文章编号:1004-9037(2014)02-0265-09

基于分段动态时间规整的语音样例快速检索

冯志远 张连海

(解放军信息工程大学信息系统工程学院,郑州,450002)

摘要:提出了一种融合下界估计和分段动态时间规整的语音样例快速检索方法。该方法针对缺乏合适的训练数据等语音资源较为有限的语言进行快速检索所设计。此方法首先提取查询样例和测试集的音素后验概率;然后,根据限制条件在测试语句中选定候选分段,并计算查询样例和每个候选分段之间实际动态时间规整得分的下界估计,再运用 K 最近邻搜索算法搜索与查询样例相似度最高的分段;最后,使用虚拟相关反馈技术对检索结果进行修正。实验结果表明:尽管此方法的检索精度略低于直接运用动态时间规整进行检索的检索精度,但其检索速度优于后者,且检索结果经过虚拟相关反馈技术修正后,其检索精度也得到有效提升。

关键词:语音样例检索;音素后验概率;分段动态时间规整;下界估计;虚拟相关反馈

中图分类号:TP391.4 **文献标志码**:A

Fast Query-by-Example Spoken Term Detection Using Segmental Dynamic Time Warping

Feng Zhiyuan, *Zhang Lianhai*

(Institute of Information System Engineering, PLA Information Engineering University, Zhengzhou, 450002, China)

Abstract: A method for query-by-example spoken term detection(QbE STD) using segmental dynamic time warping(SDTW) and lower-bound estimate(LBE) is presented. The approach is designed for low-resource situations in which limited or no in-domain training material is available. According to this method, the phone posterior probabilities of query examples and test materials should be got firstly, and then the candidate segments are selected in test materials and LBE of actual DTW scores are computed between the query example and all candidate segments in test materials quickly. The K nearest neighbor (KNN) search algorithm is chosen to search for the segments that have maximal similarity. Finally, the retrieval results can be modified by pseudo relevance feedback(PRF). The experimental result indicates that although there is a slight degradation in retrieval precision when compared with formulating DTW procedure directly, the retrieval speed of the method presented in the paper is higher than the latter, and the retrieval precision can be enhanced availably after the retrieval results modified by PRF.

Key words: query-by-example spoken term detection; phone posterior probability; segmental dynamic time warping; lower-bound estimate; pseudo relevance feedback

引 言

随着信息技术和多媒体技术的迅猛发展,在网络速度快速提升、存储成本持续降低的情况下,新闻广播、语音信箱以及会议录音等各种以语音形式存储的数据急剧增多,但由于缺乏行之有效的语音检索技术,人们难以充分有效地利用这些资源。因此,如何在浩如烟海的语音资源中快速、准确地挑选出有用的信息,对于充分利用不断积累的信息资源具有极其重要的意义。在语音检索中最重要的关键技术是语音查询词检索(Spoken term detec-

基金项目:国家自然科学基金(61175017)资助项目。

收稿日期:2012-12-11;**修订日期**:2013-03-05

tion, STD),它是根据用户输入的查询项,在语音资源中搜索和返回与之相关的语音片段。查询项的形式有两种:一是文字形式;二是波形样例形式[1]。采用前者的形式进行查询称为基于文本的语音查询词检索;采用波形样例的形式进行查询的检索方式称为语音样例检索(Query-by-example spoken term detection , QbE STD)。STD 在军事和信息安全、数字图书馆、声音分类、音乐检索[2]等很多领域都有十分广泛的应用。

现阶段 STD 往往基于大词汇量连续语音识别(Large vocabulary continuous speech recognition, LVCSR),它将查询项和测试语句转换为文本形式,例如 one-best,Lattice 等,从而将检索问题转化为字符串匹配问题[3]。当然,针对语音资源十分丰富的语言进行检索,基于 LVCSR 的 STD 取得了不错的检索精度[4-5]。但是,得到一个可信度高、鲁棒性强的 LVCSR 系统需要大量正确标注的不同声学条件(不同语者、不同说话环境等)下的语音数据用来训练其统计上的声学/语言模型。即使是语音资源十分丰富的语言,收集和正确标注大量不同声学条件下的语音数据的代价也是很大的。此外,在基于 LVCSR 的方法中,其检索精度受词汇集的覆盖范围影响较大,假如查询项中含有集外词(Out of vocabulary, OOV)时,其检索精度将会下降。针对上述问题,并且从计算量方面考虑,许多学者致力于采用基于音素的方法进行 STD 的研究[6-7]。

对语音资源较为有限的语言进行音频检索,运用上述方法更不可行,首先,这类语言的语音资源较为有限,搜集和标注语料更为困难,代价更为巨大。其次,由于不同语言之间一些音素的声学表现形式是相似的,针对此类语言的检索任务可以运用交叉语言模型的方法或者是语言独立模型的方法,但在进行检索之前,首先要运用发音词典将查询项映射为音素序列,假如测试数据归属的语言和音素识别系统存在音素差异,则映射时会产生较大偏差[8]。

针对语音资源较为匮乏的语言进行样例检索时标准 STD 技术的种种不足,一些学者提出基于模板匹配的架构。Hazen 提出了基于音素后验概率和动态时间规整的语音样例检索方法[8],此方法首先运用语音资源较为丰富的语言训练音素后验概率检测系统,提取查询样例和测试语句的音素后验概率,再运用传统的动态时间规整(Dynamic time warping, DTW)计算查询样例和测试语句的相似度,最后根据相似度的大小进行排名,从而获得检索结果;Tejedor 在此基础上提出语音样例的选取和结合的新方法[9];Chan 提出一种基于段的无监督语音样例检索方法[10],该方法首先提取查询样例和测试集的声学特征,然后运用层次凝聚聚类算法对提取的声学特征进行分段,运用 DTW 并以上述分段为单位进行语音样例和候选分段之间相似度的计算。此架构完全消除了词汇集覆盖范围的限制,虽然音素后验概率检测器对训练语料有较高要求(需要大量标注到音素级别的语料),但是对测试语料无任何要求,因此,此架构在一定程度上解决了对语音资源较为匮乏的语言进行检索的问题。

