基于汉字有限状态转移机模糊匹配的智能语音问答系统

**1 背景介绍**

近年来，随着自然语言处理技术的发展，智能问答系统受到了极大的关注，从聊天软件‘小黄鸡’的风靡，到流行于各大网络平台的应答机器人，智能问答系统在众多领域得到应用。一个优质的问答系统解决客户常见的问题，降低人工开销，并能提供24小时无间断服务。

大多数问答系统都是以文本键入的形式作为问答系统的输入，繁琐费时，特别是在移动端（如手机）等无键盘设备上或对于老年人、残疾人等操作困难人群，文本输入变得异常困难。因此，基于语音输入的问答系统具有特殊优势：用户以口述的形式输入问题，经过语音识别，把识别结果传给问答系统进行处理，这样减少了用户的操作，提高了用户体验，拓广了用户人群。基于此，语音智能问答系统已经成为未来问答系统的发展方向。

**2. 问题描述**

基于语音的智能问答系统具有快捷、方便、适用设备和人群广泛的优点。然而，将语音作为输入方式带来了新的问题。首先，语音输入的方便性带来更大的随意性，相应地要求系统的模板数量和样式也跟着增加。传统基于模板匹配的搜索方法在大量、复杂的模板面前效率代下，处理逻辑越来越复杂; 第二，由于新词不断增加，基于词匹配的搜索方法不仅受限于分词系统，也极大限制了模板共享，继而产生效率问题。最后，由于用户的声音质量的不确定性，语音识别的结果具有极大的不稳定性，这使得传统精确模板匹配的方法很难适应面向语音输入的问答方式。

本发明提出一种基于汉字有限状态转移机(Finite State Transducer, FST)的模糊搜索方法，有效地解决了以上三个困难。

**2. 发明要点**

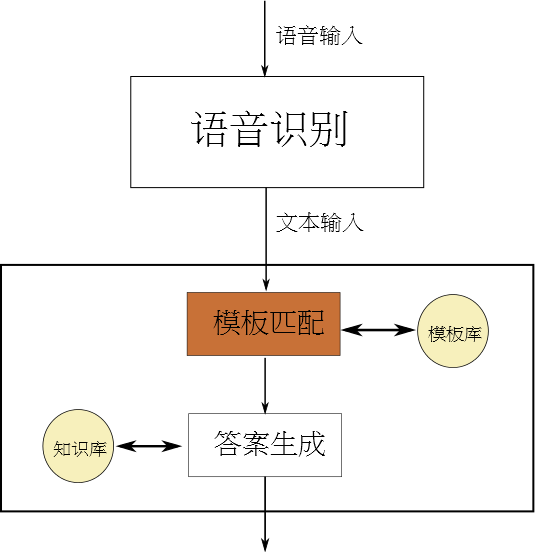
本发明提出一种基于FST的问答系统，来解决大规模模板搜索的低效性和复杂性、基于词的问答系统对分词系统的依赖性、对语音输入错误的脆弱性三个问题。具体而言，该发明包含如下主要内容：

1. **基于FST的模板共享方法。**传统基于模板匹配的问答系统需要对所有模板一一匹配。当模板数比较多且样式复杂的时候，这种匹配效率低下，算法逻辑复杂。本发明提出将模板共享压缩成FST，有效提高了搜索效率，且简化了搜索算法。
2. **基于字FST的模板匹配方法。**基于词的FST方法不仅依赖于分词方法，且对增加的新词难以进行模板共享。为解决上述困难，本发明提出基于字FST的匹配方法，将FST中的词变成字，可以有效增加路径共享，减小FST的体积，提高搜索效率。
3. **基于发音相似性的FST模糊匹配方法。**由于语音识别输出结果具有非确定性，传统精确的FST搜索无法适应语音问答的要求。我们在FST搜索过程中引入模糊匹配，允许一定的插入、删除和替换错误，并引入声音混淆矩阵来对匹配错误进行权重重估，从而解决语音问答系统的容错问题。

