基于加权有限状态机的动态匹配词图生成算法

郭宇弘* 黎 塔 肖业鸣 潘接林 颜永红
(中国科学院语言声学与内容理解重点实验室 北京 100190)

摘 要：由于现有的加权有限状态机(WFST)解码网络没有精确词尾标记，导致当前已有的词图生成算法不含精确的词尾时间点，或者仅是状态、音素级别的词图，无法应用到关键词检索中。该文提出在 WFST 静态解码器下的语音识别词图生成算法。首先从理论上分析了 WFST 解码音素图和词图的可转换关系，然后提出了字典的动态音素匹配方法解决了 WFST 网络中词尾时间点对齐的问题，最后通过令牌传递的遍历方法生成了词图。同时，考虑到计算量优化，在令牌传递过程中引入了剪枝算法，使音素图转词图的耗时不到解码耗时的 3%。得到的词图，不仅可以用于语言模型重打分，由于含有精确的词尾时间点，还可以直接应用到关键词检索系统中。实验结果表明，该文的词图生成算法具有较高的计算效率；和已有动态解码器的词图相比，词图中包含更多解码信息，在大词汇连续语音识别的重打分结果和关键词检索中都能取得更好的性能。

关键词：自动语音识别；加权有限状态机；词图生成；关键词检索
中图分类号：TP391.42 **文献标识码**：A **文章编号**：1009-5896(2014)01-0140-07
DOI: 10.3724/SP.J.1146.2013.00422

Exact Word Lattice Generation in Weighted Finite State Transducer Framework

Guo Yu-hong Li Ta Xiao Ye-ming Pan Jie-lin Yan Yong-hong
(*Key Laboratory of Speech Acoustics and Content Understanding*, *Chinese Academy of Sciences*, *Beijing* 100190, *China*)

Abstract: The existing lattice generation algorithms have no exact word end time because the Weighted Finite State Transducer (WFST) decoding networks have no word end node. An algorithm is proposed to generate the standard speech recognition lattice within the WFST decoding framework. The lattices which have no exact word end time can not be used in the keyword spotting system. In this paper, the transformation relationship between WFST phone lattices and standard word lattice is firstly studied. Afterward, a dynamic lexicon matching method is proposed to get back the word end time. Finally, a token passing method is proposed to transform the phone lattices into standard word lattices. A prune strategy is also proposed to accelerate the token passing process, which decreases the transforming time to less than 3% additional computation time above one-pass decoding. The lattices generated by the proposed algorithm can be used in not only the language model rescoring but also the keyword spotting systems. The experimental results show that the proposed algorithm is efficient for practical application and the lattices generated by the proposed algorithm have more information than the lattices generated by the comparative dynamic decoder. This algorithm has a good performance in language model rescoring and keyword spotting.

Key words: Automatic speech recognition; Weighted Finite State Transducer (WFST); Lattice generation; Keyword spotting

1 引言

作为大词表连续语音识别的核心模块，语音识别解码器负责利用上下文相关的声学模型、字典和语言模型等知识源把语音信号转换为文本。评价语音识别解码器性能的一项关键指标就是识别器的准确率。在非常理想的情况下，语音识别应具有非常高的识别准确率，此时仅仅选用语音识别的解码首选结果就可以使语音搜索、关键词检错等应用的准确率非常高。然而，考虑到现实应用经常出现的信道不匹配、说话人不匹配或者说话人发音不标准的

2013-04-01 收到，2013-07-18 改回
国家自然科学基金(10925419, 90920302, 61072124, 11074275, 11161140319, 91120001, 61271426)，中国科学院战略性先导科技专项 (XDA06030100, XDA06030500)，国家 863 计划项目(2012AA012503)和中科院重点部署项目(KGZD-EW-103-2)资助课题
*通信作者：郭宇弘 guoyuhong@hccl.ioa.ac.cn

问题，导致大词表连续语音识别(Large Vocabulary Continuous Speech Recognition, LVCSR)的首选结果在电话环境一类语音的识别错误率通常在 40%左右。在这种较低准确率的情况下仅仅使用解码的首选结果往往是不够的。识别结果可以以多候选(N-Best)或者词图等形式输出，这种多候选或者词图结果保留了识别中的更多识别信息，把它们交由后处理模块能有效提高识别结果的准确性。常见的后处理技术包括：基于词图的重打分[1]、多遍解码[2]、混淆网络[3]等。

