维普资讯 http://www.cqvip.com

大词汇量连续语音识别中搜索空间的表示及相关搜索方法的研究进展*)

杨凤芹[1,2,3] 孙吉贵[1,2] 张长胜[1,2] 张长海[1,2]

(吉林大学计算机科学与技术学院 长春 130012)[1]

(符号计算与知识工程教育部重点实验室 长春 130012)[2] (东北师范大学计算机学院 长春 130017)[3]

摘 要 本文综述了近年来大词汇量连续语音识别中搜索空间的表示及相关搜索方法的研究进展,分析了搜索空间的表示及相关搜索方法对语音识别性能产生的影响,并对本领域的研究中存在的问题和未来的发展动向进行了讨论。

关键词 大词汇量连续语音识别,搜索空间,搜索,声学模型,语言模型

Research on the Search Space Representation and the Search Methods in Large Continuous Speech Recognition

YANG Feng-Qin[1,2,3] SUN Ji-Gui[1,2] ZHANG Chang-Sheng[1,2] ZHANG Chang-Hai[1,2]

(College of Computer Science and Technology, Jilin University, Changchun 130012)[1]

(Key Laboratory for Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012)[2]

(College of Computer Science, Northeast Normal University, Changchun 130017)[3]

Abstract This paper reviews the research progress of the search space representation and the search methods in large vocabulary continuous speech recognition(LVCSR), and analyzes the affection on the recognition performance by the search space representation and the search methods. Finally, the unsolved problems and the prospect in this field are pointed out.

Keywords Large vocabulary continuous speech recognition, Search space, Search, Acoustic model, Language model

1 引言

语音识别算法的主要思路是在候选的词串中搜索使声学模型和语言模型的概率乘积最大的词串。因此,识别过程也常称作搜索(Search)或解码(Decoding)。由候选词串构成的搜索空间要受语言模型和声学模型的约束。从当前大词汇量连续语音识别的研究进展来看,为了提高识别结果的精度,通常使用上下文相关的声学模型和长跨度的语言模型,因此增加了搜索空间的复杂度。本文主要研究一遍搜索方法中的搜索空间的表示及相关的搜索算法。文章内容安排如下:第2部分运用HMM和贝叶斯概率框架对搜索问题进行了描述;第3部分介绍了该领域的发展现状;第4部分研究了搜索空间的静态表示及搜索方法;第5部分探讨了搜索空间的动态表示及搜索方法;第6部分讨论了当前研究中存在的问题及未来研究重点;最后是结论。

2 搜索问题描述

用 $O=o_1^T=o_1,\cdots,o_T$ 表示输入信号的观察向量的时间序列,用 $W=w_1^N=w_1,\cdots,w_N$ 表示由词典中的词组成的词串,那么语音识别中搜索的主要任务就是找到一个 $\hat{W}$,使它满足

$$\hat{W}=\arg\max_W P(W|O)=\arg\max_W \frac{P(O|W)P(W)}{P(O)} \quad (1)$$

其中第二步是根据Bayes法则得到的。考虑到在计算不同词串 W 时,$P(O)$ 是一个常数值,式(1)也可以写作:

$$\hat{W}=\arg\max_W\{P(O|W)P(W)\} \quad (2)$$

式子(2)的计算需要以下两种知识[1]:

(1)声学模型 $P(x_1\cdots x_T|w_1\cdots w_N)$。声学模型是识别系统的底层模型,其目标是通过模型度量,寻找语音特征向量序列对应的发音。当前常用的声学模型是隐马尔可夫模型(HMM),用HMM对识别基元进行建模,然后通过语料库对这些基元进行训练。在大词汇量连续语音识别系统中,识别基元通常是比词小的子词单位,英语识别中常用的是音素,汉语识别中常用的是声韵母、半音节或音节等。在连续语音识别系统中,人们在发音时普遍会受到上下文的影响而发生变化,这就是连续语音之间协同发音现象。为了得到比较高的识别率,通常采用上下文相关的(Context Dependent,CD)声学模型。根据发音词典对这些基元模型进行连接,就得到了词模型。

