

多 Agent 系统中含糊性的处理方法

王自强, 冯博琴

(西安交通大学计算机科学系, 西安 710049)

[摘要] 为了能够正确理解含糊语句所传达的信息概念, 分析了含糊性如何影响对话双方的知识, 尤其是含糊语句产生后, 谈话双方彼此了解的情况。提出基于 Kripke 结构的多 agent 系统中含糊性的处理方法, 多 agent 系统的信息状态用 Kripke 结构表示, 并用 tell 函数来实现系统间的通信, 而 tell 的应用受到 Grice 原理的约束。此方法能够正确地把由含糊子句传递信息中的语义含糊和感觉含糊区分开来, 因而对于研究多 agent 系统具有一定的实用价值。

[关键词] 多 agent 系统; 含糊; 处理方法; Kripke 结构

[中图分类号] TP181 **[文献标识码]** A **[文章编号]** 1009-1742(2003)09-0072-06

1 引言

在多 Agent 系统中, 如果给定的目标和环境包含有不确定性, 则 Agent 就无法更好地执行正确的行为。这种不确定性与不确定的环境、其他 Agent 的行为及智能 Agent 的行为密切相关。虽然有许多学者试图解决这一问题, 如他们提出的可信度因子法、主观贝叶斯方法及 Dempster-Shafer 的证据理论, 但是他们基本上都是采用概率和统计的方法^[1], 而这些方法常采用不确定知识的经验模型, 因而不具有逻辑系统所具有的直观语义; 并且这些方法无法表示由自然语言组成的用户目标, 例如人类语言中有一类常用来表示事物的势态、人的情态以及过程变迁的词。像“大、中、小”, “必然、可能”, “知道、认可”等是无法采用概率和统计的方法来描述的。并且多 Agent 系统中用于描述和解决现实世界问题的知识常常带有不确定性; 因而 Agent 系统应能体现和反映知识的这种不确定性, 即能适当地表示不确定知识, 利用不确定知识的推

理是非单调的、次协调的^[1]。只有如此, Agent 才能在不确定的环境中, 采取适当的行动来解决给定的问题, 或得到优化的解决方案。

由于模态逻辑通过引入模态词较好地解决了现实世界中的含糊性, 有关专家提出了利用模态逻辑来对含糊进行形式化描述, 如文献 [2] 提出的“上下文方法”, 但仅限于单个 Agent 的情况, 不适合于上述的多 Agent 系统, 于是笔者提出了基于 Kripke 结构^[3]的多 Agent 系统的信息处理方法。

2 具有含糊性的多 Agent 系统

假设在 1 个有 n 个 Agent 构成的组中, 每个成员了解外界和其他 Agent 的有关情况, 并且每个 Agent 所采取的唯一行动是告诉同组中其他成员它所知道的事情。并且, 还假设它仅告诉它所认为正确的事情, 且其他成员对它的话不产生怀疑。为了表述简单, 认为 Agent 之间的知识是不冲突的。

为了对 Agent 的信息状态进行处理, 采用了赋值的 Kripke 结构^[4]

$(M, w, V): M = \langle W, \{R_i\}_{i \in A}, A \rangle, w \in W。$

V 是评价函数, $V: W \rightarrow P^2$, P^2 是命题变量 P 的幂集, W 是可能世界的集合, A 表示一个有限的 Agent 集, $\{R_i\}_{i \in A}$ 表示对于每个属于 A 的 Agent i 在世界中所能可达的关系。并且, 只有 Agent i 在世界 w 中认为世界 w' 可能时, 可以说 wR_iw' 。另外, 用模态运算集 $\{K_i\}_{i \in A}$ 来表示知识, 其中 $(M, w) \models K_i\varphi$, 并且当对所有满足 wR_iw' 的 w' 都有 $(M, w') \models \varphi$ 成立。以下将省略下面的结论(因为它是上式的特殊情况): $(M, w) \models \varphi$, 当且仅当对于任意的 V 都有 $(M, w, V) \models \varphi$ 成立。

在文献 [4] 的基础上, 定义了群体知识 E 和常识 C :

$(M, w) \models E_G\varphi$, 当且仅当有

$(M, w) \models \bigwedge_{i \in G} K_i\varphi$ 成立;

