多语互译机译系统统一算法基础——差异函数

胡 玥 高庆狮 高小宇

(北京科技大学 信息工程学院, 北京 100083)

摘 要: 在基于语义语言理论的多语互译机器翻译系统中,为使用同一算法和程序方便地实现多种自然语言之间的互译,引入了"差异函数".该函数可将某语言功能在各种自然语言表示上的差异隐去,使翻译过程与具体的语言表示无关,从而使系统可以用统一的程序来处理那些功能相同、但在不同语言上处理方法不同的语言内容.以基数表示功能的处理差异为例,讨论了差异函数的作用及其在多种语言上的应用.

关 键 词: 计算机应用; 自然语言处理系统; 机器翻译; 差异函数

中图分类号: TP 391.1

文献标识码: A **文 章 编 号**: 1001-5965(2008)05-0604-05

United algorithm base of multi-language machine translation system: difference function

Hu Yue Gao Qingshi Gao Xiaoyu

(Information Engineering School, University of Science and Technology Beijing, Beijing 100083, China)

Abstract: In the multi-language machine translation system based on semantic linguistic theory, in order to realize the translation between different natural languages with one algorithm and one program, the 'difference function' was introduced. The function of 'difference function' is to hide the representation differences among various natural languages about a language function, so that translation process is independent of concrete languages and the multi-language translation machine system can easily deal with language content with the united algorithm and program, which requires different treatment in different language. Taking the difference of radix representation function as an example, the role of the difference function and the application of such function in multi-language were discussed.

Key words: computer application; natural language processing system; machine translation; difference function

由于自然语言表示的复杂性和多样性,迄今一般的多语机器翻译系统[1-6]很难用同一算法和程序实现多种自然语言间的互译.而基于语义语言理论[7-9]的多语互译机译系统能够用同一算法和程序方便地实现多种自然语言之间的互译,根本原因是其翻译过程与具体语言的表示无关.

不同的自然语言之所以不同,不仅仅是使用的符号不同,还在于各种语言功能的表达方式不同,即同一种语言现象,如时态、语态、否定等在不同的语言中表示的方式不同.有的使用变词尾、加助词、加前缀、加后缀,有的使用加词、换词等不同方法或不同组合来表示.在语义语言理论中这些表示上的不同大多数被纳入到具体的语义单元的表示不同之中,但是仍然有相当的一部分,像时态的表示和基数的表示等,超出基本语义单元范围.例如,汉语基数以万为单位,英语以千为单位,这些表示是因具体自然语言而异的,难于纳入基本语义单元,如不采取措施解决此类问题,多语机

收稿日期: 2007-12-20

基金项目: 国家863计划资助项目(2006AA01Z140;2006AA010101);国家自然科学基金资助项目(60736014)

作者简介: 胡 玥(1963—),女,北京人,副教授,huhuyue_001@sina.com.

译系统难以用统一的算法和程序对这部分语言现象进行翻译.

针对这个问题,本文以基数表为例引入“功能相同的差异函数”的概念.该函数可将各自然语言中关于某种语言功能表示的差异隐去,使得翻译过程与具体的语言表示无关.

1　语义单元与差异函数的引入

给定 K 个自然语言$\{I,I\in\Omega_K\}$,其中,$K\in\mathbf{N}$,Ω_K 是一个符号集.

K 个不同自然语言的句子可以互译的原因是它们的语义相同.句子的意义被称为句义,记作 SS.一个自然语言 I 表示的句子称作该句子的句义在该自然语言上的表示,记作 R_ISS.

在所有的上述 K 个自然语言上表示相同意思的单元称作同一个语义单元在不同自然语言上的表示;其“意思”称作语义单元(SU).在某自然语言 I 上表示该语义单元的单元称作在该自然语言上的语义单元表示,记作 R_ISU.

差异函数是指“功能相同的差异函数”.“功能相同”是指语言的处理功能相同,“差异”是指实现某一功能时在不同的语言中的具体处理方式不同.“功能相同的差异函数”就是对一个语言功能在各种不同的自然语言上具体实现形式不同而定义的特殊函数.“函数”是指处理算法和程序.例如,处理数,处理动作的时态,处理各种语言现象等.

