德语语音合成中的字音转换研究

王永生
WANG Yong-sheng

上海同济大学 留德预备部,上海 200092
Centre for Foreign Language Training,Tongji University,Shanghai 200092,China
E-mail:yshwangtj@hotmail.com

WANG Yong-sheng.Algorithm of grapheme-to-phoneme conversion in German speech synthesis system.Computer Engineering and Applications,2009,45(35):132-134.

Abstract: It is difficult to convert graphemes to phonemes in German Speech Synthesis system.A rule-driven algorithm based on an iterated finite state transducer is proposed in present research to automatically convert graphemes to corresponding phonemes. In this algorithm,grapheme-to-phoneme rules are first made on the basis of a given lexicon;then graphemes of words are converted to phonemes following iterations resulting from morphological rules and phonological rules.Test results show that the accuracy of the total lexicon following application of the graphemes to phonemes conversion algorithm reaches as high as 94.4%.
Key words: German speech synthesis;grapheme-to-phoneme conversion;finite state transducer

摘 要:字音转换是德语语音合成系统不得不解决的难题。可以使用基于规则驱动的迭代有限状态转录机来解决这一问题。在该算法中,首先在一个词库的基础上制定一些字音转换规则,然后在此规则的基础上通过迭代有限状态转录机将德语单词中的所有字素转换成音素。经过对整个词库进行算法测试,单词的字音转换正确率可以达到94.4%。
关键词:德语语音合成;字音转换;有限状态转录机
DOI:10.3778/j.issn.1002-8331.2009.35.040 **文章编号**:1002-8331(2009)35-0132-03 **文献标识码**:A **中图分类号**:TP391

1 前言

字母转换成音素(Letter-to-Phoneme Conversion,简称字音转换)是德语语音合成系统中的一个难点问题。在德语语音合成系统中,要合成一个德语单词的语音,首先必须知道它怎样发音。可是如何取得单词的音标和读音呢?最直接的想法就是事先将这些音标(文本形式)和读音(数字形式,通过真人录制)存储在数据库中,这样,一旦需要合成某个词的语音,通过查找数据库就可取得其音标和读音。然而随着语言学的不断发展,每年均会产生大量的新词汇,加之德语可以让人根据一定的规则随意创造复合词的特点,这就决定了德语几乎有着无限多的词汇,因而无论建立多大的词库,总有可能存在一些词在词库中找不到,如何解决这部分词的读音呢?显然必须借助德语语音合成中的字音转换技术来解决。

但问题是,在德语中,字母与音素之间并不存在完全一一对应的关系,一个字母可以不止对应一个音素,如 x 的读音是/ks/;与此同时,一个音素也可以对应多个字母,如 schreiben 中的 sch 读/ʃ/。这显然给字音转换带来了麻烦。因而必须首先在字母与音素之间建立一种一一对应的关系。事实上,德语单词与其音标之间存在一一对应关系的是字素和音素,所谓字素(Grapheme),是指由一到多个字母组成的对应着一个音素或音素丛(Phoneme Cluster)的字母组合[1]。

以单词 schreiben 为例,其字素与音素的一一对应关系如表1所示。

表1 schreiben 中字素与音素的对应

sch	r	ei	b	e	n
ʃ	r	ai	b	ə	n

这样一来,所谓的字音转换其实就是字素到音素的转换(Grapheme-to-Phoneme Conversion)。然而要将一个单词中的字素转换成对应的音素,还需解决以下两个问题:

(1)如何判断一个单词中哪些字母或字母组合是字素?

(2)如何确定字素的读音,即对应的音素?

这也正是所要讨论的重点。

2 字素音素转换规则的制定

为了研究字素音素转换的规则,事先已经创建了一个有33 800个德语单词的词库,基本涵盖了德语中最常用的一些词。并已通过算法将每个单词的字素和音素一一对应起来,如(其中6和2分别代表音素/ʃ/和/ə/,这是为了便于在计算机中以 ASCII 码字符表示音标符号):

sch-r-ei-b-e-n⇔6-r-ai-b-2-n

然后经过统计与分析,生成一张字素表(其中包括所有字

基金项目:国家社会科学基金项目资助(the National Social Science Foundation of China under Grant No.08CYY001)。
作者简介:王永生(1972-),男,博士,主要研究方向:计算语言学。
收稿日期:2008-12-16 **修回日期**:2009-01-15