(2)语言模型 $P(w_1\cdots w_N)$。它对词典中的词形成句子施加了一个概率意义上的约束,这些约束来自于句法、语法和语用。在大词汇量语音识别中,通常将 $P(w_1\cdots w_N)$ 分解成条件概率 $P(w_n|w_1\cdots w_{n-1})$ 的乘积,其中 $1\leqslant n\leqslant N$。而 $P(w_n|w_1\cdots w_{n-1})$ 可以用二元文法(Bigram)或三元文法(Trigram)模型来近似,即

$$P(w_n|w_1\cdots w_{n-1})=p(w_n|w_n-1) \quad (3)$$

$$\text{或 } P(w_n|w_1\cdots w_{n-1})=p(w_n|w_{n-2}w_{n-1}) \quad (4)$$

3 国内外研究现状

根据在搜索过程中各种知识使用的时机可将搜索分为一遍(One-pass)搜索[2]和多遍(Multi-pass)搜索[3]。一遍搜索

*)国家自然科学基金资助项目(60473003);东北师范大学自然科学青年基金资助项目(20061006)。杨凤芹 助教,博士研究生,主要从事语音识别的研究;孙吉贵 教授,博士生导师,主要从事自动推理、约束程序设计及智能信息处理的研究。

维普资讯 http://www.cqvip.com

是将所有可利用的知识合为一个统一的路径评价标准，通过一遍扫描语音信号便给出候选结果。多遍搜索则是将可利用的知识根据不同的需要，在搜索的不同阶段，单独地或部分集成地作为路径的评价标准，进行多遍扫描给出候选结果。本文的研究只限于一遍搜索。在一遍搜索中，搜索空间的扩展模式可分为静态扩展和动态扩展两种。静态扩展即在搜索之前，将搜索空间完整地表示出来，而动态扩展是一边搜索一边围绕搜索路径扩展搜索空间。

在静态扩展搜索空间方面，Hanazawa[4]等人的研究结果表明，若识别任务的有限状态文法的词集很大，并且是等概率的词集，则有向无环词图（Directed Acyclic Word Graph DAWG）是一种高效的搜索空间表示方法。如果识别器采用的是N元文法（N-gram）语言模型，那么需要利用优化技术对搜索空间进行优化，可以运用确定有限状态自动机中状态的不可区分性优化搜索空间[5]。近来 Riley 等人[6]提出了一种新的搜索空间表示方法，将知识（词典、语言模型和上下文相关的声学模型）分别表示成加权的有限状态转换器（Weighted Finite State Transducer WFST），再用诸如转换器复合、转换器确定化及加权自动机中 ε-删除等算法将这些转换器以最优的方式组合起来。这种方法可以在一遍识别中使用复杂的声学模型和语言模型。

动态扩展搜索空间方法在识别过程中根据需要为每个遇到的声学上下文或语法上下文创建词法树[7]。动态创建搜索空间的方法可分为基于词的搜索和基于时间的搜索。在基于词的搜索中，每个搜索假设依赖于前驱词。这意味着在搜索过程的早期阶段已经对前驱词的未知的终止时间进行了优化，也就是说对正在考察的词与前驱词之间的词边界进行了优化。在基于时间的搜索中[8]，对每个搜索假设来说需要明确地保留前驱词的终止时间，并在搜索的最后一步对未知的词边界进行优化，动态地创建搜索空间。这种方法的优点是声学模型和语言模型高度分离，可以灵活地选择语言模型，缺点是在识别时需要额外的计算。下面我们将具体讨论现有的搜索空间表示及搜索方法。

4 搜索空间的静态表示及搜索方法

4.1 搜索空间的静态表示

实际上，在搜索前将搜索空间完整地表示出来，一直是LVCSR中最自然的生成搜索空间的方法。如 HKT[9]中采用的就是静态扩展搜索空间的方法。HTK 先根据文法规则得到描述词和词之间转移情况的词网络。如图1就是一个简单的词网络。然后根据发音词典将词级的网络扩展成模型级的搜索网络。具体扩展方法是将词网络中的每个词转换成词结束（Word-end）节点，该节点前面是与该词的发音相对应的模型序列。例如，对图1的词网络进行扩展后得到的单音子（Monophone）搜索网络如图2所示。这种扩展方法简单直接，但当识别器采用复杂的声学模型和语言模型时，随着词典规模的不断变大，用这种方法产生的搜索网络的规模会急剧增大。为了减小网络规模，提高识别效率，需要使用一些优化技术来压缩网络。研究表明，当识别任务的有限状态文法的词集很大，并且是等概率的词集时，就可以将网络表示成DAWG[4]。DAWG是有限状态自动机的一个特例，不允许存在环。DAWG可以使不同的词之间共享音素，这样在很大程度上压缩了搜索网络的规模。图3给出了一个用DAWG表示搜索网络的例子[10]。用DAWG表示搜索网络时可能会在任意节点合并弧，所以要对传统的搜索方法做一些修改，通常采用前向搜索与回溯相结合的方法[4]。