功能相同的差异函数可以用 $F_{GI}(X)$ 来表示,其中 G 表示语言功能,I 表示在自然语言 I 上,X 代表处理对象,即变量.例如 G 表示基数在各自然语言中的处理功能,$I=$E 表示英语,$I=$C 表示汉语,X 表示任意给定的基数.又如 G 表示动作的现在进行式,$I=$E 表示英语,$I=$C 表示汉语,X 表示某个动作.

在语义语言理论中基数的处理功能是功能相同的差异函数的典型例子.下面以英汉的基数表示处理为例进行讨论.

在基数表示中,数词可以表示数,如,十、ten 是数;而数的单位词是名词,如,百、千、万、亿、hundred,thousand,million 等.数的单位名词之前要加上数词后才表示数.

1.1　基本基数在英汉递归表示上的对应关系

由于万以下的数在英汉中不存在单位差异问题,将万以下的基数称为基本基数,对基本基数的语义单元可通过如下的递归定义表示:

$\omega_0::=a_0$;

$\omega_1::=a_1|a_2|\cdots|a_9$;

$\omega_2::=b_0|b_1|\cdots|b_9$;

$\omega_3::=c_2|c_3|\cdots|c_9$;

$\omega_4::=\omega_2|\omega_3|\omega_{14}$;

$\omega_5::=\omega_6|\omega_7|\omega_8$;

$\omega_9::=\omega_{10}|\omega_{11}|\omega_{12}|\omega_{13}$.

其中,$a_1,a_2,\cdots,a_9,b_0,b_1,\cdots,b_9,c_2,c_3,\cdots,c_9,\omega_6,\omega_7$ 和 ω_8 分别为基本基数的语义单元.各语义单元的含义及对应的汉语表示 ω_i^{C} 和英语表示 ω_i^{E} 如表 1 所示.

表 1　基本基数在自然语言英、汉上的表示对应

语义单元	汉语表示	英语表示
ω_0	零	zero
a_1	一	one
⋮	⋮	⋮
c_2	二十	twenty
c_3	三十	thirty
⋮	⋮	⋮
c_9	九十	ninety
ω_6	ω_1^{C} 百	ω_1^{E} hundred
ω_7	ω_1^{C} 百零 ω_1^{C}	ω_1^{E} hundred and ω_1^{E}
ω_8	ω_1^{C} 百 ω_1^{C}	ω_1^{E} hundred and ω_1^{E}
ω_{10}	ω_1^{C} 千	ω_1^{E} thousand
ω_{11}	ω_1^{C} 千零 ω_1^{C}	ω_1^{E} thousand and ω_1^{E}
ω_{12}	ω_1^{C} 千零 ω_1^{C}	ω_1^{E} thousand and ω_1^{E}
ω_{13}	ω_1^{C} 千 ω_3^{C}	ω_1^{E} thousand and ω_3^{E}
ω_{14}	$\omega_3^{\mathrm{C}}\omega_1^{\mathrm{C}}$	$\omega_3^{\mathrm{E}}\omega_1^{\mathrm{E}}$

1.2　基本基数扩充的困难及阿拉伯表示的引入

当基数上万以后,英语以千为单位:k,kk=M,kkk=G,kkkk=T,….汉语以万为单位:万,万万=亿,万万万=万亿,万万万万=亿亿,….

将上万以后的基数称为扩充基数.由于英语和汉语在基数单位上存在差异,造成了上万以后的英汉扩充基数的语义单元表示难以用递归定义的方式直接对应.

针对上万以后基本基数的扩充语义单元表示困难的问题,可采用纯阿拉伯表示法解决部分问题.即各语言基数分别简单地与阿拉伯表示对应,以阿拉伯表示作为中间桥梁实现多语言之间的对应关系.该方法的优点是比较简单,缺点是难以靠引入有限的语义单元来彻底解决问题.下面仍以汉语和英语为例讨论基本基数的扩充表示问题.

1) 基本基数表示与阿拉伯表示的对应

万以下的基本基数语义单元的递归定义(见 1.1 节)可直接与阿拉伯表示对应.对应关系如表 2 所示.其中,$F(\omega_3^{\mathrm{A}})=$〈取 ω_3^{A} 的高位〉.

2) 汉语扩充基数与阿拉伯表示的对应

对汉语的扩充基数语义单元可通过在基本基

数定义的基础上加入以下产生式来建立它与阿拉伯表示之间的对应关系. 对应关系见表3.