素及其对应的音素),经过统计,共有 106 个字素。每个字素可能由 1 到 4 个字母组成, 如单词 Dschungel 由字素 dsch、u、n、g、e、l 组成。

然后再根据这个词库,通过一些统计方法,进行半人工的字音转换规则(以下简称 G2P 规则)的制定。如字素 b 可能的读音有两个,即/p/和/b/,那么可以制定如下规则:

b->p/*_(chen|sel|ter|tissin| se|de|te)$ 规则 1

b->p/*_C$ 规则 2

b->p/*_$ 规则 3

b->b/*_* 规则 4

所有的 G2P 规则均采用 Chomsky and Halle 的规则记法[2],形式如下:

a->b/c_d

表示"当 a 在 c 和 d 之间出现时,把 a 转换成 b"。即符号"/"前是生成式,表示由字素 a 转换成音素 b,符号"/"后是字素的上下文环境,其中的"_"表示字素本身,前后的 c 和 d 分别表示该字素前和后的字母或字母组合。

而在表示字素的上下文环境时,采用了正则表达式(Regular Expression)的一些表示方法,如算符"*"(在正则表达式中称为 Kleene *)表示零到多个字母;符号"$"是一个锚号(Anchor),表示一个单词的尾边界;还有另一个锚号"^"表示一个单词的首边界。圆括号内的符号"|"是析取符,表示"或"关系。

在规则中出现的小写字母表示字母本身,而出现的大写字母则有特定的含义,如其中的"C"表示一个辅音字母或辅音字母的组合;"S"表示一个音节。

因而规则 1 表示,如果某个单词含有字素 b,且从该字素起到单词的结尾的字母组合是 chen、sel、ter、tissin、se、de 或 te,则该字素 b 读 \p\,如 Weibchen[ˊvaip72n]中的字素 b(其中 7 表示音素/ç/)。

规则 2 表示,如果某个单词含有字素 b,且从该字素起到单词的结尾的字母或字母组合均是辅音字母,则该字素 b 读\p\,如 schreibt[6raipt]中的字素 b。

规则 3 表示,如果字素 b 位于单词的结尾,则其读 \p\,如 Dieb[di:p]。

规则 4 表示,除了上述 3 种情况外,其他情况下的 b 一律读 \b\。

并且对于同一字素的规则,其排列的顺序是将条件限制较多的规则放在前面,以便在字音转换时优先匹配。

另外,在制定规则时,规则的主体并不是只有字素,对于那些读音固定的字素组合,也可以针对其制定规则,如:

t-i-o-n->ts-i-o:-n / S_$ 规则 5

ie-r-e-n->i:-r-2-n / S_$ 规则 6

规则 5 表示,如果某个词的词尾为 tion,则其字素组合为 t-i-o-n,转换成音素 ts-i-o:-n,如单词 Addition[aditsi'o:n]中的 tion。

规则 6 表示,如果某个词的词尾为 ieren,则其字素组合为 ie-r-e-n,转换成音素 i:-r-2-n,如单词 abfrieren[ˊapfri:r2n]中的 ieren。

总共制定了 410 条这样的 G2P 规则。

3 基于规则驱动的迭代有限状态转录机的字素音素转换算法

G2P 规则制定好之后,就来看看如何进行字素到音素的转换,以单词 schreiben 为例,就是如何进行下列转换:

sch-r-ei-b-e-n⇔6-r-ai-b-2-n

可以通过有限状态转录机(Finite State Transducer,以下简称 FST)[3]来描述这个问题。

定义字素音素转换的有限状态转录机为

$$A=(\Sigma,Q,\delta,q_0,F) \quad (1)$$

其中定义 Q 为内部状态 q_0、q_1、…、q_l 的有限集合,即 $Q=\{q_i|i=0,1,\cdots,l\}$,其中 q_0 为初始状态;定义 G 为德语中所有字素组成的集合,即 $G=\{g_j|j=1,2,\cdots,m\}$,字素 g_j 由 1 到 4 个字母组成;定义 P 为德语中所有音素组成的集合,即 $P=\{p_k|k=1,2,\cdots,n\}$;定义 Σ 为字素-音素偶对 g_j-p_k 构成的有限字母表,即 $\Sigma=\{g_j\text{-}p_k|g_j\in G,p_k\in P\}$,因此,$\Sigma\subseteq G\times P$;定义 F 为最后状态的集合,即 $F\subseteq Q$;定义 δ 为状态转换函数,即给定一个状态 $q\in Q$ 和 $g_j\text{-}p_k\in\Sigma$,$\delta(q,g_j\text{-}p_k)$返回一个新状态 $q'\in Q$。