搜索空间的DAWG表示方法要求识别任务的语言模型的词集是等概率的，当语言模型不满足这个要求时就需要采用其他的优化技术来压缩网络，许多研究者运用确定有限状态自动机理论对搜索网络进行优化[6,11]。AT&T 在1998年提出了用 WFST 静态扩展搜索网络的方法[12]。经过几年的发展和完善，当前这种方法已经成为大词汇量解码器中静态表示搜索空间的最常用的方法[13~16]。

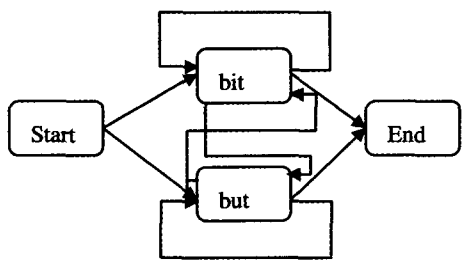

图1 一个简单的词网络

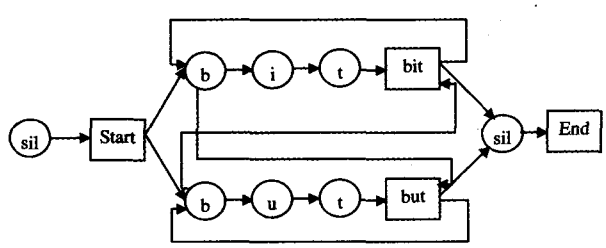

图2 Bit-But 网络的单音子扩展

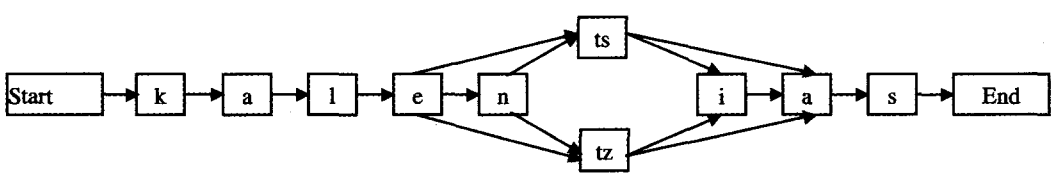

图3 用DAWG表示搜索网络的示例

WFST 是一个有限状态网络，每条弧上标有一个输入符号、一个输出符号和一个权值。输入或输出符号可以为空符号 ε，表示此弧不接收输入或不产生输出。转换器中的一条路径将该路径上的输入符号序列和输出符号序列构成一个序偶，并定义这个序偶的权值为该路径上所有权值的和[12]。

若 S 和 T 是两个加权转换器，S 以权值 u 将符号序列 x 映射到序列 y，T 以权值 v 将 y 映射为 z，则可以定义加权转换器 S 和 T 的复合为 $S\circ T$，$S\circ T$ 以权值 ω 将符号序列 x 映射到序列 z，其中 $\omega=u+v$[17]。WFST 为语音识别中知识的表示提供了一个优美的、统一的框架，这种方法可利用转换器的复合（composition）算法将多个级别的模型组合起来，并且可将搜索网络优化到 HMM 状态级。