ω_9^* ::=$\omega_1 \mid \omega_4 \mid \omega_5 \mid \omega_9$; /注意 ω_9^* 与 ω_9 的区别/

ω_{17}^* ::= $\omega_{18} \mid \omega_{19} \mid \omega_{20} \mid \omega_{21} \mid \omega_{22}$;

ω_9^{**} ::=$\omega_1^{**} \mid \omega_4^{**} \mid \omega_5^{**} \mid \omega_9$;

ω_{17}^{**} ::=$\omega_{18}^{**} \mid \omega_{19}^{**} \mid \omega_{20}^{**} \mid \omega_{21}^{**} \mid \omega_{22}^{**}$;

ω_{23}^* ::=$\omega_{24}^{**} \mid \omega_{25}^{**} \mid \omega_{26}^{**} \mid \omega_{27}^{**} \mid \omega_{28}^{**} \mid \omega_{29}^{**}$.

引入 ω_9^{**} 和 ω_{17}^{**} 是为了补齐阿拉伯表示左边的零,或者处理汉语左边的零.

表2 基本基数的阿拉伯表示

语义单元	阿拉伯表示
ω_0	0
a_1	1
⋮	⋮
a_9	9
c_3	30
⋮	⋮
c_9	90
ω_6	ω_1^{A} 00
ω_7	ω_1^{A} 0 ω_1^{A}
ω_8	ω_1^{A} 0 ω_4^{A}
ω_{10}	ω_1^{A} 000
ω_{11}	ω_1^{A} 00 ω_1^{A}
ω_{12}	ω_1^{A} 0 ω_4^{A}
ω_{13}	$\omega_1^{\mathrm{A}}\omega_5^{\mathrm{A}}$
ω_{14}	$F(\omega_3^{\mathrm{A}})$ ω_1^{A}

表3 扩充基数的汉语及阿拉伯表示

语义单元	汉语表示	阿拉伯表示
ω_{18}	$\omega_9^{*\mathrm{C}}$ 万	$\omega_9^{*\mathrm{A}}$0000
ω_{19}	$\omega_9^{*\mathrm{C}}$ 万零 ω_1^{C}	$\omega_9^{*\mathrm{A}}$000ω_1^{A}
ω_{20}	$\omega_9^{*\mathrm{C}}$ 万零 ω_4^{C}	$\omega_9^{*\mathrm{A}}$00ω_4^{A}
ω_{21}	$\omega_9^{*\mathrm{C}}$ 万零 ω_5^{C}	$\omega_9^{*\mathrm{A}}$0ω_5^{A}
ω_{22}	$\omega_9^{*\mathrm{C}}$ 万 ω_9^{C}	$\omega_9^{*\mathrm{A}}\omega_9^{\mathrm{A}}$
ω_1^{**}	零 ω_1^{C}	000ω_1^{A}
ω_4^{**}	零 ω_4^{C}	00ω_4^{A}
ω_5^{**}	零 ω_5^{C}	0ω_5^{A}
ω_9	ω_9^{C}	ω_9^{A}
ω_{18}^{**}	$\omega_9^{**\mathrm{C}}$ 万	$\omega_9^{**\mathrm{A}}$0000
ω_{19}^{**}	$\omega_9^{**\mathrm{C}}$ 万零 ω_1^{C}	$\omega_9^{**\mathrm{A}}$000ω_1^{A}
ω_{20}^{**}	$\omega_9^{**\mathrm{C}}$ 万零 ω_4^{C}	$\omega_9^{**\mathrm{A}}$00ω_4^{A}
ω_{21}^{**}	$\omega_9^{**\mathrm{C}}$ 万零 ω_5^{C}	$\omega_9^{**\mathrm{A}}$0ω_5^{A}
ω_{22}^{**}	$\omega_9^{**\mathrm{C}}$ 万 ω_9^{C}	$\omega_9^{**\mathrm{A}}\omega_9^{\mathrm{A}}$
ω_{24}^{**}	$\omega_{17}^{**\mathrm{C}}$亿	$\omega_{17}^{**\mathrm{A}}$00000000
ω_{25}^{**}	$\omega_{17}^{**\mathrm{C}}$亿零 ω_1^{C}	$\omega_{17}^{**\mathrm{A}}0000000\omega_1^{\mathrm{A}}$
ω_{26}^{**}	$\omega_{17}^{**\mathrm{C}}$亿零 ω_4^{C}	$\omega_{17}^{**\mathrm{A}}000000\omega_4^{\mathrm{A}}$
ω_{27}^{**}	$\omega_{17}^{**\mathrm{C}}$亿零 ω_5^{C}	$\omega_{17}^{**\mathrm{A}}00000\omega_5^{\mathrm{A}}$
ω_{28}^{**}	$\omega_{17}^{**\mathrm{C}}$亿零 ω_9^{C}	$\omega_{17}^{**\mathrm{A}}0000\omega_9^{\mathrm{A}}$
ω_{29}^{**}	$\omega_{17}^{**\mathrm{C}}$亿 ω_{17}^{C}	$\omega_{17}^{**\mathrm{A}}\omega_{17}^{\mathrm{A}}$