以 schreiben 为例,其字素音素转换的 FSA 如图 1 所示。

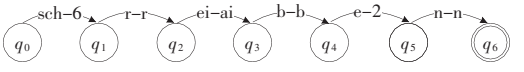

图 1 schreiben **字素音素转换的** FSA

图 1 中圆圈表示状态,带有箭头的弧(称为有向弧)表示输入的字素-音素偶对。

很显然,所要求解的问题就是,已知一个字音转换的规则库及一个德语单词(即一个字素丛),如何求解字素音素转换函数 δ。但问题是,并不知道单词中哪些字母或字母组合是字素,以及如果某个字素存在多个读音时它们与哪些音素对应。

然而,可以遵循这样的思路来解决问题。由于字素最多可由 4 个字母组成,因而首先定义一个长度为 4 的窗口,在单词上由左向右逐步移动,每次移动一个字符。一开始在这个移动窗口内的字母组合是 schr,根据事先创建的字素表,显然它不是一个字素,再判别 sch 是不是一个字素,发现它是一个字素,则搜索 G2P 规则库中关于字素 sch 的规则(其中 7 表示音素/ç/):

s-ch->s-7/(blä|häu|radie)_en$

sch->6/*_*

显然在 schreiben 中的 sch 匹配第二条规则,即其读/6/。

然后再将窗口向右移动一个字符,再进行类似的字素判别和规则匹配,直到窗口到达单词的右边界为止。

但有时仅仅从语音学(Phonology)的角度并不能完全解决字素音素转换的问题。如 ausschreiben['aus6raib2n],按照上述方法, 会错误地认为字母组合 ssch 中的 ss 和 ch 是两个字素,而事实上 ssch 应该拆分成字素 s 和 sch,分别可转换成音素/s/和/6/。但如果能先从形态学(Morphology)的角度,将前缀 aus 切分开,则可以很好地解决这类问题。另外对于德语中大量存在的复合词,如果在字素音素转换前先将其切分开,则必然会极大地降低转换的难度。

为此,提出一种基于规则驱动的迭代有限状态转录机(Rules-driven Iterated Finite State Transducer)算法,来进行字素音素的自动转换。

首先重新定义(式(1))中定义的字素音素转换的有限状态转录机 A,所要改动的仅仅是其中的集合 Σ。定义:

$$\Sigma=\{gs_j\text{-}ps_k|gs_j\in G^+,ps_j\in P^+\}$$

其中 gs_j 是由一个或多个字素组成的字素丛(Grapheme Cluster),因而 $gs_j\in G^+$,其中 G^+是字素集合 G 的正闭包(即 $G^+=G^1\cup$

$G^2 \cup \cdots \cup G^n \cup \cdots$)；

ps_k 是由一个或多个音素组成的音素丛(Phoneme Cluster)，即 $ps_k \in P^+$，其中 P^+ 是音素集合 P 的正闭包(即 $P^+=P^1 \cup P^2 \cup \cdots \cup P^n \cup \cdots$)。

因而 Σ 集合中的元素是字素丛及其对应的音素丛组成的偶对。

这里以 zwischenmenlich 为例，来说明什么是基于规则驱动的迭代有限状态转录机算法。

(1)算法初始化

初始的 FSA 只有两个状态，一个是初始状态，一个结束状态。以 zwischenmenlich 为例，其初始的 FSA 如图 2 所示。

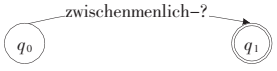

图 2 初始化时的 FSA

其中的“?”表示字素音素转换尚未成功。

(2)复合词切分

德语中存在大量的复合词，为了降低字素音素转换的难度，同时也为了提高转换的准确率，必须先将他们切分开。以 zwischenmenlich 为例，它由单词 zwischen 和 menlich 复合而成。因而经过切分后的 FSA 如图 3 所示。