**3. 发明内容和系统实现**

**3.1系统架构**

图1为语音问答系统的架构示意图。该系统分为语音识别和问答系统两个子系统。用户以口述的形式将问题作为输入传给语音识别子系统，语音识别子系统将语音输入转换为文本后传给问答子系统。问答系统得到这些文本输入后，与系统模板库中的模板进行匹配，得到最接近的匹配模板，进而通过搜索知识库得到问题的相应答案。本发明的主要目的是提高模板匹配的效率和精度,因此集中于问答系统中的模板匹配模块，如图1中阴影框所示。



**图1： 语音问答系统架构**

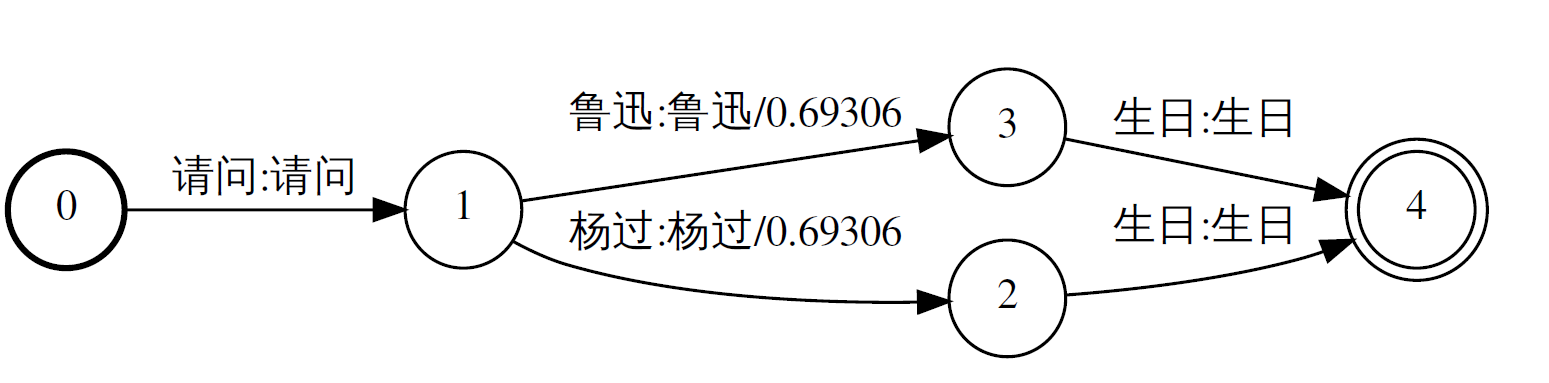
**3.2基于FST的模板共享方法**

传统的模板匹配方法将一个输入串与模板中的每个模板一一匹配，因此匹配计算量随着模板个数的增加呈线性增长。当模板数增大到一定数量时，匹配效率将大为下降。为解决计算量问题，可以将模板用树结构组织起来，从而使搜索时可以共享子串匹配，降低计算量。同样的思路可以用缓存结构实现：对搜索过的子串在缓存中保存下来，下次搜索同样的子串可以节约时间。传统模板匹配算法的另一个问题是当模板样式增多时，匹配算法必须对每种类型模板做单独处理，从而使得匹配逻辑越来越复杂。

本发明提出用有限状态转移机(FST)来进行模板共享和搜索。FST是对有限状态自动机(FSM)的扩展，用来精简表示一个有限状态转移过程。一个FST可以表示为一个有向图**(V,R)**，其中**V**为点集合，**R**为边集合。**R**中的每个边可以表示为一个五元组**(s,e,w,t,c)**，其中**s**和**e**分别为该边的初始和到达状态，**w**和**t**分别为输入元素和输出元素，**c**为该边的权值。FST可以用来表达上下文无关文法(FSG)，因此所有符合上下文无关文法的模板都可以表达为一个FST。例如，下面的模板可以表达为图2所示的FST:

**<模板1> := 请问<人名>生日**

**<人名> := 鲁迅｜杨过**



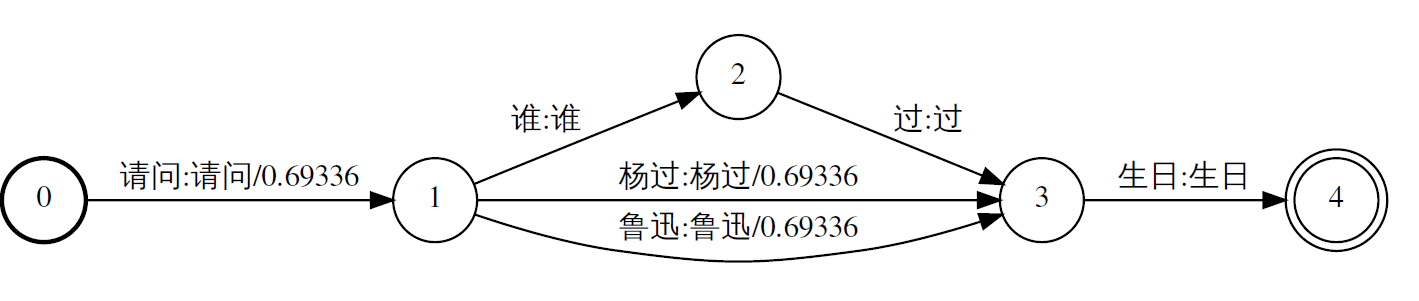
**图2：　<模板1> FST结构**

通过FST，我们可以把多个模板结合起来，形成一个大的FST，并通过确定化和最小化，使得模板可共享的部分充分共享，从而大大节约空间和计算开销。注意所有这些FST操作都有标准算法和工具可使用（如openFST），简单高效。如图3，即是将如下模板2和模板1进行合并之后FST结果:

**<模板2> :=请问　谁　过　生日**

通过图3，我们看到两个模板中能共享的部分都被共享了，这极大压缩了搜索空间，提高了搜索效率。当模板量极大的时候，这种基于共享FST的匹配方法将大大优于传统模板匹配方法。

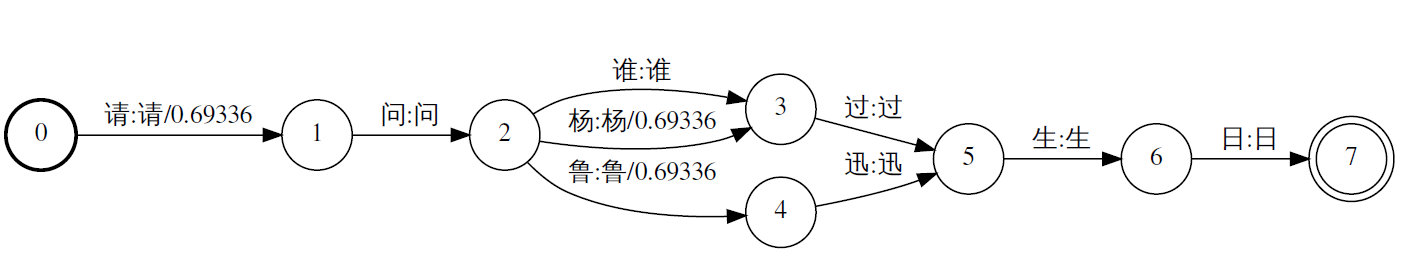
特别重要的是，基于这种FST的模板重构极大简化了搜索算法的设计：任何模板，只要它可以表达为上下文无关文法(CFG)，不论多么复杂，都可以表示为标准的FST并与其它模板组合共享，这意味着只需要一个对FST进行搜索的标准算法即可处理任何模板类型，而不需对某一类型模板单独处理，这极大简化了模板匹配算法。在3.4节我们将给出我们的一个FST搜索算法。



**图3：　<模板1>+<模板2> FST结构**

**3.3基于字FST的模板匹配方法**

基于FST的模板共享方法可以有效提高模板匹配的空间和时间效率，但对于新词却很难实现共享。例如，每天都可能有新人名、地名、产品名出现，如图（3）中的“杨过”，这些新词一方面为分词系统带来压力，另一方面这些新词在基于词的FST结构中很难实现共享。为了解决这一问题，本发明提出基于字FST的共享方法。汉语中虽然词可以无限增长，但汉字集是封闭的，这意味着任何新词拆成汉字后都有很大概率和其它词实现共享。基于字构造FST不仅可以摆脱对分词系统的依赖，而且可以实现模板的进一步共享。如图4所示，将图3中的词分解成汉字以后，“杨过”和“过生日”中的“过”即实现了共享，从而进一步压缩了搜索空间。