3) 英语扩充基数与阿拉伯表示的对应

对英语的扩充基数语义单元可通过在基本基数定义的基础上加入以下产生式来建立它与阿拉伯表示之间的对应关系. 对应关系见表4.

ω_{30} ::=$\omega_{31} \mid \omega_{32} \mid \omega_{33} \mid \omega_{34}$;

ω_{35} ::=$\omega_{36} \mid \omega_{37} \mid \omega_{38} \mid \omega_{39} \mid \omega_{40} \mid \omega_{41} \mid \omega_{42}$.

由此可见采用直接与阿拉伯数字表示法对应的方法每增加一个数的单位(包括数的单位名词和数的单位名词的重复有穷次),就需要增加一组语义单元,各自然语言都类似. 这样一来会使语义单元库中存入大量基数语义单元.

表4 扩充基数的英语及阿拉伯数表示

语义单元	英语表示	阿拉伯表示
ω_{31}	ω_5^{E}thousand	ω_5^{A}000
ω_{32}	ω_5^{E}thousand and ω_1^{E}	ω_5^{A}00ω_1^{A}
ω_{33}	ω_5^{E}thousand and ω_4^{E}	ω_5^{A}0ω_4^{A}
ω_{34}	ω_5^{E}thousandω_5^{E}	$\omega_5^{\mathrm{A}}\omega_5^{\mathrm{A}}$
ω_{36}	ω^{E}million	ω_1^{A}000000
ω_{37}	ω^{E}million andω_1^{E}	ω_1^{A}00000ω_1^{A}
ω_{38}	ω^{E}million and ω_4^{E}	ω_1^{A}0000ω_4^{A}
ω_{39}	ω^{E}million and ω_5^{E}	ω_1^{A}000ω_5^{A}
ω_{40}	ω^{E}million and ω_9^{E}	ω_1^{A}00ω_9^{A}
ω_{41}	ω^{E}million and ω_{13}^{E}	ω_1^{A}0ω_{13}^{A}
ω_{42}	ω^{E}million ω_{14}^{E}	$\omega^{\mathrm{A}}\omega_{14}^{\mathrm{A}}$
ω_{43}	$\omega_5^{\mathrm{E}}\omega_9^{\mathrm{E}}$	$F(\omega_3^{\mathrm{A}})\omega_9^{\mathrm{A}}$
ω_{44}	$\omega_6^{\mathrm{E}}\omega_{13}^{\mathrm{E}}$	$F(\omega_1^{\mathrm{A}})\omega_{13}^{\mathrm{A}}$

2 转换函数与功能相同的差异函数

2.1 直接使用基数处理功能的差异函数

直接使用基数处理功能相同的差异函数是另一种解决方法,它将具体基数处理功能加以抽象,隐去不同语言在具体表示上的差异,实现语言处理功能与具体的语言表示无关,该方法在基于语义语言理论的多语互译机译系统中能有效而彻底解决使用同一算法和程序实现多种语言基数互译的问题.

基数处理功能相同的差异函数在实现时仍然需要借用阿拉伯数作为中间变量,但此阿拉伯数是语义层面"变量",不再单独作为具体的语义单元处理. 需处理的"语义单元"是实现该功能的差异函数.