$(M, w) \models C_G\varphi$, 当且仅当有

$(M, w) \models \bigwedge_{i \in G} E_i\varphi$ 成立。

此信息结构满足自反性和串行性, 即它们满足 KD45, 对所有的 $w \in W$ 和 $i \in A$ 满足以下性质:

(D) $(M, w) \models \neg K_i \perp$, (串行性)

(4) $(M, w) \models K_i \rightarrow K_i K_i \varphi$, (正自反)

(5) $(M, w) \models \neg K_i \varphi \rightarrow K_i \neg K_i \varphi$, (反自反)

Agent 被赋予非常有限的通信能力, 即它们唯一通信行为的形式为 $\text{tell}(i, G, S)$, 其定义如下:

定义 1 tell 。

tell 是一个从赋值 Kripke 结构到另一个 Kripke 结构的函数, 它可以看作一个更新函数, 并且 tell 有 3 个参数 (i, G, S) , 其中 i 表示说话的 Agent, $G \subseteq A \setminus \{i\}$ 表示听话的 Agent, S 是表示属于某一特定语言段 L^{spec} 上的自然语言句。虽然在模型中允许多于 2 个的 tell 行为同时发生, 但是当 2 个 tell 行为同时发生时, 将需要系统并行更新。为了描述方便, 考虑 1 次只允许 1 个 tell 行为发生。

1 个从 L^{spec} 到命题逻辑 L^{pl} 的函数 τ 将产生 1 个可能的代表 S 的语义, 实际上 $\tau(S)$ 返回的是等价知识类的集合。并且那些等价的知识属于相同的类, 用 $[\varphi]$ 表示 φ 的等价类。

事实上 S 在语义上含糊并不意味着说话者和听话者认为 S 是含糊的^[5], 下面将看到如何形式化地定义语义含糊 (semantic ambiguity) 和感觉含糊 (perceived ambiguity) 之间的差别。

3 语义含糊与感觉含糊

定义 2 语义含糊。

如果 $S \in L^{\text{spec}}$, 于是说 S 在语义上是含糊的, 当且仅当有 $|\tau(S)| > 1$ 成立; 也即是至少存在 2 种不等价的方式来表示 S 的语义。

根据 Pinkal 在文献 [6] 中的解释可知: 一个句子在语义上是含糊的, 当且仅当在一定的条件下, 尽管有足够的相关事实, 但是仍然无法把 true 或者 false 作为它的真值。

例如, 句子“奥迪是一种快车”中所包含的修饰词“快”, 当解释为“对现代车而言是快车”时, 这个句子的真值为 false; 但是, 当把“快”解释为“对它同时代的车而言是快车”时, 这个句子的真值取决于和奥迪相比的车的类型。

于是可以得出如下结论: 令 p 和 q 表示 2 个命题, 命题 p 比 q 更加准确 (相对含糊而言), 当且仅当满足如下 2 个条件: 在 q 为 true (false) 的状态世界中, p 也一定为 true (false); 在 q 真值不确定的一定环境下, p 的真值可能为 true 也可能为 false。

定义 3 感觉含糊。

假设 S 表示一个语义上含糊的句子, 正如定义 2 所示, 于是有 $\text{tell}(i, G, S)(M, w, V) = (M', w', V')$ 成立, 并且下面的条件也成立:

$(M', w', V') \models \neg C_{\{i\}} \cup G\varphi$;

$(M', w', V') \models E_{\{i\}} \cup G(V_{[\varphi] \in \tau(S)} C_{\{i\}} \cup G\varphi)$ 。

于是可以说 S 是感觉含糊的。

感觉含糊的解释如下: 令 A 和 B 表示在一次谈话中的 2 个参与者, 在谈话环境 D 下, A 讲给 B 的言语 U 被 B 认为在环境 D 中是含糊的, 需要满足的条件是 B 在 D 下处理 U 时获得的结果不同于 A 对 U 的解释。

语义含糊与感觉含糊之间的区别: 语义含糊是属于语言语法的一部分, 句子在语义上是含糊的, 可能不被听话者所注意, 事实上它往往是由语言研究者所发现的; 而感觉含糊是由理解 (interpretation) 的过程所导致的结果 (从本质上讲是无效的 (defeasible) 结果), 因而可能会在错误传达 (miscommunication) 信息的情况下导致多种解释。另外, Poesio 在文献 [7] 中认为, 语义含糊着重于描述多重感觉 (multiplicity of sense) 的理论模型, 而感觉含糊着重于描述有关推理。