**图4：　<模板1>+<模板2> 汉字FST结构**

**3.4基于发音相似性的FST模糊匹配方法**

语音识别系统受口音、噪声等影响，识别结果经常出现随机错误;另一方面，传统FST的搜索方法是完全精确匹配的，这意味着直接将识别结果在FST中进行匹配很难成功。如“请问谁过生日”这句话很可能被语音识别系统识别成“请问谁的生日”，虽然绝大部分词识别正确，但一个“过”字的不匹配导致整个搜索失败。语音识别中这种小错误几乎每句都有，使得传统FST搜索方法几乎无法得到结果。

本发明提出基于发音相似性的FST模糊匹配方法，在搜索过程中允许搜索串和模板之间存在一定程度的不匹配，特别是在计算匹配测度的时候考虑匹配对之间的发音相似性。

**3.4.1 FST模糊匹配算法**

本算法扩展了图的剪枝搜索算法，是语音识别中“Token Pass”算法的一个变种。首先将模板集FST定义为 **I**,　将待查询字串也转成FST，定义为 **J** ，　FST模糊匹配算法目的是找到一条**I**和**J**间差异最小的匹配路径。其基本思路是将全局路径匹配任务分解为部分路径匹配任务，部分匹配的结果保存在一个称为token的数据结构中，通过token在 **I**和**J**两个FST中进行扩展和路径匹配。这一扩展和匹配直到某一token到达两个FST的终结状态，这一token所记录的匹配路径即为最佳匹配结果。

定义一个token 为 **v(i,j,h,s)**，其中(**i,j**)分别为该token在FST **I**和**J**　中的状态，**h**为该token在**I**和**J**中经过的历史路径，**s**为该token所经过路径的匹配距离，距离越小，说明匹配度越高。基于token的汉字FST模糊匹配算法如图5所示。

注意在图5所示的搜索算法中，距离度量D(**w1,w2**)描述了两个词**w1**和**w2** 之间的距离度量。在传统精确匹配算法中，D定义如下：

在模糊匹配中，一般将D定义为编辑距离，定义如下：

其中cs,cd,ci分别为替换错误、删除错误和插入错误的距离分值。为了使算法简单，我们引入了一种输入为空（<eps>）的边，从而使对不同错误的处理统一于距离度量D的计算中。

由此可见，我们提出的模糊匹配算法是精确匹配和基于编辑距离模糊匹配的扩展。通过自由选择距离测度D，我们可能进一步提高问答系统的性能。例如，我们可以将TF/IDF引入到D的计算中，对那些重要的领域相关词给以更高的权重; 或依据语法分析的结果，对主体词加大计算权重，使得匹配过程更关注于关键字/词的匹配与否。特别重要的是，这些权重可以附加在FST的边权重上，从而不必对搜索过程做任何改变。在本发明中，我们利用测度D上的灵活性，针对语音输入中的特殊错误模式进行有效补偿。

**定义：**

T: 当前活跃的token列表

T.top(n): token列表T中匹配距离最低的前n个token集合

merge(T): 对T中token进行合并，当多个token的状态对(i,j)相同时，保留匹配距离最小的token

prune(T): 对T中的token进行剪枝

**M.S:** FST **M** 中初始状态集合

**M.E:** FST **M** 中终止状态集合

E(**m**,**M**): 在FST **M**中对状态**m**的边扩展列表

st(**m,e, M**): 在FST **M**中对状态**m**通过边**e**所到达的状态

**e.w**: 边**e**包含的输入字符

eh(**h,(e1,e2)**): 对历史**h**加入边对(**e1,e2**)

