

计算机进行自然语言处理的意义表示模型

唐永炎

(自动化系)

摘 要 本文描述了计算机进行自然语言理解的一个意义表示模型。该模型主要包括基本行为、基本状态的种类与表示;此外,本文还给出了从语言输入生成意义表示的实现算法;最后,描述了用 GCLLSP 语言实现的 MR (意义表示器)程序。

主题词 语言处理;微机/自然语言理解

0 前 言

计算机处理自然语言的核心,即自然语言理解(Natural Language Understanding,简称NLU),是人工智能(AI)的一个十分重要但又困难的分支。AI本身试图揭示人脑思维的机制,以使机器具有智能。由于语言既与思维密切相关,又是交流的工具,所以一个理想的NLU系统,一方面将提供探索人脑思维机制的真正途径,另一方面可使人们用日常语言与计算机进行互流,从而给人类带来极大的方便。这就是计算机处理自然语言系统(NLU)研究的最终目标。

1 NLU系统研究方法

我们先考察一下人是怎样进行语言理解的。当一个人见到一句话“他到书店买了本书”时,这个人就会将这句话变成某种大脑内部的意义表示,在这表示中,将涉及到客体“他”,“书店”,“书”,并且只要多加琢磨一下这句话,就会在大脑中出现他的模样,书店的情形以及书的样子等;同时,行为“买”的意义也被表示出来,涉及到他将钱交给店员,而店员将书给他等。这样,我们就可推测出他的行为动机(也许他喜欢读书)以及他的行为后果(或许他看不成电影了)。

由以上简单的例子中可以看到,理解包括了两个方面:一是字面意义的理解,即形成某种输入的意义表示;二是通过对建立的意义表示,利用有关概念的知识,进行处理来作出“合情”推理。图1是一个NLU系统的组成示意图。

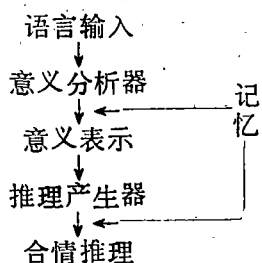


图1 NLU系统组成

图1中的“记忆”包括了有关一般世界的知识，即常识。由于常识表示与处理问题仍是AI中一个尚未完全解决的问题，完整的记忆模型尚未出现，所以本文重点是解决自然语言的意义表示模型中的有关问题。

2 意义表示模型的基本思想

在深入到模型的细节之前，先应确定以下两个问题：

2.1 模型的总体要求

在形成语言输入的意义表示时，究竟应详尽到什么程度？例如对上面例句意义表示所涉及到的客体“书”来说，人们在自己的记忆中储存了有关的大量知识，包括“书”是一个物理客体，它有形状、大小和它的组成如何、价格多少，它的用途又是怎样？并且在描述“书”的这些方面时，又涉及到许多新的客体与行为，如“纸”，“钱”，“阅读”等，同时在记忆中涉及有关它们的大量知识，如此下去，能搜索尽整个记忆。显然，在意义表示中不可能如此繁复。事实上，我们只须表示出给定输入所引出的最少的基本信息，同时这些信息又必须足够丰富，使得在推理需要时能获得记忆中的必要知识。因此，模型的总体要求应体现以下两个原则：

原则1 所表示的信息最少冗余；

原则2 所表示的信息足够丰富。

需注意的是，除非我们对记忆中各种常识概念的表示及相互关系有一完整清晰的了解，否则理论上这两个原则就不能严格成立。我们在此仍明确提出这两个原则是因为它们对下面的模型组成具有总体指导作用。

2.2 模型的基本组成

因为语言是描述人在周围世界中的活动及其所处状态的，所以模型的基本组成应包括行为，客体，状态。

依据以上两个原则，要求用来描述各种活动的行为，可将这些行为称为基本行为，对于意义表示中的客体只须用一个客体名称来指称它，这样，在需要时也就能抽出记忆中的大量有关知识，用来描述人和周围世界所处的状态的种类也需是足够基本，同时又足够丰富，称这些状态为基本状态。

