

人工智能语言、逻辑及发展趋势

胡运发 (国防科技大学计算机研究所)

【摘要】

本文综述了人工智能语言发展历史,及其在人工智能研究中的地位和作用,说明了不同的人工智能语言对应于不同的逻辑系统。逻辑系统的性质决定了人工智能语言的基本特征,表达能力和执行效率。本文试图阐明人工智能语言的发展方向。

人工智能语言的研究是十分重要的,因为实践已经证明,它是人工智能系统研究的基础环境。日本宣布第五代计算机计划后,人们又逐步认识到它也是新一代计算的关键。人工智能语言本身就是一种具有问题求解能力的智能系统,它的成功,不仅具有工具上的意义,也具有理论上和方法上的意义。

一 人工智能语言的历史概况

人工智能语言在人工智能发展的历史上起着中心的作用。因为人工智能工作者在设计一个智能系统时,往往需要提出一种新的思想,这种新思想往往比较复杂,需要强有力的语言来表达,并且这种语言可被执行,因此,研究者可以很快地验证其思想的正确与否。

许多人工智能先驱者都在人工智能语言方面作出了重要贡献。例如Newell, Simon等人早在1957年就提出一种人工智能语言,名叫IPL。他们在IPL中首次引入符号计算,联想和表处理概念。它是由于早期的人工智能研究的需要而产生的,又反过来推动了人工智能的研究和发展。实际上,许多早期的人工智能研究系统就是用IPL语言编写的。直至70年代还有人用IPL语言书写人工智能程序。

在人工智能历史上起作用最大的要算LISP语言。它是另一人工智能先驱者McCarthy在1960年创立的。此后,一直占统治

地位。差不多所有重要的人工智能系统均是用它编写的。它培养了一代又一代人工智能工作者。它被誉为人工智能数学语言。它之所以起到如此大的作用,原因在于(1)它具有描述语言的风格,人们只要书写递归函数定义而无需顾及运算的细节。(2)它强调数据和程序的统一性。因为智能行为的一个特征允许系统自我改进,程序既是工作的主体,又是工作的对象,因此这种统一性是必要的。

(3)LISP语言给出某种知识表示方法——特性表。许多知识表达方式如语义网,框架结构,脚本,幽灵等均在此基础上发展而来。(4)它具有不动点指称语义和λ-演算操作语义,保证了语言系统的正确性。

70年代,特别是80年代,另外两种人工智能语言异军突起,越来越引起人们的关注。一种是逻辑型语言,以PROLOG为代表。过去,在计算机科学中,总是把一阶逻辑作为规范语言。PROLOG的问世,说明它还可作一种程序设计语言,这是一种很了不起的变化。著名的学者Lloyd甚至称赞说它是一次革命。有人认为PROLOG可以看成一种可执行的规范语言。如Kowalski论证它用作软件工程中规范说明。在PROLOG中,人们给出仅是有关世界的断言,PROLOG系统的任务保证找出它的逻辑结论。如果说,程序的语义问题已经困扰了许多计算机科学家,的话,PROLOG则第一次清晰地指明了PR-

OLOG程序(即断言)的语义是什么。PROLOG还有其他一些特点:(1)它比LISP更富于表达能力。(2)它在知识表示,知识管理方面具有明显的优势。(3)它具有模式匹配、推理、回溯等人工智能系统中必备的基础功能。(4)和LISP一样,程序和数据是统一的。(5)程序的指称语义是模型论,操作语义是证明论。PROLOG已经成为LISP语言最强有力的竞争者。

另一种新型的语言是对象语言。典型代表是Small-talk。它是施乐的PARC中心开发的一种程序设计环境。它主张世界是由一组对象组成,对象有内部状态和内部过程。对象通过信息的传递而发生联系。某些对象可按类属关系自然地构造成层次关系并允许特征共享和继承。对象语言在模块性和并行性方面具有独特的优点。

近年来,世界范围的人工智能语言的研究变得十分活跃。逻辑型,函数型,对象型语言都在向深度和广度方向发展。在理论和实践上将三种风格合成起来的研究,也在取得重要的成果。

二 人工智能语言的基本要求

并不是任何计算机语言都可以称之为人工智能语言。那么,什么是人工智能语言呢?由于人工智能本身尚无定论,关于人工智能语言也没有统一的规定。不过我们可以根据人工智能的基本要求明确人工智能语言必须具备的一些特征。

定义本身并不是十分重要的。但是定义集中地反映了人们的认识。例如什么是人工智能有许多看法:人工智能就是研究如何使计算机更聪明;人工智能是人类思维的模拟;人工智能是人类智能的放大;人工智能是研究启发式方法的一门科学;人工智能的中心是符号和搜索等。陈火旺教授认为人工智能是人类智能人工实现方法的一门科学。作者认为陈火旺教授这一定义有两点值得注意:(1)强调的是研究人类智能而不是其他智能。(2)强调的是人工实现。根据此