D(**w1,w2**): 表示两个汉字**w1**和**w2**之间的匹配距离

<eps> : 零输入字符

**算法:**

*初始化：*

1. 对**I**和**J** 每一个状态加入自循环边，其输入和输出字符为<eps>
2. for each ( **i** ∈ **I.S** && **j** ∈ **J.S** ):

T **=** T∪ **v**(**i,j,**{},0)}

*搜索过程：*

while (True):

T’={}

//对FST **I,J**进行图扩展搜索

for each **v(i,j,h,s)** in T:

for each **e1** in E(**j,J**):

for each **e2** in E(**i,I**):

//略过双空字符扩展

if (**e1.w** == <eps>) && (**e2.w** == <eps>):

continue;

//计算新扩展出的token, 累积搜索历史和匹配距离

**v’** = (st(**i,e2,I**),st(**j,e1,J**), eh(**h**,(**e1,e2**)), **s** + **e2.c** + D(**e1.w,e2.w**))

T’= T’∪ {**v’**}

//对token列表进行合并

merge(T)

//寻找成功匹配的token

{**v**} = T’.top(1)

if (**v.i** ∈ **I.E)** && (**v.j** ∈ **J.E**):

return **v**;

//对token列表进行剪枝

T = prune(T’)

//如果剪枝结果为空列表，则匹配失败

If (size(T) == 0):

return NULL;

done

**图5：　汉字FST模糊匹配算法**

**3.4.2 基于发音相似性的距离测度方法**

在语音问答系统中，语音识别的输出虽然有显著的随机错误，但这些错误并不是杂乱无章的，其中绝大部分发音和正确发音都具有相似性，我们可以利用这种相似性来规范模糊匹配中D的计算，对那些发音相近的错误匹配给以较小的匹配距离，对发音相差较大的错误匹配给以较大的匹配距离。

基于发音相似性的模糊匹配方法的关键在于设计D(w1,w2)的计算方法。我们考虑利用音素间发音的相似性来计算字/词间的发音相似性。音素是最小的语音单位，每个音素都有自己的发音特色。音素的基元数目较少，因而计算简便，如汉语标准普通话中有35个音素，如表1所示。

|  |  |
| --- | --- |
| **辅音基元（22）** | **元音基元（13）** |
| b,c,ch,d,f,g,h,j,k,l,m,n,ng,  p,q,r,s,sh,t,x,z,zh | aI,a,Ie,eI,eN,e,Ci,Chi,  Bi,oU,o,u,v |

**表1 汉语音素**

音素对之间具有明显的相似性，这些相似性可以用混淆矩阵M来表示，其中第(i,j)个元素M(i,j)给出音素i和j的混淆程度。为了更能反映语音识别系统对相近发音的混淆规律，我们将语音识别结果表示为音素串，并与该句标准发音的音素串进行对比，可以得到某一音素被识别为另一音素的可能性，表示为：

M(i,j) = P(i|j) = C(i|j)/C(j) (3)

其中C(j)为标准发音中音素的个数，C(i|j)为语音识别结果中标准发音j被识别成音素i的个数，P(i|j)为音素j被识别成音素i的概率。基于M(i,j)，我们可以计算两个字/词(w1,w2)的距离。设w1的音素串为x,w2的音素串为y：

　　　　　　　　　　　（4）

其中，s为x和y的某种对齐方式，和为基于该对齐的x和y的第k个音素（允许空音素存在）。将D(w1,w2)用于图5中的FST模糊匹配算法，即可得到基于发音相似性的模糊匹配算法。

**4. 方案优势**

* 本发明利用FST高度共享和压缩结构将模板库进行重构，可以极大提高问答系统模板匹配的效率。这一过程不仅有标准高效的FST算法和式具支持，而且将复杂的模板搜索转化为简单的FST搜索，极大简化了模板　匹配算法。
* 通过将词FST替换为汉字FST，不仅进一步压缩了模板库的存储空间，提高了搜索效率，而且摆脱了搜索算法对分词系统的依赖，简化了对未登录词的处理。
* 基于发音相似性的模糊搜索方法，不仅可以增强系统的容错性，而且可以针对语音识别的特殊错误模式进行补偿，从而有利于对用户输入的有效理解，进而提高整个系统的性能。