检索速度是评价信息检索方法好坏的一个重要指标,而直接运用模板匹配的方法无法做到快速检索。这是因为一个查询样例或者测试语句可能含有成千上万帧,直接运用 DTW 进行检索往往耗费大量时间,且运用 DTW 进行检索时,缺乏对声学条件变异的考虑。为了满足用户对检索速度的要求,学者们对上述架构下的快速检索方法进行了研究,一些快速检索算法被相继提出。Jansen 从底层声学特征出发,提出运用局部敏感哈希、二值最近邻搜索等随机逼近算法对声学特征进行降维逼近,降低了语音样例和候选分段匹配时的运算复杂度[11];Chan 在上述模板匹配法的基础上提出基于段的无监督语音样例检索方法[10],该方法本质上是用较为稳定的特征分段代替特征分帧作为匹配单元,这样就大大降低了运算复杂度,提高了检索速度;Chan 又在上述基础上提出了一种分帧和分段相结合的检索方法[12],使得检索速度得到提高的同时,检索精度也得到有效提升;2012 年,Zhang 将上述架构与 GPU 相结合,充分利用 GPU 的并行运算能力,大大提高了检索速度[13]。

本文提出一种基于下界估计(Lower bound estimate,LBE)和分段动态时间规整(Segmental dynamic time warping, SDTW)的语音样例快速检索方法,该方法首先提取查询样例和测试语句的音素后验概率参数;然后,根据限制条件在测试语句中选定候选分段,计算查询样例和每个候选分段之间实际 DTW 得分的下界估计,并运用 K 最近邻搜索(K nearest neighbor,KNN)算法搜索与查询样例相似度最高的分段。该方法的基本思想为舍弃下界估计大于当前最佳匹配得分的候选分段,无需 DTW 匹配,通过大量减少 DTW 的匹配次数实现提高检索速度的目的。为了使检索结果更加

准确，本文还运用虚拟相关反馈（Pseudo relevance feedback，PRF）技术对检索结果进行修正，提出了基于虚拟相似度的相关区域重排序方法，从而缓解了 DTW 不能充分考虑声学条件变异的局限性。

1　音素后验概率检测

MFCC 是最广泛应用的特征参数。MFCC，PLP 等频谱参数构成了语音识别声学特征的基础，但因为这些参数只使用了 20，30 ms 左右的语音信息，所以极易受噪声的干扰。TRAP 结构描述的是长时间窗内各个子带的能量变化轨迹，这种长时性能够很好地描述语音信号在时间上的相关性，在语音识别中得到广泛应用[14]。本文将改进的 TRAP 结构[15]引入对音素后验概率的检测，完整的系统架构如图 1 所示。

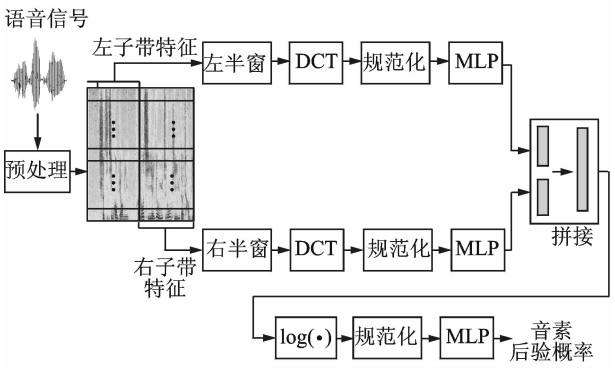

图 1　音素后验概率检测系统

Fig. 1　Detection system of phoneme posterior probability

2　动态时间规整

应用 DTW 之前，应首先定义两帧特征参数之间的距离，本文采用内积距离，给定两帧特征向量 $\boldsymbol{q}$ 和 $\boldsymbol{s}$，其内积距离

$$d(\boldsymbol{q},\boldsymbol{s})=-\log(\boldsymbol{q}^{\mathrm{T}}\boldsymbol{s}) \tag{1}$$

本文将上述内积距离定义在对数空间。在进行对数运算中，如果 $\boldsymbol{q}^{\mathrm{T}}\boldsymbol{s}=0$，则会导致 $d(q,s)=+\infty$，为避免出现此种错误，本文对 $\boldsymbol{q}$ 做近似变换，设 $\boldsymbol{q}'$ 为 $\boldsymbol{q}$ 的近似变换，$\boldsymbol{q}'$ 与 $\boldsymbol{q}$ 的变换关系为

$$\boldsymbol{q}'=(1-\lambda)\cdot\boldsymbol{q}+\lambda\boldsymbol{\mu}$$

式中：λ 为一个很小的正数，$\boldsymbol{\mu}$ 为一个与 q 维数相同且服从均匀分布的概率向量。

2.1　基于帧的动态时间规整

给定一个语音样例 $\boldsymbol{Q}=(\boldsymbol{q}_1,\cdots,\boldsymbol{q}_M)$ 和一个语音片段 $\boldsymbol{S}=(\boldsymbol{s}_1,\cdots,\boldsymbol{s}_N)$，其中 $\boldsymbol{q}_i$ 和 $\boldsymbol{s}_j$ 表示 D 维音素后验概率特征向量。给定 $\boldsymbol{Q}$ 和 $\boldsymbol{S}$，DTW 的目标就是寻找一个规划路径，使得该路径上的累积距离最小。定义规划路径为

$$\varphi=\{(\varphi_q(1),\varphi_s(1)),\cdots,(\varphi_q(K_\varphi),\varphi_s(K_\varphi))\} \tag{2}$$

