	1%	0	2	2
100	5	4%	0.4%	1	1	2	100	5	5%	1%	0	2	3
100	6	4%	0.7%	2	2	0	100	6	5%	0.5%	0	2	3
平均	3.5	2.67%	0.76%	0.5	1.5	0.67	平均	3.5	3.3%	0.87%	0	1.3	2

表3 系统识别结果

多说话人识别集							不认人识别集						
音数	字串长度	误识率		误识别次数			音数	字串长度	误识率		误识别次数		
		字串	数字	替代	删除	增补			字串	数字	替代	删除	增补
100	1	4%	4%	1	0	3	100	1	4%	4%	1	0	3
200	2	8.5%	4.25%	4	12	1	100	2	6%	3%	4	1	1
200	3	14%	5%	6	22	0	100	3	20%	7%	9	8	3
200	4	7.5%	1.875%	2	11	2	100	4	16%	5%	8	7	1
200	5	11.5%	2.3%	4	15	4	100	5	18%	3.8%	8	8	2
200	6	12%	2%	8	8	8	100	6	11%	1.8%	2	3	6
平均	3.5	9.6%	3.2%	4.2	11.3	3	平均	3.5	12%	3.6%	5.2	4.5	2.3

表中结果表明:

(1) 本系统用于特定人识别系统识别性能非常好，字串误识别率在4%以内;

(2) 随着字串长度误识率增加，大部分错误发生在删除和增补上面，多数是数字连读时，这在人的耳朵分辨中也是困难的。

在对系统进行多说话人和不认人的实验中, 训练集是由 25 个男声的发音组成, 每个男声发音 2 字串和 3 字串各 10 遍, 4 字串 20 遍, 共 1000 个音。去掉其中发音时的出错和明显的端点检测错误, 剩下 952 个音用于实际训练。

在进行多说话人实验之中, 我们从为训练集发音的 25 个男声中随机抽取了 10 个男声用于检测, 产生多说话人实验的识别集。10 个男声中每人发音 1 字串 10 遍、2 字串、3 字串、4 字串、5 字串和 6 字串各 20 遍, 共 1100 个音。

系统首先用 25 个男声发音构成的训练集进行训练, 获得 10 个数字的参考模式库。再用它作为先验知识识别 10 个男声发音构成的多说话人识别集。这一结果显示在表 3。实验过程中字串长度未知。

从表中可见:

(1) 随着字串长度的增加, 字串误识率逐渐增大, 但数字误识率并没有这种规律, 相反在字串长度为最长 (为 6) 时, 误识率反而是最小为 1. 833%。此表明, 尽管字串误识变多了, 而其中数字的错误总数占整个被误识总数的比例并未变多, 反而变少。在增补和删除错误中, 计算数字误识是根据对应数字的正确与否来计算。

(2) 从数字和字串的误识率来看, 性能是比较好的。有相当部分错误与特定人系统相似, 仍然是增补和删除类错误, 并且从实验结果看, 这些错误基本上都是发生在数字连读之时。

另一个实验是应用前面 25 个男声发音构成的训练集训练获得的参考模式库, 识别另外 5 个男声的发音。这个不认人识别集是这样构成: 每个男声发 1 字串、2 字串、3 字串、4 字串、5 字串和 6 字串各 20 遍, 总共 600 个音, 这一识别结果显示在表 3 中, 实验过程中字串长度为未知。

从表中可见:

(1) 它的识别与多说话人实验性能差不多, 这说明 25 个男声及训练数是有一定的代表性的;

(2) 这个实验结果反映出的规律与多说话人实验结果反映规律基本相同。

2. 2 讨论

在多次实验中发现, 连接数字识别主要发生在发 '28' 连读时 '8' 错成 '2', 以及 '6' 错成 '9'。前一种现象通过对发音过程分析, 发现 '2' '8' 连说时 [er] 与 [a] 的口腔形状很相似, 而且舌位不能很快由 [er] 状态恢复原位, 因此造成 '8' 误识为 '2'。第二种现象则比较容易理解, 由于同韵母而连续造成声母的混淆导致误识。

如果系统能够有效地利用声调信息, 这种错误是容易被克服的, 因为 '2' 和 '6' 是去声, '8' 是阴平, '9' 是上声。据目前许多研究发现, 连续语音中声调的检测是困难的, 因为它受发音部位 (如: 喉部、口腔、鼻腔) 在发音过程中前后协同影响所至。然而声调是语音, 尤其汉语语音的重要信息。在这方面需要进一步研究。

参考文献

1 Rabiner LR et al. High Performance Conneted Digit Recognition Using Hidden Markov Models. IEEE Tran. on ASSP, 1989

2 张世平. 汉语全音节实时语音识别系统: [博士学位论文], 清华大学无线电系, 1988, 12

3 王成友. 语音特征信息综合方法及汉语数字语音识别系统的研究: [博士学位论文], 国防科技大学, 1997, 10

4 汤叔祺. 基于 HMM 的汉语语音识别研究: [硕士学位论文], 国防科技大学, 1996, 4