4 处理含糊性的准则

对多 agent 通信行为的执行做以下约束, 这些约束是在 Grice 原理的基础上进行的, 可以看作是 Grice 原理在多 Agent 系统中的部分应用^[8]。

令一个给定的形如 (M, w, V) 的赋值 Kripke 结构表示当前的信息状态。下面准则是对 tell 应用的约束, 如果有 $\text{tell}(i, G, S)(M, w, V) = (M', w', V')$, 则有 $\exists [\varphi] \in \tau(S)$ 满足如下准则:

$$1) (M, w, V) \models K_i \varphi$$

定性准则: 不要认为你缺少足够的证据。这一点是通过要求说话者至少知道一条 S 的知识为真。

$$2) (M, w, V) \models K_i \neg C_{i|i} \cup G\varphi$$

定量准则: 为当前的信息交换贡献出你的知识。Agent i 知道至少一条知识 S 不是公共知识。

$$3) \forall [\Psi] \in \tau(S) \setminus \{[\varphi]\} (M, w, V) \models K_i (C_{i|i \cup G} \Psi \vee C_{i|i \cup G} \neg \Psi)$$

行为准则: 避免含糊。Agent i 知道说话者和听话者之间的公共知识是属于 S 的所有知识 Ψ 且 $[\Psi] \neq [\varphi]$, 并且其值的真假是确定的。

$$4) (M', w', V') \models C_{i|i \cup G} \varphi$$

评价准则: 告诉完 S 后, φ 成为听话者和说话者 i 的公共知识。

同文献[9]的规定相同, 条件分为前件 (pre-condition) 和后件 (post-condition)。前件指的是通信行为执行之前 (也就是在更新原始的 Kripke 模型之前) 必须成立的条件, 后件指的是通信行为结束后那些必须成立的结果。

准则 1 至准则 3 是把 tell 应用到赋值 Kripke 结构时的前件, 而准则 4 是其 后件, 也就是把说话者的贡献加入到公共知识中。详见文献[10]。

在对多 agent 系统进行更新时, 当把前件 (1—3) 作为强行约束时, 将产生一个满足后件 (4) 的一个状态。也就是说: 虽然 Agent 能够和语义上含糊的句子进行通信, 但是这样的句子被听话者认为是含糊的情况不可能发生。

另一方面, 如果前件 (1—3) 被作为缺省约束应用时, 如说话者违反行为准则时, 则就不能保证评价准则成立。

同样在构建智能系统时, 检测含糊性也是十分重要的, 因为它能够通知系统去实现修补策略。同时也说明了含糊和析取之间的区别, 如果处理 1 个含糊句 S 和 m 个知识的析取 $\bigvee K_{k=1}^m \varphi_k$ 的方式相

同, 这将导致更加弱的、形如 $(M', w', V') \models C_{i|i \cup G} \bigvee_{k=1}^m \varphi_k$ 的后件。

定义 4 信息递增。

假设 $\{K_i\}_{i \in A}^*$ 表示有限个元素 $\{K_i\}_{i \in A}$ 的串联集合, 其中也包括空序列 ϵ 。

如果 $\vec{K} \in \{K_i\}_{i \in A}^*$ 并且 $(M, w, V) \models \vec{K}\varphi$, 则有 $\text{tell}(i, G, S)(M, w, V) \models \vec{K}\varphi$ 成立。

这里的信息递增的定义是建立在文献 [11] 中 Groeneveld 的“描述信息递增”的基础上进行的。

5 含糊知识的更新

图 1 和图 2 的 Kripke 结构分别代表了 Agent 1 和 Agent 2 的信息状态。考虑有 2 个命题变量 p 和 q , Agent 1 知道 p 成立, 但它无法确定 q 的真值。另外, Agent 1 不知道 Agent 2 是否知道 p 或 q ; 或者 Agent 2 不知道 Agent 1 是否知道 p 或 q 。