两点可以把人工智能这门科学和仿生学,哲学,逻辑学,心理学,思维科学等学科明确地区分开来。何谓人工实现呢?作者认为所谓人工实现指的是建立一种充分有效的符号物理系统,此系统具有某种问题求解的能力。并非任何具有演算能力的物理系统都可以称为人工智能实现,例如模拟计算机不能称为人工智能系统,因为它没有符号演算能力。图灵机(如果实现的话)是符号物理系统,但它不能称为人工智能系统,因为它没有与人类求解问题相匹配的效率。普通计算机软件系统可以看作符号物理系统,但它们不具备问题求解能力(因为所谓问题求解指的是人们只需给出问题世界的描述,系统自动地产生问题的解),故也不能称人工智能系统。这种看法标准比一般人的看法高一点,但有一个好处,就是把人工智能系统和其他系统区别开来。

如果我们按上述看法来认识人工智能。人工智能语言是描述人工智能的形式系统。它必须具备如下特性:(1)通用的问题世界的描述语言。(2)能正确地完备地产生问题的解。(3)充分有效的。(4)有某种自我改进的能力。

第一条要求人工智能语言是某种通用的规范语言,它无需用户顾及解题过程中的细节。第二条要求语言本身就是某种问题求解系统,无需用户给出求解的具体步骤。第五条要求有适当的运行效率。第四条的起码要求是程序和数据的一致性和某种高阶能力。

满足上述要求的语言决不止一种语言。例如和LISP相似的还有FP, FFP, VAL, KRC, OBJ等,和PROLOG相似有LOGLISP, 和Small-talk相似的有FOOPS等。

如何从众多的人工智能语言中选择一种呢?如何创建更加符合人工智能要求的语言呢?实际上,尽管人工智能语言很多,但表达能力有强有弱,执行效率有高有低,使用起来有的方便有的不太方便。为什么会产生这

些差异呢?其根源在于任何一种人工智能语言都有一种更加深刻的逻辑系统来支撑。此种逻辑系统的性质决定了相应人工智能语言的性能。所以要弄清人工智能语言有关问题,我们还必须深入到相应的理论体系中去。

三 人工智能语言的逻辑基础

3.1 一阶逻辑作为人工智能语言

一阶逻辑语言是一种构成合式公式的形式符号系统。一般的合式公式 wff 构成规则是若 A 和 B 是一种 wff, 则 $(\neg A)$, $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$, $(A \leftrightarrow B)$, $(\forall x A)$ 和 $(\exists x A)$ 也是一合式公式 wff。不含自由变元的公式称为封闭公式。

一阶逻辑语言的语义是它的模型论。一个公式 w, 若模型 U 满足它, 则 w 是可满足的, 记作 $U \models w$ 。若任何模型都可满足它, 则它是有效的, 记作 $\models w$ 。

若任何满足 $A_1, \dots, A_n$ 的模型也一定满足 B, 则称 B 是 $A_1, \dots, A_n$ 的逻辑结论, 并且当且仅当 $A_1, \dots, A_n \rightarrow B$ 是有效公式。

一阶语言的操作语义是它的证明论。它是根据公式 $A_1, \dots, A_n$ 推导出它的逻辑结论 B 的一种形式推导方法。

由于任何一公式与它的子句具有相同的满足性, 所以一般公式的形式推导等价于子句公式的形式推导。著名的 Robinson 归结原理就是基于子句之上的一种形式推导方法。Robinson 归结方法的基本结论是一公式 w 是不可满足的, 当且仅当经过连续归结, 得到一空子句口。Robinson 归结是正确的, 即若得到一空子句, 则该公式必然是不可满足。Robinson 归结是完备的, 即任一不可满足的公式, 一定可导出空子句。

从理论上说, 一阶谓词是可以作为一种人工智能语言的。倡导逻辑程序设计的人们也是这样期待的。但是, 一般而言, 任何一种归结策略, 如果它是正确和完备的, 就可能存在某种组合爆炸问题, 因而效率是极低的。当然存在某些效率极高的归结策略, 如

线性输入归结, 但令人遗憾的是它是不完备的。Kowalski 的贡献在于他发现一阶逻辑的真子集—Horn 逻辑, 它仍然具有强大的表达能力(如表达可计算函数), 而且使用一种线性输入反驳策略—SLD 归结, 在 Horn 逻辑范围内是完备的。在此真子集范围内, 强大的表达能力和高效率演绎机制和谐地统一起来了。正是这种统一导致了 PROLOG 语言的诞生。所谓 Horn 逻辑是一种受限的子句逻辑系统, 即任一子句中, 正文字不得多于一个。