和多候选结果相比，词图形式包含了更多的信息，它不仅有多个识别词序列结果，更包含了每个词、音素的声学得分、语言得分以及时间点等信息，并且它合并了多候选的冗余信息，其表示也更加高效[4]。因此，词图在语音识别后处理中得到非常广泛的应用。例如：可以从词图里面直接抽取多候选结果；另外词图本身已经具有了图的性质，在某些场合第 1 遍解码用比较精细的模型会带来计算量过高的问题，此时可以用简单的模型在第 1 遍解码时生成词图，再用精细的模型在词图上进行 2 遍解码或者重打分则可获得更好的效果；而在关键词检索的应用中，词图或者词图的混淆网络形式可以作为检索器的输入。因此，词图成为了语音识别中第 1 遍解码和后处理模块之间的桥梁。

词图的生成过程是由解码器搜索解码网络，记录下搜索路径从而转化成相应的词图。解码网络是由各个知识源构成的一个搜索空间，一般来讲可以分为动态构建的解码网络和静态网络。基于动态网络的解码器，以前缀树的发音词典作为搜索网络，语言模型则通过动态查询的方式把得分引入解码过程之中，然后利用重入字典树或者字典树拷贝的方式对整个解码网络进行搜索[5]。动态网络解码器的优势在于，由于字典和语言模型是分离的，其占用内存较少，同时，由于搜索空间为一个前缀树的字典，字典里面有准确的词尾节点，这样，在进行词图生成的时候，可以准确地获取到词尾时间点。然而，动态网络解码器的最大缺点在于它的时间复杂度较高[6]，相对于静态网络解码器，它的速度较慢。对于当今的大规模的语音识别应用，往往需要更快的响应速度，因而解码速度更快的静态网络解码器更加适合。静态网络的解码器基于加权有限状态机(Weighted Finite State Transducer, WFST)[7]。WFST 解码器的特点是实现简单，解码速度快，对于知识源有统一的建模方式，并且它具有完善的理论框架以及成熟的优化算法。应用在语音识别的 WFST 网络输入一般为上下文相关的三音素或者隐马尔科夫模型(HMM) 状态，输出为识别词。为了让解码器网络得到充分优化加快解码速度和降低解码的内存占用，解码网络中不含词边界信息，这就为 WFST 解码器生成含有精确时间点的词图造成了一定困难。文献[8]中提出了最早的 WFST 解码器的词图生成算法，准确说文献[8]是介绍了一种记录 WFST 格式的解码路径的算法，它并不包含词的边界和时间信息，它产生的词图主要用于语言模型重打分。文献[6]提出了在构建解码网络的时候插入额外的词尾标记用于找回词尾时间信息，但额外的词尾标记会导致解码网络得不到充分优化，从而网络变大，并且，解码网络格式的变化也导致解码网络的使用缺乏兼容性，需要为生成词图的解码器重新构建网络。Povey 等人[9]提出了一种 WFST 的词图生成算法并应用在开源项目 Kaldi [10]中。但是这种算法产生的是一种 HMM 状态级别的词图，仍然不是标准的词图。文献[9]在文中提到不同解码器产生的词图在格式上不统一的问题，要做统一的比较和解释比较困难。

本文在给出了语音识别标准词图和 WFST 的解码音素图的定义之后，探索了两者之间的联系，提出在 WFST 解码器下的词图生成算法。本文首先提出了一种动态字典匹配的方法，此方法可以用来进行词的时间点对齐，解决了 WFST 解码网络没有精确词尾节点的问题。然后提出了一种基于令牌传递(token passing)的方法，把 WFST 的解码音素图转换为标准词图。由于本文提出的 WFST 词图生成算法生成的是标准的词图，可以应用到已有的重打分、关键词检索等一系列后处理应用中而无需额外操作，且由于没有对网络进行特殊处理，本算法在网络使用上具有兼容性，无需重新构建解码网络。

本文的组织结构如下：第 2 节介绍背景知识，给出了 WFST 的定义和解码框架以及标准词图的定义；第 3 节揭示了 WFST 音素图和词图的联系和映射关系；第 4 节和第 5 节分别给出了词图的生成算法和相应的实验结果及分析；最后，第 6 节给出结论。