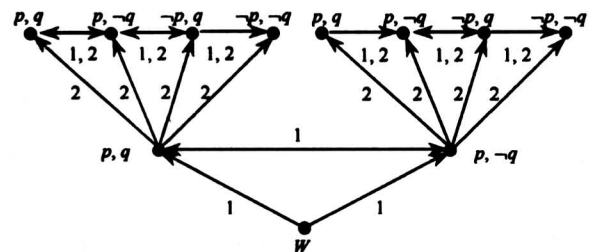


图 1 表示 agent 1 信息状态的 Kripke 结构
Fig.1 Kripke structure representing information state of agent 1

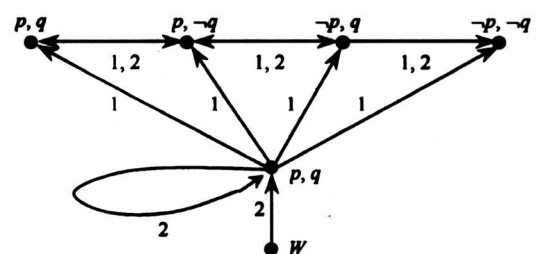


图 2 表示 agent 2 信息状态的 Kripke 结构
Fig.2 Kripke structure representing information state of agent 2

由图 1 可知, 实际上下列条件成立:

$$(M, w, V) \models K_1 p \wedge \neg K_1 q,$$

$$(M, w, V) \models \neg K_1 K_2 (C_{1,2} p \vee C_{1,2} q).$$

由图 2 可知 Agent 2 认为下列条件成立:

$$(M, w, V) \models K_2 p \wedge q,$$

$$(M, w, V) \models (K_2 (K_1 p \wedge K_1 q)).$$

如果 Agent 1 告诉 Agent 2 有关 S 的情况是:

$\tau(S) = \{p, q\}$, 于是图 1 和图 2 中的模型必须用 $\text{tell}(1, \{2\}, S)$ 更新, 这样更新后的 Agent 1 和 Agent 2 的信息状态如图 3 和图 4 所示。

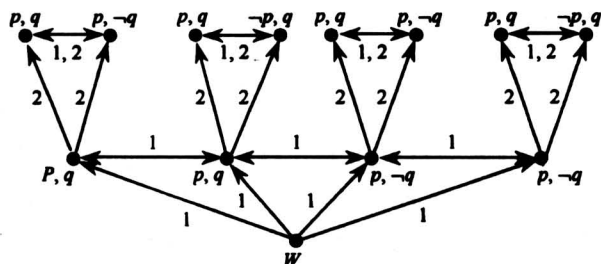


图 3 用 S 更新后的 agent 1 的信息状态
Fig.3 Information states of agent 1 after updating with S

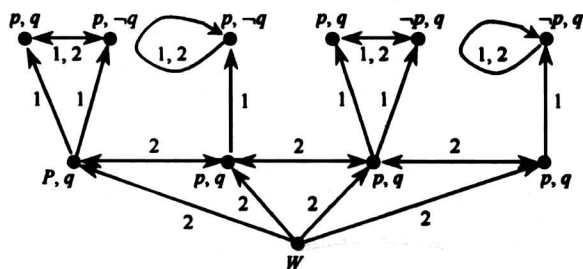


图 4 用 S 更新后的 agent 2 的信息状态
Fig.4 Information state of agent 2 after updating with S

决定 Agent 2 的结果信息是十分复杂的。在很大程度上 Agent 2 遵循合作准则, 知道 Agent 1 违反了行为准则, 但是 Agent 1 没有感觉到这一点。为了简明起见, 假设说话者充分信任听话者, 也就是说, 他们认为听话者都遵循合作准则。

对于图 4 而言, 存在 4 种可能。对于每个 S 的知识有 2 个: S 的知识之一是 Agent 2 认为 Agent 1 的认为 p 为真, 这是因为它不能确定 q 的真值, 或者由于它知道 q 不为真; 另一个 S 的知识, 对于 q 来说, 情况同上, 这里不再重述。