由上所述，意义表示模型所要解决的主要问题可归结为：

问题1 基本行为的组成、种类及表示；

问题2 基本状态的种类及表示；

问题3 如何形成这些表示。

问题 1、2 将在下节讨论, 问题3在第4节基本算法中回答。

3 模型的细节描述

3.1 基本行为的组成

每一种基本行为都有下列槽(slot)依附其上:

- 代理人(agent) 表示行为的施予者;
- 客体(object) 表示该行为的作用对象;
- 方向(direction) 表示该行为进行时的方向, 或该行为客体接受与施予的对象, 它可分解成两个槽, 从(from)和到(to);
- 工具(instrument) 表示完成该行为所借助的其它工具行为;
- 目的(intention) 表示该行为的代理人所欲达到的目的;
- 结果(result) 表示该行为所能导致的结果;
- 时态(tense) 表示该行为进行的时态。时态有下列类型, 即 $\bar{n}$ 表示否定或相反; q 表询问, r 表示持续; p 表示过去; $\square$ 缺省表示现在; f 表示将来

以上所讨论的槽的意义反映了Minsky. M在文献[1]中所提出的人们为理解一行为而常会问及的一些问题, 诸如“谁引起它?”, “它的目的是什么?”, “它的结果是什么?”, “它影响到谁?” “它怎样完成?” 等的描述。在形成意义表示时, 对某些需要复杂的推理知识才能回答的问题, 以及尚无法确定答案的问题, 赋以相应槽的值为空(nil)或在表示中省略去。

以下采用框架(frame)的形式来表示基本行为。有关框架表示的理论, 请参阅文献[1]。

3.2 基本行为的种类及表示

关于基本行为到底有多少种并不重要, 重要的是它们能全面地描述各种行为, 这一点的更详细论述可参阅文献[2]、[3]、[5]。

基本行为可归纳为如下表1中所示的11种。

3.3 基本状态的种类

种类1 能用相对数字(-10~10)作为状态值状态, 例如健康、恐惧、愤怒、饥饿、喜恶惊奇等。如健康状况良好的值为+5; 不好为0; 病危为-8; 死亡为-10等。

种类2 能用绝对数字作为状态值的状态, 例如长度、颜色、质量、速度等。

种类3 能用客体间的关系来表示的状态, 例如控制、拥有、包含、接近、位于、相触等。

种类4 用愿望, 即动机来表示的状态。这一类状态很重要。

3.4 基本状态的表示

对第1类状态, 表示的形式是:

state: 状态名

agent: A

value: D

result: R

tense: T

表1 11种基本行为

序号	action	槽 (Slot)					解 释
		agent	object	from	to	instrument	
1	移动(MOVE)	A	part(A)	B	C		A将其某一身体部位part(A)从一处B移到另一处C
2	施力(PEL)	A	B	C	D	move, propel, grasp	A将力作用于某物B使该物从C移至一新处D
3	摄取(INGEST)	A	B	C	mouth(A) in(A)	pttrans	A将食物B送到A的嘴中, 并进入身体
4	抓住(GRASP)	A	B	C	hand(A)	move	A用手抓住一客体B
5	排出(EXPEL)	A	B	in(A)	C	propel, move	将某物B排出身体, 如血、汗唾液等
6	转换(ATRANS)	A	B	C	D	pttrans, mtrans, move	从一状态到另一状态抽象的转换, 如买卖
7	位移(PTRANS)	A	B	C	D	move, propel	一客体的物理转换, 即从一处移到另一处
8	发声(SPEAK)	A	B	A	C	move (mouth(A))	A说出B这一东西
9	留意(ATTEND)	A	B	C	mind(A)		A集中注意于感官B上。为speak之反义
10	传信(MTRANS)	A	B	C	mind(D)	speak, attend	信息B被A从C处传给D处
11	生思(MBUILD)	A	input(B) output(C)	memory(A)	mind(A)	mtrans	从比较简单的思想B导致作出决定C