2　背景知识

2.1 基于 WFST 的解码框架

WFST 是一个权值定义在半环 K 上的 8 元组[7]：

$$T = (\sum, \Omega, Q, E, i, F, \lambda, \rho) \tag{1}$$

其中 $\sum$ 和 Ω 分别表示输入符号和输出符号，Q 为有限状态集，$i \in Q$ 和 $F \subseteq Q$ 分别为起始状态和最终状态集合，起始状态的权值和最终状态集的权值函

数分别为 λ 和 ρ，边集 $E=Q\times\{\Sigma\cap\{\varepsilon\}\}\times\{\Omega\cap\{\varepsilon\}\}\times\mathrm{K}\times Q$，表示了一种转换功能：从一个状态跳转到另一个状态的同时，可以把输入符号转换为对应的输出符号，同时经过这条边的权值为 $k\in\mathrm{K}$。ε 代表的是空的输入输出符号。

在语音识别中常用的权值半环有 Log 半环和 Tropical 半环[7]，为了达到更精确的识别率，本文采用 Log 半环。最终的静态解码网络的构建可以表示成为

$$F=\pi_{\varepsilon}\left(\min\left(\det\left(C'\circ(L'\circ G')\right)\right)\right) \tag{2}$$

其中 C', L', G' 分别代表了加入辅助符号的上下文相关模型、发音字典和语言模型；min, det 和 $\circ$ 分别代表 WFST 的最小化、确定化和复合操作；π_{ε} 代表的把辅助符号转换成空边的操作。生成的 WFST 的最终解码网络是一个有向有环图。一条 WFST 的路径 π 包含了一个输入序列 $l(\pi)=l(e_1)\cdots l(e_n)$ 和对应的输出序列 $O(\pi)=O(e_1)\cdots O(e_n)$，路径的权值在 Log 半环下表示成为

$$K(\pi)=\lambda+K(e_1)+\cdots+K(e_n)+\rho(e_n) \tag{3}$$

2.2 标准词图

标准词图是一个含有解码信息的有向无环图，可以定义为一个五元组：

$$L_w=(\Omega,Q_w,E_w,i_w,f_w) \tag{4}$$

其中 Ω 为输出的词的集合，和式(1)中的含义相同；Q_w 为状态集合，每个节点 $q\in Q_w$ 含有时间信息 $t(q)$；$i_w\in Q$ 和 $f_w\in Q$ 分别为起始状态和终止状态；E_w 为边集，每条边 $e\in E_w$ 是一个五元组 $e=(S(e),N(e),w,l,\boldsymbol{p})$，包含了这条的开始和目标状态 $S(e)$ 和 $N(e)$，输出词 $O(e)=w\in\Omega$，语言模型得分 l，这个词的发音序列 $\boldsymbol{p}$ 的每一个音素 c_i 和音素的声学模型得分 a_i。语言模型得分和声学模型得分均为概率的对数值，即 l 和 a_i 也为 Log 半环中的元素。一条词路径 π_w 定义为从起始状态 i_w 到终止状态 f_w 的路径，输出词序列 $O(\pi_w)$ 为其所经过的边的输出的串联，即：$O(\pi_w)=O(e_1)\cdots O(e_n)=w_1\cdots w_L$，词路径得分为词图中所有输出相同词序列的得分最大的一个，即：

$$K(\pi_w)=\max_{O(\pi_w)}\left(K(e_1)+\cdots+K(e_L)\right) \tag{5}$$

3 WFST 音素图和标准词图的转换关系

3.1 WFST 解码音素图

WFST 解码器的解码过程是一个基于令牌传递(token passing)[11,12]的维特比剪枝搜索(Viterbi beam search)[13]，在解码的过程中，所有令牌(token)经过的且和终止状态可以连通的路径都被记录下来，生成一个音素图 T_p。但是由于 T_p 记录了解码路径中状态的时间信息，所以 T_p 也可以定义为一个扩展了状态集的 WFST $T_p=(\sum,\Omega,Q',E,i,F,\lambda,\rho)$，扩展状态 $q'\in Q'$ 包含了时间信息 $t=t'(q')$。根据文献[8]，音素图的每一个状态 q' 其实对应了一个二元组 $q'=(q,t)$，$q\in Q$ 为解码网络中的状态，t 为此状态的时间。扩展后的音素图 T_p 不是解码网络 T 子图，相反，由于不同令牌到达解码网络状态的时间不同，解码网络中相同的状态由于其在音素图中的时间不同成为了不同的状态，也因此，虽然解码网络是一个有向有环图，但是音素图状态时间的先后顺序不同使音素图 T_p 成为了一个有向无环图。这是音素图转成标准词图的基础。