所谓归结是一种推理规则。令 C_1 和 C_2 分别包含正文字 L 和负文字 $\neg L'$ 的两个子句, 如果存在一置换 σ , 使得 $L\sigma = L'\sigma$, 则从 C_1 中删除 L 得 C_1' , 从 C_2 中删除 $\neg L'$ 得 C_2' , 并且令 $C_3 = (C_1' \cup C_2')\sigma$, 则 C_3 称为 C_1 和 C_2 的一个归结式。 C_1, C_2 分别称 C_3 的归结因子。产生置换 σ 的操作叫做一致化。SLD 归结限于 Horn 子句。Horn 逻辑系统是 PROLOG 语言的理论基础。出于简化和效率上考虑, PROLOG 语言对 SLD 归结进一步加以限制, 即限于深度优先的 SLD 归结。尽管深度优先原则破坏了 SLD 的完备性, 但却换取了效率。

上面我们只是简单地叙述与 PROLOG 相联系的逻辑系统—Horn 逻辑, 它从本质上决定了人工智能语言 PROLOG 的表达方式, 表达能力和效率。

并表达方式上说它就是一阶逻辑语言, 和人们自然语言十分接近, 因此方便人们使用。

从表达能力来说, 它是受限的一阶逻辑, 其能力当然低于一阶谓词逻辑能力, 但已经证明它足以表达一切可计算函数, 从可计算角度看, 能力也是很强的。

从效率的角度看, 它是所有一阶逻辑系统效率最高的系统。经过近 20 年的努力, PROLOG 系统已经达到实用水平, 并且可以和 LISP 语言的效率相比较。

PROLOG 语言还具有其他一些特征。

(1) 它是一种受限的可执行的规范语言。

(2) PROLOG 程序具有模型论语义，即程序的语义是程序的逻辑结论。操作语义是SLD归结，是正确的完备的高效的。

(3) 它具有能力很强的结构模式匹配能力。但究其本性只具有等价关系中的自反性，并不具备等价关系中其他特性，如对称性、传递性、替换性等。

(4) 经适当扩充，具有高阶能力。

3.2 函数语言的逻辑基础 函数语言是递归地构造函数项的形式符号系统。函数语言的基本概念来自数学。函数的基本特点有三：(1) 单射性，即对任一函数 f 如 $x=y$ ，则 $f(x)=f(y)$ 。(2) 变量引用的透明性，即变量在其辖域内是同一的。例如 $f(x)=x^2+2x+1$ ，式中各个 x 是具有同一值。在此条件下， $x^2+2x+1=0$ 才能找到相应的解。反之， $x^2+2x+1=0$ 就可能会失去意义，例如第一个 x 为 3，第二个 x 为 -5 也满足方程，但破坏了引用透明性，方程就没有意义了。(3) 相等性。在函数语言中，相等性具有自反性，传递性，对称性和替换性。相等性保证了函数语言的演绎能力。

函数语言的语义是不动点理论。不动点和模型之间有着密切的关系。对于 Horn 逻辑来说，它的最小不动点就是它的最小 H-模型。

函数语言的操作语义是归约。所谓归约是一种系统 $\langle S, \rightarrow \rangle$ ，其中 S 是一状态的集合， $\rightarrow$ 称为归约，它是一种 S 的二元关系，即 $\rightarrow \subseteq S \times S$ 。不同的归约系统，表达能力不同，效率也不同。由于表达能力上差异，并非任何归约系统都能够模拟其他归约系统。能够模拟其他归约系统的归约系统称为通用的归约系统。Turner 在 1979 年建议组合算子演算 (combinator calculus) 作为一种通用的归约系统，并且证明了 FP, SASL 等语言均可编译为一些组合算子。但是 Michael 论证了这一结论并不充分。 λ -演算似乎是通用的，但是由于它不可以模拟并行 or 操作，也缺乏充分的论据。

什么是有效的归约系统呢？Michael 给出的条件是：

(1) 状态 S 是可判定的 (不失一般性， S 可以是非负整数)

(2) $\{\beta | \alpha \rightarrow \beta\}$ 是有限的， α 的所有可能的分枝函数的个数 $n(\alpha) = |\{\beta | \alpha \rightarrow \beta\}|$ 是完全可计算的。

(3) 由 $t(\alpha) = \{\beta | \alpha \rightarrow \beta\}$ 的定义的转移函数 $t: S \rightarrow P(S)$ 是完全可计算的。