可见通过 tell 函数对基于 Kripke 结构的 multi-agent 系统的更新, 使系统能够正确地把含糊子句传递的信息中语义含糊和感觉含糊区分开来。

6 实例分析

例如, 为了证明 $(A \wedge B \Rightarrow A)$, 其中 $\tau(A) = \{a_1, a_2\}$, B 是不含糊的。必须证明下列 4 种情况都成立:

$$\begin{aligned} &|= (a_1 \wedge B) \rightarrow a_1; \quad |= (a_1 \wedge B) \rightarrow a_2; \\ &|= (a_2 \wedge B) \rightarrow a_1; \quad |= (a_2 \wedge B) \rightarrow a_2. \end{aligned}$$

下面给出每一种情况的证明树, 如图 5 所示。

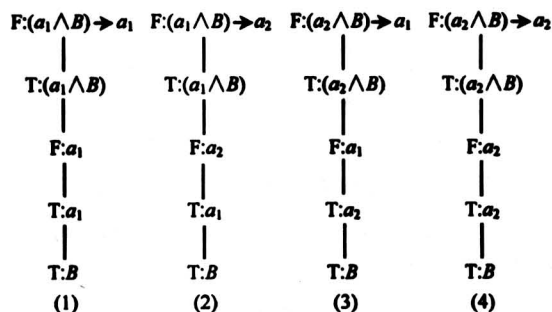


图 5 4 种情况分别对应的证明树
Fig.5 Proof trees corresponding to four case

实际上, 图 5 中的证明树在结构上是相同的, 与它们是否包含含糊信息无关, 这表明了一种策略: 可以延迟那些不含含糊的信息, 而合并它们相同的证明树。于是上述 4 种情况可以用带标号的紧凑证明树表示, 如图 6 所示。

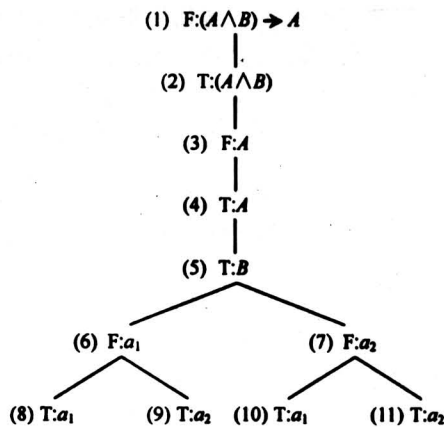


图 6 紧凑的带标号证明树
Fig.6 Compact labeled proof tree

由于 A 是含糊的, 因而不符合提出的含糊性处理规则, 故在图 6 中的 (3) 和 (4) 不能合成 1 个分支。通过对图 5 和图 6 的比较可见, 所提出的含糊性处理方法的结构简单, 处理方便。

7 相关工作分析

处理含糊性的一般方法是基于多值逻辑, 尤其是 Zadeh 提出的模糊逻辑 (被认为是泛化的多值逻辑) 倍受重视^[12]。虽然用模糊逻辑对连续的观测值和模糊量词之间的关系进行建模是一种较好的形

式化方法,但是它不适合于形式化的逻辑推理。于是 Fine 提出了运用超值 (supervaluation) 来处理含糊性的方法^[13]。近来 Williamson 把含糊认为是一种认知 (epistemic) 现象,并提出了一种不同于 Fine 的处理含糊性的方法^[14],是一种基于模态逻辑的方法,并且还融合和扩展了前人处理含糊性的有关方法。但是这些方法主要关注含糊的哲学含义,而没有提出如何在 AI 知识库中对含糊性进行管理。笔者提出基于赋值 Kripke 结构的多 agent 系统中含糊性处理方法,着重对语义含糊和感觉含糊进行了比较,提出处理含糊性的有关准则,并用具体实例说明对含糊性知识的处理,这对于研究多 agent 系统有一定的实用价值。

8 结语

在多 Agent 系统中含糊性可以在一定程度上利用 Kripke 结构加以处理,但是由于多 Agent 系统的复杂性,笔者对一些复杂情况做了简化处理,给出一个在多 Agent 系统中如何形式化定义和解决含糊性的大致轮廓。笔者所建立的方法只是初步的,还有许多问题需要解决,像如何把此方法扩展到一阶(动态)逻辑?哪些经验公式在此方法中可以继续应用?这些都是下一步应该着重研究的目标。