3.2 WFST 的音素图和标准词图的联系

基于以上分析，本文给出 WFST 音素图和词图的一种映射关系：对于一个扩展的 WFST 音素图 $T_p=(\sum,\Omega,Q',E,i,F,\lambda,\rho)$ 和一个标准词图 $L_w=(\Omega,Q_w,E_w,i_w,f_w)$，对于在 T_p 中可行路径的词输出 $O(\pi)=O(e_1)\cdots O(e_n)$，用所有输出同样词路径的得分最大值作为此词序列输出的得分，如果任何一条这样的词路径在词图 L_w 中都有一条对应的词路径，且词路径的得分相同，那么称标准词图 L_w 是由 WFST 音素图 T_p 所生成的。由于 WFST 的音素图和标准词图的定义的差别，这种映射关系可以对音素图 T_p 做如下 3 步预处理变换后，再把音素图转为词图：

(1)把 T_p 的初始状态 i 的权值 λ 累计到所有从 i 出来的边上，此时，T_p 中的 i 和 L_w 中的 i_w 对应，无需权值项 λ，在 Log 半环中的情况如图 1 所示；

(2)在 T_p 中加入一个附加的终止状态 f，把终止状态集合 F 中的所有状态 f_i 连接一条输入输出均为空的边 ε、权值为 ρ_i 的边到附加的终止状态 f 上。此时，T_p 的附加终止状态 f 和 L_w 中的 f_w 对应，也无需权值项 ρ，如图 2 所示；

(3)利用词时间点对齐方法，遍历整个预处理后的音素图 T_p，可以选出词时间点对齐后的词边界状态 $Q''\subseteq Q'$ 和词图 L_w 中的 Q_w,对应。具体的词对齐方法和完整的词图生成算法将在下一节介绍。

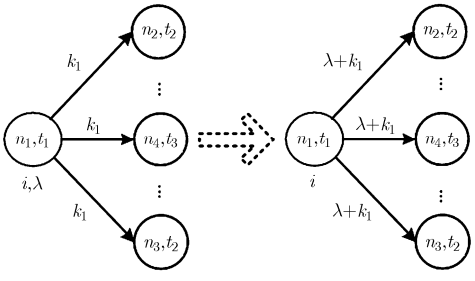


图1 音素图起始状态的权重处理

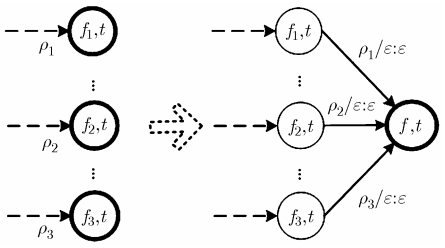


图 2 音素图终止状态集的归一处理

4 WFST 的词图生成算法

4.1 动态词匹配

词图生成的理论基础由文献[4]提出，其中的一个假设是词对无关假设，即：任何一对词的词时间点和这对词的历史无关，只与这对词本身有关。这个假设是针对于动态网络解码器提出的。对于本文上下文相关音素级别的WFST 解码器，由于WFST做了网络优化，词的历史可能不会唯一，但 WFST 输入为上下文相关的三音素模型，词对无关假设可以变为音素对无关假设，即：任何一对音素的时间点和其历史无关。因此，由 WFST 解码器记录下来的音素图中音素的时间点是准确的。但 WFST 的解码网络缺乏明确的词尾标记，解码时候产生的音素图的词输出可能在其发音音素的任何位置，需要一个词时间点对齐的方式来重新找回准确的时间点。

本文不采取直接的字典发音匹配进行词时间点对齐(即只要当前的音素序列和词的一个发音匹配即可进行对齐)。直接的字典发音匹配缺少通用性，其问题在于，当字典中的词存在多发音且一个发音是另外一个发音的前缀的时候，前缀发音总会被优先匹配，而导致长的发音无法得到匹配。例如，英语中的缩写单词“Corp.”，其发音可以是缩写的发音“k ao r p”也可以是完整发音“k ao r p er ey sh ah n ”，缩写的发音是完整发音的前缀。