式中 $\varphi_q(k)$ 和 $\varphi_s(k)$ 分别代表 $\boldsymbol{Q}$ 和 $\boldsymbol{S}$ 的特征参数序列索引。因此，给定路径 φ，则 $\boldsymbol{Q}$ 和 $\boldsymbol{S}$ 的相应匹配得分为

$$A_\varphi(\boldsymbol{Q},\boldsymbol{S})=\sum_{k=1}^{K_\varphi}d(\boldsymbol{q}_{\varphi_q(k)},\boldsymbol{s}_{\varphi_s(k)}) \tag{3}$$

式中 d 表示两向量之间的内积距离。为了避免在匹配过程中输入序列和语音片段之间出现较大的时差，需要对路径 φ 进行限制。其中，最常用的条件为

$$|\varphi_q(k)-\varphi_s(k)|\leqslant r \tag{4}$$

式中 r 为路径限制因子。由上述可知，要得到 $\boldsymbol{Q}$ 和 $\boldsymbol{S}$ 之间的最优对齐，其计算复杂度为 $O(MN)$。

2.2　分段动态时间规整

SDTW 通过在两个特征向量序列的距离矩阵中划分并检索多条路径来以达到找到其最佳的局部对齐的目的。给定两个特征向量序列 $\boldsymbol{Q}=(\boldsymbol{q}_1,\cdots,\boldsymbol{q}_M)$ 和 $\boldsymbol{S}=(\boldsymbol{s}_1,\cdots,\boldsymbol{s}_N)$，SDTW 把它们之间的距离矩阵划分为一系列交叉重叠的对角带，这样，不但避免两个匹配子段在匹配过程中时域上相差过大，而且每一个对角带对应一个不同的匹配路径，这样就直接产生多条路径以供检索。

SDTW 在进行 DTW 搜索时定义了两个限制条件。首先就是常用的调节窗条件，给定 $\boldsymbol{Q}$ 以及 $\boldsymbol{S}$，则定义在大小为 $M\times N$ 距离矩阵上的规整函数 $p(\cdot)$ 的形式为 $p(\cdot)=(i_k,j_k)$，其中，(i_k,j_k) 定义为规整路径的第 k 个坐标。根据在语音信号的特性，调节窗条件为

$$|i_k-j_k|\leqslant R \tag{5}$$

从式(5)可以看出，R 在这里与上述路径限制因子意义相同。

第二个限制条件为相邻对角带起点坐标的步长。很明显，假设固定一条规整路径的起点坐标，则调节窗条件限制的不仅仅是匹配的区域，而且还有其终点坐标。假设 $i_1=1$，$j_1=1$，则其终点坐标为 $i_{\mathrm{end}}=m$，$j_{\mathrm{end}}\in(m-R,m+R)$。因此，对每次规整过程使用不同的起点，则距离矩阵自然的划分为一系列宽度为 $2R+1$ 的对角带，如图 2 所示。为

了避免规整函数的冗余计算，本文针对起点坐标利用重叠滑动窗，具体来说，每一次向前移动 R 步进行一个新的 DTW 搜索。

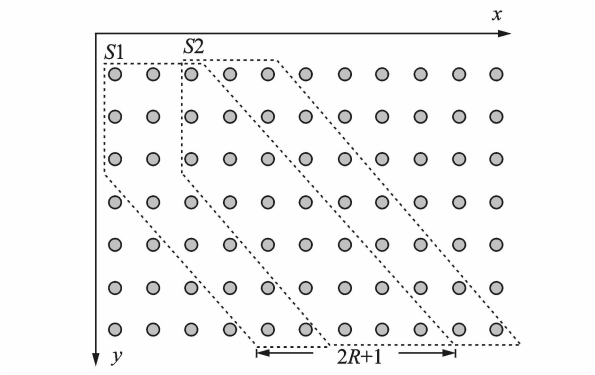

图 2 SDTW 原理图($R=2$)

Fig. 2 The schematic diagram of SDTW ($R=2$)

一般地，给定 R 以及测试语句长度 n，即其包含音素后验概率的帧数，则起点坐标为

$$(1,(k-1)\cdot R+1) \tag{6}$$

式中，$1\leqslant k\leqslant \lfloor \frac{n-1}{R} \rfloor$，$\lfloor\cdot\rfloor$ 为向下取整运算。由此看来，对于一个查询样例和一段测试语句，分段动态时间规整可以产生 $\lfloor \frac{n-1}{R} \rfloor$ 个匹配区域，即 $\lfloor \frac{n-1}{R} \rfloor$ 个规整路径，而 R 不仅是起点坐标的步长，而且还是路径限制因子。

匹配区域划定之后，使用 DTW 动态计算查询样例 $\boldsymbol{Q}$ 和匹配区域 hr 之间的相似度得分，寻找最优规划路径时，在上述条件限定下，选择使目前累计距离最小的索引对作为下一步的规划路径。在处理完最后一个索引对后，通过回溯得到最优规划路径 $P_{\text{opt}}=\{(i_1,j_1),(i_2,j_2),\cdots,(i_K,j_K)\}$。因此 $\boldsymbol{Q}$ 和 hr 的匹配距离得分 DTW($\boldsymbol{Q},hr_k$)可以利用式(7)求出

$$\text{DTW}(\boldsymbol{Q},hr_k)=\sum_{s=1}^{s=K}D(i_s,j_s) \tag{7}$$

式中 D 为查询样例和该区域归属的测试语句之间的距离得分。进一步，将其转化为相似度得分 $S(Q,hr_k)$

$$S(\boldsymbol{Q},hr_k)=1-\frac{\text{DTW}(\boldsymbol{Q},hr_k)-\min(\text{DTW}(\boldsymbol{Q},HR))}{\max(\text{DTW}(\boldsymbol{Q},HR))-\min(\text{DTW}(\boldsymbol{Q},HR))} \tag{8}$$