对于给定的基数处理功能差异函数 $F_{n\mathrm{C}}(X)$ 和 $F_{n\mathrm{E}}(X)$,由于采用阿拉伯数作为中间变量,所以当给定的数 X 在自然语言 I 上的表示为$R_I\mathrm{SU}_n$时,基数处理功能的差异函数 $F_{nI}(X)$ 的功能就相当于转换函数 $F_{I\to\mathrm{A}}(R_I\mathrm{SU}_n)$和 $F_{\mathrm{A}\to I}(R_{\mathrm{A}}\mathrm{SU}_n)$. 其中,$X$ 是任意给定的基数;C 表示汉语;E 表示英语;n 表示基数处理功能;$R_I\mathrm{SU}_n$是基数的语义单元 SU_n在自然语言 I 上的表示;$R_{\mathrm{A}}\mathrm{SU}_n$是基数的语义单元 SU_n的阿拉伯表示;$I\to$A 表示将 I 语言上的表示转换为阿拉伯表示;A$\to I$ 表示将阿拉伯表示转换为语言 I 上的表示.

设 $F_{G_1I}(R_I\mathrm{SU}_n)=F_{I\to\mathrm{A}}(R_I\mathrm{SU}_n)$是差异函

数,显然有 $F_{G_1 I}(R_I \mathrm{SU}_n)=R_\mathrm{A}\mathrm{SU}_n$. 差异函数 $F_{G_1 I}$ 的功能是把基数 n 的语义单元 SU_n 在自然语言 I 上的表示 $R_I\mathrm{SU}_n$ 转换为阿拉伯表示,这种功能对不同的语言 I 是相同的,但在不同语言上的实现方式是不同的. 同理,设另一差异函数 $F_{G_2 I}(R_\mathrm{A}\mathrm{SU}_n)=F_{\mathrm{A}\to I}(R_\mathrm{A}\mathrm{SU}_n)$,有 $F_{G_2 I}(R_\mathrm{A}\mathrm{SU}_n)=R_I\mathrm{SU}_n$. 差异函数 $F_{G_2 I}$ 的功能是把基数 n 的语义单元 SU_n 的阿拉伯表示转换成为在自然语言 I 上的表示 $R_I\mathrm{SU}_n$,这种功能对不同的语言 I 也是相同的,但在不同的语言 I 上的实现方式也是不同的.

在一个多自然语言集 Ω 上,例如{英语,汉语,日语,…},任意两种语言 I 和 J 上的基数 n 的语义单元 SU_n 表示 $R_I\mathrm{SU}_n$ 和 $R_J\mathrm{SU}_n$ 之间的转换 $R_J\mathrm{SU}_n=F_{I\to J}(R_I\mathrm{SU}_n)$ 和 $R_I\mathrm{SU}_n=F_{J\to I}(R_J\mathrm{SU}_n)$,可通过差异函数 $F_{G_1 I}$,$F_{G_1 J}$,$F_{G_2 I}$,$F_{G_2 J}$ 来实现. $F_{I\to J}(R_I\mathrm{SU}_n)=F_{G_2 J}(F_{G_1 I}(R_I\mathrm{SU}_n))$,$F_{J\to I}(R_J\mathrm{SU}_n)=F_{G_2 I}(F_{G_1 J}(R_J\mathrm{SU}_n))$.

例如,$F_{\mathrm{C}\to\mathrm{E}}$(六亿一千五百二十万八千四百三十八)=$F_{\mathrm{A}\to\mathrm{E}}(F_{\mathrm{C}\to\mathrm{A}}$(六亿一千五百二十万八千四百三十八))=$F_{\mathrm{A}\to\mathrm{E}}$(615 208 438)=six hundred fifteen million two hundred and eight thousand four hundred and thirty eight.

采用差异函数方法,可以省去在库中存入大量基数语义单元的负担,也可以比较彻底解决多语言之间用同一算法和程序实现基数互译的问题.

2.2　$F_{I\to\mathrm{A}}(R_I\mathrm{SU}_n)$和$F_{\mathrm{A}\to I}(R_\mathrm{A}\mathrm{SU}_n)$的实现

$F_{I\to\mathrm{A}}(R_I\mathrm{SU}_n)$的实现可以使用多种不同算法. 比较简单的是"+""×"法:〈数〉的单位下降时插入"+";〈数〉的单位上升时,把先前的所有比它低的数放在一对括号内,然后插入"+0";遇"零"等于"+0",相当于不加不乘;遇"and"或者","等于加.

例如,三千四百万五千六百七十二=(3×1 000+4×100)×10 000+5×1 000+6×100+70+2=3 400 5672.