本文提出动态的词匹配方式进行词时间点对齐。此方法不仅仅是记录一个词就进行词边界的对齐，而是记录多个词和多个音素，动态进行发音匹配。其方法如下：(1)当记录的词序列长度达到 3 个词时开始尝试匹配，要求第 1 个和第 2 个词的发音完全匹配，第 3 个词中已记录的发音要匹配(由于此时第 3 个词的发音可能还未记录完全，无需完全匹配)；(2)这种匹配的方式有且只有一种方法。满足这两个条件时可以确定第 1 个词的边界位置。示例如图 3 所示，词 1 包含两个多发音，前缀发音含有 4 个音素，长发音有 6 个音素，词 2 和词 3 分别含有 3 个及 2 个音素的发音。在尝试匹配时，如果某种匹配方式出现无法匹配的音素，则违背条件(1)，需要更换匹配方式，如图 3 的错误 1；如果匹配前两个词的方式不只一种，则违背了条件(2)，这有可能是因为第 1 个词的发音去掉前缀发音后的部分刚好为第 2 个词的前缀，如图 3 中的错误 2，此时需要加入后面更远的词输入进行匹配(如：加入第 4 个或以上的词)；只有条件(1)和条件(2)同时满足的匹配方法才能确定第 1 个词的边界如图 3 的正确匹配。

4.2 音素图转词图

有了前面引入的音素图的预处理变换和动态词匹配的方法之后，可以通过令牌传递遍历整个音素图 T_p 生成其对应的词图 L_w。遍历过程如下：先对所有的状态进行拓扑排序，然后根据拓扑排序的顺序处理每个状态，进行一个基于令牌传递的搜索过程。在处理每一个状态时，遍历当前挂在此状态下的每个令牌，每个令牌记录了还未被匹配的词和音素的序列，以及所经过的路径的得分和时间信息。首先对当前处理的令牌先进行动态词匹配，如果确定了一个词的位置，则在词图 L_w 中生成一条相应的边和对应的状态(如果此状态已经存在，则连接到状态上)，然后删除令牌中已匹配部分的词和音素；最后，把令牌传递到当前状态的所有后继状态上，同时把其经过的边的信息加入令牌中。

如果仅仅使用令牌传递遍历整个音素图而不引入任何的剪枝策略，此算法的复杂度将会非常高，达到了 $O(|Q|\cdot|E|)$，$|Q|$和$|E|$分别为音素图的状态及边的数目。此计算复杂度在实际应用中不可取，需要引入如下的剪枝策略，去掉冗余计算。如不采用剪枝策略，一个令牌在传到一个状态后，会把它后继的所有路径遍历一遍，另一个传到这个状态的令牌也会重复遍历一遍，造成了冗余计算。因此，要把冗余遍历的令牌剪枝。剪枝的方法为：在每一次处理令牌的时候，如果此令牌找到了动态词匹配的位置并生成了词图中的一条边，如果其目标状态已经存在，则这个令牌到达的状态前面已有令牌到

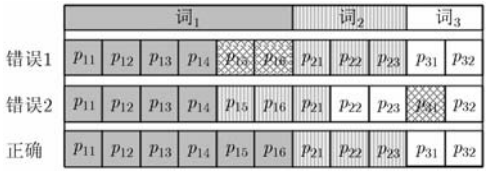


图 3 动态词匹配正确和错误的匹配示例

达，无需重复遍历，可以把此令牌剪枝掉。之所以只对生成边的令牌进行剪枝，这是因为如果对所有的令牌都进行剪枝会产生词图中的悬挂状态(无法到达终止状态)。其算法如表 1 表示。加入令牌剪枝后，算法的计算复杂度降为 $O(|Q|+|E|)$。

表1 音素图转词图的算法描述

算法1. 音素图 T_p 转词图 L_w

(1) 对音素图 T_p 的状态集 Q 进行拓扑排序
(2) 初始化令牌 token_0，把令牌挂在 T_p 的初始状态 i 上
(3) **for all** q **in** Q **do**
(4) **for all** token **in** q , s Token List **do**
(5) **if** 动态词匹确认了词边界 **then**
(6) 在词图中连接旧状态和目标状态
(7) **if** 词图目标状态已存在 **then**
(8) 删除当前 token
(9) **end if**
(10) **end if**
(11) 传播 token 并加入经过边的信息
(12) **end for**
(13) **end for**