式中：DTW($\boldsymbol{Q},HR$)={DTW($\boldsymbol{Q},hr_1$),…,DTW($\boldsymbol{Q},hr_G$)}为匹配得分集合。为避免混淆，本文将 $S(\boldsymbol{Q},hr_k)$称为原始相似度。

3 融合下界估计的分段动态时间规整

为进一步提高匹配效率，本文在分段动态时间规整的基础上提出融合下界估计的分段动态时间规整算法，该算法在应用分段动态时间规整之前。首先根据限制条件在测试语句中选定候选分段，计算查询样例和每一个候选分段之间 DTW 得分的下界估计，再运用 K 最近邻搜索(K nearest neighbor, KNN)算法搜索与查询样例最相关的分段。该算法的基本思想为舍弃下界估计大于当前最佳得分的分段，无需进行 DTW 匹配，通过大量减少 DTW 匹配次数实现检索速度的提高。

3.1 下界估计

3.1.1 定义

给定两个音素后验概率序列，$\boldsymbol{Q}=(\boldsymbol{q}_1,\cdots,\boldsymbol{q}_M)$ 和 $\boldsymbol{S}=(\boldsymbol{s}_1,\cdots,\boldsymbol{s}_N)$，其中，$\boldsymbol{q}_i=\{q_i^1,q_i^2,\cdots,q_i^D\}$，$\boldsymbol{s}_i=\{s_i^1,s_i^2,\cdots,s_i^D\}$。可以通过对 $\boldsymbol{Q}$ 求得一个序列 $\boldsymbol{U}=\{\boldsymbol{u}_1,\cdots,\boldsymbol{u}_M\}$进而得到 $\boldsymbol{Q}$ 和 $\boldsymbol{S}$ 实际 DTW 得分的下界估计，本文称 $\boldsymbol{U}$ 为 $\boldsymbol{Q}$ 的上限序列，其中，$\boldsymbol{u}_i=\{u_i^1,\cdots,u_i^D\}$，且 $u_i^j=\max(q_{i-r}^j,\cdots,q_{i+r}^j)$。$\boldsymbol{U}$ 可以看作针对 $\boldsymbol{Q}$ 的一个 D 维最大值取值器，r 为限制因子，与上述 SDTW 过程中的窗长数值大小保持一致。很明显，$\boldsymbol{U}$ 中任意元素 $\boldsymbol{u}_i$ 满足 $\sum_{j=1}^{D}\boldsymbol{u}_i^j\geqslant 1$。给定 $\boldsymbol{Q}$ 和 $\boldsymbol{S}$，则其实际 DTW 得分的下界估计 $L(\boldsymbol{Q},\boldsymbol{S})$定义为

$$L(\boldsymbol{Q},\boldsymbol{S})=\sum_{i=1}^{l}d(\boldsymbol{u}_i,\boldsymbol{s}_i) \tag{9}$$

式中：$l=\min(M,N)$，d 为两向量之间的内积距离。由式(9)可以得出 $L(\boldsymbol{Q},\boldsymbol{S})$的计算复杂度仅为 $O(l)$。

3.1.2 证明

本文采用倒推法给出不等式 $L(\boldsymbol{Q},\boldsymbol{S})\leqslant$DTW($\boldsymbol{Q},\boldsymbol{S}$)的证明。

将上述不等式左右两部分分别展开，可以得到

$$\sum_{i=1}^{l}d(\boldsymbol{u}_i,\boldsymbol{s}_i)\leqslant\sum_{i=1}^{K_\varphi}d(\boldsymbol{q}_{\varphi_q(k)},\boldsymbol{s}_{\varphi_s(k)}) \tag{10}$$

式(10)右边表示实际 DTW 得分，将其匹配路径拆分成两个部分，分别用 MA 和 UM 表示，即

$$\sum_{i=1}^{l}d(\boldsymbol{u}_i,\boldsymbol{s}_i)\leqslant\sum_{k\in MA}d(\boldsymbol{q}_{\varphi_q(k)},\boldsymbol{s}_{\varphi_s(k)})+$$

$$\sum_{k\in UM} d(\boldsymbol{q}_{\varphi_q(k)}, \boldsymbol{s}_{\varphi_s(k)}) \tag{11}$$

式中，MA 包含 l 个元素，其构建规则如下：针对不等式左边第 i 项，与之相对应的不等式右边实际规整路径中的某个元素$(\varphi_q(k), \varphi_s(k))$，假如 $\varphi_s(k)=i$，则将其选入 MA；假如实际规整路径中与第 i 帧相匹配的不止一帧，即规整路径中 $\varphi_s(k)=i$ 的元素大于 1 个，则将具有最小的 $\varphi_q(k)$ 的$(\varphi_q(k), \varphi_s(k))$选入 MA，通过这种规则，确保 MA 中含有的元素个数为 l 个。UM 包含整个规整路径中除 MA 外所有剩下的元素。由内积距离定义可知，上述不等式的每一项均为正数，因此，假如可以证明

$$\sum_{i=1}^{l} d(\boldsymbol{u}_i, \boldsymbol{s}_i) \leqslant \sum_{k\in MA} d(\boldsymbol{q}_{\varphi_q(k)}, \boldsymbol{s}_{\varphi_s(k)}) \tag{12}$$