英语相类似:

thirty four million five thousand six hundred seventy two=(30+4)×1 000 000+5×1 000+6×100+70+2=34 005 672.

$F_{\mathrm{A}\to I}(R_\mathrm{A}\mathrm{SU}_n)$与 $F_{I\to\mathrm{A}}(R_I\mathrm{SU}_n)$的实现相类似. 汉语按 4 个十进制一隔,英语按三进制一隔,汉语可以在亿、万等之后如果不空可以先加零.

3　与差异函数有关的几个问题

3.1　功能相同的差异函数与具体自然语言

以基数处理功能相同的差异函数 $F_{nI}(X)$为例,虽然日语与韩语都与汉语一样,以万、亿为单位,但是仍然有一些区别. 例如,日语使用汉字时,可以与汉语一样,但是使用假名时,会有一些日本语独特的差异. 当然可以采用复合方式:

$F_{n\,\mathrm{Japan}}(R_\mathrm{Japan}\mathrm{SU}_n)=F_{n\,\mathrm{C}}(F_{n\,\mathrm{Japan}\to\mathrm{C}}(R_\mathrm{Japan}\mathrm{SU}_n))$

其中 $F_{n\,\mathrm{Japan}\to\mathrm{C}}(R_\mathrm{Japan}\mathrm{SU}_n)$是把假名表示的数转换成为日本汉字表示的数. 韩语的基数有两种表示方法:固有数词和汉字数词. 两种表示都是使用韩语的文字. 汉字数词与汉语的数词有简单对应关系,固有数词需要添加一些变换,不过,固有数词只在百以下使用. 与日语的处理类似,可以直接使用功能差异函数,也可以采用复合方式.

法语、德语、西班牙语、葡萄牙语、意大利语、波兰语、瑞典语、俄语与英语一样,以千为单位,但是也同样各有差异.

法语把数词归到形容词类,81 表示为 quatre-vinght-un,即"4-20-1",4 个 20 加 1. 这些细小差别使用功能相同的差异函数 $F_{n\,\mathrm{F}}(R_\mathrm{F}\mathrm{SU}_n)=R_\mathrm{A}\mathrm{SU}_n$就可方便的解决表示问题.

德语基数词的组合表达次序及连接词的使用与其他语言有所差别. 例如,29 表示为"neunundzwanzig",即"9 和 20",中间没有空格. 437 表示为"4 百和 7 和 30". 其中第一个连接词"und(和)"可以省略.

西班牙语基数词的表达方式与英语基本一样,但是由于词尾随后续词的阴性阳性有所变化,所以表达形式略有小差异. 例如,100 表示为"cient";101 表示为"ciento uno",即"百　1",且"百"之后加词尾 to.

葡萄牙语基数词的表达方式与西班牙基本一样,但是词尾随后续词的阴性阳性的变化比西班牙语少,所以表达形式仍然略有小差异. 例如,2 469表示为"dois mil quatrocentos e sesenta e nove",即"2 千 4 百+60+9". 其中,dois 是 2;mil 是千;quatro 是 4;centos 是百;quatrocentos 是 4 百;e 是连接词"和".

与法语类似,意大利语把数词归到形容词中,基数词的表达方式与西班牙语、葡萄牙语类似,有些词尾变化. 例如,223 表示成为"duecentoventitré",即"2(due)百(cento)20(venti)3(tre)",注意,之间没有空格,23 中的 20 与 3 都是独立词,之间没有空格,也没有其他符号.

波兰语基数词虽然是属于千、百万的计数系统,它的表达方式与英语差不多,但是需要根据后

续名词进行性(男性、有生命阳性、无生命阳性、阳性、中性)、数(单数、复数)、格(主、属、给、宾、工具、时间地点间接补语、称呼)的词尾变化,十分复杂.其中称呼格不必考虑,有生命阳性与无生命阳性不分,合并为非男阳性(即非人阳性).波兰语基数词的功能相同差异函数可以采用复合形式,其中复杂的性、数、格词尾变化需要使用单独的功能相同差异函数,对不需要进行性、数、格变化的自然语言,对应的差异函数可以是空.