5 实验结果和分析

5.1 实验参数设置

本文实验所采用的 WFST 解码器为文献[14]描述的解码器，参与对比的解码器为动态网络解码器 TDecoder [15]。测试任务包含了大词汇连续语音识别和关键词检索两种，每种任务各有两个测试集。具体的测试集和模型参数如表 2 所示，其中电话语音和实网语音由于信道和环境的原因，语音效果较差。

5.2 词图的产生效率

本实验采用电话语音测试集和对应的模型，因为电话语音的信道较差，语音的混淆性较大，解码时产生的路径也较多，能充分体现词图的产生算法的时间效率。对于转词图过程中令牌传递剪枝和未剪枝的情况，本实验记录了这两种情况下核心函数调用的平均次数。如表 3 所示，加入剪枝后，核心函数的调用次数从上百万次降低到 1000 次左右，剪枝对于减少计算量有非常明显的效果。加入剪枝后，转词图部分所占实时率仅为 0.0011，占整个解码实时率的 2.5%，相对解码耗时的比例非常小。因此，本文提出的词图算法具有非常高的效率。仍然保持了 WFST 解码器在应用上快速的优势。

表 2 测试集和模型参数

	测试集名	时长(h)	词典条数	声学模型		语言模型(Million)	
				高斯数目	HMM 状态数	二元文法数	三元文法数
LVCSR	语音输入法	1.0	92k	16	8655	11M	19M
	电话语音	1.0	43k	12	5884	29M	33M
关键词检索	实网语音	3.4	43k	16	20309	9M	35M
	采访对话	2.5					

表 3 转词图的时间效率

未剪枝调用次数	加剪枝调用次数	转词图实时率	转词图时间比例
1142778	1072	0.0011	2.5%

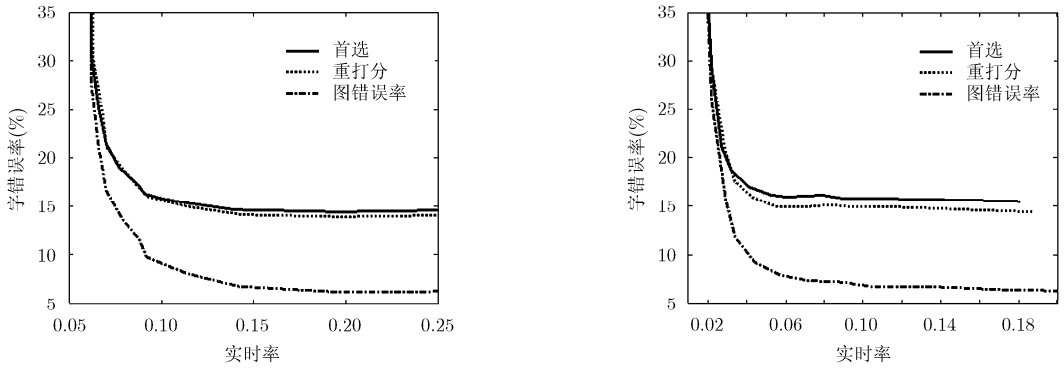
5.3 大词汇连续语音识别

本实验在语音输入法测试集和电话语音测试集上进行。参与对比的参数为解码器的首选字错误率、词图错误率和用一个大的语言模型进行词图重打分后的首选字错误率。词图错误率是指用词图中和答案最匹配的路径计算识别的错误率，它体现了词图用于重打分可以取得的错误率的下限。对于语音输入法测试集，参与重打分的语言模型为一个三元文法语言模型(3-gram)含有 2-gram 30 M 和 3-gram 58 M；对于电话语音进行重打分的模型为五元文法模型，含有 2~5 元文法共 447 M。实验结果分别见图 4 和图 5。

从两个测试集的纵向比较来看，由于电话语音的效果较差，错误率要比语音输入法高很多。但是由于电话语音所采用的重打分语言模型更精准，因此，电话语音在重打分上错误率的下降要比语音输入法更加明显。从两个解码器的横向比较来看，在语音输入法测试集上，本文解码器在首选的极限结果上略差于 TDecoder，但是无论从重打分还是词图错误率都比 TDecoder 的错误率下降更为明显；在电话语音测试集上本文解码器首选结果的极限结果好于 TDecoder，在重打分和词图错误率的下降仍然好于 TDecoder。也就是说，本文的词图生成算法相对于 TDecoder 保留了更多了词路径，包含更多的解码信息。