其中 $\alpha \rightarrow \beta$ 表示一条归约规则，直观含义是可通过一步计算，从 α 归约为 β 。

归约系统的一个重要性质是汇流性 (confluence property)。所谓汇流指的是在归约系统 $\langle S, \rightarrow \rangle$ 中如果 $\forall \alpha, \beta, \gamma \in S$ 且 $\alpha \rightarrow^* \beta$ 且 $\alpha \rightarrow^* \gamma$ 则存在 $\delta \in S$ 且 $\beta \rightarrow^* \delta$ 且 $\gamma \rightarrow^* \delta$ 。此性质可图示为



汇流性也常常称之为 Church-Rosser 性质。因为 Church-Rosser 在 λ -演算中建立了

类似的性质。汇流性是十分重要的性质。因为它保证了标准形是唯一的，并保证通过向前归约能得到它的标准形。它也是 eager 型计算机机制的必要条件。

3.3 合成语言的逻辑基础 由于多方面原因，促使人们探索合成语言各种途径。各种语言在一定意义上存在自己的优点和长处。例如，尽管逻辑语言在表达能力上强于函数语言，比较接近人们的思维形式，效率上也可以和函数语言相比较。但函数语言也有自己的优点，如语义的等价能力，而且在一定条件下，时空效率也较高。所以人们总希望通过某种途径将他们合成起来。

合成的方式有三种：(1) 扩展型。以逻辑型或函数型语言为基础，增补其不足部分的功能，例如 LogLISP 和 Funlog。(2) 环境合成型。几种不同风格的人工智能语言系统，通过共享环境的办法统一到一个复合系统之中，例如 POPLog。(3) 理论合成型。首先建立一种表达能力更强的逻辑系统，然后在其上建立一种功能更强的能够容纳多种

语言功能的新型语言。如EQlog系统, FO-OPS系统就是这样做的。EQlog是在有序类带等词的Horn逻辑系统之上建立起来的一种新型语言,它包含了函数语言的能力,逻辑语言的能力以及约束语言的能力。

实际逻辑的不同侧面往往对应程序设计语言的不同方面。例如方程逻辑抓住程序设计的函数方面。多类型逻辑抓住了程序设计中的类型方面, Horn逻辑抓住了逻辑程序方面, 自反逻辑抓住了面向对象语言的能力。这就启示我们建立一种新的逻辑体系是组合各类人工智能语言范例(Paradigms)的正确途径。美国斯坦福人工智能研究中心著名学者J. A. Goguen的主张是: 在当前, 自反的有序类的含有等词的Horn逻辑是统一对象, 函数和逻辑语言从而形成统一的简单的表达能力更强的新型人工智能语言的逻辑基础。

但是从长远的观点看问题, 为了不就事论事以及阻止逻辑系统的“人口爆炸”现象的发展, 可把机构(Institution)理论作为逻辑程序设计概念的形式基础。

所谓一个机构它是一种逻辑系统。其定义是:

①逻辑语言中的语句是机构中语句。

②逻辑语言的操作语义是机构中具有适当效率的演绎。

③逻辑语言的指称语义是由机构中的一类模型给出。(初始模型比Herbrand论域更可取)。

上述分析告诉我们人工智能语言性质取决于相应的逻辑体系的性质。研究人工智能语言的首要任务是研究相应的逻辑系统。

参考文献

- [1] J. A. Goguen & J. Meseguer, Models and Equality for Logical Programming, Lecture Notes in Computer Science, Springer-Verlag, 1987.
- [2] J. O. Michael, Equational logic as a Programming Language, MIT Press, 1985.
- [3] 陈火旺, 人工智能与逻辑学, 国防科技大学技术报告, 1988.
- [4] J. W. Lloyd, "Foundation of Logic Programming" Springer-Verlag, Berlin, 1984.

《系统软件研讨会》征文通知

中国计算机学会软件专业委员会系统软件学组将于1989年7—8月在哈尔滨召开第二次学术讨论会, 现将有关征文内容及注意事项通知如下:

一、征文内容

- 操作系统, 编译系统等理论, 实践及方法。
- 微机系统软件的开发及汉化工作。
- 引进系统的消化, 分析, 开发与创新经验及最终国产化的途径。

二、征文要求及方法

应征论文要反应本单位或个人实践或理论研究成果, 学术上要有严谨的科学性和一定的实用价值, 文章论点要明确, 文字精练, 数据可靠, 图表清晰并附有文摘, 已在全国性会议, 国内外刊物上发表过的文章, 请勿应征, 征文一律不退稿, 请作者自留底稿。截稿时间为1989年4月底, 稿件请寄往:

北京中关村中国科学院软件研究所程虎同志(邮政编码 100080)。将于6月份发出录用通知。

中国计算机学会软件专业委员会系统软件学组