维普资讯 http://www.cqvip.com

实际应用时，将各种知识分别表示成转换器的形式，最后的搜索空间即为这些转换器经过复合运算后所得到的转换器。如可构造上下文相关转换器 C、词典转换器 L 和语言转换器 G[12]。C 把上下文相关音素模型(HMM)的名字序列映射成相应的音素序列，则 C^{-1} 将音素序列映射为 HMM 名字序列。例如，图 4 中的转换器 C^{-1} 把两个单音子 x,y 转换成了上下文相关的三音子(Triphone)。它描述了从上下文不相关音素到上下文相关 HMM 的映射，图中将上下文相关的 HMM 描述为“音素/左上下文_右上下文”。状态(a,b)表示前一个音素是 a，后一个音素是 b；ε 表示音素序列的开始或结束，* 表示下一个音素是任意的。从图中可以看出，转换器通过唯一的状态序列$(\varepsilon,*)(x,y)(y,x)(x,\varepsilon)$将音素序列 xyx 映射为 $x/\varepsilon_y \quad y/x_x \quad x/y_\varepsilon$。$L$ 描述了词的发音，根据发音词典将音素序列映射成词序列。G 描述了词序列的概率，根据词序列的语言模型概率将词序列映射到自身。$G\circ L\circ G$ 中的每个路径将 HMM 名字序列和词序列构成了一个序偶，根据词序列的发音为 HMM 序列的概率为该序偶分配一个权值。$G\circ L\circ G$ 即为解码器的搜索网络。构造完整的 $G\circ L\circ G$ 网络需要转换器复合算法[18]、加权转换器的确定化算法[19,20]和加权自动机的 ε 删除算法。

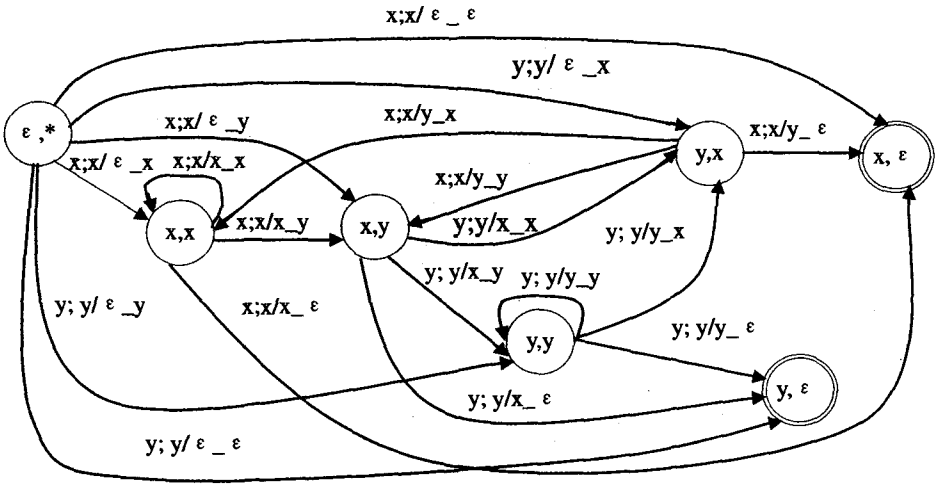

图 4　非确定的上下文相关模型 C^{-1}

4.2　静态搜索空间的搜索方法

在上述的表示方法中，事先已经把搜索空间完整地表示出来了，那么很多搜索算法都可以完成识别。实际中常用的是时间同步 Viterbi 搜索算法[21]。

时间同步 Viterbi 搜索是利用动态规划技术实现的宽度优先搜索，用 $t-1$ 时刻所有状态中的最高得分来更新 t 时刻的状态，并用回溯指针记录从 $t-1$ 时刻传递过来最高得分的状态。搜索结束时可以通过回溯指针得到最佳的状态序列。

在小规模语音识别系统中，Viterbi 搜索是非常有效的，能保证找到最优路径。然而对 LVCSR 问题来讲，穷尽的动态规划搜索需要的计算量太大，需要采用各种剪枝技术来减小搜索空间的规模。beam 剪枝是一种行之有效的剪枝技术，该方法不是在每个时刻 t 把所有的搜索路径都保留下来，而是通过设置一个阈值 T 只保留一些最好的路径。在 t 时刻首先找到具有最小代价 $D_{\min}$ 的状态，然后删除该时刻代价大于 $D_{\min}+T$ 的那些状态。

下面具体说明时间同步 Viterbi beam 搜索算法。首先定义如下变量[22]：

$D(t;s_t;w)$表示 t 时刻到达词 w 中状态 s_t 的最佳路径的代价；

$h(t;s_t;w)$表示 t 时刻到达词 w 中状态 s_t 的最佳路径的回溯指针。

有两种类型的动态规划转移规则，即词内部转移和词间转移。词内部转移的计算公式如下：

$$D(t;s_t;w)=\min_{s_{t-1}}\{d(x_t,s_t\mid s_{t-1};w)+D(t-1;s_{t-1};w)\} \quad (5)$$

$$h(t;s_t;w)=h(t-1,b_{\min}(t;s_t;w);w) \quad (6)$$

其中，$d(x_t,s_t\mid s_{t-1};w)$是从状态 s_{t-1} 转移到状态 s_t 并产生观察向量 x_t 的代价；$b_{\min}(t;s_t;w)$是 $D(t;s_t;w)$的最优前驱状态。