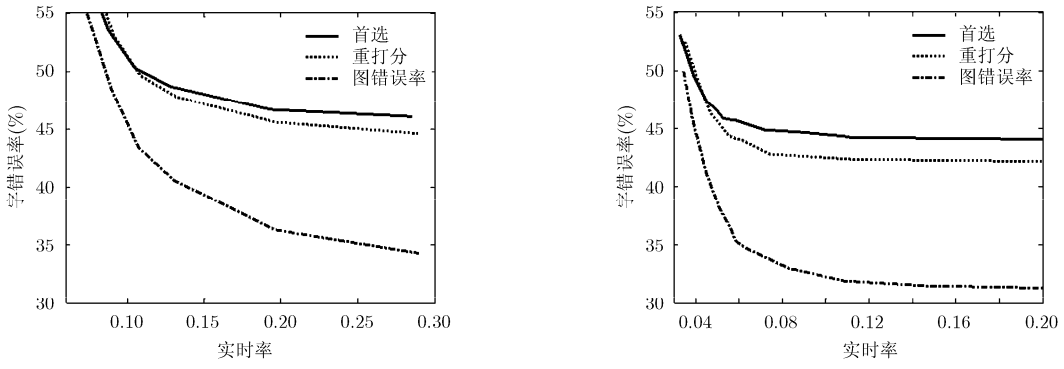
5.4 关键词检索

本文采用的关键词检索系统由文献[16]所述。图 6 给出了在两个关键词测试上，本文方法和



(a) TDecoder词图效果　　(b)本文方法的词图效果

图 4 在语音输入法测试集上的词图效果



(a) TDecoder词图效果　　(b)本文方法的词图效果

图 5 在电话语音测试集上的词图效果

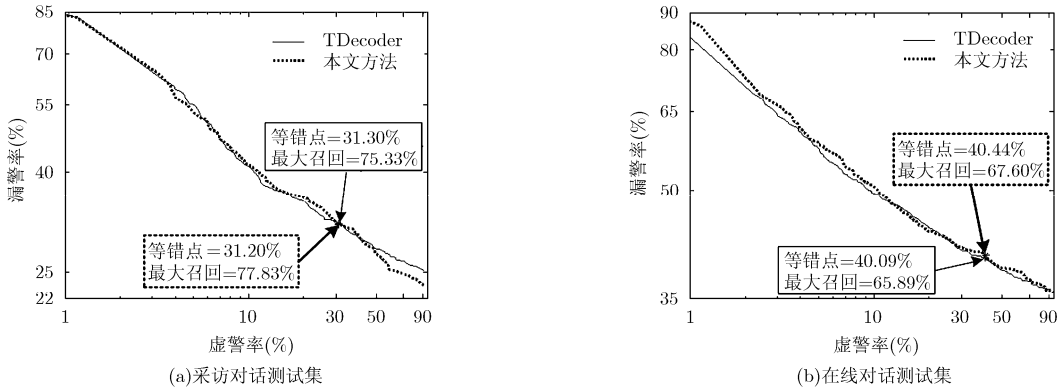
TDecoder 的关键词的检测误差图(Detection Error Tradeoff, DET)。关键词有两个重要的指标：等错点和最大召回率。等错点是 DET 图中漏警率和虚警率相同的点。召回率的定义为

$$R_{\text{Recall}} = \frac{\text{识别的关键词数}}{\text{关键词总数}} \tag{6}$$

最大召回就是关键词检索系统所能够实现的最大的召回率。DET 图和等错点以及最大召回率均由图 6 所示。为了方便比较，把本文的解码器和 TDecoder 系统的等错点调到相当的水平，可以看到，在等错点相当的时候，基于本文的词图生成算法的关键词系统具有较高的最大召回率，从而有效减少关键词检索时信息的丢失，同时仍然说明了本算法生成的词图具有更多的信息。

6 结束语

本文针对以往的 WFST 词图算法不含精确词尾时间点的问题，提出了一种在 WFST 框架下的能



(a)采访对话测试集　　(b)在线对话测试集

图 6 在关键词测试集上的 DET 图

产生含有精确词尾时间点的词图的生成算法。本文从理论上分析了 WFST 解码音素图和标准词图的转换关系，并从实际上提出动态词匹配和基于拓扑排序及令牌传递的词图生成算法。从实验结果上看，本文的词图生成算法具有较快的速度，同时，本算法生成的词图比已有动态网络解码器的词图包含更多的解码信息，在关键词，LVCSR 重打分上具有更好的表现。

参 考 文 献

[1] Shore T, Faubel F, Helmke H, *et al.*. Knowledge-based word lattice rescoring in a dynamic context[C]. Proceedings of Interspeech, Portland, 2012: 1337–1340.