对第2类状态, 表示的形式是:

state: 状态名

object: O

value: D

result: R

对第3类状态,表示形式是:

state: 关系名

object1: O1

object2: O2

result: R

对第4类状态,表示形式是:

state: GOAL

agent: A

objective: O

plan: P

表示A具有愿望(目标)O,相联于此目标的计划是P。由于对目标(动机)、计划的理解是深一层意义上的理解,我们在本模型中暂不详细讨论,有关的问题可参阅文献[3],[6]。

3.5 例子

下面试用上述模型来表示一些句子的意义。

例1 他坐火车去北京。

ACTION: PTRANS

agent: 他

object: 他

from: nil

to: 北京

instrument: ACTION: PROPEL

agent: 火车

object: 火车

from: nil

to: 北京

and

ACTION: PTRANS

agent: 他

object: 他

from: nil

to: 火车

表示他上了火车,并且火车开往北京,他去了北京。

例2 他喝了杯茶。

ACTION1: INGEST

agent: 他

object: 茶

from: 杯子

to: mouth(他)

instrument: ACTION2

ACTION2: PTRANS

agent: 他

object: 杯子

from: 桌子

to: mouth(他)

instrument: ACTION3

ACTION3: MOVE

agent: 他
object: hand(他)
from: nil
to: 杯子

例 3 他决定去上海

ACTION1: MBUILD

agent: 他
object:
input: nil
output: ACTION2

from: memory(他)

to: mind(他)

例 4 他告诉她那个故事。

ACTION1: MTRANS

agent: 他
object: 故事
from: 他
to: 她
instrument: ACTION2
and ACTION3

例 5 他在书店买了本书。

ACTION1: ATRANS

agent: 他
object: 钱
from: 他
to: 店员
result: ACTION2
and STATE1

STATE1: 拥有(ownership)

object1: 他
object2: 书

例 6 他想吃饭。

STATE1: 饥饿(hungry)

agent: 他

and ACTION4

ACTION4: GRASP

agent: 他
object: 杯子
from: nil
to: hand(他)

ACTION2: PTRANS

agent: 他
object: 他
from: nil
to: 上海

ACTION2: SPEAK

agent: 他
object: 故事
from: mind(他)
to: nil

ACTION3: ATTEND

agent: 她
object: ear(她)
from: 他
to: mind(她)

ACTION2: ATRANS

agent: 店员
object: 书
from: 店员
to: 他
result: ACTION1
and STATE2

STATE2: ownership

object1: 店员
object2: 钱

STATE2: GOAL

agent: 他

value: -5
result: STATE2

objective: ACTION; INGEST
agent: 他
object: 饭
from: nil
to: mouth(他)
in(他)

例 7 见到你真高兴, 你近来身体好吗?

ACTION: ATTEND

agent: 我
object: eye(我)
from: 我
to: 你

result: STATE1
and STATE2

STATE3: 位于(Mental Location)

object1: STATE4
object2: LTM(我)

STATE4: 健康

agent: 你
value: nil

tense: q, p

STATE1: 喜恶

agent: 我
value: +7

STATE2: GOAL

agent: 我
objective: STATE3

这里, 句子中隐含的“想”, “知道”, “身体好吗”的“知道”被表示成位于长期记忆(LTM)中这一状态。

例 8 这书多少钱?

ACTION1: ATRANS

agent: 买者

object: 钱

from: 买者

to: 卖者

result: ACTION2

tense: q

ACTION2: ATRANS

agent: 卖者

object: 书

from: 卖者

to: 买者

result: ACTION1

4 实现的基本算法

在从输入到意义表示形成的过程中, 我们利用了以下的数据结构:

INLIST: 输入正文放入的表。

ELIST: 期望表, 存放当前被输入中一单词所激活的那组期望, 初始为空。

LEXICON: 字典, 存放系统能够处理的所有单词及与之相联的那组期望。

EXPECTATION: 期望, 为在需要时执行的函数。其形式包括:

TEST: 期望得以实现所需满足的条件;