词间转移是空转移，不产生任何的观察，但需要为回溯指针创建一个新节点。设 $F(w)$为 w 的终止状态，$I(w)$为 w 的初始状态，η 为伪初始状态。词间转移具体描述如下：

$$D(t;\eta;w)=\min_{v}\{\log P(w\mid v)+D(t;F(v);v)\} \quad (7)$$

$$h(t;\eta;w)=\langle v_{\min},t\rangle::h(t,F(v_{\min});v_{\min}) \quad (8)$$

其中，$v_{\min}=\arg\min_{v}\{\log P(w\mid v)+D(t;F(v);v)\}$，::表示链接追加操作。

时间同步 Viterbi beam 搜索算法先计算所有词的内部转移，然后再计算词间空转移。因为词间空转移不产生任何的观察向量，所以在计算词间转移的时候时间索引 t 保持不变。由于 w 的初始状态 $I(w)$可以有到自身的转移，因此此时可能已经有到 w 的初始状态 $I(w)$的路径，其代价为 $D(t;I(w);w)$，所以，在继续进行词间转移之前需要进行如下检查：

如果 $D(t;\eta;w)<D(t;I(w);w)$，那么

$$D(t;I(w);w)=D(t;\eta;w) \quad (9)$$

$$h(t;I(w);w)=h(t;\eta;w) \quad (10)$$

研究实验结果表明，对于大词汇量语音识别，在每个时刻 t 只需保留完整搜索空间的一小部分，通常将 beam 设置为完整搜索空间的 5%到 10%。

5　搜索空间的动态表示及搜索方法

5.1　搜索空间的动态表示

搜索空间的动态表示不是事先将搜索空间完整地表示出来，而是在识别的过程中根据搜索的需要动态地扩展搜索空间[23,24]。这种方法将发音词典表示成前缀树，也称为词法树

维普资讯 http://www.cqvip.com

(Lexical Tree)。词法树能使所有词共享前缀,因此大大减小了搜索空间。这个词法树只需存储一次,动态扩展搜索空间时用的是虚节点和仅包含扩展所需信息的临时结构。可根据语法上下文或词的开始时间扩展搜索空间,因此有以下两种方法动态构造 N 元文法的树形搜索空间,逐渐生成词序列[21]:

(1)重入树(Re-entrant Tree),又称为基于词的搜索(Word-conditioned Search)。这种方法如图 5 所示,在扩展搜索空间的过程中依据的是语法上下文,对每个活动的语法上下文考察一个虚树副本,将这些信息记录在每个路径上。根据所采用的语言模型,在虚根节点进行路径的合并。

(2)起始同步树(Start-synchronous Tree),又称为基于时间的搜索(Time-conditioned Search)。如图 6 所示,该策略在扩展搜索空间的过程中依据的是词的起始时间,在要求后继词假设的时刻考察一个虚树副本。在给定的时刻,通过考察该时刻的虚树副本,来扩展在该时刻到达词结尾的所有路径假设。

在这两种搜索空间动态扩展的方式中都可以用时间同步的动态规划搜索,而在第二种方式中还可以用时间异步的搜索方法。

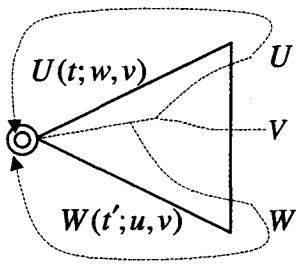

图 5 动态网络扩展的重入词法树

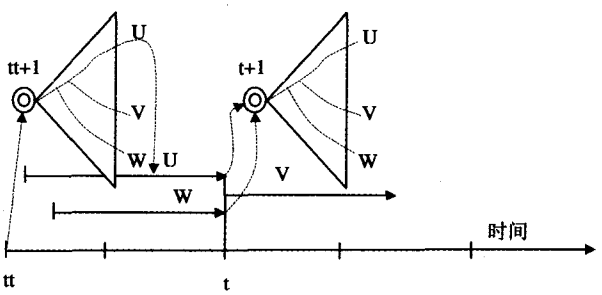

图 6 动态网络扩展的起始同步树

5.2 基于重入词法树的时间同步搜索

基于重入词法树的时间同步搜索和其它时间同步搜索方法一样,并行产生状态假设,根据输入信号的同一部分对这些假设打分,并可直接实现 beam 剪枝。为了正确处理 N 元文法概率,必须知道每个活动路径中所有 $N-1$ 个词的信息,在下一个词的结束状态进行路径合并。下面以三元文法为例来描述该搜索算法。