[2] Zhang Hao and Gildea D. Efficient multipass decoding for synchronous context free grammars[C]. Proceedings of the Association for Computational Linguistics, Columbus, 2008: 209–217.

[3] Mangu L, Brill E, and Stolcke A. Finding consensus in speech recognition: word error minimization and other applications of confusion networks[J]. *Computer Speech & Language*, 2000, 14(4): 373–400.

[4] Ortmanns S, Ney H, and Aubert X. A word graph algorithm for large vocabulary continuous speech recognition[J]. *Computer Speech & Language*, 1997, 11(1): 43–72.

[5] Demuynck K, Duchateau J, Compernolle D V, *et al.*. An efficient search space representation for large vocabulary continuous speech recognition[J]. *Speech Communication*, 2000, 30(1): 37–53.

[6] Rybach D, Schluter R, and Ney H. A comparative analysis of dynamic network decoding[C]. Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, 2011: 5184–5187.

[7] Mohri M, Pereira F C N, and Riley M. Speech Recognition with Weighted Finite-State Transducers[M]. Handbook of Speech Processing, Verlag Berlin Heidelberg, Springer, 2008: 559–582.

[8] Ljolje A, Pereira F, and Riley M. Efficient general lattice generation and rescoring[C]. Proceedings of 6th European Conference on Speech Communication and Technology, Budapest, 1999: 1251–1254.

[9] Povey D, Hannemann M, Boulianne G, *et al.*. Generating exact lattices in the WFST framework[C]. Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, 2012: 4213–4216.

[10] Povey D, Ghoshal A, Boulianne G, *et al.*. The Kaldi speech recognition toolkit[C]. Proceedings of Automatic Speech Recognition and Understanding Workshop, Hawaii, 2011: 10.1109/ASRU.2011.6163923.

[11] Young S, Russell N, and Thornton J. Token passing: a simple conceptual model for connected speech recognition systems [R]. Report of University of Cambridge, Department of Engineering, 1989: 1–23.

[12] Nolden D, Rybach D, Ney H, *et al.*. Joining advantages of word-conditioned and token-passing decoding[C]. Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, 2012: 4425–4428.

[13] Satoshi K, Takaaki H, Yoshikazu Y, *et al.*. Efficient beam width control to suppress excessive speech recognition computation time based on prior score range normalization [C]. Proceedings of Interspeech, Portland, 2012: 1649–1652.

[14] Guo Yu-hong, Li Ta, Si Yu-jing, *et al.*. Optimized large vocabulary WFST speech recognition system[C]. Proceedings of 9th International Conference on Fuzzy Systems and Knowledge Discovery, Chongqing, 2012: 1243–1247.

[15] Shao Jian, Li Ta, Zhang Qing-qing, *et al.*. A one-pass real-time decoder using memory-efficient state network[J]. *IEICE TRANSACTIONS on Information and Systems*, 2008, 91(3): 529–537.

[16] 张鹏远, 韩疆, 颜永红. 关键词检测系统中基于音素网格的置信度计算[J]. 电子与信息学报, 2007, 29(9): 2063–2066.
Zhang Peng-yuan, Han Jiang, and Yan Yong-hong. Phoneme lattice based confidence measures in keyword spotting[J]. *Journal of Electronics & Information Technology*, 2007, 29(9): 2063–2066.

郭宇弘： 男，1985 年生，博士生，研究方向为语音识别、音频信号处理.

黎 塔： 男，1983 年生，助理研究员，研究方向为大词汇连续语音识别.

肖业鸣： 男，1983 年生，博士生，研究方向为语音识别、声学模型.

潘接林： 男，1965 年生，研究员，博士生导师，主要研究领域包括大词汇连续语音识别、声学模型建模、搜索算法等.

颜永红： 男，1967 年生，研究员，博士生导师，2002 年入选中科院百人计划，现为中科院语言声学与内容理解重点实验室主任和所长助理.