ACTION: 期望的测试条件成立时所需进行的动作。

FAILLIST: 存放上次执行 ELIST 中每个期望时, 测试为假, 即失败的那些期望, 初始为空。

OLDELIST: 存放以前所有测试为假的那些期望, 初始为空。

FAILLOLDELIST: 存放 OLDELIST 中期望的测试为假, 即失败的那些期望, 初始为空。

CLIST: 存放形成的意义表示, 包括局部的与最终的。

下面写出利用以上数据结构实现意义表示模型的算法。

WHILE (INLIST 非空) DO

{ 移去 INLIST 中第一个字 WORD;

在 LEXICON 中查找 WORD, 将与它当前意义有关的那组期望激活, 存入 ELIST;

WHILE (ELIST 非空) DO

{ 移出当前 ELIST 的第一个期望, 赋给 EXPECTATION;

IF EXPECTATION 的 TEST 为真,

那么, 执行 EXPECTATION 的 ACTION 部分,

否则, 将此 EXPECTATION 放入 FAILELIST;

}

WHILE (OLDELIST 非空) DO

{ 移去当前 OLDELIST 的第一个期望,

赋给 EXPECTATION;

IF EXPECTATION 的 TEST 为真,

那么, 执行 EXPECTATION 的 ACTION 部分,

否则, 放入 FAILLOLDELIST;

}

合并 FAILLOLDELIST 与 FAILELIST,

将结果赋给 OLDELIST;

置 FAILELIST 为空;

}

期望的 TEST 部分有三种类型: 恒真, 这时必然执行相应的 ACTION; 是否在 CLIST 中包含了一特定的结构(可能有次序要求); 是否在输入中含有一特定的词或短语。

期望的 ACTION 有三种类型: 将一意义表示结构放入 CLIST; 将一子结构放入另一结构的某一槽中(同时移去该子结构); 激活另一组期望, 置入 ELIST 中。

5 意义表示器 MR (Meaning Representor) 程序

利用上述算法, 我们在 IBMPC/XT 微机上用 GCLISP 语言编到了意义表示生成器 (MR 的原型程序。该程序能将用自然语言输入的句子转换成相应的意义表示。目前该系统存储了大约 30 个基本动词, 50 个涉及客体、状态所及的其它类型的词, 它能利用这些词的含义 (以一组期望的形式存入相应字的字典中), 得出上述意义表示中的意义表示。

该程序先读入一英文句子, 然后依次对每个字找出字典中其含义在当前情形下的特定含

义——这是通过先前读入的字的期望的约束以及检查下面的字的含意而完成的——即一组在此特定句子中应激活的该字的一组期望(该字的其它含意,在当前输入中便失去了意义);然后,依次处理每个期望,保留测试失败的期望,外理先前失败的期望,如此下去,直至处理完最后一个字,这时,就形成了所输入句子的意义表示。

下面,我们详细考察一个例子。

设输入句子为: A dog went home.

字“a”的期望,在我们程序的字典中是:

TEST: (hasnew)

ACTION: (slotfiller ref def)

表示测试CLIST中是否有一新的结构填入,若有,则将此结构的ref(指称)槽赋以def(确定)值。由于“a”是第一个字,所以此期望的测试失败,故被放入FAILELIST,并且由于它只有一个期望,所以在将FAILELIST赋给OLDELIST之后,就读入第二个字。

字“dog”的期望是:

TEST: (alwayst)

ACTION: (add (dog (class physical-object)(type animate)))

这一期望表示将一局部意义,即有关“dog”的概念结构加入到CLIST中。由于它的测试是恒真(alwayst),所以相应的动作立即得到执行。且对OLDELIST无影响。此时,原OLDELIST中的期望再次得以激活,测试hasnew为真,动作得到执行,于是此“dog”结构加入了一槽ref,其值为def,同时,OLDELIST为空了。

下面读入的字是“went”,由于读字本身除了基本的PTRANS含意外,还与其它字组合形成许多别的意思,所以它的字典定义中应包括对应于所有这些含意的各自的那组期望。不失一般性,下面只描述出现在“go home”及“go to school”这两类情形中的期望:

TEST: (nextword to)

ACTION:

TEST: (nextwordtype place)

ACTION: (expl)

TEST: (nextwordtype verb)

ACTION: (currentword)

TEST: (nextwordtype place)

ACTION: (expl)

这组期望的含意是:如果下一字为to,则进一步测试下一字的类型,如为一地点,则执行expl所指的那组期望,否则若为一动词,则激活当前词(currentword)即为该动词的期望(此情形这里不予考虑),如果下一字不为to,而是一直接的地点,则也激活expl所指的那组期望:

(add(PTRANS (agent nil)

(object nil)

(from nil)

(to nil)

```
(putin PTRANS to)
((Is there animate before PTRANS)
((putin PTRANS agent)
(putin PTRANS object)))
```

这组期望的第一项必为动作(注意: 动作部分一般可递归地包含子期望), 它的含意是在 CLIST 中加入 PTRANS 结构, 并将当前词用 putin 函数放入 PTRANS 结构的 to 槽中, 接下去测试是否在 CLIST 中存在(Is there)一先于 (before) PTRANS 的类型为 animate 的结构, 若有, 则置它进入 PTRANS 的 agent 槽和 object 槽。在这里的输入句子中, 上面的测试成立, 于是在读入下一字之前的时刻, CLIST 中已形成这样的结构:

```
(PTRANS (agent(dog(class physical object) (type animate) ref def)))
(object (dog(class physical object) (type animate)(ref def)))
(form nil)
(to home))
```

由于在测试(nextwordtype place)中, 程序已读入了字“home”, 且此字是一地点, 所以就不再另对“home”作处理, 最终句子的意义表示即上面 CLIST 中的表达式。

上述例句尽管看起来比较简单, 但 MR 程序所依据的算法却对所有已定义和将要定义的字的意义分析、表示都是通用的。我们只须表示完整一个字的各种含意于相应的各组期望中, 就能不对程序其它部分作任何修改而表示出各种输入句子的意义表示。目前, MR 程序中尚未考虑词根变形, 指称确定, 特别是对形成的意义表示的推理问题, 但它对本文节 4 所讨论的基本算法的实现, 为进一步实现各项工作提供了十分必要的基础。

6 结 论

本文给出的 NLU 的意义表示模型, 是作为进一步进行语言处理与理解的必不可少的基础。所提出的形成意义表示的基本算法和具体实现为以后的工作提供了很好的准备。由于篇幅有限, 文中没有给出实现的全部细节, 也没有提及许多重要问题。但从本文可得到以下结论:

- (1) 意义表示的形成是进行 NLU 的第一步;
- (2) 意义表示以基本行为、客体名称、基本状态为框架, 可用来描述语言中的内部意义
- (3) 意义表示是作为记忆中利用常识知识进行推理的基本词汇与指针[4]

参考文献

- 1 Minsky M. A framework for representing knowledge. In Winston, P.H. The psychology of computer vision New York, McGraw-Hill, 1975
- 2 Schank R.C. Conceptual Information processing. New York, Elsevier, North

Holland, 1975

- 3 Schank R C. Abelson R P Scripts. plans, Goals and Understanding Hillsdale, New york, Erlbaum, 1977
- 4 Schank R C. Dynamic Memory. Cambridge University Press, New york 1982
- 5 Schank R C. The Cognitive computer. Addison-Wesley Publishing Company, Inc. 1984
- 6 Wilensky R. Planing and understanding. Addison-Wesky Pudlishing Company, Inc. 1983

A Meaning-representation Model in Computer Natural Language Processing

Tang Yongyan

(Dep. of Auto.)

Abstract The paper describes a meaning-representation model in computer natural language processing. The model includes mainly the types and representations of primitive acts and states. In addition, the implementation algorithm of the generation from language input to meaning representation is given. Finally, we describe the MR(Meaning Representor) program which have been implemented in GCLISP.

Subjectwords Language processing, Microcomputer/NLU