首先定义如下变量[25,26]:

$Q_{uv}(t,s)$表示以(u,v)为前驱词对,在 t 时刻到达词法树中状态 s 的最佳路径的得分;

$B_{uv}(t,s)$表示以(u,v)为前驱词对,在 t 时刻到达词法树中状态 s 的最佳路径的开始时间。

词内部搜索的动态规划迭代计算公式为

$$Q_{uv}(t,s)=\max_{\sigma}\{q(x_t,s|\sigma)\cdot Q_{uv}(t-1,\sigma)\} \tag{11}$$

$$B_{uv}(t,s)=B_{uv}(t-1,\sigma_{uv}^{\max}(t,s)) \tag{12}$$

其中 $q(x_t,s|\sigma)$是 HMM 声学模型的转移概率和输出概率的乘积,$\sigma_{uv}^{\max}(t,s)$是以(u,v)为前驱词对,在 t 时刻到达状态 s 的最优前驱状态。

在到达词边界时,根据前驱词对路径进行合并的公式为

$$H(v,w;t)=\max_{u}\{p(w|u,v)\cdot Q_{uv}(t,S_v)\} \tag{13}$$

其中,$p(w|u,v)$是词三元组(u,v,w)的三元文法概率,S_w 是词 w 在词法树中的最后一个状态。

在开始进入新的词法树搜索时,初始化如下:

$$Q_{uv}(t-1,0)=H(u,v;t-1) \tag{14}$$

$$B_{uv}(t-1,0)=t-1 \tag{15}$$

其中 $s=0$ 是为了初始化引入的一个假想的状态。

5.3 基于起始同步树的时间同步搜索

这种搜索方法依据的是前驱词的结束时间或者说是后继词的开始时间[24,26,27]。为了满足语言模型的约束,需要在可能的词边界进行路径合并。下面仍然以三元文法为例描述该搜索算法,先定义如下几个量:

$Q_\tau(t,s)$表示在词法树内部搜索过程中,在 t 时刻,到达以 τ 为开始时间的词法树中状态 s 的最佳路径的得分;

$H(u,v;\tau)$表示以 uv 为最后两个词且 τ 为词边界的词(状态)序列产生语音向量 $x_1\cdots x_\tau$ 的概率。

在开始新的词法树搜索时,按照下面的式子进行初始化:

$$Q_{t-1}(t-1,s)=\begin{cases}\max\limits_{(u,v)}H(u,v;t-1), & s=0\\ 0, & s>0\end{cases} \tag{16}$$

状态 $s=0$ 是用来初始化的。

对词(树)内部的搜索使用常规的动态规划公式为

$$Q_\tau(t,s)=\max_{\sigma}\{q(x_t,s|\sigma)Q_\tau(t-1,\sigma)\} \tag{17}$$

在词边界进行搜索路径合并的计算比较复杂,为了有效地完成计算,引入一个附加的变量:

$$\hat{H}(u,v,w;t)=\max_{\tau}\left\{\frac{H(u,v;\tau)}{\max\limits_{(u'v')}H(u',v';\tau)}\cdot Q_\tau(t,S_w)\right\} \tag{18}$$

$$H(v,w;t)=\max_{u}\{p(w|u,v)\cdot\hat{H}(u,v,w;t)\} \tag{19}$$

其中,$p(w|u,v)$是词三元组(u,v,w)的三元文法概率,S_w 是词 w 在词法树中的最后一个状态。

从以上的计算可以看出,在 t 时刻为了找到词 w 的最佳前驱词,需要执行两步优化。第一步确定词 w 的开始时间 τ;第二步根据三元文法概率 $p(w|u,v)$进行路径合并。用 t 时刻最佳路径的得分 $H(v,w;t)$启动 $t+1$ 时刻的新词法树的搜索。