参考文献

- [1] Castelfranchi C. Modeling social action for AI agents [J]. *Artificial Intelligence*, 1998, 103: 157~182
- [2] Buvac S. Semantic ambiguity and under-specification [M]. USA: Stanford CSLI Publications, 1996. 101~124
- [3] Chellas F. *Modal logic* [M]. UK: Cambridge University Press, 1990. 52~63
- [4] Popkorn S. First steps in modal logic [M]. UK: Cambridge University Press, 1994. 166~170
- [5] Fagin R, Halpern J. Reasoning about knowledge [M]. USA: MIT Press, 1995. 83~103
- [6] Pinkal M. Logic and lexicon: the semantics of the indefinite [J]. *Studies in Linguistics and Philosophy*, 1995, 56: 86~92
- [7] Poesio M, Traum D. Conversational actions and discourse situations [J]. *Computation Intelligence*, 1997, 13(3): 309~347
- [8] FIPA-97. Agent communication language: specification 2, version 0.2 [S]. Geneva, Switzerland: Foundation for Intelligent Physical Agents (FIPA), 1997. 303~316
- [9] McRoy S, Hirst G. The repair of speech act misunderstandings by abductive inference [J]. *Computational Linguistics*, 1995, 21(4): 435~478
- [10] Francez N, Berg J. A multi-agent extension of DRT [A]. *Proceedings of the 1st International Workshop on Computational Semantics (IWCS-1)* [C]. The Netherlands: Tilburg University, 1994. 81~90
- [11] Groeneveld W. Logical investigations into dynamic semantics [D]. Institute for Logic, Language and Computation, University of Amsterdam, 1995. 46~52
- [12] Zadeh L A. Fuzzy logic and approximate reasoning [J]. *Synthese*, 1975, 30: 407~428
- [13] Fine K. Vagueness, truth and logic [J]. *Synthese*, 1975, 30: 263~300
- [14] Williamson T. Vagueness, the problem of philosophy [M]. Routledge, London, 1994. 210~234

Methods of Processing Ambiguity in Multi-Agent System

Wang Ziqiang, Feng Boqin

(Department of Computer Science, Xi'an Jiaotong University, Xi'an 710049, China)

[Abstract] In order to give a more refined notion of information that is conveyed by ambiguous expressions, this paper analyzes how ambiguity affects the knowledge of the dialog participants and, especially, what they know about each other after an ambiguous sentence has been uttered. Methods of processing ambiguity based on Kripke structures in a multi-agent system is presented. The information states of multi-agent system are represented as a Kripke structures. The agents communicate with each other by means of a tell function, whose application is constrained by an implementation of some of Grice's maxims. The method can easily distinguish semantic ambiguity from perceived ambiguity in the information conveyed by ambiguous sentences, so it is of

practical value for researching on multi-agent systems.

[Key words] multi-agent system; ambiguity; processing method; Kripke structure

(cont. from p.71)

The Assessment of the Impacts of Land Use Change on the Ecosystem Carbon Sink

Wu Jianguo¹, Zhang Xiaoquan², Xu Deying²

(1. *Research Center of Impact of Climate Change of Chinese State Environmental Protection Administration, Beijing 100012, China*; 2. *Research Institute of Forest Environment and Ecology of Chinese Academy of Forestry Science, Beijing 100091, China*)

[Abstract]

Based on the analyzing of the carbon cycle under different ecosystem in Liupan mountain forest zone and the conception of carbon sink and source, the impacts of land use change on the carbon sink of ecosystem were assessed. It is found that the function of ecosystem carbon source or sink under different land uses includes carbon storage, input and output of ecosystem carbon pool, net change of carbon storage and stabilization of soil organic carbon (SOC). The ecosystem carbon storage under natural secondary forest or plantation ecosystem is higher than that under cropland or rangeland ecosystem. The storage of ecosystem carbon, SOC, active SOC pool, slow SOC, protected and unprotected fraction of SOC and labile fraction of SOC under secondary forest or plantation ecosystem are higher than those under cropland and rangeland ecosystem. Natural secondary forest ecosystem is strong carbon sink, while plantation ecosystem is weak carbon sink, and cropland or rangeland ecosystem is source. The measures of increasing ecosystem sink include increasing input of ecosystem carbon pool, decreasing output of ecosystem carbon pool and increasing stabilization of SOC in ecosystem.

[Key words] land use change; soil organic carbon; carbon cycle; ecosystem carbon sink/source