瑞典语基数词的表达方式与英语基本一样,除了一个基数词稍微有些变化外,其他都没有任何变化.例如,1 675 表示为"ettusen sexhundra sjutti(o)fem",即"1 千 6 百 7 十 5".其中,ett 是 1;tusen 是千;ettusen 是 1 000;sex 是 6;hundra 是 100;sexhundra 是 600;sju 是 7;tio 是 10;sjutti(o)是 70;fem 是 5.

俄语基数词虽然是属于千、百万的计数系统,它的表达方式与英语差不多,但是需要根据后续名词进行数(单数、复数)、性(复数不分,单数分阳性、阳性、中性)、格(主、属、给、宾、工具、时间地点间接补语、称呼)的词尾变化.虽然比波兰语简单,但还是相当复杂.与波兰语类似,基数词的功能相同差异函数可以采用复合形式,需要处理性、数、格的功能差异函数.

所有上述这些语言上的细小差别,都可以通过使用不同的"功能相同的差异函数"方便地进行解决.

3.2 功能相同的差异函数自变量 X 的确定

功能相同的差异函数的自变量 X 有两种表现形式:连续形式和不连续形式.

基数处理功能的差异函数的自变量 X 是连续表现形式的一个典型.其自变量 X 的确定十分简单,只需在子树剪枝过程中[9],利用以下简单判别算式:〈S〉::=〈数〉|〈S〉〈数〉|〈S〉〈数单位〉,就可识别 X.

动作时态处理功能相同的差异函数的自变量 X 是不连续表现形式的典型.例如,英语句子"〈C_1〉 has 〈F〉 finished reading 〈C_2〉",对应的汉语表示为:"〈C_1〉〈F〉读完〈C_2〉".其中,〈C_1〉和〈C_2〉是动词的第一和第二参数;〈F〉是限定动作的限定词.英语动作时态处理功能相同的差异函数的自变量 X="has finished reading"是不连续,汉语自变量 X="读完"是连续的.即语义单元表示:R_ESU=〈N{人}〉read〈N{读物}〉;R_CSU=〈N{人}〉>读<N{读物}〉.

过去完成时态处理 P 的功能相同的差异函数:

$F_{PE}(R_E\mathrm{SU})$ =〈N{人}〉 be * 〈限定动作的限定词〉 finished 〈read 的进行式〉〈N{读物}〉;

$F_{PC}(R_C\mathrm{SU})$ =〈N{人}〉〈限定动作的限定词〉〈read〉完〈N{读物}〉.

其中,功能相同的差异函数的自变量 $X=R_E$SU(标有下划线部分)的表现形式是不连续的.

4 结束语

本文以基数表示功能为例讨论了"功能相同的差异函数"及其在基于语义单元理论的多语机器翻译系统中的作用,该函数是系统可以用同一算法和程序方便地实现多种自然语言之间互译的重要基础之一.

参考文献(References)

[1] Lehnert W G, Ringle M H. Strategies for natural language processing[M]. London: Lawrence Erlbaum Associates, 1982

[2] Moyne J A. Understanding language man or machine[M]. New York: Plenum Press, 1985

[3] Allen J. Natural language understanding[M]. Redwood City: The Benjamin/Cummings Publishing Company Inc, 1995

[4] Manning C D, Schütze H. Foundations of statistical natural language processing[M]. London: The MIT Press, 1999

[5] Kao A, Poteet S R. Natural language processing and text mining[M]. Bellevue: Springer, 2007

[6] Tesnière L. Eléments de syntaxe structurale[M] Paris: Klinksieck, 1959

[7] 高庆狮,陈肇雄. 类人机器翻译原理[J].计算机研究与发展,1989,26(1):1-8
Gao Qingshi, Chen Zhaoxiong. Subhuman machine translation system[J]. Computer Research and Development, 1989, 26(1): 1-8(in Chinese)

[8] Gao Q S, Hu Y. Semantic language and multi-language MT approach based on SL[J]. Journal of Computer Science & Technology, 2003, 18(6): 848-852

[9] 高小宇,高庆狮,胡玥.基于语义单元表示树剪枝法的高速多语言机器翻译方法[J].软件学报,2005,16(11):1000-1919
Gao Xiaoyu, Gao Qingshi, Hu Yue. High-speed multi-language machine translation method based on pruning on the tree of representations of semantic elements[J]. Software Journal, 2005, 16(11): 1000-1919(in Chinese)

(责任编辑:文丽芳)