则式(11)中的 $\sum_{k\in UM} d(\boldsymbol{q}_{\varphi_q(k)}, \boldsymbol{s}_{\varphi_s(k)})$可以直接消去。

设$(\varphi_q(k), \varphi_s(k))$为 MA 中的一个元素，它与左边的第 i 项 $d(\boldsymbol{u}_i, \boldsymbol{s}_i)$相对应，即 $\varphi_s(k)=i$。将式(12)左右两边用内积距离形式表示

$$\sum_{i=1}^{l} -\log(\boldsymbol{u}_i, \boldsymbol{s}_i) \leqslant \sum_{i\in MA} -\log(\boldsymbol{q}_{\varphi_q(i)}, \boldsymbol{s}_{\varphi_s(i)}) \tag{13}$$

因为两边的元素个数相同，且一一对应。假设式(13)左边的每一项均小于右边与之相对应的那一项，则不等式成立。消去负号与对数运算，仅保留内积运算，为证明式(13)，仅需证明

$$\boldsymbol{u}_i \cdot \boldsymbol{s}_i \geqslant \boldsymbol{q}_{\varphi_q(i)} \cdot \boldsymbol{s}_{\varphi_s(i)} \tag{14}$$

根据 MA 的构建规则：$\varphi_s(i)=i$，因此$\boldsymbol{s}_i=\boldsymbol{s}_{\varphi_s(i)}$，且由于 DTW 全局路径限制条件$|\varphi_q(i)-\varphi_s(i)|\leqslant r$，可以得到，$|\varphi_q(i)-i|\leqslant r$ 或者是 $i-r\leqslant\varphi_q(i)\leqslant i+r$，根据 u_i^j 的定义可知 $u_i^j\geqslant q_{\varphi_q(i)}^j$，因此式(14)成立，故不等式 $L(\boldsymbol{Q},\boldsymbol{S})\leqslant$DTW$(\boldsymbol{Q},\boldsymbol{S})$也成立。

由于音素后验概率特性向量的所有元素之和为 1，因此，两帧音素后验概率特征向量的内积不大于 1，即$\boldsymbol{q}_{\varphi_q(i)}\cdot\boldsymbol{s}_{\varphi_s(i)}\leqslant 1$，假如$\boldsymbol{u}_i\cdot\boldsymbol{s}_i\geqslant 1$，其下界估计将毫无意义，下面给出$\boldsymbol{u}_i\cdot\boldsymbol{s}_i\leqslant 1$ 的证明。

设 $u_{\max}=\max(u_i^1,\cdots,u_i^D)$，则存在

$$\boldsymbol{u}_i\cdot\boldsymbol{s}_i \leqslant u_{\max}\cdot\sum_{j=1}^{D} s_i^j = u_{\max}$$

由 u_i^j 的定义可知：$u_{\max}\leqslant 1$，故可得出$\boldsymbol{u}_i\cdot\boldsymbol{s}_i\leqslant 1$，所以用此方法进行 DTW 实际得分的下界估计是有意义的。

3.2　K 最近邻搜索算法

为了在测试集找到与查询样例最为相似的 K 个语音分段，直接运用 DTW 检索则需要对测试集中每一个测试语句中的每一个候选分段进行匹配，效率十分低下。如将下界估计算法与 KNN 搜索算法融合，则能较好的提高匹配效率，KNN 搜索算法伪代码如下所示。

```
Data: Q, U and C
Result RL containing k most possible segments having Q
begin
    for each utteranc e c ∈ C do
        for each segment s ∈ c do
            lb = ComputeLB (U, s)
            PQ.push([lb, s])
    KthBest = MaxFloat
    while PQ Null and (|RL| < k  or  PQ.lb < KthBest)
    do
            v = DTW(Q, s)
            if v < KthBest
              c = FindC (s)
              if c RL
                UpdateRL (c, s, v)
              else
                indice = examineS (s)
                if indice
                  UpdateRL (c, s, v)
            KthBest = FindMax (RL)
            PQ. pop()
```

该算法基本思想就是去除任何下界估计大于当前最佳得分(KthBest)的语音片段，上述伪代码中函数 ComputeLB 计算查询样例 $\boldsymbol{Q}$ 和测试语句中每一个可能的片段 $\boldsymbol{S}$ 的下界估计，测试集中所有可能的片段根据其下界估计得分排名，并将其该片段的信息以及相应的下界估计得分存储在 PQ 中。

KNN 算法从 PQ 的最顶端开始，即从下界估计最小的分段开始。计算该片段与查询样例之间的实际 DTW 距离，如果该片段的实际 DTW 得分小于当前最佳得分，运用函数 FindC 定位该分段所属的测试语句，假如结果列表(RL)无此测试语句，则将此句加入 RL，即更新 RL(对应函数为 UpdataRL)；如果 RL 有此测试语句，则运用函数 examineS 检查结果列表中已存在属于该语句的片段与当前片段之间的帧索引差，indice 为索引差指数，如果索引差较大，表明查询样例在当前测试语句中出现不止一次，应将此片段添加进结果列表，否则予以舍弃；最后，将 KthBest 设置为结果列表中实际 DTW 得分的最大值，运用函数 Findmax 获得。假如结果列表中的语音片段个数等于 K 个且 PQ 中所有剩下分段的下界估计均大于 KthBest，则算法结束。由上述分析可知，上述方法只是排除任意一个与查询样例 Q 之间 DTW 得分的