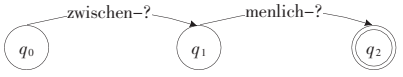

图 3 复合词切分后的 FSA

(3)基于形态规则的迭代

所谓形态规则，是指从前缀后缀的角度再次对单词进行切分。经过研究，针对德语中一些读音固定的前缀，制定一些前缀规则(简称为 P-规则)。如：

au-s-> au-s/^_*

z-u-s-a-mm-e-n->ts-u-z-a-m-2-n/^_*

再针对一些读音固定的后缀，制定一些后缀规则(简称为 S-规则)。如：

l-i-ch->l-i-7/*_$

k-ei-t->k-ai-t/*_$

总共定义了 101 条 P-规则和 41 条 S-规则。

图 3 所示的 FSA 在基于形态规则迭代后产生的 FSA 如图 4 所示。

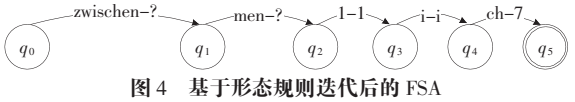

图 4 基于形态规则迭代后的 FSA

(4)基于语音规则的迭代

所谓基于语音规则的迭代，是指首先从左向右遍历整个 FSA，如果某个有向弧上的偶对 gs_j-ps_k 中的字素丛和音素丛尚未完成转换，需要进行迭代，其中字素丛 gs_j 由 n 个字母组成，即：

$gs_j=l_1l_2l_3l_4\cdots l_n$

由于德语字素最多由 4 个字母组成，因而选取一个长度为 4 的窗口，首先判断 $l_1l_2l_3l_4$ 是不是一个字素(通过查找字素表)，如果不是，再判断 $l_1l_2l_3$ 是不是字素，如果也不是，再判断 l_1l_2 是不是字素，如还不是，则 l_1 必然是字素。在成功确定某个字母或字母组合可能是字素后，再从 G2P 规则库中搜索其对应的规则，并依次进行匹配，如成功匹配某个规则，则该字素转换成功，再将窗口后移，搜索第二个字素，直到单词中的所有字素转换完毕。

4 规则测试

通过对整个词库运行上述算法，发现其中有 2 602 个词的转换存在错误，也就是单词的转换正确率为 92.3%。经过对转换错误的样本进行分析，发现有一些词的字音转换只是由于长短音的转换错误。如 Kalabrien[ka′la:bri2n]中的第 2 个 a 应该为长音/a:/，却被转换成了短音/a/。这类词有 711 个，由于这类错误并不严重，是勉强可以接受的，如果忽略这部分错误，则整个转换正确率可达到 94.4%。再分析剩下的转换错误的词，其中绝大多数是外来词，主要是来源自英语的词，如 Jeans 等。对于这些外来词，由于其读音情况要比德语单词复杂得多，且由于所占比例不大，因而没有必要在德语语音合成中研究他们的读音规则，完全可以将他们及其读音作为特例添加到一个特例库，这样在语音合成时，如果碰到这类词，就可以直接从特例库中提取其读音。

5 结论

由于德语有着几乎无限多的词汇，因而无论建立多大的词库，总有可能存在一些词在词库中找不到，解决这部分词的读音显然必须借助字音转换技术。提出了一种基于规则驱动的迭代有限状态转录机的字素音素转换算法。在该算法中，首先在一个词库的基础上制定一些字音转换规则，然后在此规则的基础上通过迭代有限状态转录机将德语单词中的所有字素转换成音素。经过对整个词库进行算法测试，单词的字音转换正确率可以达到 92.3%，如果忽略长短音转换的错误，则其正确率可达到 94.4%。

参考文献：

[1] Zhang J,Hamilton H J,Galloway B.English graphemes and their corresponding sound units[C]//Proceedings of Pacific Association for Computational Linguistics,Ohme,Japan,1997:351-362.

[2] Chomsky,N,Halle M.The sound pattern of English[M].New York:Harper and Row,1968.

[3] Jurafsky D,Martin J H.Speech and language processing:An introduction to natural language processing,computational linguistics,and speech recognition[M].[S.l.]:Prentice Hall,2000.