5.4 基于起始同步树的时间异步搜索

在 LVCSR 中,时间异步搜索是以深度优先的方式进行的,总是选择当前得分最高的路径进行扩展,并不要求这些路径是结束在同一时刻的。在选择要扩展的路径时,需要用启发式函数对当前每个路径进行估计,t 时刻路径 h 的估价函数 $f_h(t)$可定义如下[28]:

$$f_h(t)=a_h(t)+b_h^*(t) \tag{20}$$

其中,$a_h(t)$是到 t 时刻为止部分路径的得分,$b_h^*(t)$是对该部分路径扩展成合法的完整路径时最好得分的估计。该搜索方法是通过栈来实现的,可以用一个栈[29]也可以用多个栈[30]。栈搜索是最佳优先算法,从栈顶弹出最好的部分假设路径(也就是 $f_h(t)$最大的路径),用一个词扩展该路径并将扩展后得到的路径压回栈。如果 $f_h(t)$是可纳的,那么从栈中弹出的第

维普资讯 http://www.cqvip.com

一个完整的假设路径就是最有可能的语音模型。

6 存在的问题及未来研究重点

尽管已经有很多学者对大词汇量连续语音识别中搜索空间的表示和搜索方法进行了研究，并取得了很大的进展，但仍有许多方面不尽人意，比如：

(1)虽然用 WFST 静态构建搜索空间时可以非常灵活地处理知识，但在处理非常大的语言模型时对内存的需求太大，这是个亟待解决的问题。

(2)搜索空间的动态表示虽然能减少内存需求，能使语言模型和声学模型高度分离，但在识别过程中需要的计算过于复杂。

根据对大词汇量连续语音识别中搜索空间的表示和搜索方法研究的现状分析，我们认为如下方向可能成为未来该领域研究的主要方向：

(1)为了解决 WFST 静态构建搜索空间时所导致的内存需求瓶颈问题，可从以下方面入手：

(a)研究如何在搜索空间预处理阶段减少内存需求。

(b)研究是如何处理最终得到的网络：应该将其载入内存中还是留在磁盘上进行按需访问。

(2)研究搜索空间的混合表示方法，先将某些知识静态地表示出来，然后再动态地集成其它的知识，结合二者的优点，从而提高表示和搜索的效率。

(3)研究如何采用更长范围的语法约束来增强语言模型的预测能力，而不只限于二元或三元文法模型。

结论 大词汇量连续语音识别中搜索空间的表示及搜索方法依赖于所采用的声学模型和语言模型。为了提高识别的精度，通常需要采用复杂的声学模型和语言模型，因此搜索空间变得非常庞大。虽然目前提出了一些搜索空间的表示及搜索方法，但还有很多问题需要进一步的研究。语音识别系统要想在未来信息化社会中适合各种应用场合—尤其是嵌入式环境，搜索空间的表示及搜索方法必须在保证识别精度的前提下，具备内存消耗小、运算速度快的特点。这方面的研究若能取得突破，将会对语音识别的应用产生巨大的推动作用。

参 考 文 献

1 罗春华. 连续语音识别中的搜索策略研究：[硕士学位论文]. 北京：清华大学计算机科学与技术系，2001. 4～12

2 Ney H, Haeb-Umbach R, Tran B H, et al. Improvements in Beam Search for 10, 000-word Continuous Speech Recognition [A]. In: Proceedings of IEEE International Conference on Acoustic, Speech and Signal Process-ing [C], San Francisco, CA, 1992. 13～16

3 Li Z, Boulianne G, Labute P, et al. Bi-directional Graph Search Strategies for Speech Recognition [J]. Computer Speech and Language, 1996, 10(4): 295～321

4 Hanazawa K, Minami Y, Furui S. An efficient search method for large-vocabulary continuous-speech recognition [A]. In: Proceedings ICASSP [C], Munich, ALLEMAGNE, 1997, 3: 1787～1790

5 Federico M, Cettolo M, Brugnara F, et al. Language modeling for efficient beam-search [J]. Computer Speech and Language, 1995, 9(4): 353～379

6 Riley M, Pereira F, Mohri M. Transducer composition for context-dependent network expansion [A]. In: Proceedings EUROSPEECH [C], Rhodes, Greece, 1997, 3: 1427～1430

7 Beyerlein P, Ullrich M, Wilcox P. Modeling and decoding of crossword context dependent phones in the Philips large vocabulary continuous speech recognition system [A]. In: Proceedings EUROSPEECH [C], Rhodes, Greece, 1997, 3: 1163～1166