下界估计大于当前最佳得分的语音片段，减少 DTW 匹配次数，提高检索效率，而对其检索精度不会有任何影响。

具体到本文任务，在运用 LBE 之前，测试语句中每一个候选分段与查询样例进行匹配时，其参与匹配的帧数是未知的，因此，需对测试语句加上一个长度与查询样例长度相同的滑动窗，该滑动窗每隔 R(R 的大小与窗长保持一致)帧向前移动一次，即完成一次匹配分段的选取。匹配分段选定之后，再计算查询样例和该分段 DTW 得分的下界估计，而在此之前的 SDTW 算法中除了满足 2.2 节所涉及的条件之外，其匹配分段长度的选取也要满足上述条件，只有 SDTW 和 KNN-DTW 算法中的 DTW 在匹配时满足相同的条件，才能验证此算法不会影响其检索精度。

4 基于虚拟相似度的相关区域重排序方法

为了使检索结果更加准确，本文利用虚拟相关反馈技术对检索结果进行修正。具体过程如下：经过初步检索，可以得到一系列相关区域(分段)，找出第一次检索结果中排名最靠前的 N 个区域和 M 个区域，区域选择如图 3 所示。将前者 N 个区域标记为 $Q_H=\{Q_1,Q_2,\cdots,Q_N\}$，称为虚拟相关区域；将后者 M 个区域标记为 $HR=\{hr_1,hr_2,\cdots,hr_M\}$，称之为假设相关区域，其相应原始相似度得分标记为 $S_{HR}=\{S(\boldsymbol{Q},hr_1),\cdots,S(\boldsymbol{Q},hr_M)\}$，称 M 为假设相关区域总数(其实这里的 hr_t 与上面的 Q_i 意义相同，为避免混淆，才如此标记)，只选择前 M 个区域而不是全部的匹配区域，这是因为：第一，假定所有真正相关区域在此 M 个区域中全部出现，事实上，真正的相关区域很大一部分集中在排名较为靠前的一部分匹配区域中，当然，M 的取值不应过小；第二，可以大大节省检索时间。对于两个与查询样例真正相关的匹配区域来说，它们应该具有较高的相似度，因此，与 Q_H 具有较高相似度的区域，也很可能是真正相关的区域，其排名应该被提高；反之，其排名应该被降低。利用 DTW 获取第 i 个假设区域 hr_i 与 Q_H 的匹配得分 $\mathrm{DTW}_{\mathrm{prf}}(Q_H,hr_i)$

$$\mathrm{DTW}_{\mathrm{prf}}(Q_H,hr_i)=\sum_{n=1}^{N}\mathrm{DTW}_{\mathrm{prf}}(Q_n,hr_i)^2$$

由 $\mathrm{DTW}_{\mathrm{prf}}(Q_H,hr_i)$ 可以得到其虚拟相似度 $\mathrm{SIM}(Q_H,hr_i)$

$$\mathrm{SIM}(Q_H,hr_i)=1-\frac{\mathrm{DTW}_{\mathrm{prf}}(Q_H,hr_i)-\min\{\mathrm{DTW}_{\mathrm{prf}}(Q_H,HR)\}}{\max\{\mathrm{DTW}_{\mathrm{prf}}(Q_H,HR)\}-\min\{\mathrm{DTW}_{\mathrm{prf}}(Q_H,HR)\}}$$

其中，$\mathrm{DTW}_{\mathrm{prf}}(Q_H,HR)=\{\mathrm{DTW}_{\mathrm{prf}}(Q_H,hr_1),\cdots,\mathrm{DTW}_{\mathrm{prf}}(Q_H,hr_M)\}$。将虚拟相似度得分 $\mathrm{SIM}(Q_H,hr_i)$ 与原始相似度得分 $S(Q,hr_i)$ 融合，从而得到新的相似度得分 $S_{\mathrm{new}}(Q,hr_i)$

$$S_{\mathrm{new}}(Q,hr_i)=(1-a)\cdot S(Q,hr_i)+a\cdot\mathrm{SIM}(Q_H,hr_i)$$

其中 a 为虚拟相似度权重因子。最后依据 $S_{\mathrm{new}}(Q,hr_i)$ 对检索结果进行重排序。

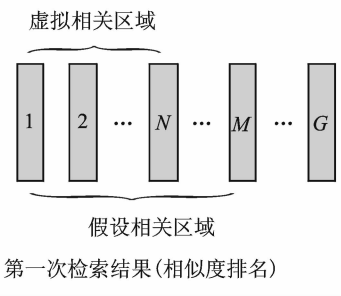

图 3 相关区域的选择

Fig. 3 The selection of the relevant area

5 实 验

5.1 实验配置

实验训练集选用 CTS 中 Switchboard Cellular 的部分语料共(10 h)作为音素后验概率检测系统的训练集。选择 TIMIT 语料库(除去 SA1 和 SA2 以及开发集共 4 640 句，总时长约 3.5 h)作为测试集，并使用 NICO Toolkit 来训练 MLP 作为音素后验概率检测器。

在音素后验概率检测实验中，选择帧长与帧移间隔分别为 25 ms 和 10 ms，然后对语音信号进行预加重、加汉明窗，将频谱转化为梅尔频标后并进行三角窗滤波，使用梅尔域的 23 个频带，时域上左右各扩展了 15 帧，加中心帧 16 帧，每帧帧移为 10 ms，相当于共用到了 310 ms 的扩展模式，每个频带取 DCT 变换后的前 10 维加上 C0 特征(能量)，因此，两个底层 MLP 各有 253 维的输入特征。两个底层 MLP 的输出维数与音素个数相等。高层 MLP 的输入维数为两个底层 MLP 的输出维数之和，即为音素个数的 2 倍，其输出维数与音素个数相等。

从测试集中随机选择 15 个查询样例，具体如表 1 所示。表 1 中各个查询样例后面括号中的数字为该样例在测试集中实际出现的次数。

表 1　查询样例汇总

Table 1　The summary of sample query

most(65)	favor(10)	drop(24)
surface(11)	artists(13)	formula(10)
products(9)	popular(23)	abruptly(8)
contagious(7)	completely(15)	shampooed(7)
warm(21)	problem(39)	children(33)

5.2　性能指标

采用信息检索领域用来评估检索算法的评估指标 MAP 以及实时系数(Realtime coefficient，RT)作为量化检索性能的指标。其中，MAP 用来衡量检索的精度；实时系数用来衡量检索的速度，其定义为对所有查询样例完成检索 CPU 所消耗的时间与测试集总时长之间的比值。

5.3　实验结果

为使下文对实验结果的描述更加方便与准确，对文中所涉及的检索方法进行编号，具体对应关系见表 2。

表 2　不同方法与其对应编号

Table 2　The corresponding numbers of different methods

检索方法	编号
DTW	方法 1
SDTW	方法 2
LBE＋SDTW	方法 3
LBE＋SDTW＋PRF	方法 4

5.3.1　*方法 1 与方法 2 的检索性能比较*

表 3 所示为方法 1 与方法 2 的检索性能对比。实验中 λ 取值为 0.01(后续实验均为此值)。从表 3 中不难看出，方法 2 的检索精度略低于方法 1，但是其检索速度大大优于后者。图 4 所示为采用方法 2 时，窗长 R 对 MAP 的影响，从图 4 中可以看出，MAP 随着 R 的变化先增大后减小。这是由于窗长过小时，过分限制了查询样例和测试语句之间的路径规整，造成检索精度的降低；而窗长过大时，可能产生具有较大时差的规整路径，也会造成检索精度的降低。所以，运用 SDTW 时选取合适的窗长十分重要。

表 3　方法 1 与方法 2 的检索性能对比

Table 3　The retrieval performance comparison of method 1 and method 2

方法	MAP/%	RT
方法 1	60.64	3.49
方法 2	60.08	0.53

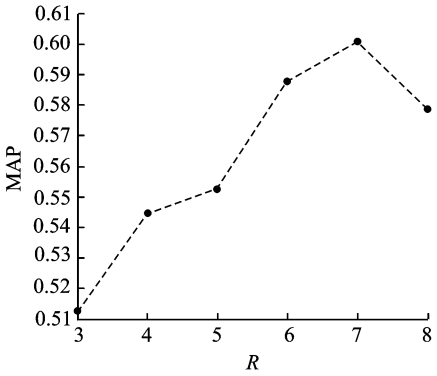

图 4　窗长对 MAP 的影响

Fig. 4　The effect of window size on MAP

图 5 所示为采用方法 2 及方法 3 时，窗长对 RT 的影响。从图 5 中可以看出，采用方法 2 时，RT 随着窗长 R 的不断增加而减小，这是由于随着窗长增大，其规整路径随之减少，故其检索时 CPU 消耗时间也随之减少，RT 与 CPU 消耗时间是正比关系，因此，RT 随着窗长 R 的不断增加而减小。而采用方法 1 时，随全局限制因子的增加，RT 并无明显变化，其平均实时系数为 3.49。因此，方法 2 在检索速度方面相对于方法 1 有很大优势。

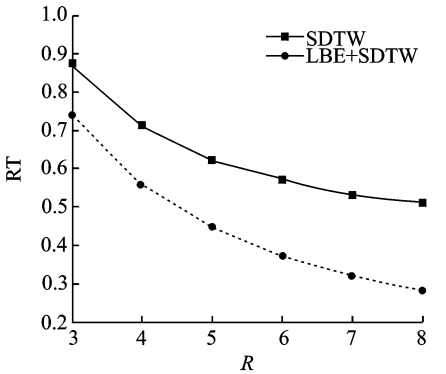

图 5　窗长对 RT 的影响

Fig. 5　The effect of window size on RT

5.3.2　*方法 2 与方法 3 的检索性能比较*

从 3.2 节的分析可知，采用方法 3 不会改变 SDTW 的检索精度(MAP 实验结果不再给出)，而

是从检索速度方面加以改善，从图 5 可以看出，采用方法 3 时，相对于方法 2，其检索速度进一步提高，这是由于方法 3 所需进行的 DTW 匹配次数远远少于方法 2，虽然计算查询样例与每一个候选分段实际 DTW 得分的下界估计需要耗费一定时间，但其时间消耗量与节省的时间（节省的时间主要为节省的 DTW 匹配所应消耗的时间）相比是很小的。

5.3.3 *方法 3 与方法 4 检索性能比较*

在方法 3 的基础上，本文使用 PRF 对检索结果进行修正，图 6 所示为使用方法 4 时，虚拟相似度权重因子 a 对 MAP 的影响，实验中窗长 R 取值为 7，假设区域总数 M 取值为查询样例实际出现次数的 3 倍，虚拟相关区域数目 N 取 2。从图 6 中可以出，虚拟相似度权重 a 为 0.5 时，MAP 达到最大为 61.56%，相对于采用方法 2 时，其 MAP 提高了 1.36%；相对于方法 1，MAP 提高了 0.92%。这是因为虚拟相似度是一种有效的置信度方法，可以对存在一定偏差的原始相似度进行修正，使得检索结果更准确。但是，这是以检索速度的降低为代价的，从图 5 可知，当 R 取值为 7 时，RT 为 0.32，而运用 PRF 之后，RT 为 0.536，很明显，运用 PRF 造成了 RT 的急剧增加。这是因为假设相关区域总数 M 取值为查询样例实际出现次数的 3 倍，在进行 KNN 搜索时，结果列表中所要得到的分段个数是运用 PRF 之前的 3 倍，这样就使得当前最佳匹配得分的不断增加，因此，DTW 匹配次数也随之急剧增加，可以看出，当语音样例实际出现的次数超过一定范围之后，即计算查询样例和候选分段之间 DTW 得分的下界估计所消耗的时间大于或者等于节省的 DTW 匹配所应消耗的时间时，方法 4 也不能保证检索速度的提高；另外，在运用 PRF 对假设区域中的每个分段进行反馈时，