8 Ney H, Ortmanns S. Dynamic programming search for continuous speech recognition. IEEE Signal Processing Magazine, 1999, 16(5): 64～83

9 Young S, Jansen J, Odell J, et al. The HTK Book, version 2. 1. Distributed with the HTK toolkit

10 Georgila K, Fakotakis N, Kokkinakis G. Large Vocabulary Search Space Reduction Employing Directed Acyclic Word Graphs and Phonological Rules [J]. International Journal of Speech Technology, 2002, 5(4): 355～370

11 Antoniol G, Carli G, Cettolo M, et al. Tools for development, test and analysis of ASRs [A]. In: Proceedings ICSPAT [C], Cambridge, MA, 1992. 1005～1010

12 Mohri M, Riley M, Hindle D, et al. Full expansion of context-dependent networks in large vocabulary speech recognition [A]. In: Proceedings ICASSP [C], WA, Seattle, 1998. 665～668

13 Allauzen C, Mohri M. An optimal pre-determinization algorithm for weighted transducers [J]. Theoretical Computer Science, 2004, 328(1-2): 3～18

14 Stoimenov E, McDonough J. Modeling Polyphone Context with Weighted Finite-State Transducers [A]. In: Proceedings ICASSP [C], Toulouse, France, 2006

15 Schuster M, Hori T. Efficient generation of high-order weighted finite state transducers for speech recognition [A]. In: Proceedings ICASSP [C], PA, USA, 2005, 1: 201～204

16 Mohri M, Pereira F, Riley M. Weighted finite-state transducers in speech recognition [J]. Computer Speech and Language, 2002, 16(1): 69～88

17 Berstel J. Transduction and Context-free Languages. Teubner Studienbucher, Stuttgart, Germany, 1979

18 Mohri M, Pereira F, Riley M. Weighted automata in text and speech processing [A]. In: ECAI'96Workshop on Extended Finite State Models of Language [C], Budapest, Hungary, 1996. 46～50

19 Mohri M. On some application of finite-state automata theory to nature language processing [J]. Journal of Natural Language Engineering, 1996(2): 1～20

20 Mohri M. Finite-State Transducers in Language and Speech Processing [J]. Computational Linguistics, 1997, 23(2): 269～312

21 Aubert X L. An overview of decoding techniques for large vocabulary continuous speech recognition [J]. Computer Speech and Language, 2002, 16(1): 89～114

22 Huang Xuedong, Acero A, Hon Hsiao-Wuen. Spoken Language Processing: A Guide to Theory, Algorithm and System Development [M]. Prentice Hall, Apr. 2001. 622～626

23 Odell J J, Woodland P C, Young S J. A one pass decoder design for large vocabulary recognition [A]. In: Proceedings of the ARPA Spoken Language Technology Workshop [C], Plainsboro, NJ, 1994. 405～410

24 Ney H, Ortmanns S. Progress in Dynamic Programming Search for LVCSR [A]. Proceedings of the IEEE [C], 2000, 88(8): 1224～1240

25 武健. 汉语语音识别中统计语言模型的构建及其应用：[硕士学位论文]. 北京：清华大学计算机科学与技术系，2000. 43～47

26 Ortmanns S, Ney H, Seide F, et al. A Comparison Of Time Conditioned and Word Conditioned Search Techniques for Large Vocabulary Speech Recognition [A]. In: International Conference on Spoken Language Processing [C], Philadephia, PA, 1996, 4: 2091～2094

27 Ney H, Ortmanns S, Lindam I. Extensions to the Word Graph Method for Large Vocabulary Continuous Speech Recognition [A]. In: Proceedings ICASSP [C], Munich, Germany, 1997, 3: 1791～1794

28 Renals S, Hochberg M M. Start-synchronous search for large vocabulary continuous speech recognition [J]. IEEE Trans Speech and Audio Processing, 1999, 7(5): 542～553

29 Paul D B. An efficient A* decoder algorithm for continuous speech recognition with a stochastic language model [A]. In: Proceedings ICASSP [C], San Francisco, Estados Unidos, 1992, 1: 25～28

30 Hochberg M, Renals S, Robinson A, et al. Large vocabulary continuous speech recognition using a hybrid connectionist-HMM system [A]. In: Proceedings of the International Conference on Spoken Language Processing [C], Yokohama, Japan, 1994. 1499～1502