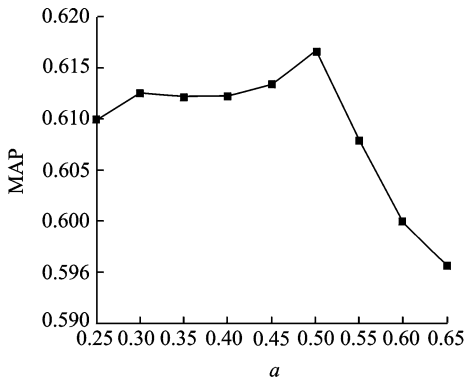

图 6 虚拟相似度权重因子对 MAP 的影响

Fig. 6 The effect of virtual similarity weighting factor on MAP

也需要一定次数的 DTW 匹配，而以上两个方面都需要消耗一定的时间，从而造成 RT 的增加。从图 6 中还可以看出当虚拟相似度取值太大时，MAP 急剧下降，这说明原始相似度对系统的检索精度也起着重要作用。因此，在对原始相似度和虚拟相似度进行融合时，需要选择合适的权重，这样才能使得相关区域的排序更加准确。

6 结束语

本文提出了一种基于下界估计和分段动态时间规整的语音样例检索方法，此方法首先提取查询样例和测试语句的音素后验概率参数；然后，计算查询样例和每个候选分段之间时间 DTW 得分的下界估计，并运用 K 最近邻搜索算法搜索与查询样例相似度最高的分段；最后，使用虚拟相关反馈技术对检索结果进行修正。实验表明，尽管其检索精度略低于直接运用 DTW 进行检索，但其检索速度大幅提高，且检索结果经 PRF 修正后，MAP 得到有效提高，然而，这是以检索速度的降低为代价的。

参考文献：

[1] Shen W, White C M, Hazen T J. A comparison of query-by example methods for spoken term detection [C]//Conference of the International Speech Communication Association 2009. Brighton, United Kingdom: [s. n.], 2009: 2143-2146.

[2] Chelba C, Hazen T J, Saraclar M. Retrieval and browsing of spoken content[J]. IEEE Signal Processing Magazine, 2008, 3(25): 39-49.

[3] Tzanetakis G, Ermolinsky A, Cook P. Pitch histograms in audio and symbolic music information retrieval[J]. Journal of New Music Research, 2003, 2(32): 143-152.

[4] Saraclar M, Sproat R W. Lattice-based search for spoken utterance retrieval [C]//Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics. Boston, America: [s. n.], 2004: 129-136.

[5] Miller D, Kleber M, Kimball O, et al. Rapid and accurate spoken term detection[C]// Conference on the International Speech Communication Association. Antwerp, Belgium: [s. n.], 2007: 314-317.

[6] Ng K. Subword-based approaches for spoken document retrieval[D]. Massachusetts Institute of Technology, 2000: 53-69.

[7] Yu Peng, Chen Kaijiang, Ma Chengyuan, et al. Vocabulary-independent indexing of spontaneous speech [J]. IEEE Trans on Speech Audio Processing, 2005, 5(13): 635-643.

[8] Hazen T J, Shen W, White C. Query-by-example spoken term detection using phonetic posteriorgram templates [C]//Automatic Speech Recognition and Understanding. Merano/Meran, Italy: [s. n.], 2009: 421-426.

[9] Tejedor J, Szöke I, Fapšo M. Novel methods for query selection and query combination inquery-by-example spoken term detection[C]// SSCS 2010. Palazzo Vecchio: [s. n.], 2010: 15-20.

[10] Chan Chunan, Lee Linshan. Unsupervised spoken term detection with spoken queries using segment-based dynamic time warping[C]//Interspeech 2010. Chiba, Japan: [s. n.], 2010: 2141-2144.

[11] Jansen A, Durme B V. Indexing raw acoustic features for scalable zero resource search [C]//Interspeech 2012. Portland Oregon: [s. n.], 2012: 524-527.

[12] Chan Chunan, Lee Linshan. Integrating frame based and segment-based dynamic time warping for unsupervised spoken-term detection with spoken queries [C]//ICASSP 2011. Prague, Czech Republic: [s. n.], 2011: 5652-5655.

[13] Zhang Yaodong, Adl K, Glass J. Fast spoken query detection using lower-bound dynamic time of graphical processing units[C]//ICASSP 2012. Kyoto, Japan: [s. n.], 2012: 5173-5176.

[14] Grezl F. Trap-based probabilistic features for automatic speech recognition[D]. Brno University of Technology, 2007: 13-19.

[15] Schwarz P. Phoneme recognition based on long temporal context[D]. Prague: Brno University of Technology, 2008: 35-40.

作者简介：冯志远(1988-)，男，硕士研究生，研究方向：语音查询词检索，E-mail: fengzhiyuande@163.com；张连海(1971-)，男，副教授，研究方向：语音信号处理、语音识别。